CNC 8060 8065

Installation manual

(Ref: 1402)

FAGOR AUTOMATION

MACHINE SAFETY

It is up to the machine manufacturer to make sure that the safety of the machine is enabled in order to prevent personal injury and damage to the CNC or to the products connected to it. On start-up and while validating CNC parameters, it checks the status of the following safety elements. If any of them is disabled, the CNC shows a warning message.

- Feedback alarm for analog axes.
- Software limits for analog and sercos linear axes.
- Following error monitoring for analog and sercos axes (except the spindle) both at the CNC and at the drives.
- Tendency test on analog axes.

FAGOR AUTOMATION shall not be held responsible for any personal injuries or physical damage caused or suffered by the CNC resulting from any of the safety elements being disabled.

HARDWARE EXPANSIONS

FAGOR AUTOMATION shall not be held responsible for any personal injuries or physical damage caused or suffered by the CNC resulting from any hardware manipulation by personnel unauthorized by Fagor Automation.

If the CNC hardware is modified by personnel unauthorized by Fagor Automation, it will no longer be under warranty.

COMPUTER VIRUSES

FAGOR AUTOMATION guarantees that the software installed contains no computer viruses. It is up to the user to keep the unit virus free in order to guarantee its proper operation.

Computer viruses at the CNC may cause it to malfunction. An antivirus software is highly recommended if the CNC is connected directly to another PC, it is part of a computer network or floppy disks or other computer media is used to transmit data.

FAGOR AUTOMATION shall not be held responsible for any personal injuries or physical damage caused or suffered by the CNC due a computer virus in the system.

If a computer virus is found in the system, the unit will no longer be under warranty.

All rights reserved. No part of this documentation may be transmitted, transcribed, stored in a backup device or translated into another language without Fagor Automation's consent. Unauthorized copying or distributing of this software is prohibited.

The information described in this manual may be changed due to technical modifications. Fagor Automation reserves the right to make any changes to the contents of this manual without prior notice.

All the trade marks appearing in the manual belong to the corresponding owners. The use of these marks by third parties for their own purpose could violate the rights of the owners.

It is possible that CNC can execute more functions than those described in its associated documentation; however, Fagor Automation does not guarantee the validity of those applications. Therefore, except under the express permission from Fagor Automation, any CNC application that is not described in the documentation must be considered as "impossible". In any case, Fagor Automation shall not be held responsible for any personal injuries or physical damage caused or suffered by the CNC if it is used in any way other than as explained in the related documentation.

The content of this manual and its validity for the product described here has been verified. Even so, involuntary errors are possible, thus no absolute match is guaranteed. Anyway, the contents of the manual is periodically checked making and including the necessary corrections in a future edition. We appreciate your suggestions for improvement.

The examples described in this manual are for learning purposes. Before using them in industrial applications, they must be properly adapted making sure that the safety regulations are fully met.

INDEX

About the product 7
Declaration of conformity 11
Declaration of conformity 13
Version history 15
Safety conditions 17
Warranty terms 21
Material returning terms 23
CNC maintenance 25

CHAPTER 1 SOFTWARE INSTALLATION.

1.1 Software installation at the CNC. 27
1.1.1 Work modes and software protection at the CNC. 29
1.2 Software installation at the PC (simulator). 32
1.3 Updating the software version. 34
1.3.1 Software update since previous versions. 35
1.4 Requirements before and after CNC setup. 36
1.5 Installation of third-party software (CNC 8065 only). 37
1.6 Software configuration. 38
1.6.1 MTB (Machine Tool Builder) folder. 39
1.6.2 USERS folder. 40

CHAPTER 2 MACHINE PARAMETERS.

2.1 Parameter matching between the CNC and the Sercos drive. 43
2.2 Parameters to verify before the startup. 45
2.3 General machine parameters. 47
2.4 General machine parameters. Execution channels. 76
2.5 Machine parameters for the axes and spindles. 99
2.6 Machine parameters for the axes. Work sets. 131
2.7 Machine parameters for JOG mode. 166
2.7.1 Example of how to set the handwheels and jog keys. 171
2.8 Machine parameters for the M function table. 175
2.9 Machine parameters for kinetics. 177
2.9.1 Kinematics configuration. 177
2.9.2 Definition of the spindle kinetics (types 1 through 8) 181
2.9.3 Definition of the table kinetics (Types 9 through 12). 185
2.9.4 Definition of the kinematics of the spindle - table (Types 13 through 16). 189
2.9.5 Definition of the spindle kinetics (types 17 through 24) 193
2.9.6 Definition of the –C– axis kinematics (Types 41 through 42). 197
2.9.7 Definition of the –C– axis kinematics (Type 43) 199
2.9.8 Definition of the OEM kinematics (Types 100 through 105) 200
2.9.9 Configuration of angular transformations. 201
2.10 Machine parameters for the magazine. 203
2.10.1 Types of tool magazine. 206
2.11 Machine parameters for HMI (Interface). 208
2.12 OEM machine parameters. 214

CHAPTER 3 INTRODUCTION TO THE PLC.

3.1 PLC program. 220
3.2 Modular structure of the PLC program. 221
3.3 PLC program execution. 222
3.4 PLC resources. 223
3.4.1 Numbering of the physical inputs and outputs. 226
3.5 Operation of a timer. 228
3.5.1 Monostable mode. TG1 input. 230
3.5.2 Delayed activation mode. TG2 input. 232
3.5.3 Delayed deactivation mode. TG3 input. 234
3.5.4 Signal limiting mode. TG4 Input. 236
3.6 Operation of a counter. 238

CHAPTER 4 PLC PROGRAMMING.

4.1 Directing instructions. 243

FAGOR

CNC 8060
CNC 8065

(REF: 1402)

4.2 Consulting instructions. 247
4.2.1 Simple consulting instructions. 247
4.2.2 Flank detection instructions. 248
4.2.3 Comparing instructions. 249
4.3 Operators and symbols. 250
4.4 Action instructions. 251
4.4.1 Assignment binary instructions. 252
4.4.2 Conditional binary instructions. 253
4.4.3 Sequence breaking action instructions. 254
4.4.4 Arithmetic action instructions. 255
4.4.5 Logic action instructions. 257
4.4.6 Specific action instructions. 259
4.4.7 Action instruction of the electronic cam. 262
4.4.8 Instructions for independent move: positioning. 264
4.4.9 Instructions for independent move: synchronization. 266
4.4.10 Instructions for coordinate latching using a probe or digital input. 268
4.5 Summary programming commands. 272

CHAPTER 5 CNC-PLC COMMUNICATION.

5.1 Auxiliary –M– functions. 276
5.1.1 Special considerations with the multi-spindle option and channels. 277
5.2 Auxiliary –H– functions. 278
5.2.1 Special considerations with the multi-spindle option and channels. 279
5.3 Auxiliary –S– function. 280
5.3.1 Special considerations with the multi-spindle option and channels. 281
5.4 Transferring auxiliary functions -M-, -H-, -S-. 282
5.4.1 Synchronized transfer. 283
5.4.2 Non-synchronized transfer. 284
5.5 Displaying PLC errors and messages. 285

CHAPTER 6 LOGIC CNC INPUTS AND OUTPUTS.

6.1 General consulting signals. 288
6.2 Consulting signals for axes and spindles. 298
6.3 Consulting signals for the spindle. 304
6.4 Consultation signals of the independent interpolator. 306
6.5 Tool manager consulting signals. 308
6.6 Keystroke consulting signals. 310
6.7 General modifiable signals. 314
6.8 Modifiable signals for axes and spindles. 323
6.9 Spindle modifiable signals. 327
6.10 Modifiable signals of the independent interpolator. 331
6.11 Tool manager modifiable signals. 332
6.12 Keystroke modifiable signals. 335

CHAPTER 7 TOOL AND MAGAZINE MANAGEMENT.

7.1 Types of tool magazine. 343
7.2 Tool table, active tool table and tool magazine table 345
7.3 Communication between manager and PLC. 346
7.3.1 Manager --> PLC communication. 347
7.3.2 PLC --> Manager communication. 348
7.3.3 Manager Emergency. 350
7.3.4 Tool monitoring. 351
7.4 Variables related to tool magazine management. 352
7.5 Tool loading and unloading from the magazines. 353
7.6 Magazine-less system. 354
7.6.1 Valid operations and marks activated by the PLC with each one of them. 355
7.6.2 Detailed description of the operations of the magazine. 356
7.6.3 Basic PLC programming. 357
7.7 Turret type magazine. 358
7.7.1 Valid operations and marks activated by the PLC with each one of them. 359
7.7.2 Detailed description of the operations of the magazine. 361
7.7.3 Communication between the PLC and the M06 subroutine. 363
7.7.4 Program of the M06 subroutine. 364
7.7.5 Basic PLC programming. 367
7.8 Synchronous magazine without changer arm. 368
7.8.1 Valid operations and marks activated by the PLC with each one of them. 369
7.8.2 Detailed description of the operations of the magazine. 371
7.8.3 Communication between the PLC and the M06 subroutine. 374
7.8.4 Program of the M06 subroutine. 375
7.8.5 Basic PLC programming. 379

FAGOR

CNC 8060
CNC 8065

(REF: 1402)

7.9 Synchronous magazine with changer arm and 1 claw ... 381
7.9.1 Valid operations and marks activated by the PLC with each one of them. ... 382
7.9.2 Detailed description of the operations of the magazine. ... 384
7.9.3 Communication between the PLC and the M06 subroutine. ... 387
7.9.4 Program of the M06 subroutine. ... 388
7.9.5 Basic PLC programming. ... 393
7.10 Synchronous magazine with changer arm and 2 claws. ... 395
7.10.1 Valid operations and marks activated by the PLC with each one of them. ... 396
7.10.2 Detailed description of the operations of the magazine. ... 398
7.10.3 Communication between the PLC and the M06 subroutine. ... 401
7.10.4 Program of the M06 subroutine. ... 402
7.10.5 Basic PLC programming. ... 407
7.11 Asynchronous magazine with changer arm. ... 409
7.11.1 Valid operations and marks activated by the PLC with each one of them. ... 410
7.11.2 Detailed description of the operations of the magazine. ... 412
7.11.3 Communication between the PLC and the M06 subroutine. ... 415
7.11.4 Program of the M06 subroutine. ... 416
7.11.5 Basic PLC programming. ... 422

CHAPTER 8 KEY CODES.

8.1 Example for simulating the keyboard from the PLC. ... 428
8.2 Assigning codes for a Spanish keyboard. ... 430

CHAPTER 9 CONCEPTS.

9.1 Configure the name and number of axes and spindles. ... 433
9.1.1 Configure the number of axes and spindles of the system. ... 434
9.1.2 Configure the number of axes and spindles of the channels. ... 435
9.1.3 Configuration examples. ... 436
9.2 Configure an axis as rotary axis. ... 446
9.3 Configure two axes as a tandem axis. ... 449
9.3.1 Tandem axis configuration. Machine parameters. ... 450
9.3.2 Effect of the preload. ... 452
9.3.3 Tandem axis configuration. Block diagram. ... 454
9.3.4 Tandem related variables. ... 456
9.3.5 Tandem adjustment procedure. ... 457
9.4 Analog axes. ... 458
9.4.1 Configure the number of the analog output and of the feedback input. ... 458
9.4.2 Configure 2 axes with the same feedback input and analog output. ... 460
9.5 Multi-axis management. ... 461
9.5.1 Configuration of a multi-axis group. Machine parameters. ... 463
9.5.2 Configuration of a multi-axis group. The PLC program generates an error. ... 466
9.5.3 Changing the set and the gear at the CNC and at the drive. ... 467
9.5.4 Configuration examples. ... 468
9.6 Home search. ... 470
9.6.1 Home search (axes and spindles). ... 472
9.6.2 Home search (gantry axes). ... 475
9.7 Software limits of the axes. ... 477
9.7.1 How to set the software travel limits. ... 479
9.7.2 Set the tolerance for an axis located at the software travel limits. ... 481
9.8 Configure a handwheel as "feed handwheel". ... 482
9.9 Configuration of the HSC mode (High Speed Cutting). ... 484
9.9.1 Configuration of the HSC mode. ... 485
9.9.2 Influence of the type of acceleration and of the filters in HSC mode. ... 488
9.9.3 Procedure for analysis and adjustment of the HSC. ... 489
9.9.4 Summary of the useful variables to analyze the HSC. ... 492
9.9.5 The loops and the variables. ... 497
9.10 Calculation of the kinematics dimensions. ... 498
9.10.1 Swivel (angular) spindle. Calculation of the dimensions using a probe. ... 499
9.10.2 Swivel (angular) spindle. Calculation of the dimensions using a dial indicator. ... 505
9.10.3 Rotary table. Calculation of the dimensions using a probe. ... 509
9.11 Management of several keyboards. ... 513
9.11.1 How to configure the feature. ... 513
9.11.2 Operation of the jog panels. ... 518
9.12 Remote OpenPCS (8065 CNC only). ... 519
9.13 Assigning a help text to the graphic softkeys and to the CNC status icon. ... 521
9.14 Remote module RCS-S. ... 522
9.14.1 Configure the module like as a node of the Sercos bus. ... 523
9.14.2 Configure the analog outputs. ... 523
9.14.3 Configuration of the feedback inputs. ... 524
9.14.4 Configure the feedback input for a handwheel. ... 524
9.14.5 Parameter setting example. ... 525

FAGOR

CNC 8060
CNC 8065

(REF: 1402)

CHAPTER 10 CNC VARIABLES.

10.1 Understanding how variables work. ... 527
10.1.1 Accessing numeric variables from the PLC. ... 529
10.2 Variables in a single-channel system. ... 530
10.3 Variables in a multi-channel system. ... 533
10.4 Variables related to general machine parameters. ... 536
10.5 Variables related to the machine parameters of the channels. ... 557
10.6 Variables related to axis and spindle machine parameters. ... 580
10.7 Variables related to the sets of machine parameters. ... 617
10.8 Variables related to machine parameters for JOG mode. ... 671
10.9 Variables related to machine parameters for M functions. ... 675
10.10 Variables related to kinematic machine parameters. ... 677
10.11 Variables related to machine parameters for the tool magazine. ... 681
10.12 Variables related to OEM machine parameters. ... 684
10.13 Variables associated with the status and resources of the PLC. ... 686
10.14 PLC consulting logic signals; general. ... 690
10.15 PLC consulting logic signals; axes and spindles. ... 701
10.16 PLC consulting logic signals; spindles. ... 706
10.17 PLC consulting logic signals; independent interpolator. ... 708
10.18 PLC consulting logic signals; tool manager. ... 710
10.19 PLC consulting logic signals; keys. ... 713
10.20 PLC modifiable logic signals; general. ... 714
10.21 PLC modifiable logic signals; axes and spindles. ... 722
10.22 PLC modifiable logic signals; spindles. ... 728
10.23 PLC modifiable logic signals; independent interpolator. ... 730
10.24 PLC modifiable logic signals; tool manager. ... 731
10.25 PLC modifiable logic signals; keys. ... 736
10.26 Variables related to the machine configuration. ... 737
10.27 Variables related to volumetric compensation. ... 745
10.28 Variables associated with the Mechatrolink bus. ... 746
10.29 Variables related to synchronized switching. ... 748
10.30 PWM related variables. ... 749
10.31 Variables related to cycle time. ... 751
10.32 Variables associated with the feedback inputs for analog axes. ... 753
10.33 Variables associated with the analog inputs and outputs. ... 755
10.34 Variables associated with the velocity command and the feedback of the drive. ... 756
10.35 Variables related to the change of gear and set of the Sercos drive. ... 758
10.36 Variables related to loop adjustment. ... 759
10.37 Variables related to the loop of the axis or of the tandem spindle. ... 767
10.38 Variables related to user tables (zero offset table). ... 769
10.39 Variables related to user tables (fixture table). ... 774
10.40 Variables related to user tables (arithmetic parameters table). ... 776
10.41 Variables related to the position of the axes. ... 780
10.42 Variables related to spindle position. ... 786
10.43 Feedrate related variables. ... 788
10.44 Variables associated with acceleration and jerk on the tool path. ... 793
10.45 Variables related to managing the feedrate in HSC mode. ... 794
10.46 Variables related to spindle speed. ... 798
10.47 Variables associated with the tool manager. ... 806
10.48 Variables related to managing the tool magazine and the tool changer arm. ... 808
10.49 Variables related to the active tool and to the next one. ... 810
10.50 Variables associated with any tool. ... 822
10.51 Variables associated with the tool being prepared. ... 831
10.52 Variables related to jog mode. ... 839
10.53 Variables related to the programmed functions. ... 845
10.54 Variables related to the electronic cam. ... 872
10.55 Variables related to the independent axes. ... 874
10.56 Variables associated with the virtual tool axis. ... 881
10.57 Variables defined by the user. ... 882
10.58 General variables of the CNC. ... 883
10.59 Variables related to CNC status. ... 886
10.60 Variables associated with the part-program being executed. ... 891
10.61 Interface related variables. ... 895

FAGOR

CNC 8060
CNC 8065

(REF: 1402)

ABOUT THE PRODUCT

BASIC CHARACTERISTICS.

Basic characteristics.	8060 M	8060 T	8065 M	8065 T
PC-based system.	Closed system		Open system	
Operating system.	Windows XPe		Windows XP	
Number of axes.	3 a 5		3 to 28	
Number of spindles.	1	2	1 to 4	
Number of tool magazines.	1	2	1 to 4	
Number of execution channels.	1	2	1 to 4	
Number of handwheels.	1 a 3		1 to 12	
Type of servo system.	Analog / Sercos Digital		Analog / Digital Sercos / Digital Mechatrolink	
Communications.	RS485 / RS422 / RS232 Ethernet			
Integrated PLC. PLC execution time. Digital inputs / Digital outputs. Marks / Registers. Timers / Counters. Symbols.	 < 1ms/K 1024 / 1024 8192 / 1024 512 / 256 Unlimited			
Block processing time.	< 1.5 ms		< 1 ms	

Remote modules.	RIOW	RIO5	RIO70	RCS-S
Communication with the remote modules.	CANopen	CANopen	CANfagor	Sercos
Digital inputs per module.	8	16 / 32	16	- - -
Digital outputs per module.	8	24 / 48	16	- - -
Analog inputs per module.	4	4	8	- - -
Analog outputs per module.	4	4	4	4
Inputs for PT100 temperature sensors.	2	2	- - -	- - -
Feedback inputs.	- - -	- - -	4 (*)	4 (**)

(*) Differential TTL / Sinusoidal 1 Vpp (**) TTL / Differential TTL / Sinusoidal 1 Vpp / SSI protocol

Customizing (on an open system only).
PC-based open system, fully customizable. INI configuration files. FGUIM visual configuration tool. Visual Basic®, Visual C++®, etc. Internal databases in Microsoft® Access. OPC compatible interface

SOFTWARE OPTIONS.

Bear in mind that some of the features described in this manual depend on the software options that are installed. The information of the following table is informative only; when purchasing the software options, only the information provided in the ordering handbook is valid.

Software options (·M· model).

	8060 M	8065 M		8065 M Power	
		Basic	Pack 1	Basic	Pack 1
Open system. Access to the administrator mode.	- - -	- - -	- - -	Option	Option
Number of execution channels	1	1	1	1	1 to 4
Number of axes	3 a 5	3 to 6	5 to 8	5 to 12	8 to 28
Number of spindles	1	1	1 to 2	1 to 4	1 to 4
Number of tool magazines	1	1	1	1 to 2	1 to 4
Limited to 4 interpolated axes	Option	Option	Option	Option	Option
IEC 61131 language	- - -	- - -	Option	Option	Option
HD graphics	Option	Option	Option	Standard	Standard
Conversational IIP	Option	Option	Option	Option	Option
Dual-purpose machines (M-T)	- - -	- - -	- - -	Option	Standard
"C" axis	Option	Standard	Standard	Standard	Standard
Dynamic RTCP	Option	- - -	Option	Option	Standard
HSSA machining system.	Option	Standard	Standard	Standard	Standard
Probing canned cycles	Option	Option	Standard	Standard	Standard
Tandem axes	Option	- - -	Option	Standard	Standard
Synchronism and cams	- - -	- - -	- - -	Option	Standard
Tangential control	- - -	- - -	Standard	Standard	Standard
Volumetric compensation (up to 10 m^3).	- - -	- - -	- - -	Option	Option
Volumetric compensation (more than 10 m^3).	- - -	- - -	- - -	Option	Option
ProGTL3 language.	Option	Option	Option	Option	Option
Part-program translator.	Option	Option	Option	Option	Option

Software options (·T· model).

	8060 T	8065 T		8065 T Power	
		Basic	Pack 1	Basic	Pack 1
Open system. Access to the administrator mode.	- - -	- - -	- - -	Option	Option
Number of execution channels	2	1	1 to 2	1 to 2	1 to 4
Number of axes	3 a 5	3 to 5	5 to 7	5 to 12	8 to 28
Number of spindles	2	2	2	3 to 4	3 to 4
Number of tool magazines	2	1	1 to 2	1 to 2	1 to 4
Limited to 4 interpolated axes	Option	Option	Option	Option	Option
IEC 61131 language	- - -	- - -	Option	Option	Option
HD graphics	Option	Option	Option	Standard	Standard
Conversational IIP	Option	Option	Option	Option	Option
Dual-purpose machines (T-M)	- - -	- - -	- - -	Option	Standard
"C" axis	Option	Option	Standard	Standard	Standard
Dynamic RTCP	- - -	- - -	- - -	Option	Standard
HSSA machining system.	Option	Option	Standard	Standard	Standard
Probing canned cycles	Option	Option	Standard	Standard	Standard
Tandem axes	Option	- - -	Option	Standard	Standard
Synchronism and cams	Option	- - -	Option	Option	Standard
Tangential control	- - -	- - -	- - -	Option	Standard
Volumetric compensation (up to 10 m^3).	- - -	- - -	- - -	Option	Option
Volumetric compensation (more than 10 m^3).	- - -	- - -	- - -	Option	Option
ProGTL3 language.	- - -	- - -	- - -	- - -	- - -
Part-program translator.	Option	- - -	- - -	- - -	- - -

DECLARATION OF CONFORMITY

The manufacturer:

Fagor Automation S. Coop.

Barrio de San Andrés Nº 19, C.P.20500, Mondragón -Guipúzcoa- (Spain).

Declares:

The manufacturer declares under their exclusive responsibility the conformity of the product:

8060 CNC

Consisting of the following modules and accessories:

CN60-10H, CN60-10HT, CN60-10V, CN60-10VT
OP-PANEL, OP-PANEL+SPDL RATE
Remote Modules RIOW, RIO5, RIO70, RCS-S, RIOR.

Note. Some additional characters may follow the model references indicated above. They all comply with the directives listed here. However, compliance may be verified on the label of the unit itself.

Referred to by this declaration with following directives:

Low-voltage regulations.

IEC 60204-1:2005/A1:2008 Electrical equipment on machines. Part1. General requirements.

Regulation on electromagnetic compatibility.

EN 61131-2: 2007 PLC. Part 2. Equipment requirements and tests.

According to the European Community Directives 2006/95/EC on Low Voltage and 2004/108/EC on Electromagnetic Compatibility and their updates.

In Mondragón, February 1st, 2014.

Fagor Automation, S. Coop.

Director Gerente
Pedro Ruiz de Aguirre

FAGOR

CNC 8060
CNC 8065

(REF: 1402)

DECLARATION OF CONFORMITY

The manufacturer:

Fagor Automation S. Coop.

Barrio de San Andrés Nº 19, C.P.20500, Mondragón -Guipúzcoa- (Spain).

Declares:

The manufacturer declares under their exclusive responsibility the conformity of the product:

8065 CNC

Consisting of the following modules and accessories:

8065-M-ICU, 8065-T-ICU
MONITOR-LCD-10K, MONITOR-LCD-15, MONITOR-SVGA-15
HORIZONTAL-KEYB, VERTICAL-KEYB, OP-PANEL
BATTERY
Remote Modules RIOW, RIO5, RIO70, RCS-S.

Note. Some additional characters may follow the model references indicated above. They all comply with the directives listed here. However, compliance may be verified on the label of the unit itself.

Referred to by this declaration with following directives:

Low-voltage regulations.

IEC 60204-1:2005/A1:2008 Electrical equipment on machines. Part1. General requirements.

Regulation on electromagnetic compatibility.

EN 61131-2: 2007 PLC. Part 2. Equipment requirements and tests.

According to the European Community Directives 2006/95/EC on Low Voltage and 2004/108/EC on Electromagnetic Compatibility and their updates.

In Mondragón, September 1st, 2013.

Fagor Automation, S. Coop.

Director Gerente
Pedro Ruiz de Aguirre

VERSION HISTORY

Here is a list of the features added to each manual reference.

Ref. 1402

Software V01.00 (8060)
8060. First version.

SAFETY CONDITIONS

Read the following safety measures in order to prevent harming people or damage to this product and those products connected to it. Fagor Automation shall not be held responsible of any physical damage or defective unit resulting from not complying with these basic safety regulations.

Before start-up, verify that the machine that integrates this CNC meets the 89/392/CEE Directive.

PRECAUTIONS BEFORE CLEANING THE UNIT

If the CNC does not turn on when actuating the start-up switch, verify the connections.

Do not get into the inside of the unit.	Only personnel authorized by Fagor Automation may manipulate the inside of this unit.
Do not handle the connectors with the unit connected to AC power.	Before manipulating the connectors (inputs/outputs, feedback, etc.) make sure that the unit is not connected to AC power.

PRECAUTIONS DURING REPAIR

In case of a malfunction or failure, disconnect it and call the technical service.

Do not get into the inside of the unit.	Only personnel authorized by Fagor Automation may manipulate the inside of this unit.
Do not handle the connectors with the unit connected to AC power.	Before manipulating the connectors (inputs/outputs, feedback, etc.) make sure that the unit is not connected to AC power.

PRECAUTIONS AGAINST PERSONAL DAMAGE

Interconnection of modules.	Use the connection cables provided with the unit.
Use proper cables.	To prevent risks, use the proper cables for mains, Sercos and Bus CAN recommended for this unit. In order to avoid electrical shock at the central unit, use the proper power (mains) cable. Use 3-wire power cables (one for ground connection).
Avoid electrical overloads.	In order to avoid electrical discharges and fire hazards, do not apply electrical voltage outside the range selected on the rear panel of the central unit.
Ground connection.	In order to avoid electrical discharges, connect the ground terminals of all the modules to the main ground terminal. Before connecting the inputs and outputs of this unit, make sure that all the grounding connections are properly made. In order to avoid electrical shock, before turning the unit on verify that the ground connection is properly made.
Do not work in humid environments.	8060. In order to avoid electrical discharges, always work under 90% of relative humidity (non-condensing) and 50 ºC (122 ºF). 8065. In order to avoid electrical discharges, always work under 90% of relative humidity (non-condensing) and 45 ºC (113 ºF).
Do not work in explosive environments.	In order to avoid risks or damages, do no work in explosive environments.

PRECAUTIONS AGAINST PRODUCT DAMAGE

Working environment.

This unit is ready to be used in industrial environments complying with the directives and regulations effective in the European Community.

Fagor Automation shall not be held responsible for any damage suffered or caused by the CNC when installed in other environments (residential or homes).

Install the unit in the right place.

It is recommended, whenever possible, to install the CNC away from coolants, chemical product, blows, etc. that could damage it.

This unit complies with the European directives on electromagnetic compatibility. Nevertheless, it is recommended to keep it away from sources of electromagnetic disturbance such as:

- Powerful loads connected to the same AC power line as this equipment.
- Nearby portable transmitters (Radio-telephones, Ham radio transmitters).
- Nearby radio/TV transmitters.
- Nearby arc welding machines.
- Nearby High Voltage power lines.

Enclosures.

The manufacturer is responsible of assuring that the enclosure involving the equipment meets all the currently effective directives of the European Community.

Avoid disturbances coming from the machine.

The machine must have all the interference generating elements (relay coils, contactors, motors, etc.) uncoupled.

Use the proper power supply.

Use an external regulated 24 Vdc power supply for the keyboard and the remote modules.

Grounding of the power supply.

The zero volt point of the external power supply must be connected to the main ground point of the machine.

Analog inputs and outputs connection.

Use shielded cables connecting all their meshes to the corresponding pin.

Ambient conditions.

8060. The storage temperature must be between +5 ºC and +55 ºC (41 ºF and 131 ºF).

8065. The storage temperature must be between +5 ºC and +45 ºC (41 ºF and 113 ºF).

The storage temperature must be between -25 ºC and 70 ºC (-13 ºF and 158 ºF).

Central unit enclosure.

Make sure that the needed gap is kept between the central unit and each wall of the enclosure.

Use a DC fan to improve enclosure ventilation.

Main AC power switch.

This switch must be easy to access and at a distance between 0.7 and 1.7 m (2.3 and 5.6 ft) off the floor.

PROTECTIONS OF THE UNIT ITSELF

Remote modules.

All the digital inputs and outputs have galvanic isolation via optocouplers between the CNC circuitry and the outside.

SAFETY SYMBOLS

Symbols that may appear on the manual.



Danger or prohibition symbol.
It indicates actions or operations that may hurt people or damage products.



Warning symbol.
It indicates situations that certain operations could cause and the suggested actions to prevent them.



Obligation symbol.
It indicates actions and operations that must be carried out.



Information symbol.
It indicates notes, warnings and advises.

Symbols that the product may carry.



Ground protection symbol.
It indicates that that point must be under voltage.

WARRANTY TERMS

INITIAL WARRANTY

All products manufactured or marketed by FAGOR carry a 12-month warranty for the end user which could be controlled by the our service network by means of the warranty control system established by FAGOR for this purpose.

In order to prevent the possibility of having the time period from the time a product leaves our warehouse until the end user actually receives it run against this 12-month warranty, FAGOR has set up a warranty control system based on having the manufacturer or agent inform FAGOR of the destination, identification and on-machine installation date, by filling out the document accompanying each FAGOR product in the warranty envelope. This system, besides assuring a full year of warranty to the end user, enables our service network to know about FAGOR equipment coming from other countries into their area of responsibility.

The warranty starting date will be the one appearing as the installation date on the above mentioned document. FAGOR offers the manufacturer or agent 12 months to sell and install the product. This means that the warranty starting date may be up to one year after the product has left our warehouse so long as the warranty control sheet has been sent back to us. This translates into the extension of warranty period to two years since the product left our warehouse. If this sheet has not been sent to us, the warranty period ends 15 months from when the product left our warehouse.

This warranty covers all costs of material and labour involved in repairs at FAGOR carried out to correct malfunctions in the equipment. FAGOR undertakes to repair or replace their products within the period from the moment manufacture begins until 8 years after the date on which it disappears from the catalogue.

It is entirely up to FAGOR to determine whether the repair is or not under warranty.

EXCLUDING CLAUSES

Repairs will be carried out on our premises. Therefore, all expenses incurred as a result of trips made by technical personnel to carry out equipment repairs, despite these being within the above-mentioned period of warranty, are not covered by the warranty.

Said warranty will be applied whenever the equipment has been installed in accordance with instructions, has not be mistreated, has not been damaged by accident or by negligence and has not been tampered with by personnel not authorised by FAGOR. If, once servicing or repairs have been made, the cause of the malfunction cannot be attributed to said elements, the customer is obliged to cover the expenses incurred, in accordance with the tariffs in force.

Other warranties, implicit or explicit, are not covered and FAGOR AUTOMATION cannot be held responsible for other damages which may occur.

WARRANTY ON REPAIRS

In a similar way to the initial warranty, FAGOR offers a warranty on standard repairs according to the following conditions:

PERIOD	12 months.
CONCEPT	Covers parts and labor for repairs (or replacements) at the network's own facilities.
EXCLUDING CLAUSES	The same as those applied regarding the chapter on initial warranty. If the repair is carried out within the warranty period, the warranty extension has no effect.

When the customer does not choose the standard repair and just the faulty material has been replaced, the warranty will cover just the replaced parts or components within 12 months.

For sold parts the warranty is 12 moths length.

SERVICE CONTRACTS

The SERVICE CONTRACT is available for the distributor or manufacturer who buys and installs our CNC systems.

MATERIAL RETURNING TERMS

When sending the central nit or the remote modules, pack them in its original package and packaging material. If the original packaging material is not available, pack it as follows:

1. Get a cardboard box whose three inside dimensions are at least 15 cm (6 inches) larger than those of the unit. The cardboard being used to make the box must have a resistance of 170 Kg (375 lb.).
2. Attach a label indicating the owner of the unit, person to contact, type of unit and serial number. In case of malfunction also indicate symptom and a brief description of the problem.
3. Wrap the unit in a polyethylene roll or similar material to protect it. When sending a central unit with monitor, protect especially the screen.
4. Pad the unit inside the cardboard box with poly-utherane foam on all sides.
5. Seal the cardboard box with packing tape or industrial staples.

CNC MAINTENANCE

CLEANING

The accumulated dirt inside the unit may act as a screen preventing the proper dissipation of the heat generated by the internal circuitry which could result in a harmful overheating of the unit and, consequently, possible malfunctions. Accumulated dirt can sometimes act as an electrical conductor and short-circuit the internal circuitry, especially under high humidity conditions.

To clean the operator panel and the monitor, a smooth cloth should be used which has been dipped into de-ionized water and /or non abrasive dish-washer soap (liquid, never powder) or 75º alcohol. Do not use highly compressed air to clean the unit because it could generate electrostatic discharges.

The plastics used on the front panel are resistant to grease and mineral oils, bases and bleach, dissolved detergents and alcohol. Avoid the action of solvents such as chlorine hydrocarbons, venzole, esters and ether which can damage the plastics used to make the unit's front panel.

PRECAUTIONS BEFORE CLEANING THE UNIT

Fagor Automation shall not be held responsible for any material or physical damage derived from the violation of these basic safety requirements.

- Do not handle the connectors with the unit connected to AC power. Before handling these connectors (I/O, feedback, etc.), make sure that the unit is not connected to main AC power.
- Do not get into the inside of the unit. Only personnel authorized by Fagor Automation may manipulate the inside of this unit.
- If the CNC does not turn on when actuating the start-up switch, verify the connections.

SOFTWARE INSTALLATION.

1

1.1 Software installation at the CNC.

Fagor delivers the CNC with the software properly installed in a compact flash type disk. Both at a CNC and at a PC (simulator), the necessary files for the CNC are located in the following folder. See *"1.6 Software configuration."* on page 38.

8060 CNC C:\FAGORCNC and its corresponding subfolders.

8065 CNC C:\CNC8070 and its corresponding subfolders.



The CNC software must not be re-installed or modified in any way without the express consent from Fagor Automation. Fagor Automation shall not be held responsible for any personal injuries, physical or material damage suffered or caused by the CNC due to software manipulation.

Software protection at the CNC. System with a write-protected (read-only) disk.

Fagor delivers the unit with the disk that is write-protected except for the folders or files that must be unprotected for the normal operation of the CNC. The disk has been pre-configured with three access modes, each offers a different protection level. See *"1.1.1 Work modes and software protection at the CNC."* on page 29.

Software setting.

It is up to manufacturer to set up the CNC and adapt it to his machine. The manufacturer can also customize the CNC's look using the screen customizing program FGUIM. Before using this tool, read the relevant documentation carefully.

CNC setting; the MTB folder.

Both at a CNC and at a PC (simulator), the OEM has the folder ..\MTB to save the modifications done at the CNC; such as the PLC program, the machine parameters, etc. The CNC manages the MTB folders as follows.

- No MTB folder appears when installing the software for the first time. On system startup and depending on the validation code, the system will move the corresponding folder from ..\CONFIGURATION to the CNC8070 and will rename it MTB.
- When changing the validation code, the system will return the MTB folder to the ..\CONFIGURATION folder and will give it its old name (MTB_M, MTB_T, etc); then, it will move the corresponding folder from ..\CONFIGURATION to CNC8070 and will rename it MTB.
- If the OEM has created the MTB folder manually, for example by copying it from a backup, the system will not make any changes on power-up nor when modifying the validation code.

Changing the language of the help files.

Fagor supplies the CNC with the help files installed in English. The CD that comes with the product contains the help files in different languages. You can change the help files installed by default with the ones provided on the CD.

Locate the folder *Help files* inside the CD, select one of the available languages and copy all the files to its CNC location. The help files installed at the CNC (or at the PC, if it is a simulator) are located in the following folder.

8060 CNC	C:\FagorCnc\Fagor\MMC\Help
8065 CNC	C:\Cnc8070\Fagor\MMC\Help

The help files can only be in one language at a time at the CNC. The language of the help files may be different from the one chosen for the interface.

1.1.1 Work modes and software protection at the CNC.

Fagor delivers the unit with a protected compact flash type disk that is write-protected except for the folders or files that must be unprotected for the normal operation of the CNC. The changes made to protected folders or files will be operative until the unit is turned off and back on, the CNC will then restore the initial configuration. The changes made to unprotected areas of the disk will remain.

The disk has been pre-configured with three access modes, each offers a different protection level. The unit shows the active work mode with an icon on the task bar of the operating system, next to the clock. When the CNC is turned on, the status bar shows the active work mode with icons.

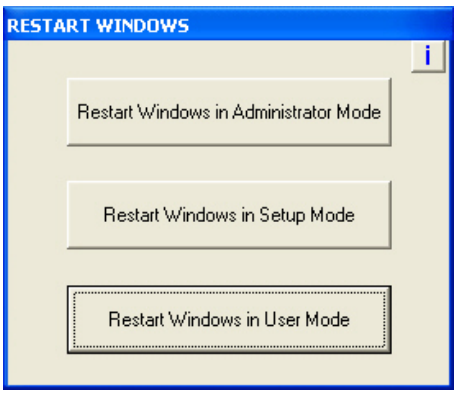
Icon.	Work mode.
	Administrator mode.
	Setup mode.
	User mode.



These work modes will not be available when upgrading the software from a version where the unit has a disk that is not write-protected (not read-only). Since it is not write-protected, the changes made to it will remain.

Work mode change (diskMonitor window).

- Having the CNC application turned off, the work mode change is done using the [ALT][D] or with icon indicating the active work mode, on the task bar of the operative system (next to the clock). In either case, the unit shows the DiskMonitor window to switch from one work mode to another. This process implies restarting the unit and if the manufacturer has set it up this way, it will also be necessary to enter the corresponding password.
- Having the CNC started up in any work mode, the [ALT][D] hotkey shows the DiskMonitor window to change from one work mode to another. This process implies restarting the unit and if the manufacturer has set it up this way, it will also be necessary to enter the corresponding password.
- The desired work mode may be selected from the utilities mode, from the softkey menu. This process implies restarting the unit and if the manufacturer has set it up this way, it will also be necessary to enter the corresponding password.



Protecting or unprotecting the folders.

In the previous window, when selecting the -i- at the top right hand side, the unit shows the list of folders and files that are unprotected at the time. In administrator mode and the list of folders being visible, pressing [CTRL]+[ALT]+[TAB]+[SHIFT] it is possible to protect or unprotect any folder or file of the system that is not key for the proper operation of the CNC.

Administrator mode.

The access to the administrator mode is enabled with the validation code ("Open system" software option). If you don't have this software option, (i.e. you have a "closed system") you will not be able to access the administrator mode and, therefore, you will not be able to install third-party software.



This mode must only be used to install non-Fagor software, to install the CNC (also possible from the setup mode), to update the operative system or change the system configuration. The CNC application does not start up in this mode.



The unit shows the following image on the desk, with red background, indicating the active work mode and warning that it is not a protected mode.

Protection level.

There is not protection level in administrator mode, the whole disk is unprotected.

Protection password.

The access to this mode is protected with the password "administrator mode", defined in the utilities mode. When starting the unit up in this work mode, it will request the access password.

Setup mode.



The setup mode must only be used to update the CNC software and to set up the machine; it does not allow installing third-party software. This mode may be used to access the operative system.



The unit shows the following image on the desk, with yellow background, indicating the active work mode and warning that it is not a protected mode.

Protection level.

The setup mode has an intermediate protection level where everything that may be changed while setting the machine up is unprotected; folders ..\MTB, ..\USERS, ..\DIAGNOSIS and the Windows register.

Protection password.

The access to this mode is protected with the password "machine parameters", defined in the utilities mode. When starting the unit up in this work mode, it will request the access password.

User mode.



It is the usual work mode for the user, once the setup is completed. This mode does not allow updating the CNC or accessing the operative system. Some utilities of the operative system (task manager, clock) will be available from the diagnosis mode.

Part-programs must be saved in the "..\USERS" folder; the CNC considers the files saved in other folders as temporary files and will be deleted when the CNC is turned off. Files from flash, pendrives, ethernet, etc. can only be managed from the explorer of the utilities mode.

Protection level.

The user mode has the maximum protection level where only the folders and files that may be changed during the normal operation of the machine are unprotected.

Protection password.

The access to this mode is not protected with the password.

Restrictions for working in Setup mode and switching to User mode.

The purpose of the following restrictions is for the OEM to deliver his machines with the CNC in protected work mode (User mode).

SETUP mode

On power-up, the CNC will show a message indicating that it is in an unprotected mode and that the setup has not been completed yet. In this situation, the CNC is no longer under Fagor warranty. After a certain period of time, with the next reset, the CNC shows the message again.

When closing the application, the CNC asks if the setup is completed..

- If YES is selected and there are passwords, the CNC makes a backup copy and switches over to USER mode (process OK).
- If YES is selected and there are no passwords, the CNC issues a warning message and does not close the application.

USER mode (work mode).

On power-up, the CNC checks for passwords and a backup copy. If any of these two is missing, the CNC will show a message indicating that setup has not been completed yet. In this situation, the CNC is no longer under Fagor warranty. The CNC checks this at every reset.

This situation can be reached when accessing the USER mode from SETUP mode through "DiskMonitor".

1.2 Software installation at the PC (simulator).

The CNC may be installed in the specific hardware that will later be mounted onto the machine or at a table-top PC that will be used as a simulator for training purposes. The CNC installed at a PC offers all the features and functions but it can only be used in simulator mode and cannot be connected to any type of machine.

The CNC installation CD contains all that is necessary to install the software and the documentation needed to install, set up and operate the CNC. The CNC software must be installed in the hard disk of the PC; it cannot be executed directly from the CD. The installation will start automatically when inserting the CD in the CD drive; if not, double-click on the setup70_Vxx_xx.exe file, where Vxx_xx indicates the version to be installed. Then, follow the instructions displayed on the screen.

Both at a CNC and at a PC (simulator), the necessary files for the CNC are located in the following folder. See *"1.6 Software configuration."* on page 38.

8060 CNC C:\FAGORCNC and its corresponding subfolders.

8065 CNC C:\CNC70 and its corresponding subfolders.

Minimum PC requirements.

In order for the CNC to run properly, its hardware must meet certain requirements.

- Windows® XP operating system.
- Internet Explorer 5.5 or newer.
- Pentium III microprocessor at 800 MHz.
- 512 Mb of RAM memory.
- 6x CD-ROM unit.
- Minimum screen resolution 800x600.

Installing the licenses.

Local license or network license.

During the installation, you will have to choose the type of license; local license (LOCAL) or network license (NET). The type of license indicates where the dongle (hardware key) is connected. For the dongle (hardware key) of he parallel port, always choose the local license.

- Choose the local license if the dongle (hardware key) will be connected to the PC where the CNC software has been installed.
- Choose the net license if the dongle (hardware key) will be connected to a server. When the CNC simulator needs to be authenticated, it will automatically look for the server throughout the net. The CNC software needs not be installed at the server; however, the software that enables the dongle (hardware key) must be installed at the server. This software is included on the CD.

About the dongle (hardware key).

Fagor supplies two types of dongle (hardware key) to be connected either to the parallel port or to the USB port. Once the software has been installed, in order to use the CNC, the dongle (hardware key) supplied with the CD must be connected to the PC. If the net license was chosen when installing the software, the dongle (hardware key) may be connected to a server.

CNC setting; the MTB folder.

Both at a CNC and at a PC (simulator), the OEM has the folder ..\MTB to save the modifications done at the CNC; such as the PLC program, the machine parameters, etc. The CNC manages the MTB folders as follows.

- No MTB folder appears when installing the software for the first time. On system startup and depending on the validation code, the system will move the corresponding folder from ..\CONFIGURATION to the CNC8070 and will rename it MTB.
- When changing the validation code, the system will return the MTB folder to the ..\CONFIGURATION folder and will give it its old name (MTB_M, MTB_T, etc); then, it

will move the corresponding folder from ..\CONFIGURATION to CNC8070 and will rename it MTB.

- If the OEM has created the MTB folder manually, for example by copying it from a backup, the system will not make any changes on power-up nor when modifying the validation code.

Changing the language of the help files.

The help files are installed in English. The CD that comes with the product contains the help files in different languages. You can change the help files installed by default with the ones provided on the CD.

Locate the folder *Help files* inside the CD, select one of the available languages and copy all the files to its CNC location. The help files installed at the CNC (or at the PC, if it is a simulator) are located in the following folder.

8060 CNC C:\FagorCnc\Fagor\MMC\Help

8065 CNC C:\Cnc8070\Fagor\MMC\Help

The help files can only be in one language at a time at the CNC. The language of the help files may be different from the one chosen for the interface.

1.3 Updating the software version.

The updates must be carried out using the software supplied by Fagor Automation. Updating the software maintains the set up of the machine parameters, PLC program, tool table and tool magazine data. Before updating the software, check the list of possible incompatibilities between versions.

Before updating the sofware.

It is recommended to always have a backup copy of the full configuration (ASCII files) such as machine parameter tables, tool tables, active-tools table and tool magazine tables as well as the PLC program. Should any anomaly occur during the installation, these file will help restore the CNC configuration.

Software update.

To update the software, close all the programs that may be running, including the CNC. The software must be installed in setup mode.

The installation will start automatically when inserting the CD in the CD drive; if not, double-click on the setup70_Vxx_xx.exe file, where Vxx_xx indicates the version to be installed. Then, follow the instructions displayed on the screen.

When starting the process, the CNC offers the possibility of installing from scratch; in this case, the installation process will not keep the CNC configuration like the machine parameters, PLC program, etc.

Updating remote nodes of the CAN bus.

Every time the CNC is powered up, it verifies the versions of the remote nodes detected in the CAN bus and automatically updates all these devices if necessary. When done loading, it goes on with the usual start-up process.

If the loading is not successful, and, consequently, the software coherence between all the elements of the CAN bus cannot be guaranteed, the CNC will display the corresponding error message every time [RESET] is pressed.

1.3.1 Software update since previous versions.

Updating from a version older than V4.25.

Unit change in machine parameter FTIMELIM, related to HSC machining.

In earlier versions, parameter FTIMELIM was programmed in milliseconds. From this version on, this parameter is set in percentage.

1.4 Requirements before and after CNC setup.

The CNC setup is carried out in setup mode. Fagor delivers the unit ready to start up in this mode. When powering the unit up, the desktop will show an image showing this circumstance.

After starting the CNC application, proceed with setup. When starting the application, it will display a warning indicating that it is operating in an unsafe mode.

When closing the application with the key sequence [ALT]+[F4] and choosing the options to "close" or to "close and make a backup copy", the CNC shows a second dialog box asking whether setup has been completed or not. If the "NO" is chosen, the next time the unit is turned on, it will start in setup mode again. If the "Yes" option is chosen, the CNC makes a backup copy of the MTB folder and, the next time it is turned on, the unit will start in user mode.



The unit must be delivered to the user ready to start up in user mode.

The unit will remain in setup mode until the manufacturer changes it to user mode in response to the dialog displayed when shutting the application down. If while in user mode, it is necessary to change something that has to do with setup, the setup mode must be accessed manually. See *"1.1.1 Work modes and software protection at the CNC."* on page 29.

When starting the application up, either in setup or user mode, the CNC display can show warnings whose cause must be solved by the manufacturer during setup.

1.5 Installation of third-party software (CNC 8065 only).

The CNC you have purchased is an industrial PC equipped with Windows XP Embedded operating system that permits the installation of third-party applications. The installation of this software must meet the following requirements.

Any malfunction of the equipment due to the installation of third-party software frees Fagor Automation from any responsibility.

Installation in the hard disk.

The CNC that you purchased may have either just a compact flash or a compact flash and a hard disk. To install any software, the CNC must have enough free memory in the compact flash. To avoid ending up with little free memory in the compact flash, it may be necessary to install third-party software in a hard disk.

Also, in order to install third-party software, you must have an open system; i.e. with access to the administrator. The access to the administrator mode is enabled with the validation code. If you don't have this software option, you will not be able to access the administrator mode and, therefore, you will not be able to install third-party software. Software installation done in setup mode or user mode will disappear when turning the CNC off.

The access to the administrator mode is protected with the password "administrator mode", defined in the utilities mode. To obtain the relevant password, contact the supplier of your machine.

When installing third-party software, remember that the results of the software, generated programs, etc. must be saved in an unprotected folder, for example CNC8070\USERS. Never use a protected foder because it will be temporary and it will disappear when turning the CNC off.

System requirements.

Before installing the software, make sure that the CNC meets all the requirements of the software, both the CPU and the memory. You can consult this information in the CNC's diagnosis mode.

Also check that there is enough free memory space in the disk for your application.

Remember that the equipment, the operating system and the CNC (depending on the configuration of the machine) may consume between 50% and 60% of the available resources. Once the software has been installed, the CNC being up and running, check the status of the system resources and check that the CNC is running properly, screen refresh, etc.

1.6 Software configuration.

The necessary files for the CNC are located in the following folder.

8060 CNC C:\FAGORCNC and its corresponding subfolders.

8065 CNC C:\CNC70 and its corresponding subfolders.

Folder.	Contents.
BACKUPSDIR	**Backup of previous installations.** Every time the user installed a full new version, the installer saves a copy of the previous installation in this folder.
CONFIGURATION	**MTB folders of the system.** This folder contains the various MTB folders of the system; MTB_T for lathe and MTB_M for mill. When booting system and depending on the validation code, the system will move the corresponding folder from CONFIGURATION to MTB.
DIAGNOSIS	**Information for the diagnosis.** This folder contains relevant information for proper error diagnosis, including the reportfagor.zip file.
DRIVERS	**CNC drivers.** This folder contains some necessary drives for proper operation of the CNC.
FAGOR	**Version folder.** This folder contains the software corresponding to the CNC version installed. Software updates are carried out in this directory and they do not affect the contents of directories MTB and USERS. Do not change the contents of this directory. Only authorized personnel from Fagor Automation may modify the contents of this directory. Fagor Automation shall not be held responsible of the performance of this CNC if the contents of this directory have been changed.
MTB	**OEM folder.** This folder is especially directed at machine manufacturers. This folder contains the modifications made by the OEM at the CNC like, for example, the PLC program, machine parameters, new screens, integrating external applications, etc.
TMP	**Temporary files.** The CNC uses this folder to save the temporary files it generates while operating. The CNC deletes the contents of this folder on power-up.
USERS	**User folder.** This folder is especially directed at users. The purpose of this folder is to provide the user with a memory space for storing part-programs, profiles, etc. as they are generated.

The CNC deletes the contents of the TMP folder on power-up. Do not use this folder to save files that you'd like to keep.

1.6.1 MTB (Machine Tool Builder) folder.

This folder is especially directed at machine manufacturers. This folder contains the modifications made by the OEM at the CNC like, for example, the PLC program, machine parameters, new screens, integrating external applications, etc.

Folder.	Contents.
DATA	This folder contains the following files or subfolders. • The databases for machine parameters, tables, etc. and the safety backup (in ASCII) of those tables. • The files related to the data of the editor cycles (dat files). • The data the CNC must maintain after it has been turned off (coordinates, zero offsets, etc.). • Each language of the ..\LANG\ subfolder contains the file cncError.txt that contains the OEM messages and errors in different languages. If an error text is not in the folder of the language active at the CNC, it looks for it in ..\LANG\ENGLISH\cncError.txt; if it is not there, the CNC will issue the relevant error message.
DRIVE	This folder contains the information regarding the DDSSETUP mode.
GRAFDATA	This folder contains the information regarding graphics.
KINEMATIC	This folder contains the information regarding the OEM kinematics.
MMC	This folder contains the CNC custom setting made by the machine builder. • The "...\CONFIG" subfolder contains the configuration files (ini files) and the files that may be modified using the screen customizing tool (Fguim). • The ..\IMAGES subfolder contains the images, icons, videos, etc. that the OEM uses to customize the CNC. • Each language of the ..\LANG subfolder contains the texts used by the scripts. Each language of the ..\LANG subfolder contains the file with the texts for the language active at the CNC. When changing the CNC language, the CNC replaces this file with the one corresponding language.
PLC	This folder contains the information regarding the PLC. • Each language of the ..\LANG\ subfolder contains the messages and errors of the PLC in different languages. • The ..\PROJECT subfolder contains the files that make up the PLC project and the object file. • The ..\WATCH contains the data saved from the monitoring and logic analyzer services.
RELEASE	This folder contains the components (ocx files) used by the OEM to create his own application.
SUB	This folder contains the following files or subfolders. • The OEM subroutines (tool change, home search, etc.). • Each language of the ..\HELP folder contains the help files associated with OEM subroutines and the pcall.txt file that contains the list of OEM subroutines. If these files are not in the folder of the language active at the CNC, the editor will not provide them as help.
TUNING	This folder contains the information regarding the setup assistance.

1.6.2 USERS folder.

This folder is especially directed at users. The purpose of this folder is to provide the user with a memory space for storing part-programs, profiles, etc. as they are generated. We recommend to save these programs in the folders created for this purpose in order to make it easier and faster to find them and make safety backup copies.

Since the disk is write-protected, the programs created by the user must be saved in this folder; the only one that is not protected. Any program saved in a protected folder will be temporary and will be deleted when the CNC is turned off.

Folder.	Contents.
HELP	Each language of the ..\HELP folder contains the help files associated with user-defined global subroutines and the pcall.txt file that contains the list of user subroutines.
POCKET	This folder contains the profiles that have been created using the profile editor and are related to the editor cycles.
PRG	This folder contains the part-programs created by the user who may create new subfolders and store the programs in a more orderly fashion. The subfolder "...\PRG_8055_TO_8070" contains the programs translated from the 8055 format into the 8070 language. The CNC saves the converted (translated) program with the same name but with the extension m55 (milling program) or t55 (lathe program).
PROFILE	This folder contains the profiles that have been created using the profile editor.
REPORTS	This folder contains the bmp files that the CNC generates when printing a graphic to a file and the pm reports generated from the diagnosis mode.
SUB	This folder contains the subroutines created by the user, whose location must be fixed for the CNC (for example, the generic user subroutines G500-G599) and the subroutine program_start)

MACHINE PARAMETERS.

2

This chapter describes the meaning of all the machine parameters of the CNC. Some parameters depend on the CNC model (8060/8065) and others are common to both. The CNC shows the right parameters depending on the model and the active software options.

In order for the machine tool to be able to properly execute the programmed instructions and interpret the elements connected to it, the CNC must know the specific machine data. Some of this data may be the number of axes, feedrates, accelerations, feedback, type of tool magazine, tool changer, etc. This data is set by the machine manufacturer and must be communicated to the CNC through the machine parameters. The machine parameters of the CNC are shown grouped as follows.

Machine parameters.	Meaning.
General parameters.	These parameters set the name of the axes and spindles, power-up conditions, subroutines associated with specific functions, etc. Some of these parameters must be defined first because they configure the axis parameter tables. For example, the number and name of the axes and spindles, etc.
Axis and spindle parameters.	These parameters indicate the properties of the axes and spindles, travel limits, moving conditions, related handwheels, probing, compensations, etc. Each axis or spindle may have up to 4 work sets. The following must be set for each one: feedrates and gains, home searches, accelerations, etc.
Jog parameters.	These parameters set the handwheels and the JOG keys.
Parameters of the M function table.	These parameters set the new M functions, the type of synchronization and the subroutine associated with each M function of the system.
Parameters for the Kinematics table.	These parameters indicate the type and characteristics of each kinematics.
Parameters for the magazines.	These parameters indicate the number of tool magazines and number of tool pockets (positions), etc.
HMI parameters.	These parameters are used to define the communication environment (interface) between the operator and the CNC.
OEM parameters.	These parameters set the reading/writing of drive variables, editing cams, defining a group of generic parameters so the manufacturer can use them like machine parameters, etc.

Icons associated with machine parameters.

Parameter validating method.



This icon may appear next to the parameter name and indicates that the CNC must be restarted for the parameter to assume its new value. The parameters that do not have this icon next to them can assume the new value by just pressing the "Validate" softkey.

Parameter matching.



This icon may appear next to the parameter name and indicates that it involved in the parameter matching between the CNC and the drive.

Selection of parameter values.

The icons appearing next to the parameter value permit accessing the list of pre-established values, a data table, a set of parameters or to refer to a file. For those parameters that do not have an icon next to them, a value must be edited within the indicated range.

Icon.	Meaning.
	The parameter has a list of options.
	To access a data table.
	To access a group of parameters.
	The parameter refers to a file.

Default value of the parameters at a simulator.

When the CNC is installed as a simulator at a PC, the default value of certain machine parameters may be different from those mentioned here. The changes basically have to do with the maximum axis feedrates, accelerations and jerk. The probe data has also been changed in order to have, in jog mode, the part centering and tool calibration cycles.

2.1 Parameter matching between the CNC and the Sercos drive.

Some parameters of the CNC and of the drive must be set equivalently to ensure proper system operation. While initializing the Sercos ring, on CNC power-up and when validating the machine parameters of the axes and spindles, the CNC updates at the drives the value of the parameters necessary to ensure that they are equivalent.

The CNC does not match the parameters when the drive is an ACSD.

Parameter matching and speed ranges (gears).

The number of parameter sets to define the speed ranges (gears) may be different at the CNC and at the drive. If there is a parameter set at the CNC, the parameter matching updates parameters NP121, NP122, NP131 and NP133 of each set of the drive with the values defined at the CNC. The parameter matching will save the CNC's default set values in the rest of the sets of the drive.

Understanding the parameter matching table.

The table indicates which CNC parameters are involved in parameter matching as well as the drive parameters that are affected. The different columns of the table have the following meaning.

CNC	List of CNC machine parameters.
DRIVE	List of drive parameters that are equivalent to each CNC parameter.
Sercos	It indicates writing the parameter at the drive depends or not on the type Sercos configuration, position or velocity).
Feedback	It indicates whether or not writing the parameter at the drive depends on the type of axis feedback, internal or external.

CNC	DRIVE	Sercos	Feedback	Remarks
AXISTYPE AXISMODE	PP76			PP76=65; Linear axis. PP76=66; Rotary axis without module. PP76=194; Rotary axis with module.
PITCH	NP123			
INPUTREV	NP121.x			It affects all the gears.
OUTPUTREV	NP122.x			It affects all the gears.
NPULSES2	PP115 (bit 0) NP117		External. External.	B0=0; If rotary encoder (NPULSES2<>0). B0=1; If linear encoder (NPULSES2==0). NP117; If rotary encoder (NPULSES2<>0).
PITCH2	NP133 NP117 NP118		External. External. External.	NP133; If rotary encoder (NPULSES2<>0). NP117; If linear encoder (NPULSES2==0). NP118; If linear encoder (NPULSES2==0).
INPUTREV2	NP131.x		External.	It affects all the gears. Only if rotary encoder (NPULSES2<>0).
OUTPUTREV2	NP132.x		External.	It affects all the gears. Only if rotary encoder (NPULSES2<>0).
FBACKSRC SINMAGNI	GP10			GP10=0; External feedback is not being used. GP10=1; TTL signal (SINMAGNI==0). GP10=2; Vpp signal (SINMAGNI<>0).
FBACKDIFF	PP5	Position.	External.	
FBMIXTIME	PP16	Position.	External.	
AXISCH LOOPCH	PP115 (bit 3)	Position.	External.	B3=0; Feedback sign AXISCH==LOOPCH. B3=1; Feedback sign AXISCH<>LOOPCH.
AXISCH	PP55 (bit 0,2,3)	Position.		B1=0 B2=0 B3=0; It does not change the sign of the feedback reading (AXISCH==NO) B1=1 B2=1 B3=1; It does not change the sign of the feedback reading (AXISCH==YES)

CNC	DRIVE	Sercos	Feedback	Remarks
I0TYPE	PP115 (bit 1,5)		External.	B1=0 B5=0; If distance-coded I0. B1=1 B5=0; if increasing distance-coded I0. B1=1 B5=1; if decreasing distance-coded I0.
I0CODDI1 I0CODDI2	NP166 NP165			Only when using distance-coded I0's. Only when using distance-coded I0's.
REFDIREC DECINPUT FBACKSRC	PP147 (bit 0) PP147 (bit 5) PP147 (bit 3) PP147 (bit 1)			B0=0; Positive home search direction. B0=1; Negative home search direction. B5=0; A home switch is being used. B5=1; A home switch is not being used. B3=0; Internal feedback. B3=1; External feedback. B1=0; The DECEL signal of the CNC is always positive.
REFFEED1	PP41			
REFFEED2	PP1			
REFVALUE	PP52 PP54	Position. Position.	Internal. External.	
REFSHIFT	PP150 PP151	Position. Position.	Internal. External.	
ABSOFF	PP177 PP178		Internal. External.	Only when using distance-coded I0's or absolute feedback.
PROGAIN	PP104			
BACKLASH	PP58	Position.		
BACKANOUT	PP2	Position.		
BACKTIME	PP3	Position.		Only if BACKANOUT<>0
REVEHYST	PP15			
PEAKDISP	PP14			
NPARSETS	GP6			
Activate software limits.	PP55 (bit 4)			B4=1; Check the limits. B4=0; Ignore the limits (for spindles, rotary axes with module and when both parameters LIMIT+ and LIMIT- are set to 0).
MODLIMUP	PP103			Only if it is a spindle or a rotary axis with module.
SZERO	SP42			Only if it is a spindle.
INPOSW	PP57			
MAXFLWE	PP159			Only if following error monitoring is active.
Calculation of the coordinate module.	PP76 (bit 7)			B7=0; On rotary axes or spindles working in velocity-Sercos, the CNC calculates the module of the coordinate.

2.2 Parameters to verify before the startup.

Te ensure the proper performance of the CNC and prevent any damage to the machine, the machine parameters must be properly set, especially the ones related to alarms, travel limits, following errors, feedrate and speed. After the setup, make sure that the parameters related to these aspects are set with operative values.

For safety reasons, verify that these parameters are properly set; wrong values, even if they are within the permitted range could cause the machine to perform poorly.

Here are the parameters that you should at least check.

General machine parameters.

Parameter	Meaning
WARNCOUPE	Gantry axis. Maximum difference allowed between the following errors of both axes before issuing a warning.
MAXCOUPE	Gantry axis. Maximum difference allowed between the following errors of both axes.
MINCORFEED	Minimum feedrate for the HSC at the corners.
MAXFEED	Maximum machining feedrate. This parameter should be set to a value other than ·0·.

Machine parameters for the axes and spindles.

Parameter	Meaning
FBACKDIFF	Maximum difference between feedbacks. The CNC takes this parameter into consideration the axes set as position Sercos and with external feedback or external+internal feedback. This parameter should be set to a value other than ·0·.
DSYNCVELW	Synchronization of axes and spindles. This parameter is defined for the slave element of the synchronization and indicates the velocity margin admitted for the synchronization to be OK.
DSYNCPOSW	Synchronization of axes and spindles. This parameter is defined for the slave element of the synchronization and indicates the position margin admitted for the synchronization to be OK.
LIMIT+ LIMIT-	Axis travel limits. Both parameter should be set to a value other than ·0·.
TENDENCY	Activation of tendency test. It detects axis runaway due to positive feedback. It should be activated during machine setup.
PROBERANGE	Maximum braking distance. This parameter sets the maximum braking distance for the probe after probing to avoid breaking it (ceramic, etc).
PROBEFEED	Maximum probing feedrate.
PROBEDELAY PROBEDELAY2	Delay for the probe 1 and 2 signal. This parameter is only set when there is a short delay from the probing instant to when the CNC actually receives the signals (infrared communication, etc.).
REPOSFEED	Repositioning feedrate after a tool inspection.
POSFEED	Positioning feedrate of the independent axis.
JOGFEED	JOG mode. Feedrate in continuous jog mode.
JOGRAPFEED	JOG mode. Rapid feedrate in continuous jog mode.
MAXMANFEED	JOG mode. Maximum feedrate in continuous jog mode.
MAXMANACC	JOG mode. Maximum acceleration in JOG mode.
INCJOGFEED	JOG mode. Feedrate in incremental jog mode.
FBACKAL	Feedback alarm activation. The feedback alarm must be activated.
G00FEED	Feedrate in G00.

Parameter	Meaning
MAXFEED	Maximum machining feedrate. This parameter should be set to a value other than ·0·.
FLWEMONITOR	Type of monitoring of the following error (axis lag). Monitoring must be activated.
MINFLWE	This parameter indicates the maximum amount of following error allowed when the axis is stopped.
MAXFLWE	For standard monitoring of the following error, this parameter indicates the maximum amount of following error allowed when the axis is moving. For "linear" monitoring of the following error, this parameter indicates the value starting at which the following error will be dynamically monitored.
FEDYNFAC	For "linear" monitoring of the following error, this parameter indicates the percentage error, deviation of the real following error value with respect to the theoretical value.

2.3 General machine parameters.

CHANNEL CONFIGURATION.

NCHANNEL
Number of CNC channels.

Possible values: From 1 to 4.
Default value: 1.
Associated variable: (V.)MPG.NCHANNEL

The use of channels is oriented to machines like dual-spindle lathes where each channel has one of the spindles and two axes; machines with feeders, where the machine and the feeder will be different channels; tool magazine loading and unloading system controlled as an axis.

The CNC can have a single execution channel (single channel system) or several (multi-channel system). Each channel is a different work environment that can act upon a part of or the whole CNC system. The difference between a multi-channel system and several independent CNC's is that the channels can not only act independently, but also together; in other words, they can communicate, synchronize with each other and carry out coordinated actions.

The axes and spindles of a channel.

A channel can have a group of axes and spindles that act independently or in parallel with the rest of the channels. It is also possible to configure a channel without assigning axes or spindles to it at first; then, later on, axes and spindles may be added to it or removed from it via part-program or in the MDI/MDA.mode.

Operation of a channel.

In order to be able to move an axis or spindle, it must be assigned to a channel. Each channel can only control its axes and spindles, although via part-program or MDI/MDA it can command movements to axes or spindles of other channels.

Each channel can execute a different program, be in an different work mode and have its own data. The channels can share information through variables and arithmetic parameters and, if necessary, they may be synchronized via part-program.

CONFIGURING THE AXES OF THE SYSTEM

NAXIS
Number of axes governed by the CNC.

Possible values: From 1 to 28.
Default value: 3.
Associated variable: (V.)MPG.NAXIS

This parameter sets the number of axes of the system, whether they are servo-controlled or not. When counting axes, we will ignore the spindles.

Bear in mind that the number of axes does not depend on the number of channels. A channel may have one, several or no axes associated with it.

AXISNAME
List of system axes.

This parameter shows the table to define the names of the axes. Parameter NAXIS sets the number of axes of the system.

AXISNAME	
AXISNAME n	Name of the axes.

AXISNAME n
Name of the axes.
Possible values: X, X1··X9, ·· , C, C1··C9.
Default value: Starting from AXISNAME1; X, Y, Z...
Associated variable: (V.)MPG.AXISNAMEn
Parameter included in the AXISNAME table.

The axis name is defined by 1 or 2 characters. The first character must be one of the letters X - Y - Z - U - V - W - A - B - C. The second character is optional and will be a numerical suffix between 1 and 9. This way, the name of the axes may be within the range X, X1…X9,...C, C1…C9. For example X, X1, Y3, Z9, W, W7, C...

When defining the axes, bear in mind that the order in which they are defined determines their logic number. The first axis of the table will be logic axis -1- and so on. As with the axis name, the logic number permits identifying the axis in PLC variables, marks, etc.

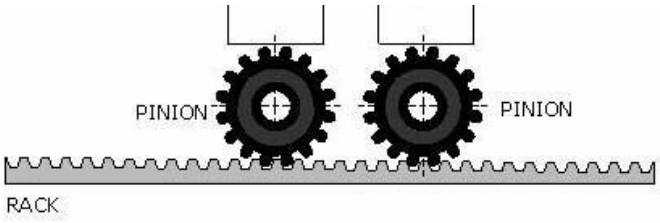
AXISNAME n	Logic order.
AXISNAME 1	Logic number ·1·.
AXISNAME 2	Logic number ·2·.
AXISNAME 3	Logic number ·3·.

CONFIGURATION OF A TANDEM SYSTEM.

TANDEM
Table of tandem axes or spindles.

This parameter shows the table to define the tandem pairs of the system. A tandem axis consists of two electric motors mechanically coupled to each other forming a single transmission system (axis or spindle).

Managing tandem spindles and axes requires a drive version V6.14 or newer.



The following aspects may be pointed out in this configuration:

- A tandem axis helps provide the necessary torque to move a servo drive system when a single motor is not capable of supplying enough torque to do it.
- Applying a pre-load torque between the main motor and the slave motor reduces the backlash.

Requirements of the tandem systems

Each tandem pair (master and slave) must meet the following requirements:

- Each master tandem axis (spindle) admits one single slave tandem axis (spindle).
- A preload may be applied between the two motors.
- Each motor may have a different rated torque.
- The turning direction of each motor may be different from the other's.
- The torque distribution between both motors may be different from 1:1 ratio. For example, on motors whose rated torque is different.
- Both axes and drives must be of the same type (same AXISTYPE and DRIVETYPE parameters for both axes).
- The channel changing permission (parameter AXISEXCH) must be the same in both axes.
- Both axes and drives must have the same software limits (same LIMIT+ and LIMIT- parameters for both axes).

- When the axes are rotary, both axes must be of the same type (same AXISMODE and SHORTESTWAY parameters for both axes).
- The fast and slow home search speeds (parameters REFFEED1 and REFFEED2) must be the same for both axes.

TANDEM n
Configuration of each tandem pair.

This parameter shows the parameter table to define the tandem pairs of the system. Each tandem pair has the following machine parameters to configure it.

TANDEM	
MASTERAXIS	Tandem pair. Master or main axis/spindle.
SLAVEAXIS	Tandem pair. Slave axis/spindle.
TORQDIST	Tandem pair. Torque distribution.
PRELOAD	Tandem pair. Preload between both motors.
PRELFITI	Tandem pair. Filter time to apply the preload.
TPROGAIN	Tandem pair. Proportional gain (Kp) for the tandem.
TINTIME	Tandem pair. Integral gain (Ki) for the tandem.
TCOMPLIM	Tandem pair. Compensation limit.

MASTERAXIS
Tandem pair. Master or main axis/spindle.

Possible values: Any axis defined in the parameter AXISNAME or spindle defined in SPDLNAME.
Associated variable: (V.)MPG.TMASTERAXIS[nb]
Parameter included in the TANDEM table.

This parameter sets which is the master axis or spindle of the tandem pair. The master motor of the tandem, besides generating torque, it is in charge of positioning.

The CNC closes the position loop only with the position of the master axis of the tandem. The velocity command of the master axis/spindle of the tandem is also sent to the slave axis/spindle of the tandem closing the velocity loop. The tandem control changes the velocity command of the master axis/spindle and that of the slave axis/spindle according to the torque distribution and the selected preload.

SLAVEAXIS
Tandem pair. Slave axis/spindle.

Possible values: Any axis defined in the parameter AXISNAME or spindle defined in SPDLNAME.
Associated variable: (V.)MPG.TSLAVEAXIS[nb]
Parameter included in the TANDEM table.

This parameter sets which is the slave axis or spindle of the tandem pair. The slave motor only provides torque. The CNC closes the position loop only with the position of the master axis of the tandem.

TORQDIST
Tandem pair. Torque distribution.

Possible values: from 1 to 99%.
Default value: 50 %.
Associated variable: (V.)MPG.TORQDIST[nb]
Parameter included in the TANDEM table.

This parameter sets the torque percentage supplied by each motor to obtain the total necessary torque in the tandem. The parameter indicates the percentage to be provided by the master motor. The difference between the value of this parameter and 100% is the percentage applied to the slave motor.

If the motors are identical and they're both supposed to output the same torque, this parameter should be set to 50%.

PRELOAD
Tandem pair. Preload between both motors.

Possible values: Within ±100%.
Default value: 0 (it disables the preload).
Associated variable: (V.)MPG.PRELOAD[nb]
Parameter included in the TANDEM table.

The preload is the amount of torque applied before, in both directions, to both motors of the tandem to set a traction between them in order to eliminate the backlash when the tandem is in rest position. The parameter indicates the percentage of the rated torque of the master motor that will be applied as preload.

In order for the two motors to supply opposite torques, the preload value must be greater than the maximum torque needed at all times, including accelerations.



Applying the preload necessarily implies mechanically joining the master and slave axes that make up the tandem; otherwise, the motors will move even without control command.

PRELFITI
Tandem pair. Filter time to apply the preload.

Possible values: From 0 to 65535 milliseconds (setting it to 0 disables the filter).
Default value: 0 ms.
Associated variable: (V.)MPG.PRELFITI[nb]
Parameter included in the TANDEM table.

First order filter that sets the time it takes the CNC to apply the preload gradually. This filter eliminates the torque steps at the input of the tandem compensator when setting a preload parameter, hence avoiding a step in the velocity command of the master motor and slave motor of the tandem.

TPROGAIN
Tandem pair. Proportional gain (Kp) for the tandem.

Possible values: from0 to 100%.
Default value: 0 (no proportional gain is applied).
Associated variable: (V.)MPG.TPROGAIN[nb]
Parameter included in the TANDEM table.

The proportional controller generates an output proportional to the torque error between the two motors.

S	Speed.
S.max	Maximum speed.
Kp	Proportional gain.
T.nom	Rated torque.
T.err	Torque error between motors.
T.mst	Torque of the master motor.
T.slv	Torque of the slave motor.

$$S = Kp \times T.err$$

$$Kp = \left(\frac{S.max}{T.nom}\right) \times TPROGAIN$$

$$T.err = (-T.mst + T.slv + PRELOAD)$$

Example:

A tandem axis has a maximum speed of 2000 rpm and a rated torque of 20 Nm. TPROGAIN has been set to 10%.

Kp = (2000 rpm / 20 Nm) · 0.1= 10 rpm/Nm.

TINTTIME
Tandem pair. Integral gain (Ki) for the tandem.

Possible values: From 0 to 65535 milliseconds.
Default value: 0 (no integral gain is applied).
Associated variable: (V.)MPG.TINTTIME[nb]
Parameter included in the TANDEM table.

The integral controller generates an output proportional to the integral of the torque error between the two motors.

S	Speed.
S.max	Maximum speed.
Kp	Proportional gain.
Ki	Integral gain.
T.nom	Rated torque.
T.err	Torque error between motors.
T.mst	Torque of the master motor.
T.slv	Torque of the slave motor.

$$S = Ki \times \sum T.err$$

$$Ki = \left(\frac{ControlTime}{IntegralTime}\right) \times Kp$$

$$T.err = (-T.mst + T.slv + PRELOAD)$$

TCOMPLIM
Tandem pair. Compensation limit.

Possible values: from0 to 100%.
Default value: 0 (disables the tandem axis),
Associated variable: (V.)MPG.TCOMPLIM[nb]
Parameter included in the TANDEM table.

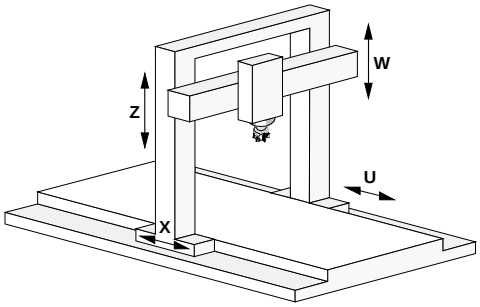
This parameter limits the maximum compensation applied by the tandem. This limit is also applied to the integral.

This parameter refers to the master motor. It is defined as percentage of the maximum speed of the master motor. If programmed with a "0" value, the output of the tandem control will be zero, thus disabling the tandem.

CONFIGURATION OF A GANTRY AXIS..

GANTRY
Gantry-axes table.

This parameter shows the table to define the gantry axes of the system. A gantry axis is a pair of axes that, due to the way the machine is built, must move at the same time and in synchronism. At the CNC, only the movements of one of the axes must be programmed (the master or main gantry axis). The other axis (slave gantry axis) is not programmable; it is controlled by the CNC.



Example of a bridge-type milling machine with two gantry axes, X-U Z-W.

Requirements of the gantry axes.

Each pair of axes (master and slave) must meet the following requirements:

- The master axis must be defined in the AXISNAME table before the slave axis. Both axes must belong to the same channel. The first three axes of the channel cannot be slaves.
- Both axes and drives must be of the same type (same AXISTYPE and DRIVETYPE parameters for both axes).
- The channel changing permission (parameter AXISEXCH) must be the same in both axes.
- Both axes and drives must have the same software limits (same LIMIT+ and LIMIT- parameters for both axes).
- When the axes are rotary, both axes must be of the same type (same AXISMODE and SHORTESTWAY parameters for both axes). Rotary axes that only turn in one direction (parameter UNIDIR = YES) cannot be gantry.
- Hirth axes (HIRTH = YES) cannot be gantry axes.
- The I0 type (parameter I0TYPE) must be the same for both axes either non-distance-coded or distance-coded (increasing or decreasing).
- When the gantry axis does not have distance-coded reference marks (I0), either both axes or just the master axis may have a home switch (parameter DECINPUT). The fast and slow home search speeds (parameters REFFEED1 and REFFEED2) must be the same for both axes.
- If cross compensation is active and the axis to be compensated is gantry, 2 compensation tables must be included, one for the master axis of the gantry and another one for the slave axis.

GANTRY n
Configuration of the gantry axes.

This parameter shows the parameter table to define the gantry axes of the system. Each gantry axis has the following machine parameters to configure it.

GANTRY	
MASTERAXIS	Gantry axis. Master or main axis,
SLAVEAXIS	Gantry axis. Slave axis.
WARNCOUPE	Gantry axis. Difference between the following errors of both axes for issuing a warning.
MAXCOUPE	Gantry axis. Difference between the following errors of both axes for issuing a warning.
DIFFCOMP	Gantry axis. Compensate for the coordinate (position) difference between the two axes after G74.
MAXDIFF	Gantry axis. Maximum position value (coordinate) difference between both axes to compensate for.

MASTERAXIS
Gantry axis Master or main axis,

Possible values: Any axes defined in AXISNAME.
Associated variable: (V.)MPG.MASTERAXIS[nb]
Parameter included in the GANTRY table.

This parameter sets which is the master axis of the gantry pair.

SLAVEAXIS
Gantry axis. Slave axis.

Possible values: Any axes defined in AXISNAME.
Associated variable: (V.)MPG.SLAVEAXIS[nb]
Parameter included in the GANTRY table.

This parameter sets which is the slave axis of the gantry pair.

WARNCOUPE
Gantry axis. Difference between the following errors of both axes for issuing a warning.

Possible values: From 0.0001 to 99999.9999 mm or degrees / from 0 to 3937.00787 inch.
Default value: 1.0000 mm or degrees / 0.03937 inch.
Associated variable: (V.)MPG.WARNCOUPE[nb]
Parameter included in the GANTRY table.

This parameter sets the maximum difference allowed between the following errors of both axes before issuing a warning. This warning lets the user act upon the machine before the CNC issues the error.

The value of parameter WARNCOUPE must be lower than that of parameter MAXCOUPE.

MAXCOUPE
Gantry axis. Maximum difference allowed between the following errors of both axes.

Possible values: From 0.0001 to 99999.9999 mm or degrees / from 0 to 3937.00787 inch.
Default value: 1.0000 mm or degrees / 0.03937 inch.
Associated variable: (V.)MPG.MAXCOUPE[nb]
Parameter included in the GANTRY table.

This parameter indicates the maximum difference allowed between the following errors of both axes. If the difference exceeds this value, the CNC will display an error message.

DIFFCOMP
Gantry axis. Compensate for the coordinate (position) difference between the two axes after G74.

Possible values: Yes / No.
Default value: Yes.
Associated variable: (V.)MPG.DIFFCOMP[nb]
Parameter included in the GANTRY table.

This parameter enables the correction of the position difference between the master and the slave axes after homing. The correction of the position difference is applied using the mark DIFFCOMP(axis).

To compensate the position value, the slave axis will move until reaching the position of the master axis at the feedrate set by parameter REFFEED2. This process can only be interrupted with RESET.

MAXDIFF
Gantry axis. Maximum position value (coordinate) difference between both axes to compensate for.

Possible values: From 0 to 99999.9999 mm or degrees / from 0 to 3937.00787 inch.
Default value: 0.0000 mm, degrees or inches.
Associated variable: (V.)MPG.MAXDIFF[nb]
Parameter included in the GANTRY table.

To avoid correcting coordinate differences that are too big between the master and the slave axes, this parameter sets the maximum coordinate difference allowed to be corrected between both axes. The CNC will only correct the coordinate difference when it is smaller than the value set in this parameter.

CONFIGURATION OF A MULTI-AXIS GROUP.

MULTIAXIS
Table of multi-axis groups.

A multi-axis group may be used to control several axes with a single drive, controlling one or more motors. This parameter shows the multi-axis group definition table for defining the number of groups of the system and the composition of each group.

MULTIAXIS	
MULNGROUPS	Number of multi-axis groups in the system.
GROUP n	Multi-axis groups.

Requirements and limitations of a multi-axis group.

The axes associated with the same drive must meet the following requirements.

- Any Sercos axis of the system may be part of a multi-axis group, except those that belong to gantry or tandem axes.
- Since the axes are controlled by the same drive, all the axes of a group must have the same Sercos address (parameter DRIVEID).
- All the axes of a group must work in the same Sercos operating mode, position or speed, (parameter OPMODEP). If the external feedback (direct feedback) of the axis is connected to a drive, the axis may be operated in either position-Sercos mode or velocity-Sercos mode (velocity-Sercos mode is recommended). If the external feedback (direct feedback) of the axis is connected to local or remote feedback input, the axis can only work in velocity-Sercos mode.
- If the external feedback is local or remote, the CNC homes the axis as if it were analog; the drive does not know that the axis has been homed and does not update its position value, it only receives the velocity command. If the external feedback is connected to another drive, it will not be possible to home the axis.
- The axis being governed by the drive is selected via PLC (mark SWITCH(axis)). The PLC must either park the axes not governed by the drive or set them as DRO axes.
- The CNC keeps monitoring (following error, tendency test, etc.) all the axes sharing the drive even if the axis is not selected.

MULNGROUP
Number of multi-axis groups in the system.

Possible values: From 0 to 16.
Default value: 0.
Associated variable: (V.)MPG.MULNGROUP
Parameter included in the MULTIAXIS table.

This parameter sets the number of multi-axis groups of the system.

GROUP n
Multi-axis groups.

This parameter shows the parameter table to define each multi-axis group. The table shows a GROUP_n parameter for each multi-axis group with the following machine parameters for configuring the groups.

GROUP n	
MULNAXIS	Number of axes and/or spindles making up the multi-axis group.
MULNAXISNAME n	Name of the axes and/or spindles of the group.

MULNAXIS
Number of axes and/or spindles making up the multi-axis group.

Possible values: From 1 to 8.
Default value: 2.
Associated variable: (V.)MPG.MULNAXIS[nb]
Parameter included in the MULTIAXIS // GROUP n table.

This parameter sets the number of axes and/or spindles making up the multi-axis group. Since each axis or spindle of the CNC can have several parameter sets, the number of axes of the group is limited by the 8 parameter sets available at the drive. The parameter sets of the drive are distributed among all the axes and/or spindles that make up the multi-axis group, so a group may be formed by 8 axes with 1 parameter set for each, 1 spindle with 4 sets plus 2 axes with 2 sets each, etc.

MULAXISNAME n
Name of the axes and/or spindles of the group.

Possible values: Any Sercos axis of the system, except tandem or gantry axes
Associated variable: (V.)MPG.MULAXISNAMExn[nb]
Parameter included in the MULTIAXIS // GROUP n table.

Any Sercos axis of the system may be part of a group, except those that belong to gantry or tandem axes. The first axis of the group will be the master axis, the one the drive will use by default; the order of the rest of the axes is irrelevant. On system power-up, the CNC will use the parameters of the master axis to run the parameter matching with the drive.

CONFIGURING THE SPINDLES OF THE SYSTEM

NSPDL
Number of spindles governed by the CNC.

Possible values: From 0 to 4.
Default value: 1.
Associated variable: (V.)MPG.NSPDL

This parameter sets the number of spindles of the system, whether they are servo-controlled or not.

Bear in mind that the number of spindles does not depend on the number of channels. A channel may have one, several or no spindles associated with it.

SPDLNAME
List of spindles of the system.

This parameter shows the table to define the names of the spindles. Parameter NSPDL sets the number of spindles of the system.

SPDLNAME	
SPDLNAME n	Name of the spindles.

SPDLNAME n
Name of the spindles.

Possible values: S, S1··S9.
Default value: Starting from SPDLNAME1; S, S1...
Associated variable: (V.)MPG.SPDLNAMEn
Parameter included in the SPDLNAME table.

The spindle name is defined by 1 or 2 characters. The first character must be the letter -S- . The second character is optional and will be a numerical suffix between 1 and 9. This way, the name of the spindles may be within the range S, S1 ... S9.

When defining the spindles, bear in mind that the order in which they are defined determines their logic number. The logic numbering of the spindles continues from the last logic axis; hence, in a 3-axis system, the first spindle of the table will be logic spindle ·4· and so on.

AXISNAME	SPDLNAME	Logic order.
AXISNAME 1		Logic number ·1·.
AXISNAME 2		Logic number ·2·.
AXISNAME 3		Logic number ·3·.
	SPDLNAME 1	Logic number ·4·.

TIME SETTING (SYSTEM).

LOOPTIME
CNC cycle (loop) time.

Possible values: From 0 to 20 ms.
Default value: 4 ms.
Associated variable: (V.)MPG.LOOPTIME

This parameter sets the CNC's loop time. Loop time greatly depends on the number of inputs, outputs and analog axes of the bus. The shorter the loop time, the less time will the CPU have to process the data.

Influence of the analog axes on parameter LOOPTIME.

Use the following orientative values:

4 ms.	Up to 8 analog axes.
5 ms.	Up to 12 analog axes.
6 ms.	Up to 16 analog axes.
8 ms.	Up to 20 analog axes.
10 ms.	Up to 24 analog axes.

Influence of the CAN bus on parameter LOOPTIME.

From the CAN bus point of view, the minimum loop time for a particular configuration mainly depends on the traffic requirements of the system; in other words, by the time consumed by the transmission of messages per cycle. The factors affect it are the following:

- Total number of remote feedbacks enabled.
- Total number of analog inputs, analog outputs and keyboard handwheels.
- Total number and changing frequency of digital inputs and outputs.
- Frequency of the PLC's PRG module (PRGFREQ).
- CAN transmission speed.

For an exact measurement of the minimum loop time required for a particular configuration, the most reliable thing to do is to use an off-the-shelf tool for a logic analysis of the CAN bus. This tool may be used to set parameter LOOPTIME with a value that covers the communicating requirements reliably, but without saturating the bandwidth of the system (we recommend not to take more than 60% of the bus) to prevent possible transient peaks of traffic or asynchronous eventualities.

If the value of parameter LOOPTIME is too low, the bus will behave as follows.

CANfagor	If the CNC's loop time is too short to manage the desired configuration, it will issue several errors indicating cycle overlapping and timeout in cyclic CAN readings.
CANopen	On CNC startup, it runs an estimate of the amount of time required and it will issue a warning if it is too short.

In either case, it would be necessary to increase the value of parameter LOOPTIME or decrease the communication time of the system (increasing the CAN speed, increasing the value of parameter PRGFREQ or removing physical elements).

PRGFREQ
Frequency of the PLC's PRG module.

Possible values: from 1 to 100 cycles.
Default value: 2 cycles.
Associated variable: (V.)MPG.PRGFREQ

This parameter indicates how often (every how many CNC cycles) a full cycle of the PLC program is executed. This parameter also sets the refreshing frequency of the digital inputs and outputs as well as analog inputs.

Thus, with a sampling period LOOPTIME = 4 ms of and a frequency of PRGFREQ = 2, the PLC program will be executed every 4 x 2 = 8 ms.

SERCOS BUS CONFIGURATION

SERBRATE
Sercos transmission rate.

Possible values: 2/4/8/16 Mbps (megabits per second).
Default value: 4 Mbps.
Associated variable: (V.)MPG.SERBRATE

This parameter indicates the Sercos transmission speed used when communicating with the drives and RCS-S modules (Sercos counter). All elements of the Sercos ring must have the same value.

Speeds of 8 Mbps and 16 Mbps require a Sercos board that can work at these speeds; otherwise the speed will be limited to 2 Mbps and 4 Mbps.

Sercos communication at 8 and 16 MHz requires a drive version V6.05 or higher.

SERPOWSE
Sercos optical power.

Possible values: From 1 to 6 ("Sercos I" board) / from 1 to 8 ("Sercos II" board).
Default value: 4 ("Sercos I" board) / 2 ("Sercos II" board).
Associated variable: (V.)MPG.SERPOWSE

This parameter sets the Sercos power or the intensity of the light going through the optic fiber. The optical power value depends on the total length of the cable being used. Set this parameter with the same value used by the drives.

Use the following approximate values. Assigning other values, e.g. a value of 6 for a length of 3 m, can cause communication errors due to fiber optic signal distortion.

Recommended values ("Sercos I" board).

SERPOWSE	Cable length.
2	Less than 7 meters.
4	From 7 to 15 meters.
6	More than 15 meters.

Recommended values ("Sercos II" board).

SERPOWSE	Cable length.	Recommended cable type.
1 to 4	Less than 15 meters.	SFO / SFO-FLEX
5 to 6	From 15 to 30 meters.	SFO-FLEX
7	From 30 to 40 meters.	SFO-FLEX
8	More than 40 meters.	SFO-V-FLEX

MECHATROLINK BUS CONFIGURATION.

MLINK
Mechatrolink mode.
Possible values: No / Mlink-I / Mlink-II.
Default value: No.
Associated variable: (V.)MPG.MLINK

This parameter may be used to set the Mechatrolink bus in either Mlink-I mode or Mlink-II mode. All the elements of the bus must be operating in the same mode (Mlink-I or Mlink-II); therefore, the selection will be conditioned by the mode available at the slaves connected to the bus.

Limitations of a system with a Mechatrolink bus.

- The CNC does not run any parameter matching process with Mechatrolink drives. The parameters of the drive must be set with the assistance of external tools provided by the drive manufacturer, through the interface of the device itself or writing the parameters from the DDSSetup and Tuning modes of the CNC.
- Second feedback cannot be used; therefore, the feedbacks cannot be combined (parameter FBMIXTIME) or swapped (parameter FBACKSRC). No configuration similar to semi-Sercos is possible either.
- Tandem axes or spindles are not possible.
- Axes or spindles cannot be parked.
- Bode is not available.
- Set changing is not possible.
- The coordinate cannot be initialized.
- The dynamic changing of feed forward, AC forward and Kv will depend on the devices connected to the bus.

DATASIZE
Telegram size in Mlink-II mode.
Possible values: 17 / 32 bytes.
Default value: 32 bytes.
Associated variable: (V.)MPG.DATASIZE

This parameter sets the telegram size for the Mlink-II mode. All the elements of the bus must be operating with the same telegram size; therefore, the selection will be conditioned by the properties of the slaves connected to the bus.

CAN BUS CONFIGURATION.

CANMODE
Type of CAN bus.
Possible values: CANfagor / CANopen.
Default value: CANfagor.
Associated variable: (V.)MPG.CANMODE

CANfagor bus.

Using the CANfagor type bus requires the definition of the maximum length of the bus with parameter CANLENGTH.

CANopen bus.

When using the CANopen type bus, the work speed is defined in each node. All the nodes must work at the same speed. The work speeds at the bus depend on the total length of the bus.

Length (m)	20	40	100	500
Speed (KHz)	1000	800	500	250

CANLENGTH
CANfagor bus cable length.

Possible values: Up to 20, 30, 40, 50, 60, 70, 80, 90, 100, 110, 120, 130 and more than 130 meters.
Default value: Up to 20 meters.
Associated variable: (V.)MPG.CANLENGTH

The work speeds at the bus depend on the total length of the bus.

Length (m)												
20	30	40	50	60	70	80	90	100	110	120	130	>130
Speed (KHz)												
1000	888	800	727	666	615	571	533	500	480	430	400	250

SERIAL LINE CONFIGURATION.

RSTYPE
Type of serial line.

Possible values: RS232 / RS485 / RS422.
Default value: RS232.
Associated variable: (V.)MPG.RSTYPE

Type of serial line. For the HBLS (portable operator panel) select RS422.

Standard configuration for RS232 and full-duplex configuration for RS422.

In the RS485, the CNC uses the same signal to control "send data" and "receive data", therefore sending is disabled while receiving. In order to ensure proper communication, an 8-ms delay is required from when the CNC stops sending till it is ready to receive data. At the slaves connected to the CNC, this delay time must be set from when data is received till it is sent. An external RS232/RS485 adapter should be used if this delay cannot be set at the slave.

DEFAULT CONDITIONS (SYTEM).

INCHES
Default work units (millimeters, inch).

Possible values: Millimeters / inches.
Default value: Millimeters.
Associated variable: (V.)MPG.INCHES

This parameter indicates the work units assumed by the CNC by default; i.e. on power-up, after executing an M02 or M30 or after a reset. To change the units from the part-program, use function G70 or G71.

ARITHMETIC PARAMETERS.

MAXLOCP
Maximum local arithmetic parameter.

Possible values: From 0 to 99.
Default value: 25.
Associated variable: (V.)MPG.MAXLOCP

See general machine parameter MINLOCP.

MINLOCP
Minimum local arithmetic parameter.
Possible values: From 0 to 99.
Default value: 0.
Associated variable: (V.)MPG.MINLOCP

Parameters MINLOCP and MAXLOCP define the group of local arithmetic parameters to be used. Local parameters may only be accessed from the program or subroutine where they have been programmed. There are seven groups of local parameters in each channel.

MAXGLBP
Maximum global arithmetic parameter.
Possible values: From 100 to 9999.
Default value: 299.
Associated variable: (V.)MPG.MAXGLBP

Check general machine parameter MINGLBP.

MINGLBP
Minimum global arithmetic parameter.
Possible values: From 100 to 9999.
Default value: 100.
Associated variable: (V.)MPG.MINGLBP

Parameters MAXGLBP and MINGLBP define the group of global arithmetic parameters to be used. Global parameters may be accessed from any program or subroutine called upon from the channel. There is a group of global parameters in each channel. The value of these parameters is shared by the program and the subroutines.

ROPARMAX
Maximum global read-only arithmetic parameter.
Possible values: From 100 to 9999.
Default value: 0.
Associated variable: (V.)MPG.ROPARMAX

See general machine parameter ROPARMIN.

ROPARMIN
Minimum global read-only arithmetic parameter.
Possible values: From 100 to 9999.
Default value: 0.
Associated variable: (V.)MPG.ROPARMIN

Parameters ROPARMAX and ROPARMIN may be used to protect a group of global arithmetic parameters so they cannot be modified. If both parameters are set to "0", there will be no protected parameters.

MAXCOMP
Maximum arithmetic parameter common to all the channels.
Possible values: From 10000 to 19999.
Default value: 10025.
Associated variable: (V.)MPG.MAXCOMP

Check general machine parameter MINCOMP.

MINCOMP
Minimum arithmetic parameter common to all the channels.
Possible values: From 10000 to 19999.
Default value: 10000.
Associated variable: (V.)MPG.MINCOMP

Parameters MAXCOMP and MINCOMP the group of local arithmetic parameters common to all the channels to be used. The common parameters may be accessed from any channel. The value of these parameters is shared by all the channels.

BKUPCUP
Number of common non-volatile arithmetic parameters.

Possible values: From 0 to 20.
Default value: 0.
Associated variable: (V.)MPG.BKUPCUP

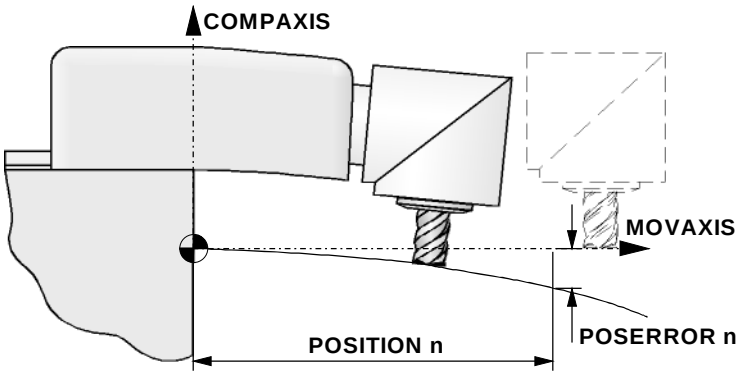
This parameter indicates the number of common arithmetic parameters that are saved in RAM memory with a battery. The CNC saves the first parameters, from the first one to the one set in this parameter.

Non-volatile data means that their values must be maintained from one session to another and after turning the CNC off. The CNC saves this data when the CNC is turned off, when power is removed, when a hardware error occurs, etc.

CROSS COMPENSATION TABLE.

CROSSCOMP
Cross compensation tables.

This parameter shows the cross compensation table. Cross compensation is used when an axis suffers position variations due to the movement of another axis.



Define the tables in the order used when measuring; otherwise, the result will be different. The CNC calculates the compensation to be applied to each axis considering the order in which the tables are defined.

Although by default the CNC always activates the cross compensation tables defined in it, these tables may be disabled via PLC using the DISCROSS mark. The PLC has a mark for each table, the DISCROSS1 mark is for table 1, DISCROSS2 for table 2 and so on.

Import cross compensation tables.

In order to save time and eliminate transcription errors, instead of entering the data manually, it is possible to adapt the format of a text file that contains the result of the measurement and import it later on.

See the operating manual for further information on how to import cross compensation tables.

CROSSCOMP n
Configuration of cross compensation tables.

The CNC shows the possible tables according to the number of axes defined. Each table has the following machine parameters to configure it.

CROSSCOMP	
MOVAXIS	Axis whose movement affects another axis (master).
COMPAXIS	Axis suffering the effects of the movement (compensated).
NPCROSS	Number of points in the table.
TYPCROSS	Type of compensation.

CROSSCOMP	
BIDIR	Bi-directional compensation.
REFNEED	Home search required for compensation.
DATA	Table to define the compensation values.

MOVAXIS
Axis whose movement affects another axis (master).

Possible values: Any axes defined in AXISNAME.
Associated variable: (V.)MPG.MOVAXIS[tbl]
Parameter included in the CROSSCOMP table.

This parameter sets which is the master axis in the cross compensation.

COMPAXIS
Axis suffering the effects of the movement (compensated).

Possible values: Any axes defined in AXISNAME.
Associated variable: (V.)MPG.COMPAXIS[tbl]
Parameter included in the CROSSCOMP table.

This parameter sets which is the compensated axis in the cross compensation.

NPCROSS
Number of points in the table.

Possible values: From 0 to 1000.
Default value: 0 (there is no table).
Associated variable: (V.)MPG.NPCROSS[tbl]
Parameter included in the CROSSCOMP table.

Each cross compensation table can have up to 1000 points.

TYPCROSS
Type of compensation (type of coordinates).

Possible values: Real / Theoretical.
Default value: Real.
Associated variable: (V.)MPG.TYPCROSS[tbl]
Parameter included in the CROSSCOMP table.

This parameter determines whether the cross compensation will be applied on to theoretical or real coordinates.

BIDIR
Bi-directional compensation.

Possible values: Yes / No.
Default value: No.
Associated variable: (V.)MPG.BIDIR[tbl]
Parameter included in the CROSSCOMP table.

This parameter indicates whether the compensation is bidirectional or not; i.e. if the compensation is different in each direction. If the compensation is not bidirectional, it applies the same compensation in both directions.

REFNEED
Home search required for compensation.

Possible values: Yes / No.
Default value: No.
Associated variable: (V.)MPG.REFNEED[tbl]
Parameter included in the CROSSCOMP table.

This parameter indicates whether both axes must be homed or not before applying the compensation.

DATA
Table to define the compensation values.

This parameter shows the list of compensation points and values. The CNC only lets access this table when parameters are MOVAXIS, COMPAXIS and NPCROSS set.

The DATA table must indicate the amount of error to be compensated in specific positions of the moving axis. The number of points is set by parameter NPCROSS. Parameters POSITION, POSERROR and NEGERROR must be set for each point. Parameter NEGERROR only if the table has been defined with bidirectional compensation (BIDIR = YES).

CROSSCOMP DATA	
POSITION	Position of the master axis.
POSERROR	Amount of error to compensate when moving in the positive direction.
NEGERROR	Amount of error to compensate when moving in the negative direction.

POSITION
Position of the master axis.

Possible values: Within ±99999.9999 mm or degrees / within ±3937.00787 inch.
Default value: 0.
Associated variable: (V.)MPG.POSITION[tbl][pt]
Parameter included in the CROSSCOMP // DATA table.

This parameter sets the position of the master axis in each point of the compensation. The points of the table must be ordered by their position on the axis and the table must begin by the most negative point (or least positive) to be compensated. For axis positioning outside this area, the CNC will apply the compensation that was defined for the nearest end.

POSERROR
Amount of error to compensate when moving in the positive direction.

Possible values: Within ±99999.9999 mm or degrees / within ±3937.00787 inch.
Default value: 0.
Associated variable: (V.)MPG.POSERROR[tbl][pt]
Parameter included in the CROSSCOMP // DATA table.

This parameter indicates the amount of error on the compensated axis when the master axis moves in the positive direction. If the table has not been defined with bi-directional compensation, this amount of error will also be applied on movements in the negative direction.

The machine reference point must have "0" error.

NEGERROR
Amount of error to compensate when moving in the negative direction.

Possible values: Within ±99999.9999 mm or degrees / within ±3937.00787 inch.
Default value: 0.
Associated variable: (V.)MPG.NEGERROR[tbl][pt]
Parameter included in the CROSSCOMP // DATA table.

This parameter indicates the amount of error on the compensated axis when the master axis moves in the negative direction. This compensation is only available if the table has been defined with bidirectional compensation.

The machine reference point must have "0" error.

VOLUMETRIC COMPENSATION TABLES.

VOLCOMP
Volumetric compensations tables.

This parameter shows the volumetric compensation table.

VOLCOMP n
Configuration of volumetric compensation tables.

This parameter shows the parameter table to define the volumetric compensations of the system. Each table has the following machine parameters to configure it.

VOLCOMP	
VCOMPAXIS1 VCOMPAXIS2 VCOMPAXIS3	Axes to be compensated.
VCOMPFILE	File containing volumetric compensation data.

VCOMPAXIS1
VCOMPAXIS2
VCOMPAXIS3
Axis to be compensated.

Possible values: Any axes defined in AXISNAME.
Associated variable: (V.)MPG.VCOMPAXIS1[tbl] / (V.)MPG.VCOMPAXIS3[tbl]
Parameter included in the VOLCOMP table.

These parameters set the axes to be compensated with volumetric compensation. The axes associated with the same compensation may belong to different channels and they may be interchanged from one channel to another while the compensation is active. One axis may be included in several compensations and these compensations may be active at the same time.

VCOMPFILE
File containing volumetric compensation data.

Associated variable: (V.)MPG.VCOMPFILE[tbl]
Parameter included in the VOLCOMP table.

File containing the configuration tables for volumetric compensation. The tables cannot be edited at the CNC and will be supplied by the company in charge of setting up the volumetric compensation.

EXECUTION TIMES.

MINAENDW
Minimum duration of the AUXEND signal.

Possible values: From 0 to 65535 ms.
Default value: 10 ms.
Associated variable: (V.)MPG.MINAENDW

This parameter has the following meanings.

- This parameter sets the time that the AUXEND signal must stay active for the CNC to consider it a valid signal. AUXEND is the synchronization signal that the PLC sends to the CNC to indicate that the –M–, –S–, –T– functions have been executed.
- For –M– functions (which do not need synchronization), this parameter indicates the duration of the MSTROBE signal.
- For –H– functions (which do not need synchronization), this parameter indicates the duration of the HSTROBE signal.

The value assigned to this parameter must be equal to or greater than the PLC's input frequency (LOOPTIME x PRGFREQ).

REFTIME
Estimated home searching time.
Possible values: From 0 to 1000000 ms.
Default value: 0. ms.
Associated variable: (V.)MPG.REFTIME

See general machine parameter TTIME.

HTIME
Estimated time for an H function.
Possible values: From 0 to 1000000 ms.
Default value: 0. ms.
Associated variable: (V.)MPG.HTIME

See general machine parameter TTIME.

DTIME
Estimated time for a D function.
Possible values: From 0 to 1000000 ms.
Default value: 0. ms.
Associated variable: (V.)MPG.DTIME

See general machine parameter TTIME.

TTIME
Estimated time for a T function.
Possible values: From 0 to 1000000 ms.
Default value: 0. ms.
Associated variable: (V.)MPG.TTIME

In Editing - Simulation mode, there is an option that allows calculating the time required to execute a part with the machining conditions established in the program. To fine tune that calculation, one may define these parameters that indicate the estimated time for processing particular functions.

The values are generic, for any H, D, T function or for homing one or several axes at a time. Spindle machine parameter SPDLTIME indicates the estimated time for executing an S function and the machine parameter of the M function table –M– MTIME indicates the time estimated to execute an M function.

NUMBERING OF THE DIGITAL INPUTS AND OUTPUTS.

NDIMOD
Total number of digital input modules.
Possible values: From 0 to 64.
Default value: 0 (no numbering is defined).
Associated variable: (V.)MPG.NDIMOD

This parameter indicates the number of these modules connected to the same CAN bus. In remote modules with CANopen protocol each double module of digital inputs and outputs counts as two.

After defining this value, it is possible to set the numbering of the digital inputs of each module. If this parameter is not set, the CNC numbers the digital inputs sequentially according to the order of the modules in the bus.

DIMODADDR
Table of digital input modules.

This parameter shows the list of digital input modules connected to the same bus CAN. When inserting a new module, the CNC will assign the numbering of the table to the first modules and the next valid base index after the highest one assigned until then to the last module.

DIMOD n
Base index of the digital input modules.

Possible values: From 0 to 1009.
Default value: The first valid value.
Associated variable: (V.)MPG.DIMODADDR[nb]
Parameter included in the DIMODADDR table.

This parameter sets the base index from which the digital inputs of the module are numbered.

Remote modules (for bus CAN with CANfagor protocol).

The values of the base index must be comply with the formula "16n + 1" (i.e. 1, 17, 33, etc.). If an invalid base index is entered, it assumes the nearest previous valid one. The base indexes may follow any order, they do not have to be sequential.

Remote modules (for bus CAN with CANopen protocol).

The values of the base index must be comply with the formula "8n + 1" (i.e. 1, 9, 17, 25, etc.). If an invalid base index is entered, it assumes the nearest previous valid one. The base indexes may follow any order, they do not have to be sequential.

The double input/output module must be assigned two base indexes for the inputs and two base indexes for the outputs; one for each board;

NDOMOD
Total number of digital output modules.

Possible values: From 0 to 64.
Default value: 0 (no numbering is defined).
Associated variable: (V.)MPG.NDOMOD

This parameter indicates the number of these modules connected to the same CAN bus. In remote modules with CANopen protocol each double module of digital inputs and outputs counts as two.

After defining this value, it is possible to set the numbering of the digital outputs of each module. If this parameter is not set, the CNC numbers the digital outputs sequentially according to the order of the modules in the bus.

DOMODADDR
Table of digital output modules.

This parameter shows the list of digital output modules connected to the same bus CAN. When inserting a new module, the CNC will assign the numbering of the table to the first modules and the next valid base index after the highest one assigned until then to the last module.

DOMOD n
Base index of the digital output modules.

Possible values: From 0 to 1009.
Default value: The first valid value.
Associated variable: (V.)MPG.DOMODADDR[nb]
Parameter included in the DOMODADDR table.

This parameter sets the base index from which the digital outputs of the module are numbered.

Remote modules (for bus CAN with CANfagor protocol).

The values of the base index must be comply with the formula "16n + 1" (i.e. 1, 17, 33, etc.). If an invalid base index is entered, it assumes the nearest previous valid one. The base indexes may follow any order, they do not have to be sequential.

Remote modules (for bus CAN with CANopen protocol).

The values of the base index must be comply with the formula "8n + 1" (i.e. 1, 9, 17, 25, etc.). If an invalid base index is entered, it assumes the nearest previous valid one. The base indexes may follow any order, they do not have to be sequential.

The double input/output module must be assigned two base indexes for the inputs and two base indexes for the outputs; one for each board;

NUMBERING OF ANALOG INPUTS FOR TEMPERATURE SENSORS PT100.

NPT100
Total number of active PT100 inputs.

Possible values: From 0 to 10.
Default value: 0 (there are no active PT100 inputs).
Associated variable: (V.)MPG.NPT100

This parameter indicates the number of PT100 inputs active in the CAN bus. A PT100 input is active if it has one of these temperature sensors connected to it.

PT100
Table of active PT100 inputs.

This parameter shows the list of PT100 inputs active in the CAN bus.

PT100 n
Analog input associated with the PT100 input.

Possible values: From 0 to 32.
Default value: 0 (Do not activate the PT100 input).
Associated variable: (V.)MPG.PT100[nb]

When numbering the analog inputs, the CNC considers the PT100 inputs as analog inputs. Therefore, when numbering the inputs, the CNC considers that each module has 6 analog inputs; the four analog inputs plus the two PT100 inputs.

	Group ·1· (address = 1)	**Group ·2· (address = 2)**
Analog inputs.	1 ·· 4	7 ·· 10
PT100 inputs.	5 ·· 6	11 ·· 12

This parameter indicates the number of the analog input associated with each active PT100 input. Each parameter can take any valid value; there is no need to follow a particular order.

For example, in a system with two modules of analog inputs and 3 active PT100 inputs (the two of the first module and the first one of the second module), the PT100 parameters may be are set in one of the following ways.

	NPT100	**PT100 1**	**PT100 2**	**PT100 3**
Option 1	3	5	6	11
Option 2	3	5	11	6
Option 3	3	6	5	11
Option 4	3	6	11	5
Option 5	3	11	5	6
Option 6	3	11	6	5

PROBE SETTING.

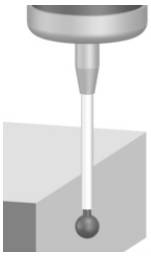
PROBE
There is a probe.

Possible values: Yes / No.
Default value: No.
Associated variable: (V.)MPG.PROBE

This parameter indicates whether there is a probe on the machine or not. The CNC may have configured two probes, it will usually be a tabletop probe to calibrate tools and a touch probe to measure the part.



Tabletop probe.



Measuring probe.

PROBEDATA
Probe parameters.

This parameter shows the table of parameters needed to set up the probe. If the CNC uses a tabletop probe, besides these parameters it is necessary to define the probe position.

PROBEDATA	
PROBETYPE1	Probe type 1.
PROBETYPE2	Probe type 2.
PRBDI1	Number of the input associated with probe 1.
PRBDI2	Number of the input associated with probe 2.
PRBPULSE1	Logic level to activate probe 1.
PRBPULSE2	Logic level to activate probe 2.

PROBETYPE1
Probe type 1.

Possible values: Remote / Local.
Default value: Remote.
Associated variable: (V.)MPG.PROBETYPE1
Parameter included in the PROBEDATA table.

This parameter indicates the type of probe depending on where it is connected; a remote probe if it is connected to a digital input at the remote modules or to a local probe if it is connected to a probe input at the central unit.

PROBETYPE2
Probe type 2.

Possible values: Remote / Local.
Default value: Remote.
Associated variable: (V.)MPG.PROBETYPE2
Parameter included in the PROBEDATA table.

This parameter indicates the type of probe depending on where it is connected; a remote probe if it is connected to a digital input at the remote modules or to a local probe if it is connected to a probe input at the central unit.

PRBDI1
Number of the input associated with probe 1.

Possible values: From 1 to 1024 for the remote probe / 1 or 2 for the local probe.
Default value: 0 (no probe is connected).
Associated variable: (V.)MPG.PRBDI1
Parameter included in the PROBEDATA table.

This parameter indicates the number of the input that the probe is connected to. For remote probes, this parameter indicates the digital input number; for local probes, it indicates the local probe input number.

The digital input will be a physical input of the module or a logic input of the PLC. The PLC considers logic inputs the inputs that do not exist physically; for example, if inputs 1 through 256 are numbered at the remote modules, the PLC considers logic inputs the ones from 257 to 1024.

Two remote probes cannot be connected to the same module if they are going to be involved in simultaneous probing operations; for example, a probe in each channel and both making a probing movement.

PRBDI2
Number of the input associated with probe 2.

Possible values: From 1 to 1024 for the remote probe / 1 or 2 for the local probe.
Default value: 0 (no probe is connected).
Associated variable: (V.)MPG.PRBDI2
Parameter included in the PROBEDATA table.

This parameter indicates the number of the input that the probe is connected to. For remote probes, this parameter indicates the digital input number; for local probes, it indicates the local probe input number.

The digital input will be a physical input of the module or a logic input of the PLC. The PLC considers logic inputs the inputs that do not exist physically; for example, if inputs 1 through 256 are numbered at the remote modules, the PLC considers logic inputs the ones from 257 to 1024.

Two remote probes cannot be connected to the same module if they are going to be involved in simultaneous probing operations; for example, a probe in each channel and both making a probing movement.

PRBPULSE1
Logic level to activate probe 1.

Possible values: Positive / Negative.
Default value: Positive.
Associated variable: (V.)MPG.PRBPULSE1
Parameter included in the PROBEDATA table.

This parameter indicates whether the probe functions are active high (24 V or 5 V) or active low (0V) of the signal provided by the probe. In any case, the probe signal must be at least 20 ms long for the CNC to consider it valid.

PRBPULSE2
Logic level to activate probe 2.

Possible values: Positive / Negative.
Default value: Positive.
Associated variable: (V.)MPG.PRBPULSE2
Parameter included in the PROBEDATA table.

This parameter indicates whether the probe functions are active high (24 V or 5 V) or active low (0V) of the signal provided by the probe. In any case, the probe signal must be at least 20 ms long for the CNC to consider it valid.

SHARED PLC MEMORY.

PLCDATASIZE
Size of the PLC's shared data area.

Possible values: From 0 to 500.000 bytes.
Default value: 0.
Associated variable: (V.)MPG.PLCDATASIZE

This parameter may be used to define a memory area to exchange data between a PLC program written in C language and an external application.

MANAGEMENT OF LOCAL I/O.

NLOCOUT
Number of local digital outputs.

Possible values: From 0 to 8.
Default value: 0.
Associated variable: (V.)MPG.NLOCOUT

The central unit has set of 8 local digital signals, identified as LI/O1 through LI/O8, that may be configured both as input and output. This parameter indicates how many of these signals, starting with LI/O1, are configured as digital outputs; the rest of the signals will act as digital inputs. The following table shows the logict numbers of pins LI/O1-LI/O8 according to parameter NLOCOUT.

Pin.	NLOCOUT								
	8	7	6	5	4	3	2	1	0
LI/O8	O8	O7	O6	O5	O4	O3	O2	O1	I16
LI/O7	O7	O6	O5	O4	O3	O2	O1	I15	I15
LI/O6	O6	O5	O4	O3	O2	O1	I14	I14	I14
LI/O5	O5	O4	O3	O2	O1	I13	I13	I13	I13
LI/O4	O4	O3	O2	O1	I12	I12	I12	I12	I12
LI/O3	O3	O2	O1	I11	I11	I11	I11	I11	I11
LI/O2	O2	O1	I10	I10	I10	I10	I10	I10	I10
LI/O1	O1+	I9	I9	I9	I9	I9	I9	I9	I9

For the PWM and synchronized switching, only local outputs associated with pins LI/O1 and LI/O2 may be used because they have been set for Laser applications. That requires setting parameter NLOCOUT = 8.

EXPSCHK
Activate the 24 V monitoring at the local digital outputs.

Possible values: ON / OFF.
Default value: ON.
Associated variable: (V.)MPG.EXPSCHK

When a local digital output is active, the connector must be supplied with 24 V DC. This parameter enables the detection of these 24 V at the connector.

The 24 V monitoring must be active when a local output has been set; if no local digital output is active, the 24 V monitoring must be deactivated.

SYNCHRONIZED SWITCHING.

SWTOUTPUT
Local digital output associated with synchronized switching.

Possible values: From 0 to NLOCOUT (Number of local digital outputs).
Default value: 0.
Associated variable: (V.)MPG.SWTOUTPUT

Local digital output associated with synchronized switching. For synchronized switching, only local outputs associated with pins LI/O1 (local output 1) and LI/O2 (local output 2) may be used because they have been set for Laser applications. That requires setting parameter NLOCOUT = 8.

Synchronized switching refers to the process to control a local digital output of the CNC depending on the type of movement programmed on the axes. The selected digital output is activated when switching from G0 to G1/G2/G3. The selected digital output is canceld when switching from G1/G2/G3 to G0.

In order for this feature to work better, the parameters of the axes involved in the movements must be set with the same values (gain, filters, acceleration and deceleration, feed forward, etc.) and must also work with the least amount of following error (lag) as possible.

SWTDELAY
Delay of the device associated with synchronized switching.

Possible values: From 0 to 100 ms.
Default value: 0.
Associated variable: (V.)MPG.SWTDELAY

This parameter sets the delay between digital output on/off and the device that is connected to it.

PWM (PULSE-WIDTH MODULATION).

PWMOUTPUT
Local digital output associated with the PWM.

Possible values: From 0 to 2.
Default value: 0.
Associated variable: (V.)MPG.PWMOUTPUT

Local digital output associated with the PWM. For the PWM, only local outputs associated with pins LI/O1 (local output 1) and LI/O2 (local output 2) may be used because they have been set for Laser applications. That requires setting parameter NLOCOUT = 8.

PWMOUTPUT	Meaning.
0	PWM missing.
1	Local output 1 (pin LI/O1).
2	Local output 2 (pin LI/O2).

The output for the PWM and for synchronized switching may be the same one and they may be used at the same time (same value at parameters SWOUTPUT and PWMOUTPUT). However, this configuration is not recommended for certain types of Laser generators.

PWMCANCEL
Cancel the PWM after an M30 or a reset.

Possible values: Yes / No.
Default value: No.
Associated variable: (V.)MPG.PWMCANCEL

This parameter indicates whether the CNC cancels the PWM after executing M02, M30 or after a reset.

When there is a CNC error in the channel that has the PWM active (or in any channel if it is managed via PLC), it the parameter is set to "Yes", the CNC deactivates the PWM. If the parameter is set to "No", the CNC keeps the status of the output; i.e. it does not cancel the

output if it is active. Once the PWM is deactivated for some error, that PWM will not be reactivated when the error disappears; therefore, they have to be reactivated either by program or via PLC.

BACKUP OF NON-VOLATILE DATA.

BKUPREG
Number of non-volatile PLC registers.

Possible values: From 0 to 20.
Default value: 0.
Associated variable: (V.)MPG.BKUPREG

This parameter indicates the number of PLC registers that are saved in RAM memory with a battery. The CNC saves the first registers, from the first one to the one set in this parameter.

Non-volatile data means that their values must be maintained from one session to another and after turning the CNC off. The CNC saves this data when the CNC is turned off, when power is removed, when a hardware error occurs, etc.

BKUPCOUN
Number of non-volatile PLC counters.

Possible values: From 0 to 20.
Default value: 0.
Associated variable: (V.)MPG.BKUPCOUN

This parameter indicates the number of PLC counters that are saved in RAM memory with a battery. The CNC saves the first counters, from the first one to the one set in this parameter.

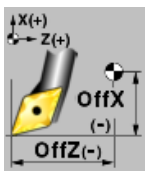
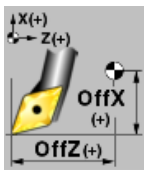
Non-volatile data means that their values must be maintained from one session to another and after turning the CNC off. The CNC saves this data when the CNC is turned off, when power is removed, when a hardware error occurs, etc.

TOOL OFFSET AND WEAR.

TOOLOFSG
Sign criteria for tool offsets and tool wear

Possible values: Positive / Negative.
Default value: Negative.
Associated variable: (V.)MPG.TOOLOFSG

This parameter sets the sign criteria for the offsets and their wear. The offsets are used to define the tool dimensions in each axis. The dimensions of the turning tools are defined using these offsets; either these offsets or tool length and radius may be used for the dimensions of the rest of the tools.

TOOLOFSG	Meaning.
Negative. 	Tool calibration returns a negative offset. The offset wear must be entered with a positive value.
Positive. 	Tool calibration returns a positive offset. The offset wear must be entered with a negative value.

In the tool table, it is possible to define whether the wear value being entered must be incremental or absolute.

SPINDLE SYNCHRONIZATION.

SYNCCANCEL
Cancels the spindle synchronization.

Possible values: Yes / No.
Default value: Yes.
Associated variable: (V.)MPG.SYNCCANCEL

This parameter indicates whether the CNC cancels spindle synchronization or not after executing M02, M30 or after an error or reset.

DEFINE THE NUMBER OF JOG PANELS AND THEIR RELATIONSHIP WITH THE CHANNELS.

NKEYBD
Number of jog panels.

Possible values: From 1 to 8.
Default value: 1.
Associated variable: (V.)MPG.NKEYBD

Number of jog panels.

KEYBDCH
List of jog panels of the system.

This parameter shows the table to assign the jog panels to the channels.

KEYBDCH	
KEYBDnCH	Channel to which the jog panel is assigned.

The CNC numbers the operator panels following the order (sequence) that they occupy in the CAN bus (Address switch). The first jog panel will be the one with the lowest number and so on.

KEYBDnCH
Channel to which the jog panel is assigned.

Possible values: Active channel / CH1 / CH2 / CH3 / CH4.
Default value: Active channel.
Associated variable: (V.)MPG.KEYBDCH[jog]
Parameter included in the KEYBDCH table.

Each jog panel may be assigned to a particular channel or always to the active channel. When a jog panel is assigned to a channel, it is always operative even if the channel is not the active one. When there are several jog panels assigned to the same channel, operations may be carried out from any of them indistinctively.

PLC TYPE.

PLCTYPE
PLC type.
Possible values: IEC / IEC+Fagor / Fagor.
Default value: IEC.
Associated variable: (V.)MPG.PLCTYPE

This parameter selects the type of PLC to be used; the Fagor one, the standard IEC-61131 or both. The following table shows who is affected by the startup (RUN) and stop (STOP) of the PLC depending on this parameter as well as the status of the mark PLCREADY.

PLCTYPE	RUN		STOP		PLCREADY	
	IEC	Fagor	IEC	Fagor	IEC	Fagor
IEC	Yes	No	Yes	No	ON	- - -
IEC+Fagor	Yes	Yes	Yes	Yes	ON	ON
Fagor	No	Yes	No	Yes	OFF	ON

RENAME THE AXES AND THE SPINDLES.

RENAMECANCEL
Cancel the name change for the axes and the spindles.
Possible values: Yes / No.
Default value: Yes.
Associated variable: (V.)MPG.RENAMECANCEL

The #RENAME instruction may be used to rename the axes and spindles via part-program or MDI/MDA mode. This parameter indicates whether the CNC keeps or cancel the names of the axes and spindles (#RENAME) after executing M02 or M30, after a reset or at the beginning of a new part-program in the same channel.

After turning the CNC off and on, the axes and spindles always keep their new name, except after a checksum error or after validating the machine parameters that involve restoring the original configuration of channels, axes or spindles. In either case, the axes and the spindles will recover their original names.

ZERO OFFSETS.

FINORG
Fine definition of zero offsets.
Possible values: Yes / No.
Default value: No.
Associated variable: (V.)MPG.FINEORG

This parameter enables the possibility to set each zero offset in the zero offset table with a coarse (or absolute) value and a fine (incremental) one. When executing function G159, the CNC assumes as new zero offset the sum of both parts.

REMOTE MODULE RCS-S (SERCOS COUNTER).

NSERCOUNT
Number of Sercos counter nodes in the bus.
Possible values: From 0 to 8.
Default value: 0.

Number of RCS-S modules in the bus.

SERCOUNTID n
Table of thumbwheel identifiers associated to the Sercos counters.
Possible values: From 1 to 15.

These parameters may be used to assign a logic number to the RCS-S modules of the bus based on their physical address, set by its rotary switch (node number). Define in each parameter the address (node number) of the RCS-S module.

Although the Sercos ring may have up to 32 nodes, the RCS-S modules always occupy positions from 1 to 15.

ACCESS TO THE PARAMETER TABLE OF THE CHANNEL.

CHANNEL n
Parameter table of channel.

This parameter shows the machine parameter table of the channel.

2.4 General machine parameters. Execution channels.

CHANNEL CONFIGURATION.

GROUPID
Group the channel belongs to.
Possible values: From 0 to 2.
Default value: 0 (it does not belong to any group).
Associated variable: (V.)[ch].MPG.GROUPID

Two or more channels may be configured to form a group. The channels of the same group behave as follows.

- Each channel may be in a different work mode, except in jog and automatic modes. Toggling between the jog mode and automatic mode of a channel will affect all the channels of the group that are in any of these modes; the channels that are in a different mode will not be affected.
- A reset in any of the channels of the group affects all of them.
- Any error in any of the channels of the group interrupts the execution in all of them.

CHTYPE
Type of channel.
Possible values: CNC / PLC / CNC+PLC.
Default value: CNC.
Associated variable: (V.)[ch].MPG.CHTYPE

A channel may be governed from the CNC, from the PLC or from both.

Channels governed by the PLC have neither jog mode nor MDI/MDA mode. The automatic and EDISIMU modes are available, but programs cannot be executed or simulated.

If during setup, it is necessary to display these work modes or execute or simulate programs, set the channel as being governed from the CNC+PLC and once setup is completed, set it back as a PLC channel.

HIDDENCH
Hidden channel.
Possible values: Yes / No.
Default value: No.
Associated variable: (V.)[ch].MPG.HIDDENCH

Hidden channels are not displayed and cannot be selected.

A hidden channel cannot be reset from the operator panel; to reset it, either group it with another channel or reset it from the PLC mark RESETIN.

CONFIGURING THE AXES OF THE CHANNEL.

CHNAXIS
Number of axes of the channel.
Possible values: From 0 to 28.
Default value: 3.
Associated variable: (V.)[ch].MPG.CHNAXIS

This parameter sets the number of axes of the channel, whether they are servo-controlled or not.

A channel may have initially associated with it one, several or no of the axes defined in the system. In any case, the number of axes assigned to the channel cannot be higher than the number of axes of the system, defined by parameter NAXIS. The sum of the axes assigned to the channels cannot exceed the number of axes of the system either.

It is possible to change the configuration of the axes of a channel via part-program, (defining a new configuration, adding or removing axes) using the instructions #SET AX, #FREE AX and #CALL AX.

CHAXISNAME
List of the axes of the channel.

This parameter shows the table to define the names of the axes. Parameter CHNAXIS sets the number of axes of the channel.

CHAXISNAME	
CHAXISNAME n	Name of the axes.

CHAXISNAME n
Name of the axes.

Possible values: Any axes defined in AXISNAME.
Default value: Starting from CHAXISNAME1; X, Y, Z...
Associated variable: (V.)[ch].MPG.CHAXISNAMEn
Parameter included in the CHAXISNAME table.

Any axis defined by parameter AXISNAME may belong to the channel.

When defining the axes, bear in mind that the order in which they are defined determines their index in the channel. The index of the first axis of the table will be ·1· and so on. As with the axis name, the index in the channel permits identifying the axis in PLC variables, marks, etc.

CHAXISNAME n	Index in the channel.
CHAXISNAME 1	Index ·1·.
CHAXISNAME 2	Index ·2·.
CHAXISNAME 3	Index ·3·.

The order of the axes and work planes (mill model).

The order of the axes in the channel sets the main work planes, those selected with functions G17, G18 and G19. Function G20 may be used to form any work plane with the axes of the channel.

Plane.	Abscissa axis.	Ordinate axis.	Longitudinal axis.
G17	CHAXISNAME 1	CHAXISNAME 2	CHAXISNAME 3
G18	CHAXISNAME 3	CHAXISNAME 1	CHAXISNAME 2
G19	CHAXISNAME 2	CHAXISNAME 3	CHAXISNAME 1

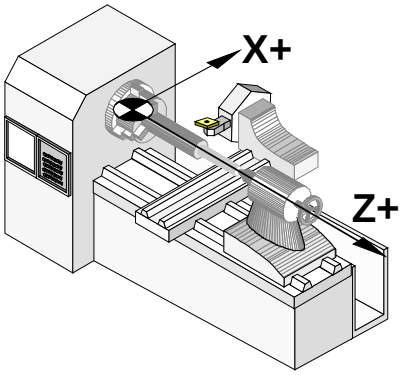
The order of the axes and work planes (lathe model).

The order of the axes of the channel and parameter GEOCONFIG set the main work planes. Refer to the parameter for further detail.

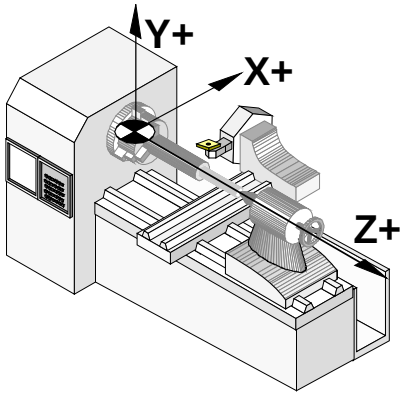
GEOCONFIG
Geometrical configuration of the axes of the channel.

Possible values: Plane / Trihedron.
Default value: Trihedron.
Associated variable: (V.)[ch].MPG.GEOCONFIG

Not being used at the mill model. On the lathe model, this parameter indicates the axis configuration of the machine, trihedron or plane.



Configuration of "plane" type axes.



Configuration of trihedron type axes.

Configuration of "Trihedron" type axes.

This configuration has three axes forming a Cartesian XYZ type trihedron like on a milling machine. There may be more axes, besides those forming the trihedron; that may be part of the thihedron or be auxiliary axes, rotary axes, etc.

With this configuration, the planes behave in the same way as on a milling machine except that the usual work plane will be G18 (if it has been configured like that in parameter IPLANE). The order of the axes in the channel sets the main work planes, those selected with functions G17, G18 and G19. Function G20 may be used to form any work plane with the axes of the channel.

Plane.	Abscissa axis.	Ordinate axis.	Longitudinal axis.
G17	CHAXISNAME 1	CHAXISNAME 2	CHAXISNAME 3
G18	CHAXISNAME 3	CHAXISNAME 1	CHAXISNAME 2
G19	CHAXISNAME 2	CHAXISNAME 3	CHAXISNAME 1

The CNC displays the ·G· functions associated with the work planes.

Configuration of "plane" type axes.

This configuration has two axes forming the usual work plane on a lathe. There may be more axes, but they cannot be part of the trihedron; there must be auxiliary, rotary, etc.

With this configuration, the work plane is always G18 and will be formed by the first two axes defined in the channel. If the X (firs axis of the channel) and Z (second axis of the channel) have been defined, the work plane will be the ZX (Z as abscissa and X as ordinate).

The work plane is always G18; machine parameter IPLANE is not applied and it is not possible to change planes via part-program. The ·G· functions associated with the work planes have the following effects.

Function.	Meaning.
G17	It does not change planes and shows a warning about it.
G18	It has no effect (except when function G20 is active).
G19	It does not change planes and shows a warning about it.
G20	It is permitted if it does not change the main plane; i.e. it can only be used to change the longitudinal axis.

The CNC does not display the ·G· functions associated with the work planes because it is always the same plane.

Configuration of "plane" type axes. Arc programming.

Programming the arc center I K depends on the active work plane.

- With function G18, in circular interpolations, the arc center I is associated with the first axis of the channel (usually X) and K with the second axis of the channel (usually Z).
- With function G20, in circular interpolations, the arc center I is associated with the abscissa axis (usually Z) and K with the ordinate axis of the channel (usually X).

Configuration of "plane" type axes. The longitudinal axis.

In this configuration, the second axis of the channel is considered as longitudinal axis. If the X (first axis of the channel) and Z (second axis of the channel) axes have been defined, the work plane will be the ZX and Z will be the longitudinal axis. Tool length compensation is applied on this longitudinal axis when using milling tools. With lathe tools, tool length compensation is applied on all the axes where a tool offset has been defined.

When using milling tools on a lathe, the longitudinal compensation axis may be changed by means of the #TOOLAX instruction or the G20 function.

Configuration of "plane" type axes. Axis swapping.

The axes may be swapped, but it must be borne in mind that the previous behavior stays the same for the first and second axes of the channel resulting from the swap.

CONFIGURING THE SPINDLES OF THE CHANNEL.

CHNSPDL
Number of spindles of the channel.

Possible values: From 0 to 4.
Default value: 1.
Associated variable: (V.)[ch].MPG.CHNSPDL

This parameter sets the number of spindles of the channel, whether they are servo-controlled or not.

A channel may have initially one, several or no spindles associated with it. In any case, the number of spindles assigned to the channel cannot be higher than the number of spindles of the system, defined by parameter NSPDL. The sum of the spindles assigned to the channels cannot exceed the number of spindles of the system either.

It is possible to change the configuration of the spindles of a channel via part-program, (defining a new configuration, adding or remove spindles) using the instructions #SET SP, #FREE SP and #CALL SP.

CHSPDLNAME
List of the spindles of the channel.

This parameter shows the table to define the names of the spindles. Parameter CHNSPDL sets the number of spindles of the channel.

CHSPDLNAME	
CHSPDLNAME n	Name of the spindles.

CHSPDLNAME n
Name of the spindles.

Possible values: Any spindle defined in SPDLNAME.
Default value: Starting from CHSPDLNAME1; S, S1...
Associated variable: (V.)[ch].MPG.SPDLNAMEn
Parameter included in the CHSPDLNAME table.

Any spindle defined by parameter SPDLNAME may belong to the channel.

On CNC power-up and after a reset, the channel assumes as master spindle the first spindle defined by the machine parameters of the channel (original master). If this spindle is parked or "handed out" to another channel, the channel assumes as master spindle the next spindle defined by the machine parameters and so on.

When defining the spindles, bear in mind that the order in which they are defined determines their index in the channel. The index of the first spindle of the table will be ·1· and so on. As with the spindle name, the index in the channel permits identifying the spindle in PLC variables, marks, etc.

CHSPDLNAME	Index in the channel.
CHSPDLNAME 1	Index ·1·.
CHSPDLNAME 2	Index ·2·.
CHSPDLNAME 3	Index ·3·.

CONFIGURATION OF THE C AXIS.

CAXNAME
Default name of the C axis.

Possible values: Any valid axis name; X, X1··X9, ·· , C, C1··C9.
Default value: C.
Associated variable: (V.)[ch].MPG.CAXNAME

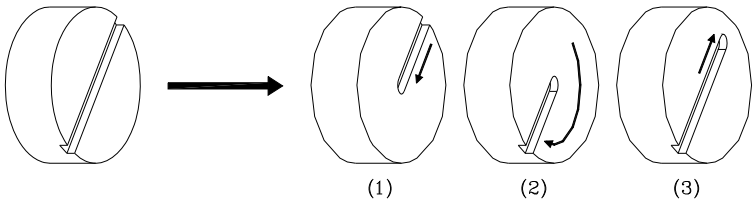
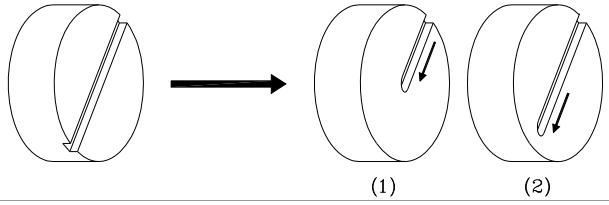
This parameter must be defined whenever there is an axis or spindle set as C axis.

When setting more than one C axis, use the program instruction #CAX to indicate the one that is active. Only one C axis may be active in each channel.

ALIGNC
"C" axis alignment for diametrical machining.

Possible values: Yes / No.
Default value: Yes.
Associated variable: (V.)[ch].MPG.ALIGNC

This parameter indicates whether the tool can machine the whole surface diametrically in a single run (ALIGNC = No) or the "C" axis must be aligned (ALIGNC = Yes).

ALIGNC	Meaning.
Yes.	 (1) Machining up to the center. (2) Rotate the "C" axis 180°. (3) Resume machining by moving the tool back.
No.	

TIME SETTING (CHANNEL).

PREPFREQ
Maximum number of blocks to prepare per cycle.

Possible values: From 1 to 8.
Default value: 1.
Associated variable: (V.)[ch].MPG.PREPFREQ

While executing a program, the CNC reads in advance the blocks to be executed in order to calculate the path to follow; this is known as block preparations. The number of blocks to be prepared depends on the type of machining operation; it is not the same to process blocks

generated with a CAD-CAM and the blocks coming from a program edited manually or with canned cycles.

This parameter sets the maximum number of blocks the CNC must process in each cycle (parameter LOOPTIME). The CNC will try processing the number of blocks set in this parameter and, if not possible, it will only process the number of blocks that it can.

Besides preparing the blocks, the CNC carries out several tasks in each cycle. Increasing the value of parameter PREPFREQ unnecessarily can jeopardize the rest of the task. Before modifying the value of this parameter, check with the service department.

ANTIME
Anticipation time.
Possible values: From 0 to 10000000 ms.
Default value: 0.
Associated variable: (V.)[ch].MPG.ANTIME

This parameter is used on punch presses that have an eccentric cam as a punching system. This parameter indicates how long before the axes reach position, the anticipation logic signal ADVINPOS of the channel is activated .

This signal may be used to start the movement of the punch before the axes reach the position. This reduces idle time, thus increasing the number of punches per minute.

If the total duration of the movement is lower than the value in the parameter, the anticipation signal ADVINPOS will be activated immediately.

If set to zero, the anticipation signal ADVINPOS is always active.

CONFIGURATION OF THE HSC MODE (CHANNEL).

HSC
Parameter table of the HSC mode..

This parameter shows the table to define the HSC work mode.

FEEDAVRG
Calculating feedrate average.
Possible values: Yes / No.
Default value: Yes.
Associated variable: (V.)[ch].MPG.FEEDAVRG

This parameter enables the adjustment of the feedrate according to the block reading speed and their size. This adjustment eliminates the need to slow down due to a poor supply of small blocks; although as a result, the feedrate reached is lower, the overall machining time will improve. The block reading speed depends on machine parameter PREPFREQ.

This parameter is only valid when the acceleration profile is trapezoidal or square sine (parameter SLOPETYPE), that is the default acceleration profile for the HSC CONTERROR mode.

SMOOTHFREQ
Smoothing frequency in the interpolation..
Possible values: From 0 to 500.0000.
Default value: 0 (it is not active).
Associated variable: (V.)[ch].MPG.SMOOTHFREQ

This parameter sets the smoothing frequency in path interpolation. This parameter avoids accelerating and decelerating throughout a path, beyond a particular frequency when generating an averaged speed.

This parameter is only valid when the acceleration profile is trapezoidal or square sine (parameter SLOPETYPE), that is the default acceleration profile for the HSC CONTERROR mode.

CORNER
Maximum angle of the corner to machine it in square corner mode.

Possible values: From 0 to 180.0000º.
Default value: 0.
Associated variable: (V.)[ch].MPG.CORNER

This parameter indicates the maximum angle between two paths, under which the machining operation is carried out in square corner mode.

HSCFILTFREQ
Filter frequency (CONTERROR mode).

Possible values: From 0 to 500.0000.
Default value: 0 (not active).
Associated variable: (V.)[ch].MPG.HSCFILTFREQ

This parameter activates an automatic FIR filter for all the axes of the channel during the execution in CONTERROR mode. This filter may be used to smooth the response of the axes generating a smoother tool path and, if necessary, reduce the feedrate so the path error is adjusted to the programmed path.

This parameter inserts a variable (non constant) phase shift depending on the frequencies. This phase shift can cause a change on the path if it is not executed at the same feedrate as, for example, when changing the feedrate override percentage or when moving back and forth on the same path.

FASTFACTOR
Default feedrate percentage (FAST mode).

Possible values: From 0 to 100.
Default value: 100.
Associated variable: (V.)[ch].MPG.FASTFACTOR

This parameter indicates the feedrate at the corners and indicates the feedrate percentage to be reached, of the maximum that the CNC can reach in FAST mode. This parameter indicates the default value and it may be changed from the part program.

FTIMELIM
Time difference permitted in feedrate interpolation (FAST mode).

Possible values: from0 to 100000.0000 %.
Default value: 200 %.
Associated variable: (V.)[ch].MPG.FTIMELIM

The feedrate interpolation in HSC FAST mode improves the machining quality. Parameter FTIMELIM allows limiting the time it takes the CNC to interpolate the feedrate in each segment. When programmed with a 0 value, the CNC does not interpolate the feedrate and, therefore, the execution is faster. When increasing the value of this parameter, it increases the maximum total execution time allowed; e.g. a value of 200% or 300% will allow an execution time twice or three times as long in the interpolated segment.

In rather large blocks where the feedrate adaptation represents a longer additional time than the one indicated in FTIMELIM, the CNC will adapt the feedrate using the maximum dynamics to avoid unnecessary waste of time.

MINCORFEED
Minimum feedrate at the corners.

Possible values: From 0 to 200000.0000 mm/min / from 0 to 7874.01575 inch/min.
Default value: 0.
Associated variable: (V.)[ch].MPG.MINCORFEED

This value should not be modified because it is possible to exceed the dynamics of the axes.

FSMOOTHFREQ
Smoothing frequency in the interpolation (FAST mode).

Possible values: From 0 to 500.0000.
Default value: 20.
Associated variable: (V.)[ch].MPG.FSMOOTHFREQ

This parameter sets the smoothing frequency in path interpolation for the HSC FAST mode. This parameter avoids accelerating and decelerating throughout a path, beyond a particular frequency when generating an averaged speed. It is only applied when the HSC FAST mode is active.

FASTFILTFREQ
Filter frequency (FAST mode).

Possible values: From 0 to 500.0000.
Default value: 50.
Associated variable: (V.)[ch].MPG.FASTFILTFREQ

This parameter activates an automatic "low pass" filter for all the axes of the channel while executing in HSC FAST mode that permits smoothing the response of the axes by generating a smoother path. This filter has the drawback of rounding the corners slightly.

This parameter inserts a constant phase shift regardless of the frequencies.

FREQRES
First resonance frequency of the machine.

Possible values: From 0 to 500.0000.
Default value: 0.
Associated variable: (V.)[ch].MPG.FREQRES

Resonance frequency that the CNC must eliminate when generating the velocity command.

This parameter is only valid when the acceleration profile is trapezoidal or square sine (parameter SLOPETYPE), that is the default acceleration profile for the HSC CONTERROR mode.

SOFTFREQ
Path filter frequency for profiles of linear acceleration.

Possible values: From 0 to 500.0000 Hz
Default value: 25 Hz.
Associated variable: (V.)[ch].MPG.SOFTFREQ

This parameter may be used to smooth the velocity profile in HSC FAST mode, which improves machining time and surface quality. This parameter must be set using the FineTune application.

HSCROUND
Default value of the maximum path error in HSC.

Possible values: From 0.0001 to 99999.9999 mm / from 0 to 3937.00787 inches
Default value: 0.06 mm / 0.00236 inch.
Associated variable: (V.)[ch].MPG.HSCROUND

Maximum path error in HSC.

SURFFILFREQ
Axis filter frequency (SURFACE mode).

Possible values: From 0 to 500.0000 Hz
Default value: 20 Hz.
Associated variable: (V.)[ch].MPG.SURFFILFREQ

This parameter activates an automatic filter for all the axes of the channel while executing in HSC SURFACE mode that permits smoothing the response of the axes by generating a smoother path.

HSCDEFAULMODE
Default mode when programming #HSC ON.
Possible values: SURFACE / CONTERROR / FAST.
Default value: SURFACE.
Associated variable: (V.)[ch].MPG.HSCDEFAULMODE

Default mode to activate HSC machining.

VIRTUAL TOOL AXIS.

A virtual axis of the tool is a fictitious axis that always moves in the direction in which the tool is oriented. The purpose of the virtual tool is to facilitate the movement in the direction of the tool when it is not aligned with with the axes of the machine. This way and depending on the applied kinematics, it will move the corresponding X Y Z axes so the tool moves along its axis. This function facilitates drilling operations, withdrawing the tool in its direction as well as increasing or decreasing the depth of the pass while machining a part.

VIRTAXISNAME
Virtual tool axis name.
Possible values: Any valid axis name; X, X1··X9, ·· , C, C1··C9.
Default value: none.
Associated variable: (V.)[ch].MPG.VIRTAXISNAME

Name identifying the virtual axis of the tool.

The virtual tool axis must be a linear axis, must belong to the channel and cannot be part of the main trihedron when it is active. Since it is an axis of the channel, it may be moved like any other axis in the various work modes: automatic, jog, tool inspection, axis repositioning, etc. The virtual tool axis has travel limits that may be set by machine parameter and by program.

VIRTAXCANCEL
Automatic cancellation of the virtual tool axis after an M30 and Reset.
Possible values: Yes / No.
Default value: No.
Associated variable: (V.)[ch].MPG.VIRTAXCANCEL

This parameter indicates whether the CNC cancels or not the virtual tool axis after executing an M02/M30 and after a reset.

DEFAULT CONDITIONS (CHANNEL).

The following parameters indicate the conditions assumed by the channel on power-up, after executing an M02 or M30 or after a reset.

KINID
Default kinematics number.
Possible values: From 0 to 6 or nothing.
Default value: Nothing (the CNC does not assume any kinematics on power-up).
Associated variable: (V.)[ch].MPG.KINID

This parameter indicates the kinetics number (not type) active by default. If set to ·0·, on power-up the CNC restores the kinematics that was active when it was turned off. If no value is set, the CNC does not activate any kinematics by default.

There can be up to 6 different kinematics defined at the CNC. To select another kinematics from the part-program, use the #KIN ID instruction.

CSCANCEL
Cancel the inclined plane on start-up.

Possible values: Yes / No.
Default value: Yes.
Associated variable: (V.)[ch].MPG.CSCANCEL

This parameter indicates whether on start-up the CNC cancels the inclined plane (#CS/#ACS) that was active when the CNC was turned off.

LINKCANCEL
Cancel axis coupling by default.

Possible values: Yes / No.
Default value: Yes.
Associated variable: (V.)[ch].MPG.LINKCANCEL

This parameter indicates whether the active axis couplings (#LINK) are canceled or not at the end of the part-program, after an emergency or after a reset.

MIRRORCANCEL
Cancel mirror image (G11/G12/G13/G14) after M30 and reset.

Possible values: Yes / No.
Default value: Yes.
Associated variable: (V.)[ch].MPG.MIRRORCANCEL

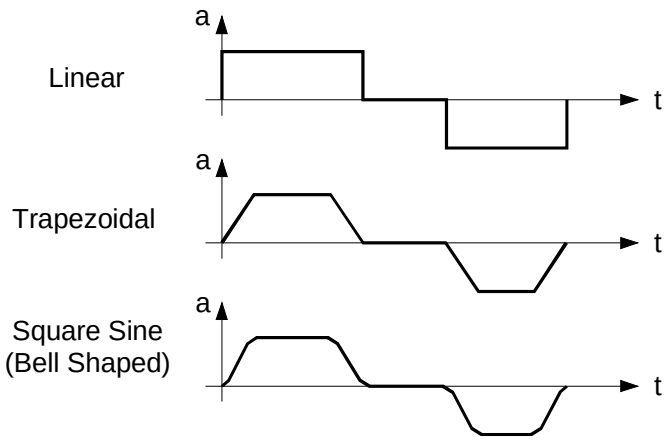
This parameter indicates whether mirror image is cancelled or not at the end of the part program or after a reset.

SLOPETYPE
Default acceleration type.

Possible values: Linear / Trapezoidal / Square sine (bell shaped).
Default value: Square sine (bell shaped).
Associated variable: (V.)[ch].MPG.SLOPETYPE

This parameter indicates the type of acceleration applied by default in automatic movements. There are three types of acceleration, namely: linear, trapezoidal and square sine (bell shaped). It is recommended to use square-sine type acceleration. When working in manual (JOG) mode, the CNC always applies linear acceleration.

The dynamics of trapezoidal and square sine (bell shaped) accelerations are similar. Trapezoidal acceleration may be used to program ramps in order to smooth out the acc/dec changes. The square sine acceleration is an improvement to the trapezoidal acceleration; it smoothes out the jerk so the movements are softer and the axis mechanics suffers less.



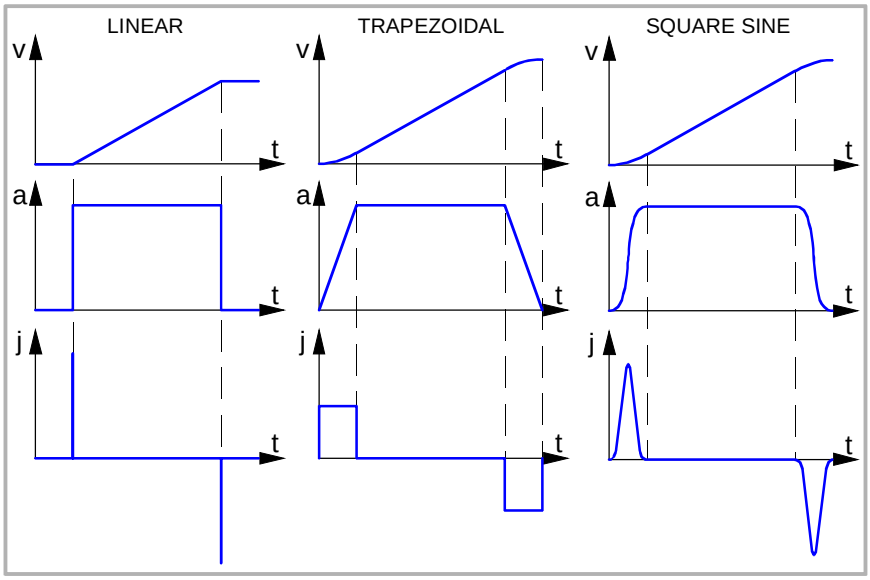
Depending on the type of acceleration selected, the machine parameters will show the ones needed to configure the acceleration. To select a different acceleration via part-program, for the automatic mode, use the instruction #SLOPE.

Description of the types of acceleration.

Square-sine acceleration provides the system's best response. The movements are smoother and the axis mechanics does not suffer as much. Linear acceleration provides the poorest response.

However, the smoother the system's response, the slower the movements. Linear acceleration provides the fastest movements and the square sine the slowest.

The figure below shows the graphs for velocity (v), acceleration graph (a) and jerk (j) for each case. As acceleration represents the velocity change per time unit, the jerk represents the acceleration change per time unit.



IPLANE
Main plane (G17/G18) by default.

Possible values: G17 / G18.
Default value: G17.
Associated variable: (V.)[ch].MPG.IPLANE

This parameter indicates the main work plane assumed by the CNC by default. The axes that form the work plane depend on machine parameter CHAXISNAME.

Plane.	Abscissa axis.	Ordinate axis.
G17	CHASIXNAME1	CHASIXNAME2
G18	CHASIXNAME3	CHASIXNAME1
G19	CHASIXNAME2	CHASIXNAME3

To change the work plane via part-program, use function G17, G18, G19 or G20.

ISYSTEM
Type of coordinates (G90/G91) by default.

Possible values: G90 / G91.
Default value: G90.
Associated variable: (V.)[ch].MPG.ISYSTEM

This parameter indicates the type of coordinates assumed by the CNC by default. The coordinates of a point may be defined either in absolute coordinates (G90) referred to part zero or in incremental coordinates (G91) referred to the current position.

To change the type of coordinates via part-program, use function G90 or G91.

IMOVE
Type of movement (G0/G1) by default..

Possible values: G0 / G1.
Default value: G01.
Associated variable: (V.)[ch].MPG.IMOVE

This parameter indicates the type of movement assumed by the CNC by default. Movements in G0 are carried out in rapid as set by parameter G00FEED. Movements in G1 are carried out at the feedrate active at the CNC.

To change them from the part-program, use function G0 or G1.

IFEED
Type of feedrate (G94/G95) by default.

Possible values: G94 / G95.
Default value: G94.
Associated variable: (V.)[ch].MPG.IFEED

This parameter indicates the type of feedrate assumed by the CNC by default.

- With G94, the feedrate is assumed in mm/min or degrees/min or inches/min.
- With G95, the feedrate is assumed in mm/rev or degrees/rev or inches/rev.

The typical configuration for the mill model will be G94. The typical configuration for the lathe model will be G95.

To change the type of feedrate via part-program, use function G93 or G94.

FPRMAN
Function G95 admitted in jog mode.

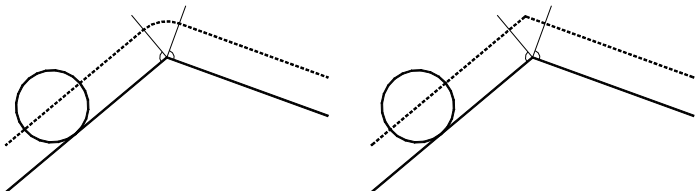
Possible values: Yes / No.
Default value: No.
Associated variable: (V.)[ch].MPG.FPRMAN

This parameter indicates whether function G95 (feedrate in mm/rev or inches/rev) is admitted in jog mode or not.

IRCOMP
Tool radius compensation mode (G136/G137) by default.

Possible values: G136 / G137.
Default value: G136.
Associated variable: (V.)[ch].MPG.IRCOMP

This parameter indicates the compensation mode assumed by the CNC by default. Being the radius compensation active, the compensated paths may be blended together using circular paths (G136) or linear paths (G137).



To change the type of compensation via part-program, use function G136 or G137.

COMPCANCEL
How to cancel tool radius compensation.

Possible values: With motion / Without motion.
Default value: Without motion
Associated variable: (V.)[ch].MPG.COMPCANCEL

This parameter indicates whether tool radius compensation is canceled in the first motion block, even if the plane axes are not involved, or if it requires a movement of the plane axes.

LCOMPTYP
Keep the longitudinal axis when changing planes (G17/G18/G19).

Possible values: Yes / No.
Default value: No.
Associated variable: (V.)[ch].MPG.LCOMPTYPE

This parameter indicates which will be the longitudinal axis when changing planes (G17/G18/G19). In either case, the longitudinal axis may be changed via part-program using

G20 functions or #TOOL AX.

LCOMPTYP	Meaning.
Yes.	When changing work planes (G17/G18/G19), the CNC el CNC keeps the longitudinal axis that was active before the plane change.
No.	When changing the work planes (G17/G18/G19), the CNC assumes the axis perpendicular to the plane as the new longitudinal axis.

ICORNER
Type of corner (G5 / G7 / G50) by default.

Possible values: G50 / G5 / G7.
Default value: G50.
Associated variable: (V.)[ch].MPG.ICORNER

This parameter indicates the type of corner assumed by the CNC by default. There are three types of corners, namely: square (G7), rounded (G5) and semi-rounded (G50). To change the type of corner via part-program, use function G5, G7 or G50.

- When working in square corner, the CNC starts executing the next movement when the axis gets into the in-position zone defined by parameter INPOSW.
- When working in round corner, it is possible to control the corner of the programmed profile.
- When working in semi-rounded corner, the CNC starts executing the next movement once the theoretical interpolation of the current move is completed.

If G5is selected, parameter ROUNTYPE must be set.

ROUNDTYPE
Rounding type in G5 (by default).

Possible values: Chordal error / % Feedrate.
Default value: Chordal error.
Associated variable: (V.)[ch].MPG.ROUNDTYPE

This parameter indicates the type of rounding applied by default when working in round corner. To change the type of rounding via program, use the #ROUNDPAR instruction.

The rounding may be executed by limiting the chordal error or the feedrate. The chordal error (#ROUNDPAR [1]) defines the maximum deviation allowed between the programmed point and the resulting profile. The feedrate (#ROUNDPAR [2]) defines the percentage of the active feedrate to be used for machining. Depending on the option selected, either parameter MAXROUND or ROUNDFEED will have to be set.

MAXROUND
Maximum rounding error in G5.

Possible values: From 0.02 to 99999.9999 mm or degrees / from 0.00079 to 3937.00787 inch.
Default value: 0.1000 mm or degrees / 0.00394 inch.
Associated variable: (V.)[ch].MPG.MAXROUND

This parameter sets the maximum deviation allowed between the programmed point and the profile resulting from rounding the corner. The CNC takes it into account if ROUNDTYPE = Chordal error.

This parameter also sets the maximum contour allowed for the HSC mode when not programming CONTERROR (CONTERROR mode) or E (FAST mode).

ROUNDFEED
Percentage of feedrate in G5.

Possible values: From 0 to 100.
Default value: 100.
Associated variable: (V.)[ch].MPG.ROUNDFEED

This parameter sets the percentage of the active feedrate to be used for machining. The CNC takes it into account if ROUNDTYPE = % Feedrate.

ARC CENTER CORRECTION.

CIRINERR
Maximum absolute radius error permitted.
Possible values: From 0 to 99999.9999 mm or degrees / from 0 to 3937.00787 inch.
Default value: 0.0100 mm or degrees / 0.00039 inch.
Associated variable: (V.)[ch].MPG.CIRINERR

See general machine parameter CIRINFACT.

CIRINFACT
Maximum relative radius error permitted.
Possible values: from0 to 100.0 %.
Default value: 0.1 %.
Associated variable: (V.)[ch].MPG.CIRINFACT

These parameters set the conditions for correcting the center position in circular interpolations. On circular interpolations, the CNC calculates the radius of the starting point and end point of the tool path. Theoretically, they should be the same; but these parameters may be used to set the maximum difference allowed between both radius.

Parameter CIRINERR indicates the maximum absolute error allowed. Parameter CIRINFACT indicates the maximum relative error allowed (% of the radius).

Both parameters are taken into account. The CNC will show the relevant error message when this difference between them is greater than CIRINERR and greater than —CIRINFACT x Radius—..

This function is controlled via program using functions G264 and G265.

BEHAVIOR OF THE FEEDRATE AND THE FEEDRATE OVERRIDE.

MAXOVR
Maximum override (%) permitted.
Possible values: From 0 to 255.
Default value: 200.
Associated variable: (V.)[ch].MPG.MAXOVR

This parameter indicates the maximum percentage to be applied to the programmed axis feedrate (feedrate override).

The percentage applied to the programmed feedrate may be set by program, via PLC or by the switch of the panel. The one set by program has the highest priority and the one set by the switch has the lowest priority.

Different values may be set for each axis via PLC and by program. The one selected at the switch is common to all of them.

RAPIDOVR
Override acts in G00 (from 0 to 100%).
Possible values: Yes / No.
Default value: Yes.
Associated variable: (V.)[ch].MPG.RAPIDOVR

This parameter indicates whether the feedrate % may be modified (between 0% and 100%) or not when working in G0. If not allowed, the percentage will stay fixed at 100%.

Regardless of the value assigned to this parameter, the override always attends to the 0% position and never acts over 100%. It is always possible to change the % of feedrate when moving in jog mode.

FEEDND
Apply the programmed feedrate to all the axes of the channel.

Possible values: Yes / No.
Default value: No.
Associated variable: (V.)[ch].MPG.FEEDND

This parameter indicates whether the programmed feedrate is applied to all the axes of the channel or only to the main axes.

FEEDND	Meaning.
Yes.	The programmed feedrate will be the result of composing the movements onto all the axes of the channel.
No.	If a movement has been programmed on any of the main axes, the programmed feedrate will be the result of composing the movement only onto these axes. The rest of the axes move at their corresponding feedrate to end the movement of them all at the same time. The programmed feedrate is limited only if an axis could exceed its MAXFEED. If none of the main axes are programmed, the programmed feedrate will be reached on the axis moving the farthest so they can all reach their destination at the same time.

OVERRIDE OF THE DYNAMICS FOR HSC.

MINDYNOVR
Minimum override for the dynamics in HSC.

Possible values: From 10 to 100 %.
Default value: 30 %.
Associated variable: (V.)[ch].MPG.MINDYNOVR

The screen of the automatic mode offers a slider for changing the dynamics defined by program for an HSC machining operation. This parameter sets the minimum percentage to be applied using the slider.

MAXDYNOVR
Maximum override for the dynamics in HSC.

Possible values: From 100 to 500 %.
Default value: 200 %.
Associated variable: (V.)[ch].MPG.MAXDYNOVR

The screen of the automatic mode offers a slider for changing the dynamics defined by program for an HSC machining operation. This parameter sets the maximum percentage to be applied using the slider.

STEPDYNOVR
Override step for the dynamics in HSC.

Possible values: from0 to 100 %.
Default value: 10 %.
Associated variable: (V.)[ch].MPG.STEPDYNOVR

The screen of the automatic mode offers a slider for changing the dynamics defined by program for an HSC machining operation. This parameter sets the incremental step to move the cursor of the slider.

MOVEMENT OF THE INDEPENDENT AXES.

IMOVEMACH
Movement of the independent axis referred to machine coordinates

Possible values: Yes / No.
Default value: No.
Associated variable: (V.)[ch].MPG.IMOVEMACH

This parameter indicates whether the movements of the independent axes are referred to machine coordinates (IMOVEMACH = YES) or to part coordinates (IMOVEMACH = NO), before the coordinate transformations.

XFITOIND
The transfer inhibit of the channel affects the independent axes.

Possible values: Yes / No.
Default value: Yes.
Associated variable: (V.)[ch].MPG.XFITOIND

This parameter indicates whether the transfer inhibit of the channel (_XFERINH mark) affects the movements of the independent axis programmed from the PLC or not. The transfer inhibit of the channel always affects the movements of the independent axis programmed from the CNC.

XFITOIND	Meaning.
Yes	The _XFERINH mark affects the movement of the independent axis programmed from the PLC and from the CNC.
No	The _XFERINH mark does not affect the movements of the independent axis programmed from the PLC; but it does affect the ones programmed from the CNC.

DEFINITION OF THE SUBROUTINES.

SUBTABLE
OEM-subroutines table.

OEM subroutines are those associated with T, G74, G180 to G189, G380 to G399 functions and interruption subroutines. These subroutines must always be in the following folder; otherwise, an error message will be issued.

C:\CNC8070\MTB\SUB

When OEM subroutines have the .ram extension, the CNC will load them into RAM memory on power-up.

TOOLSUB
Subroutine associated with "T".

Possible values: any text with up to 64 characters.
Associated variable: (V.)[ch].MPG.TOOLSUB

This subroutine is executed automatically every time a T function (tool selection) is executed.

REFPSUB (G74)
Subroutine associated with function G74.

Possible values: any text with up to 64 characters.
Associated variable: (V.)[ch].MPG.REFPSUB

Function G74 (home search) may be programmed in two ways. indicating the axes and the order they will be homed or by programming G74 alone (without axes).

When executing a block that only contains the G74 function (without axes), it calls the subroutine indicated in this parameter. This subroutine must contain the axes and the order (sequence) to search home.

This subroutine is also called when homing the axis in JOG mode without selecting the axes.

OEMSUB (G18x)
Subroutines associated with functions G180 through G189.

Possible values: any text with up to 64 characters.
Associated variable: (V.)[ch].MPG.OEMSUB1··10

These parameters indicate the number of the subroutines associated with functions G180 through G189. Every time one of these functions is executed, its associated subroutine is called upon.

OEMSUB (G38x)
Subroutines associated with functions G380 through G399.

Possible values: any text with up to 64 characters.
Associated variable: (V.)[ch].MPG.OEMSUB11··30

These parameters indicate the number of the subroutines associated with functions G380 through G399. Every time one of these functions is executed, its associated subroutine is called upon.

INT1SUB··INT4SUB
Interruption subroutines.

Possible values: any text with up to 64 characters.
Associated variable: (V.)[ch].MPG.INT1SUB / (V.)[ch].MPG.INT4SUB

These parameters indicate the name of the interruption subroutines associated with marks INT1 to INT4 of the PLC. When the PLC activates one of these marks, the channel interrupts the execution of the program and executes the corresponding interruption subroutine. If the program is interrupted (STOP) or no program is in execution (channel in READY state), the execution of the subroutine depends on parameter SUBINTSTOP.

SUBPATH
Path of the program subroutines.

Associated variable: (V.)[ch].MPG.SUBPATH

This parameter indicates the directory, by default, containing the user subroutines and the necessary subroutines to execute programs of the 8055 MC and 8055 TC.

User subroutines.

User subroutines are those associated with part-programs. These subroutines must be in an unprotected folder for the user mode. When calling any of these subroutines (instructions #PCALL, #CALL, etc.) without indicating the path, the subroutine looks for them in this order and in the following directories:

1. Folder selected with the #PATH instruction.
2. Folder of the program in execution.
3. Folder indicated in the machine parameter SUBPATH.

When the call indicates the whole path, it will only look for it in the indicated folder.

Subroutines 9998 to 9999.

In order to be able to execute programs of the 8055 MC/TC, the CNC must have two subroutines called 9998 and 9999 written in 8070 CNC language.

9998 Subroutine that the CNC will execute at the beginning of each program.

9999 Subroutine that the CNC will execute at the end of each program.

Each program in the language of the 8055 MC/TC has a call to the corresponding subroutine at the beginning and at the end. Both subroutines must be defined, even if no operation is to be carried out at the beginning or at the end of the part-program, in which case the subroutines will be empty (except for the end-of-subroutine block). If any of these subroutines is missing, the CNC will issue an error message every time when trying to execute a part-program.

TABLETOP PROBE POSITION.

PROBEDATA
Channel related probe parameters.



This parameter shows the table to define the position of the tabletop probe on the first three axes of the channel according to parameter CHAXISNAME. The table shows the following parameters to configure it.

PROBEDATA	
PRB1MAX	Maximum probe coordinate (first axis of the channel).
PRB1MIN	Minimum probe coordinate (first axis of the channel).
PRB2MAX	Maximum probe coordinate (second axis of the channel).
PRB2MIN	Minimum probe coordinate (second axis of the channel).
PRB3MAX	Maximum probe coordinate (third axis of the channel).
PRB3MIN	Minimum probe coordinate (third axis of the channel).

PRB1MAX
Maximum probe coordinate (first axis of the channel).
Possible values: Within ±99999.9999 mm / within ±3937.00787 inch.
Default value: 0.
Associated variable: (V.)[ch].MPG.PRB1MAX

See general machine parameter PRB3MIN.

PRB1MIN
Minimum probe coordinate (first axis of the channel).
Possible values: Within ±99999.9999 mm / within ±3937.00787 inch.
Default value: 0.
Associated variable: (V.)[ch].MPG.PRB1MIN

See general machine parameter PRB3MIN.

PRB2MAX
Maximum probe coordinate (second axis of the channel).
Possible values: Within ±99999.9999 mm / within ±3937.00787 inch.
Default value: 0.
Associated variable: (V.)[ch].MPG.PRB2MAX

See general machine parameter PRB3MIN.

PRB2MIN
Minimum probe coordinate (second axis of the channel).
Possible values: Within ±99999.9999 mm / within ±3937.00787 inch.
Default value: 0.
Associated variable: (V.)[ch].MPG.PRB2MIN

See general machine parameter PRB3MIN.

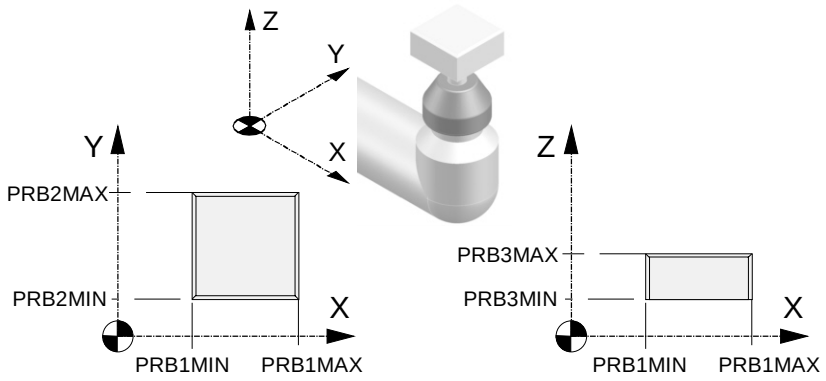
PRB3MAX
Maximum probe coordinate (third axis of the channel).
Possible values: Within ±99999.9999 mm / within ±3937.00787 inch.
Default value: 0.
Associated variable: (V.)[ch].MPG.PRB3MAX

See general machine parameter PRB3MIN.

PRB3MIN
Minimum probe coordinate (third axis of the channel).
Possible values: Within ±99999.9999 mm / within ±3937.00787 inch.
Default value: 0.
Associated variable: (V.)[ch].MPG.PRB3MIN

These parameters define the position of the tabletop probe used for tool calibration. They must be defined in absolute coordinates referred to machine reference zero.

For a LATHE model CNC, the coordinates must be given in radius.



BLOCK SEARCH.

FUNPLC
Send the M, H, S functions to the PLC during block search..
Possible values: Yes / No.
Default value: No.
Associated variable: (V.)[ch].MPG.FUNPLC

With block search, it is possible to recover the program history up to a particular block, that may be in a program or in a subroutine, in such way that if program execution is resumed at that block, it will do so with the same conditions as if it were executed from the beginning.

This parameter determines whether or not the CNC sends the M, H, S functions to the PLC during block search. The M function table has an MPLC field to define whether the CNC sends the function to the PLC or not. All M functions set in the table will be sent out to the PLC or not depending on this field regardless of the setting of machine parameter FUNPLC.

FUNPLC	Meaning.
Yes	In this case, the functions are sent out to the PLC during block search as they are being read. Once block search ends and after repositioning the axes, the CNC shows the "MHSF" softkey for executing any function of this type, thus modifying the machining conditions.
No	In this case, the functions are not sent out to the PLC during block search. After the search, the CNC screen shows the history of those functions so the user can enable them in the desired order.

INTERRUPTION SUBROUTINES.

SUBINTSTOP
Execute interruption subroutines while the program is stopped or while no program is in execution.
Possible values: Yes / No.
Default value: No.
Associated variable: (V.)[ch].MPG.SUBINTSTOP

This parameter determines whether the CNC must execute the interruption subroutine when the program is interrupted (STOP) or no program is in execution (channel in READY state). In order to execute the subroutine when no program is in execution, the channel must be in automatic mode; the subroutine cannot be executed from jog mode.

MACHINING FEEDRATE.

MAXFEED
Maximum machining feedrate on the tool path.

Possible values: From 0 to 500000.0000 mm/min / from 0 to 19685.03937 inch/min.
Default value: 0.
Associated variable: (V.)[ch].MPG.MAXFEED

This parameter sets the maximum machining feedrate (movements in G01/G02/G03) along the path; when set to ·0·, the feedrate is not limited This parameter cannot be set with a value higher than that of parameter G00FEED.

When trying to exceed the maximum feedrate via part-program, via PLC or from the operator panel, the CNC limits the feedrate to the maximum value set in MAXFEED without showing any error message or warning.

If the machining feedrate is not limited, the CNC assumes for all the movements the one set in machine parameter G00FEED as the maximum feedrate.

How to limit temporarily the maximum feedrate via PLC

The PLC has the variable (V.)[ch].PLC.G00FEED that may be used to limit the feedrate in the channel for any type of movement (G00, G01, etc). This variable limits the feedrate of the path and affects all the axes whether they move interpolated or one axis at a time.

The CNC assumes the change immediately and the change stays active until the variable takes a value of ·0· restoring the limits set by machine parameters.

DEFAULTFEED
Assume MAXFEED for movements in G1/G2/G3 without active feedrate.

Possible values: Yes / No.
Default value: No.
Associated variable: (V.)[ch].MPG.DEFAULTFEED

This parameter sets whether or not the axes move at the feedrate set in MAXFEED when programming a movement in G01/G02/G03 without programming the feedrate. If this parameter is set to ·No·, the CNC will issue an error message when programming a movement in G01/G02/G03 without programming the feedrate.

The CNC takes this parameter into account when MAXFEED has been set with a value other than ·0·.

RAPID TRAVERSE FOR THE AUTOMATIC MODE.

RAPIDEN
It is used to enable rapid traverse for the automatic mode.

Possible values: Disabled / EXRAPID or rapid key / EXRAPID and rapid key.
Default value: Disabled.
Associated variable: (V.)[ch].MPG.RAPIDEN

This parameter indicates whether rapid traverse may be enabled or not during the execution of a program for the programmed movements. Depending on the selected option, activating rapid traverse will require activating the PLC mark EXRAPID or pressing the "rapid" key or both.

RAPIDEN	Meaning.
Disabled.	Rapid traverse is not available for the automatic mode.
EXRAPID or rapid key.	To activate the rapid feed, just activate the PLC mark EXRAPID or press the "rapid" key of the jog panel.
EXRAPID and rapid key.	To activate the rapid feed, just activate the PLC mark EXRAPID and press the "rapid" key of the jog panel.

When activating rapid traverse while executing a program, each axis will assume the feedrate set by its parameter FRAPIDEN, as long as G0 or a thread is not active. Movements in G0 are carried out at the feedrate set in parameter G00FEED. The threads are executed at the

programmed feedrate. The rapid traverse will also be limited by channel parameter FRAPIDEN.

If "EXRAPID or rapid key" is the chosen option for RAPIDEN and the CNC has several channels, the EXRAPID mark will only affect the corresponding channel. The rapid key, on the other hand, affects simultaneously all the channels that may be affected at the time. If the active channel is in jog mode and another channel is executing a program, when pressing the rapid key in the active channel (jog mode), the rapid traverse will also be applied in the channel that is executing the program.
In a system where the rapid key may affect several channels at the same time, we recommend to choose the "EXRAPID and rapid key" option or manage by PLC the behavior of the rapid key according to the active channel and the selected work mode (jog or automatic).

FRAPIDEN
Rapid traverse in the channel for movements in automatic mode.
Possible values: From 0 to 500000.0000 mm/min / from 0 to 19685.03937 inch/min.
Default value: 0.
Associated variable: (V.)[ch].MPG.FRAPIDEN

Maximum feedrate in the channel for the program being executed, when rapid traverse for the automatic mode is active (parameter RAPIDEN). If the parameter is set with a "0" value, the feedrate is not limited.

This parameter does not the movements programmed in G00 or the threads. Movements in G0 are carried out at the feedrate set in parameter G00FEED. The threads are executed at the programmed feedrate.

Rapid traverse cannot exceed the value set in axis parameters G00FEED and FRAPIDEN or the maximum feedrate set by PLC (variable (V.)PLC.G00FEED). Rapid traverse cannot exceed the value set in axis parameter MAXFEED of the channel and the active feedrate set by PLC (variable (V.)PLC.F).

MAXIMUM ACCELERATION AND JERK ON THE TOOL PATH.

MAXACCEL
Maximum acceleration on the machining path.
Possible values: From 0.0010 to 600000000.0000 mm/min / from 0.00004 to 23622047.24409 inch/min.
Default value: There is no maximum feedrate.
Associated variable: (V.)[ch].MPG.MAXACCEL

This parameter may be used to smooth the feedrate on the tool path by limiting the maximum acceleration on it. Regardless of the value of this parameter, the CNC always respects the dynamics of the axes involved in the path; i.e. it respects the acceleration set for each axis.

How to change temporarily the maximum acceleration defined.

The variable "(V.)[ch].G.MAXACCEL" may be used to temporarily change at a particular time and in real time the value set by the machine parameter. The CNC assumes the change immediately and stays active until M30 or reset is executed, in which case the CNC restores the value set by the machine parameter. If the variable takes the value of ·0·, the CNC does not apply any acceleration limit on the tool path, not even the one set by machine parameter.

MAXJERK
Maximum jerk on the machining path.
Possible values: From 0.0010 to 6E11 mm/min / from 0.00004 to 2.362E10 inch/min.
Default value: There is no maximum feedrate.
Associated variable: (V.)[ch].MPG.MAXJERK

This parameter may be used to smooth the feedrate on the tool path by limiting the maximum jerk on it. Regardless of the value of this parameter, the CNC always respects the dynamics of the axes involved in the path; i.e. it respects the jerk set for each axis.

How to change temporarily the maximum jerk defined.

The variable "(V.)[ch].G.MAXJERK" may be used to temporarily change at a particular time and in real time the value set by the machine parameter. The CNC assumes the change immediately and stays active until M30 or reset is executed, in which case the CNC restores

the value set by the machine parameter. If the variable takes the value of ·0·, the CNC does not apply any jerk limit on the tool path, not even the one set by machine parameter.

MAXIMUM FREQUENCY ON THE TOOL PATH.

MAXFREQ
Maximum frequency generated on the machining path.
Possible values: From 0 to 500 Hz
Default value: 0.
Associated variable: (V.)[ch].MPG.MAXFREQ

This parameter may be used to set the parameters the damping (smoothing) filter for machining in G5. This filter improves the machining of CAD parts without having to activate the HSC feature. This parameter must be set using the FineTune application.

"RETRACE" FUNCTION.

RETRACAC
Permit enabling the "retrace" function.
Possible values: Yes / No.
Default value: No.
Associated variable: (V.)[ch].MPG.RETRACAC

This parameter indicates whether the "retrace" function may be used at the CNC or not. Once the "retrace" function is permitted, it may be activated from the PLC using the RETRACE mark.

NRETBLK
Maximum number of blocks allowed for the "retrace" function.
Possible values: From 0 to 300.
Default value: 75.
Associated variable: (V.)[ch].MPG.NRETBLK

This parameter sets the maximum number of blocks that may be retraced (executed backwards) with the "retrace" function. Once the CNC has retraced all the blocks, it goes on executing forward.

It is recommended not to increase the value of this parameter unnecessarily. In order to be able to execute the "retrace" function, the CNC stores the information of the last blocks executed. The higher the value assigned to this parameter, the greater the information the CNC must store.

RETMFUNC
Treatment of the M functions with the "retrace" function.
Possible values: Ignore / Cancel.
Default value: Ignore.
Associated variable: (V.)[ch].MPG.RETMFUNC

This parameter sets the behavior of the "retrace" function when executing M functions. When the CNC finds an M function, it can either ignore it and keep executing blocks in retrace or cancel the "retrace" function.

This parameter does not affect the following "M" functions.

- Functions M00 and M01 are always executed; they are sent to the PLC and [CYCLE START] must be pressed to resume execution in retrace.
- Functions M03 and M04 are always ignored; the CNC does not start the spindle nor does it change its turning direction.
- Function M05 only cancels the retrace function if it implies stopping the spindle; in other words, if the spindle is not running, an M05 function does not cancel the retrace function.

MASTER SPINDLE.

MASTERSPDL
Master spindle maintained.

Possible values: Temporary / maintained.
Default value: Temporary.
Associated variable: (V.)[ch].MPG.MASTERSPDL

This parameter indicates whether the channel keeps the current master spindle or recovers its original master spindle after executing M02, M30 or after an emergency or reset or restarting the CNC.

MASTERSPDL	Meaning.
Temporary.	The channel recovers its original master spindle if it is free; otherwise, it selects as master the firs spindle available in the original configuration.
Maintained.	The channel keeps the active master spindle.

When a channel does not keep its master spindle, on CNC power-up and after a reset, the channel assumes as master spindle the first spindle defined by the machine parameters of the channel (original master). If this spindle is parked or "handed out" to another channel, the channel assumes as master spindle the next spindle defined by the machine parameters and so on. If the channel does not have spindles of the original configuration (the one defined by the machine parameters) because they are parked or "handed out", it assumes as master spindle the first one of the current configuration that is not parked.

Exchanging spindles between channels.

When spindles are being exchanged between channels, the behavior of this parameter also depends on parameter AXISEXCH that sets whether the channel change of a spindle is temporary or permanent. If the current master spindle of the channel is a spindle "loaned" by another channel and its permission to change channels is temporary (AXISEXCH = Temporary), the spindle returns to its original channel.

2.5 Machine parameters for the axes and spindles.

The CNC only shows the parameters for the selected type of axis and drive. That's why it displays some characters next to each parameter indicating the corresponding type of axis and drive.

BELONGING TO THE CHANNEL.

AXISEXCH
Channel change permission.
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: No / Temporary / Maintained.
Default value: No.
Associated variable: (V.)[ch].MPA.AXISEXCH.xn

Initially, each channel has some axes and spindles assigned to it. The CNC can change channel axes and spindles or just change the configuration of a channel by modifying the position of its axes and spindles or eliminating some of them.

In order for the CNC to be able to change the channel axes and spindles, they must have permission. Parameter AXISEXCH sets whether the axis or the spindle has permission to change channels and if it does, whether the change is temporary or permanent; in other words, whether the change is maintained after an M02, M30, a reset or after restarting the CNC.

AXISEXCH.	Meaning.
No.	• The CNC can modify the position of the axis or spindle in its channel or even remove it from the channel. • The CNC cannot change the axis or the spindle of the channel. • The CNC does not maintain the changes when restarting the part-program, after a reset or after restarting the CNC. The axis or spindle returns to its original position, set by machine parameters.
Temporary.	• The CNC can modify the position of the axis or spindle in its channel or even remove it from the channel. • The CNC can change the axis or the spindle of the channel. • The CNC does not maintain the changes when restarting the part-program, after a reset or after restarting the CNC. The axis or spindle returns to its channel and original position, set by machine parameters.
Maintained.	• The CNC can modify the position of the axis or spindle in its channel or even remove it from the channel. • The CNC can change the axis or the spindle of the channel. • The CNC maintains the changes when restarting the part-program, after a reset or after restarting the CNC. The axis or spindle stays in its new channel, but it can change the position to let the original axes of the channel return.

The original configuration (the one defined in the machine parameters) of a channel with axes and spindles of the type "AXISEXCH = Maintained" may be restored either by validating the machine parameters or by undoing the changes, for example using a part-program. It must be borne in mind that validating the machine parameters restores the configuration of all the channels.

TYPE OF AXIS AND DRIVE.

AXISTYPE
Type of axis.
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Linear / Rotary / Spindle.
Default value: Linear.
Associated variable: (V.)[ch].MPA.AXISTYPE.xn

The axes defined here may be configured as gantry or tandem axes.

DRIVETYPE
Type of drive.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drives, Sercos, Mlink (Mechatrolink) and simulated drives.
Possible values: Analog / Sercos / Mlink / Simulated.
Default value: Simulated.
Associated variable: (V.)[ch].MPA.DRIVETYPE.xn

This parameter sets the type of drive, analog, Sercos, Mlink (Mechatrolink) or simulated. The CNC does not allow having Sercos and Mechatrolink drives in the same system; but it does allow combining these drives with analog drives. The simulated axis option must be used when there is no physical axis or during setup. With the Simulated option, the CNC simulates all the movements, it assumes the theoretical coordinate as real and does not output velocity commands.

The simulated axes are not activated with the validation code. It is possible to use as many simulated axes as you wish as long as the sum of the simulated axes and the physical axes does not exceed the maximum number of axes possible (as set by parameter NAXIS).

Type of drive when the CNC has been installed as simulator at a PC.

When the CNC is installed as a simulator at a PC, the drive may be a simulated type, Sercos or Mechatrolink; it cannot be analog. In either case, the CNC simulates all the movements and assumes the theoretical coordinate as the real one.

POSUNITS
Feedback system units.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for Sercos, Mlink (Mechatrolink) and simulated drives.
Possible values: Measurement / Pulses.
Default value: Measurement.
Associated variable: (V.)[ch].MPA.POSUNITS.xn

This parameter sets the units system used by the feedback system.

POSUNITS	Meaning.
Measurement.	The communication with the drive is in millimeters or degrees.
Pulses.	The communication with the drive is in pulses.

Refer to the drive manual for further detail.

CONFIGURING A SERCOS DRIVE.

SERCOSDATA
Sercos drive data.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for a Sercos drive.

This parameter shows the table to define the communication with the Sercos drive.

SERCOSDATA	
DRIVEID	Sercos drive address.
OPMODEP	Main operating mode of the Sercos drive.
FBACKSRC	Type of feedback.
FBACKDIFF	Maximum difference between feedbacks.
ULTRAIPO	Parameter not being used at this time.
FBACKTIME	Time constant for combined feedback.

DRIVEID
Sercos drive address.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for a Sercos drive.
Possible values: From 1 to 32.
Default value: 1.
Associated variable: (V.)[ch].MPA.DRIVEID.xn
Parameter included in the SERCOSDATA table.

This parameter indicates the position (node) occupied by the drive in the Sercos connection.

OPMODEP
Drive's main operating mode.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for a Sercos drive.
Possible values: Position / Speed.
Default value: Position.
Associated variable: (V.)[ch].MPA.OPMODEP.xn
Parameter included in the SERCOSDATA table.

This parameter indicates the operating mode of the Sercos drive; velocity command or position command. Refer to the drive manual for further detail.

We recommend to set the axes (except tandem axes) to operate in position-Sercos mode and the spindles in velocity-Sercos mode. However, tandem axes must necessarily work in velocity-Sercos mode. All the axes that will be interpolated with the tandem axis must also operate in velocity-Sercos mode.

Operating mode of the drive in velocity-Sercos.

- The velocity command is sent out to the drive in ten-thousandths rpm of the motor When needing to use the SANALOG from the PLC, its value will also be given in ten-thousandths rpm of the motor.
- The CNC receives the drive's feedback as absolute coordinate in ten-thousandths of a millimeter or thousandths of a degree (same as in Sercos position mode).
- The CNC controls the home search. The home search is carried out without stopping the spindle if it was turning. If it is turning in M03 or M04 and M19 is programmed, the spindle decelerates down to REFFEED1 and starts searching home.
- The following error and the backlash compensation are calculated at the CNC.
- The CNC applies feed-forward and AC-forward.
- When the spindle is turning in open loop, the coordinate is simulated.

Operating mode of the drive in position-Sercos.

- The command is sent out to the drive in absolute coordinates, in ten-thousandths of a millimeter (linear axes) or thousandths of a degree (rotary axes). When needing to use the SANALOG from the PLC, its value will also be given in ten-thousandths of a millimeter or thousandths of a degree. The drive smoothes the command using a cubic interpolation and closes the position loop every 256 microseconds.
- The feedback from the drive to the CNC arrives as absolute coordinate, in ten-thousandths of a millimeter or thousandths of a degree (same as in velocity-Sercos mode).
- The drive controls the home search. Before starting the home search, the spindle stops if it is turning.
- The drive calculates the following error (lag).
- The drive applies feed-forward and AC-forward.

FBACKSRC
Feedback type.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for a Sercos drive.
Possible values: Internal / External / Internal+External.
Default value: Internal.
Associated variable: (V.)[ch].MPA.FBACKSRC.xn
Parameter included in the SERCOSDATA table.

Type of feedback used to close the position loop. Refer to the drive manual for further detail.

When using internal feedback, the position value is taken from motor feedback whereas when using external feedback, it is taken from direct feedback. When using internal+external feedback, the feedback may be swapped via PLC using the FBACKSEL(axis) mark.. When using external or internal+external feedback, we recommend setting parameter FBACKDIFF.

When the type of feedback is external or internal+external, the CNC can enable the combined feedback; parameter FBMIXTIME.

FBACKSRC	FBMIXTIME	Type of feedback.
Internal	(Not being used)	Internal feedback.
External	0	External feedback.
External	Value greater than ·0·.	The combined (mixed) feedback is active.
Internal+External	0	The feedback type may be swapped from the PLC using the mark FBACKSEL(axis). Internal or external feedback.
Internal+External	Value greater than ·0·.	The feedback type may be swapped from the PLC using the mark FBACKSEL(axis). Internal feedback or combined (mixed) feedback.

When the type of feedback is internal+external, the CNC uses the internal feedback (motor feedback) on power-up, when resetting the drive and when initializing the Sercos ring.

Tandem axes do not admit the internal+extenrnal type of feedback. In a tandem axis, both axes must have the same type of feedback, external or internal, but usually, the master axis uses external feedback and the slave axis the internal one.

FBACKDIFF
Maximum difference between feedbacks.

Parameter valid for linear and rotary axes.
Parameter valid for a position-Sercos drive.
Possible values: Within ±99999.9999 mm or degrees / within ±3937.00787 inch.
Default value: 0 (no monitoring).
Associated variable: (V.)[ch].MPA.FBACKDIFF.xn
Parameter included in the SERCOSDATA table.

The CNC takes this parameter into account only on axes with external or external+internal feedback.

When the system has dual feedback, this parameter may be used to monitor the difference between both feedbacks. If the difference exceeds the set value, the CNC will display the corresponding error message. When defined with a "0" value, there will be no monitoring.

ULTRAIPO
(Not being used at this time)

Parameter not being used at this time.

FBMIXTIME
Time constant for combined (mixed) feedback.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for a Sercos drive.
Possible values: From 0 to 3200.0 ms.
Default value: 0.
Associated variable: (V.)[ch].MPA.FBMIXTIME.xn
Parameter included in the SERCOSDATA table.

This parameter may be used to set the time constant to be used when combining feedbacks; i.e. it sets the delay between the coordinates entered in the position loop of the rotary encoder and that of the linear encoder. The CNC takes this parameter into account only when using external or external+internal feedback.

If the parameter is set to a value other than ·0·, the combined feedback is enabled; if set to ·0·, only the external feedback is enabled. When changing this parameter, the parameter validation will force resetting, initializing, the Sercos ring.

This parameter requires a drive version V6.13 or newer.

On machines that have a lot of backlash and use external feedback to obtain greater accuracy may suffer some instability. This kind of machines, with internal feedback, run smoothly, but they may loose accuracy; using external feedback on the other hand, the machine is more accurate but it may run more abruptly. The combination of the two feedbacks achieves the desired result of accuracy and smoothness.

The CNC uses the combined feedback to calculate the velocity command, but it uses the external (direct) feedback to calculate the compensations, circularity test, etc.

CONFIGURING A MECHATROLINK DRIVE.

MLINKDATA
Data of the Mechatrolink drive.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for a Mechatrolink drive.

This parameter shows the table to define the communication with the Mechatrolink drive.

MLINKDATA	
DRIVEID	Drive select.
OPMODEP	Main operating mode of the drive.
OPTION	Activate the drive options.

DRIVEID
Drive address.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for a Mechatrolink drive.
Possible values: From 1 to 14 (Mlink-I) / From 1 to 30 (Mlink-II).
Default value: 1.
Associated variable: (V.)[ch].MPA.DRIVEID.xn
Parameter included in the MLINKDATA table.

This parameter indicates the position (node) occupied by the drive in the Mechatrolink bus. The drives must occupy consecutive positions starting at ·1·. Although the relationship between the node number and the logic number of the axis or spindle is irrelevant, for clarity sake we recommend to configure the nodes in the same order the axes and spindles are defined at the CNC.

OPMODEP
Drive's main operating mode.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for a Mechatrolink drive.
Possible values: Position / Velocity / Inverter.
Default value: Position (axes) / Velocity (spindles).
Associated variable: (V.)[ch].MPA.OPMODEP.xn
Parameter included in the MLINKDATA table.

This parameter indicates the type of drive and the operating mode; servo with a velocity command, servo with a position command or inverter control; the latter is only available for spindles. We recommend to set the servos of the axes to operate in position mode and the spindles in velocity mode. The inverter option must be used when the spindle is connected to this type of device.

OPTION
Activate the drive options

Parameter valid for linear and rotary axes and spindles.
Parameter valid for a Mechatrolink drive.
Possible values: From $00000000 to $0000FFFF.
Default value: $00000000.
Parameter included in the MLINKDATA table.

This parameter lets configure different options of the Mechatrolink drive, such as activating acc/dec filters, activate the first or second feedback, choose either a P or a PI control for the velocity loop, etc.

The drive options are coded in 16-bit format. Refer to the drive manual for the meaning of each bit.

HIRTH AXIS.

HIRTH
Axis with Hirth toothing

Parameter valid for linear and rotary axes.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Yes / No.
Default value: No.
Associated variable: (V.)[ch].MPA.HIRTH.xn

A Hirth axis is the one that can only be positioned at positions multiple of a given value.

Even if the position of a Hirth axis does not coincide with its pitch, the axis may be moved to a valid position in both automatic and jog modes. If the position to move the axis does not match its pitch, the CNC will issue an error message.

HPITCH
Hirth axis pitch.

Parameter valid for linear and rotary axes.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 99999.9999 mm or degrees / from 0 to 3937.00787 inch.
Default value: 1 mm or degrees / 0.03937 inch.
Associated variable: (V.)[ch].MPA.HPITCH.xn

When the axis is set as Hirth axis, this parameter indicates its pitch. Parameter REFVALUE will be taken as initial coordinate for the next increments.

Any stop or movement in continuous jog will stop the axis in coordinates multiple of HPITCH. Movements in incremental jog will only be valid if they are greater than HPITCH and will be limited to multiples of this value. The positions of the switch for jog movements may be customized at the axis parameter table so they match the desired steps.



The way the decimals of a Hirth axis will be displayed on the screen will be configured using FGUIM.

AXIS CONFIGURATION FOR LATHE TYPE MACHINES.

FACEAXIS
Face axis on a lathe.

Parameter valid for a linear axes.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Yes / No.
Default value: No.
Associated variable: (V.)[ch].MPA.FACEAXIS.xn

See axis machine parameters LONGAXIS.

LONGAXIS
Longitudinal axis on a lathe.

Parameter valid for a linear axes.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Yes / No.
Default value: No.
Associated variable: (V.)[ch].MPA.LONGAXIS.xn

On turning machines, you must indicate which one is the longitudinal axis and which one is the cross axis.

Typical lathe setting:		
X axis.	FACEAXIS = Yes	LONGAXIS = No
Z axis.	FACEAXIS = No	LONGAXIS = Yes
Rest of the axes.	FACEAXIS = No	LONGAXIS = No

Typical Mill setting:		
All the axes.	FACEAXIS = No	LONGAXIS = No

These parameters only affect the axes that can work in lathe mode. If it is a combined machine and any or several milling axes can work as lathe, it is necessary to set these parameters. In some cases, the same axis can work as a face axis (FACEAXIS=Yes) and as longitudinal axis (LONGAXIS=Yes) and consequently, both parameters must be set to Yes.

SYNCHRONIZATION OF AXES AND SPINDLES.

SYNCSET
Set of parameters for synchronization.

Parameter valid for rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 4.
Default value: 1.
Associated variable: (V.)[ch].MPA.SYNCSET.xn

Parameter set assumed by the axis or spindle when it is synchronized. If the parameter is set to ·0· (zero), the CNC will try to synchronize the spindle without forcing the set change.

- A slave axis or spindle always assumes the set defined in its parameter SYNCSET.
- If the master and slave spindles are in the same channel, master spindle assumes the set defined in its parameter SYNCSET. If both spindles are in different channels, the parameter set of the master spindle must be selected before synchronizing.

DSYNCVELW
Speed synchronization window.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 200000.0000 mm/min / from 0 to 7874.01575 inch/min / from 0 to 36000000.0000 degrees/min / from 0 to 100000 rpm.
Default value: 100 mm/min / 3.937 inch/min / 3600 degrees/min / 10 rpm.
Associated variable: (V.)[ch].MPA.DSYNCVELW.xn

This parameter is defined for the slave element of the synchronization and indicates the velocity margin admitted for the synchronization to be OK.

When the spindles are synchronized in speed, the slave spindle turns at the same speed as the master spindle (taking the ratio into account). If the value defined in this parameter is exceeded, the SYNSPEED signal goes low; the movement is not stopped and no error message is issued.

When synchronizing axes, the slave axis moves at the same feedrate as the master (considering the ratio). If the difference between the synchronism speed calculated for the slave axis and its real speed exceeds the value set in this parameter, it cancels the INSYNC mark of the PLC.

DSYNCPOSW
Position synchronization window.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 99999.9999 mm/min / from 0 to 3937.00787 inch/min / from 0 to 99999.9999 degrees/min.
Default value: 0.0100 mm/min or degrees/min / 0.00039 inch/min.
Associated variable: (V.)[ch].MPA.DSYNCPOSW.xn

This parameter is defined for the slave element of the synchronization and indicates the position margin admitted for the synchronization to be OK.

When the spindles are synchronized in position, the slave spindle follows the master keeping the programmed offset (bearing the ratio in mind) and it activates the SYNCHRONP mark of the PLC. If the value set in this parameter is exceeded, it cancels the SYNPOSI mark of the PLC; it does not stop the movement or issue any error message.

When synchronizing axes, the slave axis follows the master keeping the offset (considering the ratio). If the difference between the synchronism position calculated for the slave axis and its real position exceeds the value set in this parameter, it cancels the INSYNC mark of the PLC.

CONFIGURATION OF THE ROTARY AXES.

AXISMODE
Work mode of the rotary axis.

Parameter valid for a rotary axes.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Linearlike / Module.
Default value: Module.
Associated variable: (V.)[ch].MPA.AXISMODE.xn

This parameter indicates how the rotary axis will behave in terms of number of turns or the display of coordinates.

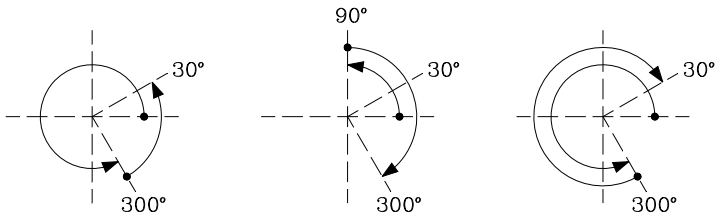
Behavior when AXISMODE = Module.

The axis behaves like a rotary axis. Movements in G0/G1 and G90/G91 may be programmed.

- For movements in G90, more than a complete turn may be programmed or values outside the module, but the whole travel must always be less than a complete turn.

  If the axis is neither SHORTESTWAY nor UNIDIR, the programmed sign indicates the turning direction and the absolute value of the coordinate indicates the target position.

SHORTESTWAY = No
UNIDIR = No



B300 B30 | B90 B−300 | B300 B−30

- In movements in G91, the sign indicates the turning direction and the absolute value of the coordinate indicates the distance to move.

The module limits (travel limits of the rotary axis) are set by set parameters MODUPLIM and MODLOWLIM. For analog and simulated axes, the lower limit MODLOWLIM must be lower than the upper limit MODUPLIM, for example 0º to 360º, 0º to 400º or -230 to 95; it is not possible, for example -100º to -230º or 360º to 0º. For Sercos axes, the module limits must be 0º and 360º.

The coordinates are always displayed within the module limits, by default 0 and 360º.

Parameters SHORTESTWAY and UNIDIR must be set. Parameters LIMIT+ and LIMIT- have not meaning.

Behavior when AXISMODE = Linearlike.

It behaves like a linear axis. Movements in G0/G1 and G90/G91 may be programmed.

The reading is free and in degrees (not affected by mm/inch). There are travel limits set by "LIMIT+" and "LIMIT-".

Parameters SHORTESTWAY, UNIDIR and those for set MODUPLIM and MODLOWLIM do not apply.

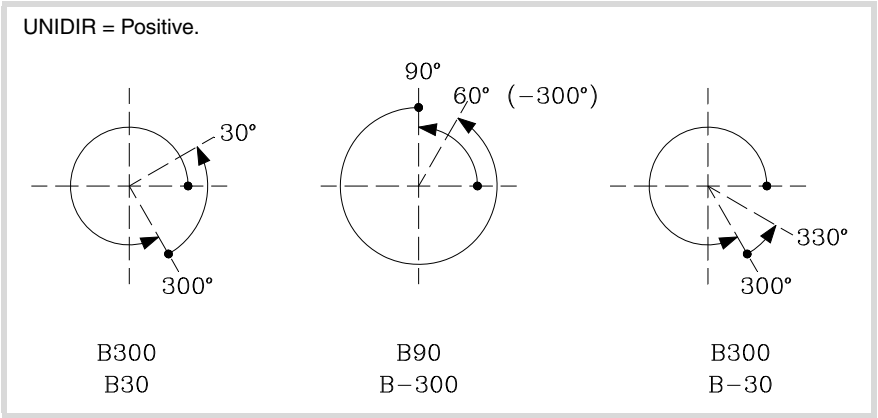
UNIDIR
Single turning direction.

Parameter valid for a rotary axes.
Parameter valid for analog drive, Sercos and simulated.
Possible values: No (both directions) / Positive / Negative.
Default value: No (both directions).
Associated variable: (V.)[ch].MPA.UNIDIR.xn

The CNC takes this parameter into consideration only when AXISMODE = Module and SHORTESTWAY = No.

This parameter indicates whether G00/G01 movements of the rotary axes in G90 may be carried out in both directions or they must always turn in the same direction (either positive

or negative). If the axis is not UNIDIR, the programmed sign indicates the turning direction and the absolute value of the coordinate indicates the target position.



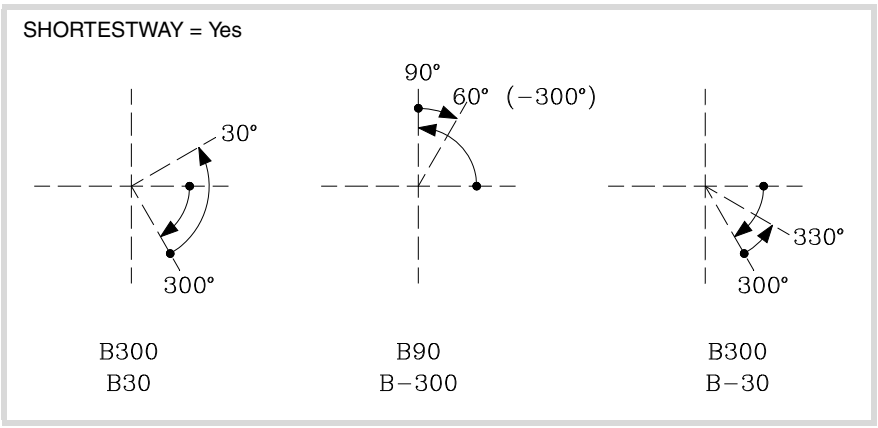
The movements in G91 are carried out in the programmed direction. If it is a UNIDIR axis, the programmed direction must be the same as the one preset for the axis; otherwise an error message will be issued because the axis cannot turn in the opposite direction. Likewise, the error will also come up when programming a mirror image on these axes.

SHORTESTWAY
Via the shortest way.

Parameter valid for a rotary axes.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Yes / No.
Default value: No.
Associated variable: (V.)[ch].MPA.SHORTESTWAY.xn

The CNC takes this parameter into consideration only when AXISMODE = Module and UNIDIR = No.

This parameters indicates whether the rotary axis movements in G90 are carried out via the shortest way or not. Otherwise, the programmed sign will indicate the turning direction whereas the absolute value of the coordinate will indicate the target position.



The movements in G91 are carried out in the programmed direction.

MODULE CONFIGURATION (ROTARY AXES AND SPINDLE).

MODCOMP
Module compensation.

Parameter valid for rotary axes and spindles.
Parameter valid for analog drives, velocity-Sercos drives and simulated drives.
Possible values: Yes / No.
Default value: No (without compensation).
Associated variable: (V.)[ch].MPA.MODCOMP.xn

The CNC takes this parameter into consideration only when AXISMODE = Module.

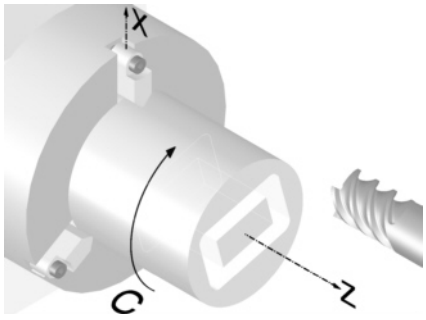
The module compensation must be activated when the axis resolution is not exact. Range parameters MODNROT and MODERR set the compensation to be applied to obtain the exact reading. The CNC applies module compensation throughout the entire revolution.

CONFIGURATION OF THE C AXIS.

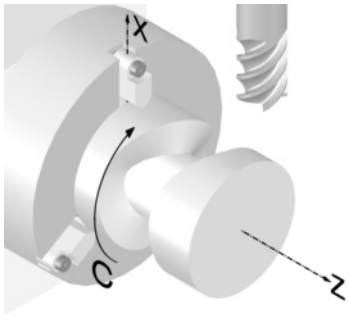
CAXIS
Possibility to work as a C axis.

Parameter valid for rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Yes / No.
Default value: No.
Associated variable: (V.)[ch].MPA.CAXIS.xn

This parameter indicates whether the axis or spindle can work as C axis.



XC work plane.



ZC work plane.

CAXSET
Set of parameters to work as C axis.

Parameter valid for rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 1 to 4.
Default value: 1.
Associated variable: (V.)[ch].MPA.CAXSET.xn

The CNC takes this parameter into consideration only when CAXIS = Yes.

This parameter indicates which set of parameters (NPARSETS) the axis or spindle uses when it works as C axis.

PERCAX
"C" axis kept active

Parameter valid for rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Yes / No.
Default value: No.
Associated variable: (V.)[ch].MPA.PERCAX.xn

The CNC takes this parameter into consideration only when CAXIS = Yes.

This parameter indicates whether the CNC keeps the C axis active or not after executing M02, M30 or after an emergency or reset. After turning the CNC off, the CNC deactivates the C axis.

CONFIGURATION OF THE SPINDLE.

AUTOGEAR
Automatic gear change.

Parameter valid for spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Yes / No.
Default value: No.
Associated variable: (V.)[ch].MPA.AUTOGEAR.xn

This parameter indicates whether the gear change is automatically generated by activating (if necessary) the auxiliary functions M41, M42, M43 and M44 when programming the speed.

LOSPDLIM
Lower "rpm OK" percentage.

Parameter valid for spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 255.
Default value: 50.
Associated variable: (V.)[ch].MPA.LOSPDLIM.xn

See axis machine parameters UPSPDLIM.

UPSPDLIM
Upper "rpm OK" percentage.

Parameter valid for spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 255.
Default value: 150.
Associated variable: (V.)[ch].MPA.UPSPDLIM.xn

When working with M3 and M4, the REVOK signal is set high when the actual spindle rpm are between these percentages.

The REVOK signal may be used to manage the Feedhold signal and avoid machining at lower or higher rpm than the ones programmed.

SPDLTIME
Estimated time for an S function.

Parameter valid for spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 1000000 ms.
Default value: 0 ms.
Associated variable: (V.)[ch].MPA.SPDLTIME.xn

In EDISIMU mode, there is an option that allows calculating the time required to execute a part with the machining conditions established in the program.

To fine tune that calculation, one may define this parameter that indicates the estimated time for processing the S function.

When assigning a value other than "0", the CNC interprets that the "S" value must be passed on to the PLC using signals SSTROBE + SFUN1.

SPDLSTOP
Functions M2 and M30, an error or a reset stop the spindle.

Parameter valid for spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Yes / No.
Default value: Yes.
Associated variable: (V.)[ch].MPA.SPDLSTOP.xn

This parameter indicates whether the spindle is stopped or not when executing M02, M30, when doing a reset or when an error occurs. Otherwise, it will be necessary to program function M5.

The spindle errors and an emergency always stop the spindle.

SREVM05
With G84, the spindle must be stopped to change the turning direction.

Parameter valid for spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Yes / No.
Default value: No.
Associated variable: (V.)[ch].MPA.SREVM05.xn

This parameter indicates whether the spindle must be stopped (with M5) when reversing the spindle in tapping cycle.

M19SPDLEREV
The SPDLEREV mark (reverse turning direction) affects the spindle in M19.

Parameter valid for spindles.
Parameter valid for an analog drive.
Possible values: Yes / No.
Default value: No.
Associated variable: (V.)[ch].MPA.M19SPDLEREV.xn

This parameter defines whether the PLC mark SPDLEREV (reverse turning direction) also affects the spindles in M19. The SPDLEREV mark always affects the spindles in open loop (M3/M4), regardless of how this parameter has been set.

STEPOVR
Spindle override step.

Parameter valid for spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 255.
Default value: 5.
Associated variable: (V.)[ch].MPA.STEPOVR.xn

This parameter sets the incremental step used to change the programmed spindle speed with the spindle override keys of the operator panel. Parameter ignored when the operator panel has a spindle speed override switch.

MINOVR
Minimum override (%) permitted for the spindle.

Parameter valid for spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 255.
Default value: 50.
Associated variable: (V.)[ch].MPA.MINOVR.xn

This parameter sets the minimum percentage allowed to apply to the spindle speed when it is changed from the operator panel (keys or speed override switch).

MAXOVR
Maximum override (%) permitted for the spindle.

Parameter valid for spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 255.
Default value: 150.
Associated variable: (V.)[ch].MPA.MAXOVR.xn

This parameter sets the maximum percentage allowed to apply to the spindle speed when it is changed from the operator panel (keys or speed override switch).

SPINDLE OVERRIDE CHANGE WHILE THREADING.

THREADOVR

Maximum variation allowed for the override while threading.

Parameter valid for spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: from0 to 100%.
Default value: 0 (The override cannot be changed).
Associated variable: (V.)[ch].MPA.THREADOVR.xn

If this parameter is set with a value other than ·0·, it is possible to change the spindle override during electronic threading (G33) and in the threading canned cycles of the ·T· model (G86, G87 and their equivalent of the cycle editor).

The parameter sets the maximum spindle override variation possible, both to increase it and to reduce it. For example, if set to ·30·, the override may be varied between 80% and 130%. The limits set by machine parameters MINOVR and MAXOVR can never be exceeded.

In order to avoid damaging the thread when varying the spindle override, you should use a feed-forward value close to 100% on the axes involved in threading so the following error is as small as possible. The CNC allows modifying the spindle override while threading if it detects that the feed forward (parameter FFWTYPE) is not active in a gear of the axes involved or if the active feed forward is lower than 90% Feed forward is defined with parameter FFGAIN and may be modified from the PLC using variables.

Threading canned cycles of the ·T· model. Considerations related to changing the override value.

- In threading cycles, it is possible to change the spindle speed during the threading passes, except in those done all the way to the final depth of the thread (one or two final passes) which are done at the override value active at the beginning of the pass.
- Even if the override is changed, the CNC respects the pitch and entry of the thread.
- It is recommended not to change the override on threads with flank infeed.

Electronic threading G33. Considerations related to changing the override value.

- If more than one G33 have been programmed for the same thread, all the threading operations must start at the same speed; otherwise, the entry point (start) to the thread will not be the same in all the threads. The CNC permits changing the spindle override during the thread cutting pass.
- If more than one G33 have been programmed for a multi-start (multi-entry) thread, all the threading operations must start at the same speed; otherwise, the angle between the starts (entry points) to the thread will not be the same as the one programmed. The CNC permits changing the spindle override during the thread cutting pass.

OVRFILTER

Time to make the override change effective.

Parameter valid for spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 1000000 ms.
Default value: 0.
Associated variable: (V.)[ch].MPA.OVRFILTER.xn

The CNC takes this parameter into account only if THREADOVR other than 0 (zero).

This parameter acts as a filter to make the override change effective The override change is applied progressively during the indicated period of time.

SOFTWARE AXIS LIMITS.

LIMIT+
Positive software limit.
Parameter valid for linear and rotary axes.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Within ±99999.9999 mm or degrees / within ±3937.00787 inch.
Default value: 99999.9999 mm or degrees / 3937.00787 inch.
Associated variable: (V.)[ch].MPA.POSLIMIT.xn

See axis machine parameters LIMIT-.

LIMIT-
Negative software limit.
Parameter valid for linear and rotary axes.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Within ±99999.9999 mm or degrees / within ±3937.00787 inch.
Default value: -99999.9999 mm or degrees / -3937.00787 inch.
Associated variable: (V.)[ch].MPA.NEGLIMIT.xn

On rotary axes, the CNC takes this parameter into account when AXISMODE = Linearlike.

On linear and rotary axes, these parameters set the axis travel limits. If both are set to "0", there will be no software limits.

The software limits are always set in radius, regardless of the setting of parameter DIAMPROG.

SWLIMITTOL
Software limit tolerance.
Parameter valid for linear and rotary axes.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 99999.9999 mm or degrees / from 0 to 3937.00787 inch.
Default value: 0.1000 mm or degrees / 0.00394 inch.
Associated variable: (V.)[ch].MPA.SWLIMITTOL.xn

This parameter indicates the maximum variation or oscillation allowed for an axis located at the limit.

This parameter indicates the maximum variation or oscillation allowed to a real coordinate of an axis referred to the software limits, before issuing an error indicating travel limit overrun. The programmed theoretical movement of the axis is only possible up to the exact limit, but the real axis coordinate is allowed this margin before the error is issued.

When they are DRO axes, the error is also issued when the real coordinate exceeds the limit over the tolerance.

When no theoretical movement has been programmed, the limit overrun error will only be issued when exceeding the tolerance in a sampling period (cycle time); for example, when hitting the axis causing it to overrun the limits abruptly. In any other case, if no theoretical movement has been programmed for the axis, the error will not be issued even if it overruns the limits.

RUNAWAY PROTECTION AND TENDENCY TEST.

TENDENCY
Activation of tendency test.
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drives and simulated drives.
Possible values: Yes / No.
Default value: No.
Associated variable: (V.)[ch].MPA.TENDENCY.xn

This parameter enables two tests:

- Axis tendency watch.

  This test watches that the real movement coincides with the theoretical movement, from the time period set in parameter ESTDELAY. If no theoretical movement has been programmed for the axis, the test detects whether the axis is moving in one direction longer than the time indicated in parameter ESTDELAY. If no theoretical movement has been programmed for the axis, the test detects whether it is moving in the opposite direction longer than the time indicated by ESTDELAY.

- Runaway protection.

  This test watches the real movement of the axis from startup to detect if runs away considering the time defined in the parameter TENDENCY.

If this alarm is canceled, the CNC shows on power-up a message indicating that this safety is disabled. This situation can only be allowed during setup; once setup is completed, this alarm must be enabled.

TENDTIME
Time to detect axis runaway.
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drives and simulated drives.
Possible values: From 0 to 65535.
Default value: 0.
Associated variable: (V.)[ch].MPA.TENDTIME.xn

When the runaway test is active, this parameter defines the time the CNC must wait before issuing an error message. We recommend to set a short time (about 4 sampling periods) to prevent the axis from moving too far.

PLC OFFSET.

PLCOINC
PLC offset increment per cycle.
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 99999.9999 mm or degrees / from 0 to 3937.00787 inch.
Default value: 0 (they are assumed instantaneously).
Associated variable: (V.)[ch].MPA.PLCOINC.xn

The PLC offset is an additional compensation over the real coordinate of the axis. The CNC applies this compensation in a transparent way for the user who will not see it reflected in the coordinates. A typical utility is to correct the axis dilatations due to temperature.

This parameter indicates whether the CNC assumes the PLC offsets instantaneously in a sampling period (cycle time) or in steps.

Example:

It is set PLCOINC = 0.001 mm (one micron per CNC cycle). If the PLC Offset had an initial value of 0.25 mm and the new value is 0.30 mm, the PLC offset applied per cycle will be:

0.250 0.251 0.252 0.253 · · · 0.297 0.298 0.299 0.300

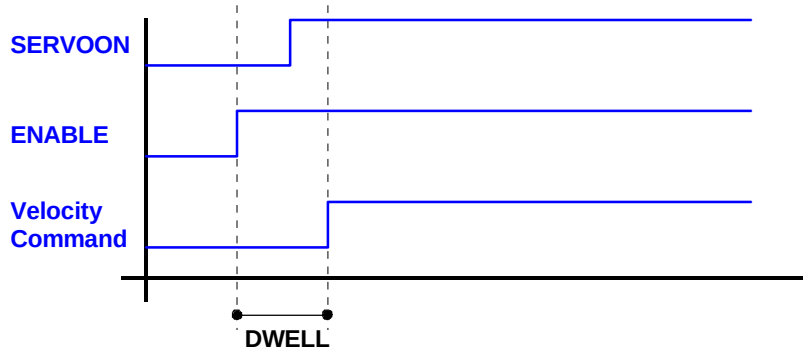
The PLC offset to be applied is set in the variable (V.).PLCOF.xn. The PLC offset applied so far may be checked in the variable (V.)A.ACTPLCOF.xn.

DWELL FOR DEAD AXES.

DWELL
Dwell for dead axes.
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 1000000 ms.
Default value: 0 (no dwell).
Associated variable: (V.)[ch].MPA.DWELL.xn

When an axis has a brake, for example very heavy vertical axes, it is only governed while it is moving. When it is governed by the CNC (movements), it is referred to as being alive and, when not moving (brake on), it is referred to as being "dead".

To bring it to "life", release the brake and close the position loop. The time required for this operation must be defined by the DWELL parameter.



The ENABLE signal indicates to the PLC to let the axis move and the SERVOON signal that the drive is ready.

The CNC waits the amount of time set in parameter DWELL before generating a movement on the "dead" axis if it does not have the SERVOON signal active. Once the dwell has started, the CNC will wait the time set in parameter DWELL before starting the movement, even if the SERVOON signal has been activated before.

Special care must be taken with the delayed disconnection of the axis, when it becomes a "dead" axis. If the SERVOON signal has been turned off and a new movement of the axis begins, the CNC will not start timing but it could issue an error indicating that the axis is locked up if as a result of disconnecting logically, the SERVOON signal ends up dropping.

On a tandem axis, both the master and slave axes must be enabled so they can be moved. In this case, the CNC only applies the time set in DWELL to the SERVOON of the master axis; if the enabling of the slave axis is slower, the PLC maneuver must verify that both axes are enabled before generating the movement.

If an axis is going to work as a "dead" axis in a program with path joints, G5, G50 or HSC, it will be necessary to enable the axis via PLC as "dead" axis (DEAD(axis) signal). This way, the CNC knows at all times that it will have to apply a dwell before every movement of that axis.

RADIUS / DIAMETER.

DIAMPROG
Programming in diameters.
Parameter valid for a linear axes.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Yes / No.
Default value: No.
Associated variable: (V.)[ch].MPA.DIAMPROG.xn

The CNC takes this parameter into consideration only when FACEAXIS = Yes.

On turning machines, the coordinates of the cross axis may be programmed in either radius or diameter. To change the type of coordinates via program, use function G151 or G152.

HOME SEARCH.

REFDIREC
Homing direction.
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Negative / Positive.
Default value: Positive.
Associated variable: (V.)[ch].MPA.REFDIREC.xn

For analog and simulated axes, parameter REFDIREC indicates the axis moving direction to begin the home search. For a Sercos axis, parameter REFDIREC indicates the motor turning direction, which does not necessary have to be the same as the positive or negative feedback reading (pulse counting) direction of the axis.

This parameter has no effect on spindles that are stopped and have no home switch. If the spindle is turning, the CNC respects its turning direction to start searching home. If the spindle has a home switch and is turning, the CNC stops the spindle and starts the home search in the direction indicated by REFDIREC.

DECINPUT
The axis/spindle has a home switch.
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drives and Sercos drives.
Possible values: Yes / No.
Default value: Yes.
Associated variable: (V.)[ch].MPA.DECINPUT.xn

If the axis has a home search but no distance-coded reference marks, the axis moves up to the home switch, then it reverses the movement and when it releases the home switch, the home switch is enabled. If the axis has a home switch and distance-coded reference marks, the axis needs not reach the home switch in order to find the reference mark.

On spindles with home switch, the home search algorithm runs two passes over the home switch. The first pass is run at the speed set in REFFEED1 and it serves to calculate the position of the home switch. The second pass is run at the same speed until reaching the home switch, going over it a the speed set in REFFEED2 and doing the usual home search.

When there is a home switch, the CNC does not search home on the go, depending on whether it is turning in M3 or M4; the home search is always done in the direction set by REFDIREC.

REFINI
Home search in the first movement.
Parameter valid for spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Yes / No.
Default value: Yes.
Associated variable: (V.)[ch].MPA.REFINI.xn

This parameter determines whether the CNC homes the spindle in its first movement or not. The CNC takes this parameter into account only when parameters NPULSES y NPULSES2 have been set with a value other than 0.

CONFIGURATION OF THE PROBING MOVEMENT.

PROBEAXIS
The axis can participate in probing movements.
Parameter valid for linear and rotary axes.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Yes / No.
Default value: No.
Associated variable: (V.)[ch].MPA.PROBEAXIS.xn

This parameter indicates whether the axis is involved in a probing move (G100).

PROBERANGE
Maximum braking distance.

Parameter valid for linear and rotary axes.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 99999.9999 mm or degrees / from 0 to 3937.00787 inch.
Default value: 1.0000 mm or degrees / 0.03937 inch.
Associated variable: (V.)[ch].MPA.PROBERANGE.xn

This parameter sets the maximum braking distance for the probe after probing to avoid breaking it (ceramic, etc). The CNC issues an error messages when this distance is exceeded.

PROBEFEED
Maximum probing feedrate.

Parameter valid for linear and rotary axes.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 36000000.0000 mm/min or degrees/min / from 0 to 1417322.83465 inch/min.
Default value: 100.0000 mm/min or degrees/min / 3.93701 inch/min.
Associated variable: (V.)[ch].MPA.PROBEFEED.xn

The value of this parameter must be smaller than the feedrate needed to brake within the distance set by PROBERANGE with the acceleration and jerk values of the axis. Otherwise, it will show a warning when validating the axis parameters indicating the maximum feedrate that may be reached.

PROBEDELAY
Delay for the "probe 1" signal.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Within ±100000.0000 ms.
Default value: 0 (no delay).
Associated variable: (V.)[ch].MPA.PROBEDELAY.xn

See axis machine parameters PROBEDELAY2.

PROBEDELAY2
Delay for the "probe 2" signal.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Within ±100000.0000 ms.
Default value: 0 (no delay).
Associated variable: (V.)[ch].MPA.PROBEDELAY2.xn

Parameter PROBEDELAY corresponds to the probe set by PRBID1 and PROBEDELAY2 to the probe set by PRBID2.

In some types of probes, there is a short delay of a few milliseconds from the probing instant to when the CNC actually receives the signals (infrared communication, etc.). In these cases, it must indicate the time elapsed from when the probing takes place till the CNC receives the signal.

Probe calibration cycle "#PROBE 2" may be used to set this parameter. After it is executed, the cycle returns, in arithmetic parameter P298 the best value to be assigned to parameter PROBEDELAY for the axes and the spindle.

REPOSITIONING OF THE AXES IN TOOL INSPECTION.

REPOSFEED
Maximum repositioning feedrate.
Parameter valid for linear and rotary axes.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 200000.0000 mm/min or degrees/min / from 0 to 7873.992 inch/min.
Default value: 0.
Associated variable: (V.)[ch].MPA.REPOSFEED.xn

Repositioning feedrate after a tool inspection. If not defined, the CNC assumes as repositioning feedrate the one defined for the jog mode (JOGFEED).

The value of parameter REPOSFEED must always be lower than G00FEED, MAXMANFEED and JOGRAPFEED.

CONFIGURATION OF THE INDEPENDENT AXIS.

POSFEED
Positioning feedrate.
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 36000000.0000 mm/min or degrees/min / from 0 to 1417322.83465 inch/min.
Default value: 1000.
Associated variable: (V.)[ch].MPA.POSFEED.xn

Positioning feedrate of the independent axis.

CONFIGURE THE MAXIMUM SAFETY LIMIT FOR THE FEEDRATE AND FOR THE SPEED.

FLIMIT
Maximum safety limit for axis feedrate.
Parameter valid for linear and rotary axes.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 500000.0000 mm/min / from 0 to 19685.03937 inch/min / from 0 to 72000000.0000 degrees/min.
Default value: 0 (there is no limitation).
Associated variable: (V.)[ch].MPA.FLIMIT.xn

This parameter sets the maximum safety limit for axis feedrate to be applied, for example, when performing maintenance on the machine, operations with disabled safety (open doors), etc. The set value is activated via PLC (marks FLIMITAC or FLIMITACCHC1) and the CNC applies it on the block being executed. When deactivating the limits, the CNC restores the programmed feedrate.

If the parameter is set with a "0" value, the feedrate is not limited. The feedrate safety limit is applied to the movements in automatic (G0, G1, etc.) and in jog mode (jog, handwheels, etc.). This parameter affects neither the threading operations nor the independent axis movements, which are executed at the programmed feedrate.

When there is a maximum feedrate set by PLC (variable (V.)PLC.G00FEED), the CNC applies the most restrictive feedrate.

SLIMIT
Maximum safety limit for spindle speed.
Parameter valid for spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 200000.0000 rpm.
Default value: 0 (there is no limitation).
Associated variable: (V.)[ch].MPA.SLIMIT.sn

This parameter sets the maximum safety limit for spindle speed to be applied, for example, when performing maintenance on the machine, operations with disabled safety (open doors),

etc. The set value is activated via PLC (marks SLIMITAC or SLIMITACSPDL) and the CNC applies it on the block being executed. When deactivating the limits, the CNC restores the programmed speed.

If the parameter is set with a "0" value, the speed is not limited. The safety limit is also applied to the spindles controlled by PLC (PLCCNTL mark), except when the spindle is digital and controlled in position.

It is also possible to set the maximum machining turning speed by program (function G192) or by PLC (variable (V.)PLC.SL.sn). When a maximum machining speed has been set, the CNC applies the most restrictive speed limit; either the one set for machining or the safety one.

Safety speed limit.	Machining speed limit.	Active speed limit.
0	100	100
50	0	50
50	100	50
150	100	100

JOG MODE.

MANUAL
Manual operating mode parameters

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.

This parameter shows the parameter table to define the movements in jog mode. On spindles, these parameters will only be applied when the spindle is interpolated with axes during rigid tapping or when the spindle works as C axis.

JOGFEED	Feedrate in continuous jog mode.
JOGRAPFEED	Rapid feedrate in continuous jog mode.
MAXMANFEED	Maximum feedrate in continuous jog mode.
MAXMANACC	Maximum acceleration in continuous JOG mode.
INCJOGDIST	Distance to travel in incremental jog.
INCJOGFEED	Feedrate in incremental jog.
MPGRESOL	Handwheel resolution.
MPGFILTER	Filter time for the handwheel.
MANPOSSW	Maximum positive travel with G201.
MANNEGSW	Maximum negative travel with G201.
MANFEEDP	Maximum % of jog feedrate in G201.
IPOFEEDP	Maximum % of execution feedrate in G201.
MANACCP	Maximum % of jog acceleration in G201.
IPOACCP	Maximum % of execution acceleration in G201.

JOG MODE. CONTINUOUS JOG.

JOGFEED
Feedrate in continuous jog.
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 200000.0000 mm/min or degrees/min / from 0 to 7873.992 inch/min.
Default value: 1000.0000 mm/min or degrees/min / 39.37008 inch/min.
Associated variable: (V.)[ch].MPA.JOGFEED.xn
Parameter included in the JOG table.

This parameter sets the feedrate for movements in continuous jog mode.

JOGRAPFEED
Rapid feedrate in continuous jog.
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 200000.0000 mm/min or degrees/min / from 0 to 7873.992 inch/min.
Default value: 10000.0000 mm/min or degrees/min / 393.70079 inch/min.
Associated variable: (V.)[ch].MPA.JOGRAPFEED.xn
Parameter included in the JOG table.

This parameter sets the rapid feedrate for movements in continuous jog mode.

MAXMANFEED
Maximum feedrate in continuous jog.
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 200000.0000 mm/min or degrees/min / from 0 to 7873.992 inch/min.
Default value: 10000.0000 mm/min or degrees/min / 393.70079 inch/min.
Associated variable: (V.)[ch].MPA.MAXMANFEED.xn
Parameter included in the JOG table.

This parameter sets the maximum feedrate for movements in continuous jog mode.

MAXMANACC
Maximum acceleration in continuous JOG.
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 1.0000 to 1000000.0000 mm/s² or degrees/s² / from 0.03937 to 3937.00787 inch/s².
Default value: 1000.0000 mm/s² or degrees/s² / 39.37008 inch/s².
Associated variable: (V.)[ch].MPA.MAXMANACC.xn
Parameter included in the JOG table.

This parameter sets the maximum acceleration for movements in continuous jog mode.

JOG MODE. INCREMENTAL JOG.

INCJOGDIST
Table for incremental jog distances.
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.

This parameter shows the table to define the axis movements in each position of the jog selector switch. The table shows five parameters, one per each position of the operator panel.

INCJOGDIST1	Switch position ·1·.
INCJOGDIST2	Switch position ·10·.
INCJOGDIST3	Switch position ·100·.
INCJOGDIST4	Switch position ·1000·.
INCJOGDIST5	Switch position ·10000·.

INCJOGDIST
Distance traveled in incremental jog.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0.0001 to 99999.9999 mm or degrees / from 0 to 3937.00787 inch.
Associated variable: (V.)[ch].MPA.INCJOGDIST[pos].xn

In each parameter, you must set the distance the axis moves every time the jog key is pressed. The most typical values are those set by default.

Parameter.	Distance.
INCJOGDIST1	0.0010 mm or degrees / 0.00003937 inch.
INCJOGDIST2	0.0100 mm or degrees / 0.00039370 inch.
INCJOGDIST3	0.1000 mm or degrees / 0.00393700 inch.
INCJOGDIST4	1.0000 mm or degrees / 0.03937007 inch.
INCJOGDIST5	10.000 mm or degrees / 0.39370078 inch.

INCJOGFEED
Table for incremental jog feedrates.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.

This parameter shows the table to define the axis feedrate in each position of the jog selector switch. The table shows five parameters, one per each position of the operator panel.

INCJOGFEED1 Switch position ·1·.
INCJOGFEED2 Switch position ·10·.
INCJOGFEED3 Switch position ·100·.
INCJOGFEED4 Switch position ·1000·.
INCJOGFEED5 Switch position ·10000·.

INCJOGFEED n
Feedrate in incremental jog.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 200000.0000 mm/min or degrees/min / from 0 to 7873.992 inch/min.
Default value: 1000.0000 mm/min or degrees/min / 39.37008 inch/min.
Associated variable: (V.)[ch].MPA.INCJOGFEED[pos].xn

In each parameter, you must set the axis feedrate.

JOG MODE. HANDWHEELS.

MPGRESOL
Table for handwheel resolution.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.

This parameter shows the table to define the resolution of the handwheel in each position of the jog selector switch. The table shows three parameters, one per each position of the operator panel.

MPGRESOL1 Switch position ·1·.
MPGRESOL2 Switch position ·10·.
MPGRESOL3 Switch position ·100·.

MPGRESOL n
Handwheel resolution in each switch position.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Within ±99999.9999 mm or degrees / within ±3937.00787 inch (·0· not admitted).
Associated variable: (V.)[ch].MPA.MPGRESOL[pos].xn

In each parameter, you must set the distance the axis moves per feedback pulse provided by the handwheel. If the resolution is negative, the moving direction is reversed (movement in the opposite direction to that indicated by the A and B signals of the handwheel). The most typical values are those set by default.

Parameter.	Resolution.
MPGRESOL1	0.0010 mm or degrees / 0.00003937 inch.
MPGRESOL2	0.0100 mm or degrees / 0.00039370 inch.
MPGRESOL3	0.1000 mm or degrees / 0.00393700 inch.

Example:
The handwheel has a graduated disk with 100 lines and we would like a feed of 0.001 mm per line at switch position ·1·.

- A 100-line handwheel provides 1 pulse per line.
  MPGRESOL1 = 0.0010 mm.
- A 200-line handwheel provides 2 pulses per line.
  MPGRESOL1 = 0.0005 mm.
- A 25-line handwheel provides 1 pulse every 4 lines.
  MPGRESOL1 = 0.0040 mm.

MPGFILTER
Filter time for the handwheel.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 1 to 1000.
Default value: 10.
Associated variable: (V.)[ch].MPA.MPGFILTER.xn

This filter smoothes the handwheel movements avoiding sudden variations. This parameter indicates the number of CNC cycles used to are the handwheel pulses read.

JOG MODE. MANUAL INTERVENTION.

MANPOSSW
Maximum positive travel with G201.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Within ±99999.9999 mm or degrees / within ±3937.00787 inch.
Default value: 99999.9999 mm or degrees / 3937.00787 inch.
Associated variable: (V.)[ch].MPA.MANPOSSW.xn
Parameter included in the JOG table.

See axis machine parameters MANNEGSW.

MANNEGSW
Maximum negative travel with G201.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Within ±99999.9999 mm or degrees / within ±3937.00787 inch.
Default value: -99999.9999 mm or degrees / -3937.00787 inch.
Associated variable: (V.)[ch].MPA.MANNEGSW.xn
Parameter included in the JOG table.

When using function G201, Manual mode laid over the Automatic mode, these parameters indicate how far the axis may be moved in both directions.

MANFEEDP
Maximum % of jog feedrate in G201.
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 100.
Default value: 20.
Associated variable: (V.)[ch].MPA.MANFEEDP.xn
Parameter included in the JOG table.

See axis machine parameters IPOFEEDP.

IPOFEEDP
Maximum % of execution feedrate in G201.
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 100.
Default value: 80.
Associated variable: (V.)[ch].MPA.IPOFEEDP.xn
Parameter included in the JOG table.

When using function G201, Manual mode laid over the Automatic mode, both parameters indicate the maximum feedrate to be used in each mode. The sum of both parameters should not exceed 100 in order so as not to exceed the dynamic limits of the machine under certain conditions.

When activating function G201, the feedrate assumes instantaneously the value set by IPOFEEDP.

MANACCP
Maximum % of jog acceleration in G201.
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 100.
Default value: 20.
Associated variable: (V.)[ch].MPA.MANACCP.xn
Parameter included in the JOG table.

See axis machine parameters IPOACCP.

IPOACCP
Maximum % of execution acceleration in G201.
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 100.
Default value: 80.
Associated variable: (V.)[ch].MPA.IPOACCP.xn
Parameter included in the JOG table.

When using function G201, Manual mode laid over the Automatic mode, both parameters indicate the maximum acceleration to be used in each mode. The sum of both parameters should not exceed 100 in order so as not to exceed the dynamic limits of the machine under certain conditions.

When activating function G201, the feedrate assumes instantaneously the value set by IPOACCP.

Considering the following values for the Y axis:

G00FEED: 1000 mm/min.
JOGFEED: 100 mm/min.
MAXMANFEED: 120 mm/min.
IPOFEEDP: 50%
MANFEEDP: 50%

When executing the following blocks:

N10 G201 #AXIS [Y]
N20 G1 Y100 F1000

At block N20, the maximum execution feedrate of the Y axis is not 1000 mm/min (G00FEED), but 500 mm/min due to the 50% limitation of IPOFEED over G00FEED. Therefore, in spite of the programmed feedrate "F1000", the axis will move at 500 mm/min due to the limitation in G201.

If while the execution, the Yaxis is moved via JOG panel, a 100 mm/min feedrate (JOGFEED) should be added. However, the maximum jogging feedrate will be 60 min/min because it has been limited to 50% of MANFEEDP over MAXMANFEED.

Therefore, the Y axis will move at 560 mm/min when combining the automatic and jog modes.

LEADSCREW ERROR COMPENSATION.

LSCRWCOMP
Leadscrew error compensation.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Yes / No.
Default value: No.
Associated variable: (V.)[ch].MPA.LSCRWCOMP.xn

This parameter indicates whether the axis uses leadscrew error compensation or not.

LSCRWDATA
Leadscrew compensation table.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.

This parameter shows the table to define the leadscrew error compensation. The table shows the following parameters to configure it.

LSCRWDATA	
NPOINTS	Number of points in the table.
TYPLSCRW	Type of compensation.
BIDIR	Bi-directional compensation.
REFNEED	Home search required for compensation.
DATA	Table to define the compensation values.

NPOINTS
Number of points in the table.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 1000.
Default value: 0 (there is no table).
Associated variable: (V.)[ch].MPA.NPOINTS.xn
Parameter included in the LSCRWCOMP table.

The leadscrew error compensation table can have up to 1000 points.

TYPLSCRW
Type of compensation (type of coordinates).

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Real / Theoretical.
Default value: Real.
Associated variable: (V.)[ch].MPA.TYPLSCRW.xn
Parameter included in the LSCRWCOMP table.

This parameter determines whether the leadscrew error compensation will be applied on to theoretical or real coordinates.

BIDIR
Bi-directional compensation.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Yes / No.
Default value: No.
Associated variable: (V.)[ch].MPA.BIDIR.xn
Parameter included in the LSCRWCOMP table.

This parameter indicates whether the compensation is bidirectional or not; i.e. if the compensation is different in each direction. If the compensation is not bidirectional, it applies the same compensation in both directions.

REFNEED
Mandatory home search.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Yes / No.
Default value: No.
Associated variable: (V.)[ch].MPA.REFNEED.xn
Parameter included in the LSCRWCOMP table.

This parameter indicates whether the axis must be homed or not before applying the compensation.

DATA
Leadscrew error compensation at each point.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.

This parameter shows the list of compensation points and values. The CNC only lets access this table when parameter NPOINTS has been set.

The DATA table must indicate the amount of error to be compensated in specific positions of the axis. The number of points is set by parameter NPOINTS. Parameters POSITION, POSERROR and NEGERROR must be set for each point. Parameter NEGERROR only if the table has been defined with bidirectional compensation (BIDIR = YES).

LSCRWCOMP DATA	
POSITION	Axis position.
POSERROR	Amount of error to compensate when moving in the positive direction.
NEGERROR	Amount of error to compensate when moving in the negative direction.

POSITION
Position of each point.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Within ±99999.9999 mm or degrees / within ±3937.00787 inch.
Default value: 0.
Associated variable: (V.)[ch].MPA.POSITION[pt].xn
Parameter included in the LSCRWCOMP // DATA table.

Each parameter of the table represents a profile point to be compensated. The position that occupies the point in the profile will be referred to machine zero. When defining the various points in the table, the following requirements must be met.

- The points of the table must be ordered by their position on the axis and the table must begin by the most negative point (or least positive) to be compensated. For axis positioning outside this area, the CNC will apply the compensation that was defined for the nearest end.
- The maximum slope of the table is ·1·; i.e. the compensation steps must not exceed the position step between two consecutive points.

Import leadscrew compensation tables.

In order to save time and eliminate transcription errors, instead of entering the data manually, it is possible to adapt the format of a text file that contains the result of the measurement and import it later on.



See the operating manual for further information on how to import leadscrew error compensation tables.

POSERROR
Error in the positive direction.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Within ±99999.9999 mm or degrees / within ±3937.00787 inch.
Default value: 0.
Associated variable: (V.)[ch].MPA.POSERROR[pt].xn
Parameter included in the LSCRWCOMP // DATA table.

This parameter indicates the amount of error when the axis moves in the positive direction. If the table has not been defined with bi-directional compensation, this amount of error will also be applied on movements in the negative direction.

NEGERROR
Error in the negative direction.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Within ±99999.9999 mm or degrees / within ±3937.00787 inch.
Default value: 0.
Associated variable: (V.)[ch].MPA.NEGERROR[pt].xn
Parameter included in the LSCRWCOMP // DATA table.

This parameter indicates the amount of error when the axis moves in the negative direction. This parameter is only available if the table has been defined with bidirectional compensation.

FILTERS TO ELIMINATE RESONANCE FREQUENCY.

FILTER
Filter table.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.

This parameter shows the tables to configure the frequency filters. Up to 3 different filters may be defined for each axis or spindle, hence being possible to eliminate several resonance frequencies.

Frequency filters may be applied to axes and spindles. The filters defined for the spindle will only be applied when it works as a C axis or it is doing a rigid tapping.

The CNC offers several filters. The "anti-resonance" filter (notch filter) that limits a zone of the passing band and the low-pass filters that eliminate the banding passing from a particular frequency on.

One of them is usually used, although both types of filters can also be applied to the same axis or spindle when the resonance frequency is within the bandwidth of a "low passing" filter.

In order to obtain a good part finish, it is recommended to set all the axes that interpolate with each other with the same filter type and with the same frequency.

FILTER n
Filter configuration.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.

Each table has the following machine parameters to configure it.

FILTER	
ORDER	Filter order.
TYPE	Filter type.
FREQUENCY	Cut-off or center frequency.
NORBWIDTH	Normal bandwidth.
SHARE	Percentage of signal going through the filter.

ORDER
Filter order.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 10 for low passing filter / from 0 to 5 for anti-resonance filter / from 0 to 50 for FAGOR low passing filter.
Default value: 0 (No filter is applied).
Associated variable: (V.)[ch].MPA.ORDER[nb].xn
Parameter included in the FILTER table.

The down ramp is dampened down; the higher the order the greater the drop. A higher order implies more calculation; therefore, we recommend to consult our Technical support department before modifying this parameter.

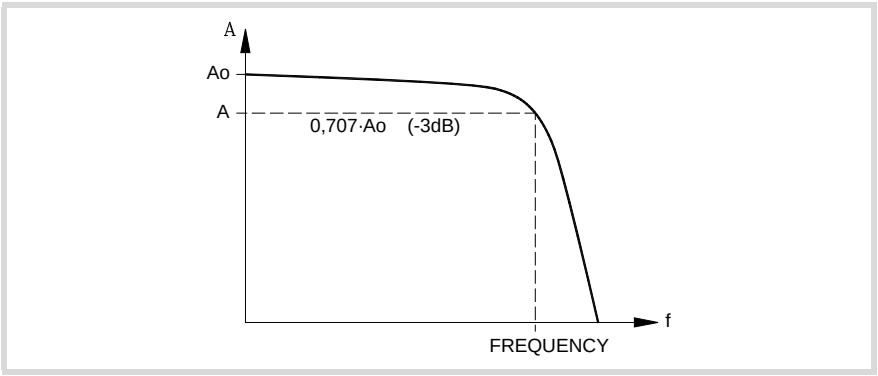
TYPE
Filter type.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Low Passing / Anti-resonance (notch filter) / Low passing FAGOR.
Default value: Low Passing.
Associated variable: (V.)[ch].MPA.TYPE[nb].xn
Parameter included in the FILTER table.

Usually, rigid and robust machines have a passing band of up to 30 Hz; i.e. they are capable of responding up to this frequency. The rest of the machines may have resonance at lower frequencies (10 Hz or less) that are generated on usual paths.

"Low passing" filter.

If the machine shows the resonance frequency at the end of its passing band, it will be enough to eliminate it using a "low passing" filter.



These filters are also used to limit a jerk whose parameter has been set too high for the machine but necessary to make it run faster. This way, the CNC smoothes the movements out although it has the drawback of slightly rounding the corners off.

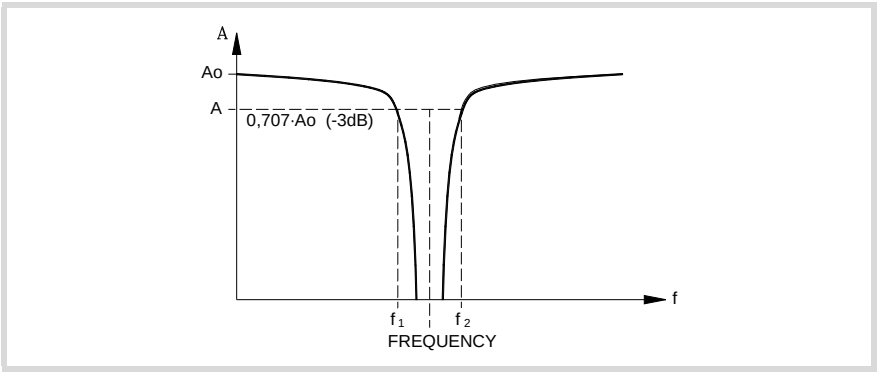
- The "low passing" filter, for the same frequency requires a lower order and consecuently adds a shorter delay.

  This type of filter inserts a variable (non constant) phase shift depending on the frequencies. This phase shift can cause a change on the path if it is not executed at the same feedrate as, for example, when changing the feedrate override percentage or when moving back and forth on the same path.

- The "FAGOR low passing" filter inserts a constant phase shift regardless of the frequency. This type of filter needs a higher order to apply the same amount of filtering.

Anti-resonance filter (band-rejection, notch filter).

If the machine shows the resonance frequency in the middle of its passing band, it will be a good idea to eliminate it using a "notch" filter.



FREQUENCY
Cut-off or center frequency.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: from0 to 500.0 Hz.
Default value: 30.0 Hz.
Associated variable: (V.)[ch].MPA.FREQUENCY[nb].xn
Parameter included in the FILTER table.

On "low passing" filters, this parameter indicates the break point frequency or frequency where the amplitude drops 3 dB or it reaches 70% of the nominal amplitude.

-3 dB = 20 log (A/Ao) ==> A = 0,707 Ao

On the band-rejection filter (notch filter), this parameter indicates the center frequency or the frequency where the resonance reaches its maximum value.

NORBWIDTH
Normal bandwidth.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 100.0
Default value: 1.0
Associated variable: (V.)[ch].MPA.NORWIDTH[nb].xn
Parameter included in the FILTER table.

This parameter is calculated with the following formula. The f1 and f2 values correspond to the break frequency where the amplitude drops 3 dB or it reaches 70% of the nominal amplitude.

-3 dB = 20 log (A/Ao) ==> A = 0,707 Ao

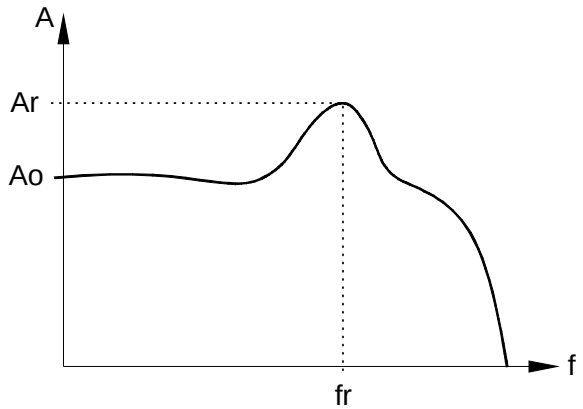
$$\text{NORBWIDTH} = \frac{\text{FREQUENCY}}{(f_2 - f_1)}$$

SHARE
Percentage of signal going through the filter.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 100.
Default value: 100.
Associated variable: (V.)[ch].MPA.SHARE[nb].xn
Parameter included in the FILTER table.

This parameter indicates the percentage of signal going through the filter. This value must be equivalent to the percentual overshooting of the resonance because it must compensate for it.

Example of a calculation for a particular response of the machine.



SHARE=100(Ar-Ao)/Ar

WORK SETS.

NPARSETS
Number of parameter sets available.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 1 to 4.
Default value: 1.
Associated variable: (V.)[ch].MPA.NPARSETS.xn

Up to 4 different parameter sets may be defined to indicate the dynamics of the axis in each one of them (feedrates, gains, accelerations, etc.).

DEFAULTSET
Set of default parameters on power-up.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 4.
Default value: 1.
Associated variable: (V.)[ch].MPA.DEFAULTSET.xn

This parameter indicates the parameter set assumed by the CNC on power-up, after executing an M02 or M30 or after a reset. When defined with a "0" value, the parameter set is always maintained.

To select a parameter set on the axis or spindle via part-program, use function G112. To select a parameter set on a spindle and make a gear change, use functions M41 through M44.

SET n
Work sets.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.

It shows the machine parameter table of the set.

2.6 Machine parameters for the axes. Work sets.

Up to 4 different parameter sets may be defined to indicate the dynamics of the axis in each one of them (feedrates, gains, accelerations, etc.). Each set only shows the parameters for the selected type of axis and drive.

FEEDBACK RESOLUTION.

PITCH
Leadscrew pitch.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 99999.9999 mm or degrees / from 0 to 3937.00787 inch.
Default value: 5 mm / 0.19685 inch / 360º.
Associated variable: (V.)[ch].MPA.PITCH[set].xn

The meaning of this parameter depends on the type of feedback.

- On a linear axis with a rotary encoder and leadscrew, the parameter defines the leadscrew pitch.
- On a linear axis with a linear encoder (scale), the parameter defines the pitch of the scale.
- On a rotary axis, the parameter sets the number of degrees per turn of the encoder.

Type of axis.	PITCH
Axis with a 5 mm pitch leadscrew.	5 mm.
Axis with a 20 µm pitch Fagor scale.	0.020 mm.
Rotary axis with a 1/10 gear ratio.	36º.

INPUTREV
Turns of the motor shaft

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 1 to 32767.
Default value: 1.
Associated variable: (V.)[ch].MPA.INPUTREV[set].xn

See machine parameter OUTPUTREV.

OUTPUTREV
Turns of machine axis.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 1 to 32767.
Default value: 1.
Associated variable: (V.)[ch].MPA.OUTPUTREV[set].xn

The possible gear ratio between the motor and the encoder may also be entered directly through parameter PITCH; in that case, parameters INPUTREV and OUTPUTREV must be set to ·1· If the gear ratio is not an integer multiple, we recomend to set the real values of the gear ratio in INPUTREV and OUTPUTREV instead of leaving a inaccurate number in PITCH.

NPULSES
Number of encoder pulses.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drives, velocity-Sercos drives and simulated drives.
Possible values: From 0 to 1000000.
Default value: 1250.
Associated variable: (V.)[ch].MPA.NPULSES[set].xn

This parameter sets the number of degrees per turn of the encoder. When using a gear reduction on the axis, the whole assembly must be taken into account when defining the number of pulses per turn. With linear encoders (scales), set NPULSES = 0

If parameters NPULSES and NPULSES2 are set with a value other than ·0·, parameter REFINI determines whether the CNC homes the spindle on its first movement or not.

PITCH2
Leadscrew pitch (external feedback).

Parameter valid for linear and rotary axes and spindles.
Parameter valid for a Sercos drive.
Possible values: From 0 to 99999.9999 mm or degrees / from 0 to 3937.00787 inch.
Default value: 5 mm / 0.19685 inch / 360º.
Associated variable: (V.)[ch].MPA.PITCH2[set].xn

The meaning of this parameter depends on the type of feedback.

- On a linear axis with a rotary encoder and leadscrew, the parameter defines the leadscrew pitch.
- On a linear axis with a linear encoder (scale), the parameter defines the pitch of the scale.
- On a rotary axis, the parameter sets the number of degrees per turn of the encoder.

Type of axis.	PITCH
Axis with a 5 mm pitch leadscrew.	5 mm.
Axis with a 20 µm pitch Fagor scale.	0.020 mm.
Rotary axis with a 1/10 gear ratio.	36º.

INPUTREV2
Turns of the motor shaft (external feedback)

Parameter valid for linear and rotary axes and spindles.
Parameter valid for a Sercos drive.
Possible values: From 1 to 32767.
Default value: 1.
Associated variable: (V.)[ch].MPA.INPUTREV2[set].xn

See machine parameter OUTPUTREV2.

OUTPUTREV2
Turns of machine axis (external feedback).

Parameter valid for linear and rotary axes and spindles.
Parameter valid for a Sercos drive.
Possible values: From 1 to 32767.
Default value: 1.
Associated variable: (V.)[ch].MPA.OUTPUTREV2[set].xn

Parameters INPUTREV2 and OUTPUTREV2 set the gear ratio for the second feedback.

NPULSES2
Number of pulses of the encoder (external feedback).

Parameter valid for linear and rotary axes and spindles.
Parameter valid for a Sercos drive.
Possible values: From 0 to 1000000.
Default value: 1250.
Associated variable: (V.)[ch].MPA.NPULSES2[set].xn

This parameter sets the number of degrees per turn of the encoder for the second feedback. When using a gear reduction on the axis, the whole assembly must be taken into account

when defining the number of pulses per turn. With linear encoders (scales), set NPULSES2 = 0.

If parameters NPULSES and NPULSES2 are set with a value other than ·0·, parameter REFINI determines whether the CNC homes the spindle on its first movement or not.

SINMAGNI
Multiplying factor for sinusoidal feedback signal.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drives and simulated drives.
Possible values: From 0 to 255.
Default value: 0.
Associated variable: (V.)[ch].MPA.SINMAGNI[set].xn

This parameter indicates the multiplying factor applied to the sinewave feedback of the axis. For square feedback signals, set SINMAGNI = 0; the CNC always applies a x4 factor.

Parameters SINMAGNI, PITCH and NPULSES set the axis feedback resolution as shown in the table below.

Type of encoder.	PITCH	NPULSES	SINMAGNI
Rotary axis. Rotary encoder. Square signals.	Rotation of the axis per turn of the encoder	Number of pulses.	0
Rotary axis. Rotary encoder. Sinusoidal signals.	Rotation of the axis per turn of the encoder	Number of pulses.	Multiplying factor.
Linear axis. Rotary encoder. Square signals.	Leadscrew pitch.	Number of pulses.	0
Linear axis. Rotary encoder. Sinusoidal signals.	Leadscrew pitch.	Number of pulses.	Multiplying factor.
Linear encoder. Square signals.	Pitch of encoder.	0	0
Linear encoder. Sinusoidal signals.	Pitch of encoder.	0	Multiplying factor.

ABSFEEDBACK
Absolute feedback system.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Yes / No.
Default value: No.
Associated variable: (V.)[ch].MPA.ABSFEEDBACK[set].xn

This parameter indicates whether the axis uses absolute feedback or not. With this system, the axis is considered to be homed right on power-up and no movement is generated when programming a home search. The PLC mark REFPOIN is always active.

FBACKAL
Activate feedback alarm.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for an analog drive.
Possible values: Yes / No.
Default value: Yes.
Associated variable: (V.)[ch].MPA.FBACKAL[set].xn

This parameter may be used to activate the feedback alarm. This alarm can tell the difference between the feedback cables being disconnected or broken for differential TTL signals and sinusoidal signals.

When a feedback alarm occurs, the PLC turns the REFPOIN mark off.

If this alarm is canceled, the CNC shows on power-up a message indicating that this safety is disabled. This situation can only be allowed during setup; once setup is completed, this alarm must be enabled.

LOOP SETTING.

LOOPCH
Velocity command sign change.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Yes / No.
Default value: No.
Associated variable: (V.)[ch].MPA.LOOPCH[set].xn

See machine parameter AXISCH.

AXISCH
Feedback sign change.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Yes / No.
Default value: No.
Associated variable: (V.)[ch].MPA.AXISCH[set].xn

If the axis runs away, the CNC issues a following error message. Change the value of parameter LOOPCH. If it does not run away, but the counting direction is not the desired one, change the values of both parameters AXISCH and LOOPCH.

INPOSW
In-position zone.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0.0001 to 99999.9999 mm or degrees / from 0.00000 to 3937.00787 inch.
Default value: 0.0100 mm or degrees / 0.00039 inch.
Associated variable: (V.)[ch].MPA.INPOSW[set].xn

The in-position zone is defined as the zone before and after the programmed position where the axis is considered to be in position. Parameter INPOSW defines the width of both zones.

BACKLASH COMPENSATION IN MOVEMENT REVERSAL.

BACKLASH
Amount of backlash to be compensated.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Within ±3.2768 mm or degrees / within ±0.12901 inch.
Default value: 0.
Associated variable: (V.)[ch].MPA.BACKLASH[set].xn

With linear encoders (scales), set BACKLASH = 0.

When an axis has backlash and reverses its moving direction, there is a delay from the instant the motor starts turning to the moment when the axis actually moves. This usually happens on axes with encoder and on old machines whose leadscrew assembly is defective (worn out).

Use a dial indicator to measure this backlash. Move the axis in a direction and set the dial indicator to "0". Move the axis in the opposite direction in incremental mode until detecting that the axis is moving. The amount of backlash is the difference between the commanded distance and what it actually moved.

BACKLASH COMPENSATION IN MOVEMENT REVERSALS WITH AN ADDITIONAL COMMAND PULSE.

BAKANOUT
Additional velocity command pulse.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drives and Sercos drives.
Possible values: Within ±32767 with an analog drive / within ±1000 rpm with a Sercos drive.
Default value: 0 (Not applied).
Associated variable: (V.)[ch].MPA.BAKANOUT[set].xn

Additional velocity command pulse to make up for the possible leadscrew backlash when reversing the moving direction. Every time the movement is inverted, the CNC will apply to that axis the velocity command corresponding to the movement plus the additional velocity command pulse set in this parameter. The duration of this additional velocity command depends on parameters BAKTIME and PEAKDISP.

With an analog drive, the additional velocity command is given in units of the D/A converter with an integer within ± 32767. - 10 V for a value of -32767 and +10V for a value of 32767.

BAKANOUT	1	3277	32767
Velocity command.	0.3 mV	1 V	10 V

On analog axes, when LOOPCH=Yes, BAKANOUT must be set with a negative value so the CNC applies the right reversal peak for movement reversal.

When setting an additional command pulse, parameters BAKTIME, ACTBAKAN and PEAKDISP must also be set.

BAKTIME
Duration of the additional velocity command pulse.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drives and Sercos drives.
Possible values: From 0 to 100 ms.
Default value: 0.
Associated variable: (V.)[ch].MPA.BAKTIME[set].xn

The CNC only takes this parameter into consideration when working with an additional velocity command pulse; BAKANOUT other than zero. Parameter BAKTIME indicates the duration of the additional velocity command pulse to make up for backlash in movement reversals.

ACTBAKAN
Application of the additional velocity command pulse.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drives and Sercos drives.
Possible values: Always / G2-G3.
Default value: Always.
Associated variable: (V.)[ch].MPA.ACTBAKAN[set].xn

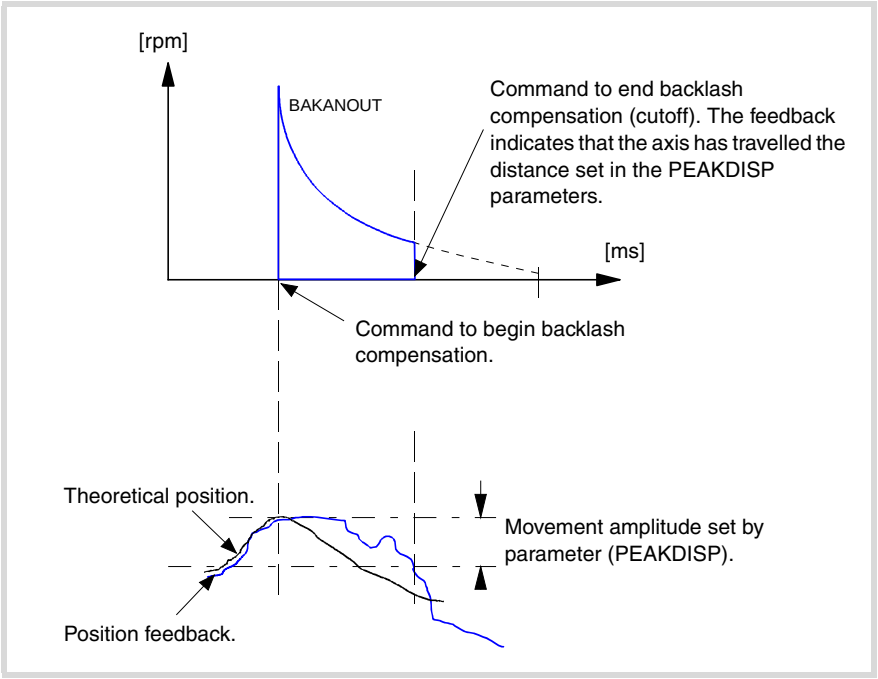
The CNC only takes this parameter into consideration when working with an additional velocity command pulse; BAKANOUT other than zero. Parameter ACTBAKAN determines when the additional command pulse is applied to compensate for backlash peaks.

PEAKDISP
Backlash peak cuttoff distance.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drives and Sercos drives.
Possible values: From 0 to 99999.9999 mm or degrees / from 0 to 3937.00787 inch.
Default value: 0.0050 mm or degrees / 0.00508 mm.
Associated variable: (V.)[ch].MPA.PEAKDISP[set].xn

The CNC only takes this parameter into consideration when working with an additional velocity command pulse; BAKANOUT other than zero. Parameter PEAKDISP sets the real

distance traveled by the axis after the theoretical movement reversal, where the CNC cuts the reversal peak off on that axis (additional command pulse).



REVEHYST
Hysteresis for applying the additional command pulse in movement reversals.

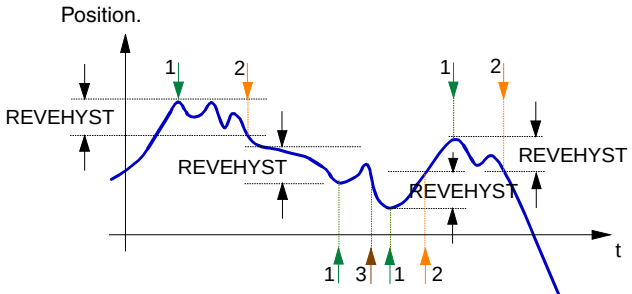
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drives and Sercos drives.
Possible values: From 0 to 99999.9999 mm or degrees / from 0 to 3937.00787 inch.
Default value: 0.0000 mm, degrees or inches.
Associated variable: (V.)[ch].MPA.REVEHYST[set].xn

This parameter limits the use of backlash compensation when the CNC detects a movement reversal so it is not applied every time the CNC receives a reversal command.

This parameter sets the value that the axis position must vary after the first movement reversal (hysteresis) for the CNC to consider applying backlash compensation. If the axis does not exceed this margin, the CNC does not apply backlash compensation.

Considerations

- If REVEHYST=0, Backlash compensation by reversal peak will always be applied on each reversal.
- If REVEHYST other than 0, and parameter PEAKDISP is used to cut the backlash peak off, we recommend to set REVEHYST to a lower value than that of PEAKDISP so the CNC applies the backlash peak.
- When having axes set as DRO axes, the value of the BACKLASH parameter will be taken into consideration. In these cases, especially when using sinusoidal feedback, we recommend to set REVEHYST to a value other than 0 in order to apply backlash.



1 ► Reversal of the position command.
2 ► Limit set in REVEHYST. Beginning of backlash compensation.
3 ► Cancellation of backlash compensation.

If REVEHYST= 0.0005 mm, the CNC will not activate the compensation in reversals occurring after the first one as long as the axis position does not change (vary) at least in the amount set by this parameter, taking as reference the position it occupied in the first position command reversing order.

In other words, if the CNC receives a command (order) to reverse when the position command has only changed 0.0002 mm from the position where the first reversal command took place, the CNC does not apply the compensation (the value set in REVEHYST has not been exceeded) and it simply reverses the moving direction.

Only when the position command variation reaches 0.0005 mm, the CNC will apply the compensation. After applying a compensation, the CNC will take the new order to reverse as a new reference to evaluate the position variation and determine when the value set in parameter REVEHYST has been reached again and start compensating again.

ADJUSTMENT OF RAPID TRAVERSE G00 AND MAXIMUM SPEED.

G00FEED
Feedrate in G00.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 500000.0000 mm/min / from 0 to 36000000 degrees/min / from 0 to 19685.03937 inch/min / from 0 to 100000.0000 rpm.
Default value: 10000.0000 mm/min / 1080000 degrees/min / 393.70079 inch/min / 3000.0000 rpm.
Associated variable: (V.)[ch].MPA.G00FEED[set].xn

Rapid positioning (traverse, G00) are always carried out at the maximum speed possible. The one indicated by G00FEED.

How to limit temporarily the maximum feedrate via PLC

The PLC has the variable (V.)[ch].PLC.G00FEED that may be used to limit the feedrate in the channel for any type of movement (G00, G01, etc). This variable limits the feedrate of the path and affects all the axes whether they move interpolated or one axis at a time.

The CNC assumes the change immediately and the change stays active until the variable takes a value of ·0· restoring the limits set by machine parameters.

MAXFEED
Maximum axis machining feedrate.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 500000.0000 mm/min / from 0 to 500000.0000 degrees/min / from 0 to 19685.03937 inch/min / from 0 to 1388.8889 rpm.
Default value: 0.
Associated variable: (V.)[ch].MPA.MAXFEED[set].xn

This parameter sets the maximum machining feedrate (movements in G01/G02/G03) of the axis; when set to ·0·, the feedrate is not limited This parameter cannot be set with a value higher than that of parameter G00FEED.

When trying to exceed the maximum feedrate via part-program, via PLC or from the operator panel, the CNC limits the feedrate to the maximum value set in MAXFEED without showing any error message or warning.

If the machining feedrate is not limited, the CNC assumes for all the movements the one set in machine parameter G00FEED as the maximum feedrate.

Variable.	Machine parameters.		Feedrate.	
(V.)PLC.G00FEED	G00FEED (axis)	MAXFEED (axis)	G00	G01, G02, ···
0	10000	0	10000	10000
0	10000	6000	10000	6000
4000	10000	6000	4000	4000
7000	10000	6000	7000	6000
12000	10000	6000	10000	6000

MAXVOLT
Velocity command to reach G00FEED.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drives and velocity-Sercos drives.
Possible values: from0 to 10000.0000 mV.
Default value: 9500 mV (9.5 V).
Associated variable: (V.)[ch].MPA.MAXVOLT[set].xn

This is the velocity command the CNC must output so the axis can reach its maximum rapid traverse feedrate G00FEED.

MAXFREQ
Frequency required to reach G00FEED.

Parameter valid for spindles.
Parameter valid for a Mechatrolink inverter.
Possible values: From 0,0010.0000 to 10000.0000 Hz.
Default value: 50 Hz.
Associated variable: (V.)[ch].MPA.MAXFREQ[set].xn

When the drive is an inverter, this parameter indicates the frequency to be provided by the CNC for the spindle to reach the speed set in parameter G00FEED.

MAXRPM
Maximum motor rpm.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for a Mechatrolink speed and inverter.
Possible values: From 0 to 100000 rpm.
Default value: 3000 rpm.
Associated variable: (V.)[ch].MPA.MAXRPM[set].xn

Maximum motor rpm.

RAPID TRAVERSE FOR THE AUTOMATIC MODE.

FRAPIDEN
Rapid traverse of the axis for movements in automatic mode.

Possible values: From 0 to 500000.0000 mm/min / from 0 to 19685.03937 inch/min.
Default value: 0.
Associated variable: (V.)[ch].MPA.FRAPIDEN[set].xn

Maximum feedrate of the axis for the program being executed, when the rapid traverse for the automatic mode is active (parameter RAPIDEN). If the parameter is set with a "0" value, the feedrate is not limited. The value of this parameter must always be lower than that of axis parameter G00FEED.

This parameter does not the movements programmed in G00 or the threads. Movements in G0 are carried out at the feedrate set in parameter G00FEED. The threads are executed at the programmed feedrate.

The rapid traverse cannot exceed the value set in axis parameters G00FEED or FRAPIDEN of the channel or the maximum feedrate set by PLC (variable (V.)PLC.G00FEED). Rapid traverse cannot exceed the value set in axis parameter MAXFEED of the channel and the active feedrate set by PLC (variable (V.)PLC.F).

Variable.	**Machine parameters.**		**Rapid feed.**	
(V.)PLC.G00FEED	G00FEED (axis)	FRAPIDEN (axis)	G00	G01, G02, ···
0	10000	0	10000	10000
0	10000	6000	10000	6000
4000	10000	6000	4000	4000
7000	10000	6000	7000	6000
12000	10000	6000	10000	6000

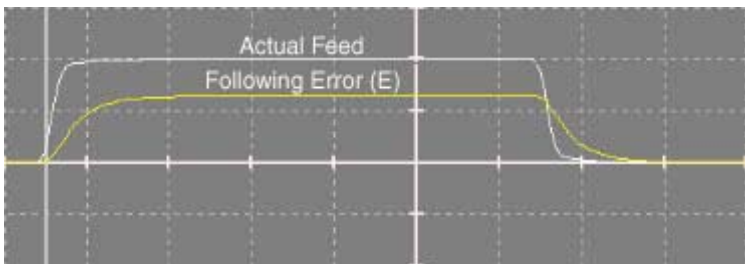
GAIN SETTING.

PROGAIN
Proportional gain.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0.0 to 100.0 (1000/min).
Default value: 1.
Associated variable: (V.)[ch].MPA.PROGAIN[set].xn

This parameter sets the following error (difference between the theoretical instantaneous position and the actual - real - axis position) for a particular feedrate.

The scale for the following error of 800 µm per square.



Example:
To obtain a following error (E) of 1 mm for a feedrate (F) of 1000 mm/min. (a gain of 1).
F = E x PROGAIN
F / E = 1000 (mm/min) / 1 (mm) = 1000 / min
PROGAIN = 1

FFWTYPE
Pre-control type

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: OFF / Feed Forward / AC-Forward / Feed Forward + AC-Forward.
Default value: OFF
Associated variable: (V.)[ch].MPA.FFWTYPE[set].xn

This parameter indicates the type of pre-control for adjusting the gains.

In order to be able to modify the override while threading (parameter THREADOVR), the feed forward must be active and higher than 90%. Feed forward may be defined with a lower value than 90% in parameter FFGAIN and may be modified later on from the PLC using variables.

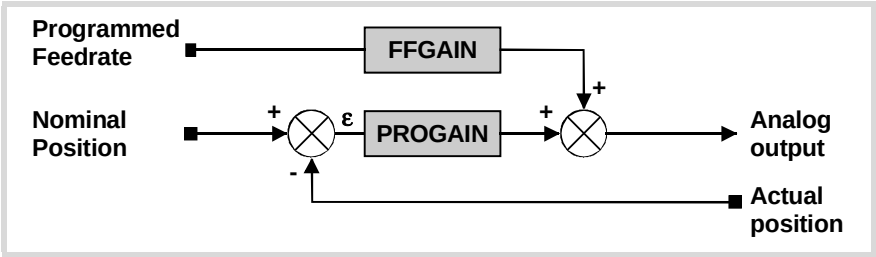
FFGAIN
Percentage of Feed-Forward in automatic.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: from0 to 120 %.
Default value: 0.
Associated variable: (V.)[ch].MPA.FFGAIN[set].xn

The CNC only takes this parameter into account when working with feed forward; parameter FFWTYPE if it is an analog or simulated drive and OPMODEP if it is a Sercos drive. This parameter should only be used when working with non-linear acceleration and deceleration.

An axis that is going to work as an independent axis should have the same feed forward value in automatic and in jog modes; in other words, parameters FFGAIN and MANFFGAIN should be the same.

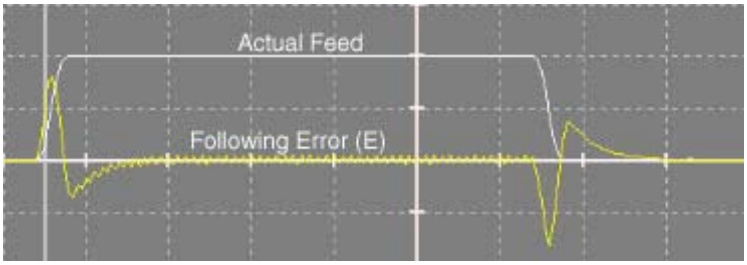
This parameter helps improve the position loop minimizing the amount of following error "ε". This parameter sets the portion of the velocity command is proportional to the programmed feedrate and part of it is proportional to the following error "ε".



The best adjustment is obtained when minimizing the amount of following error as much as possible without reversing its sign, maintaining the moving direction of the axis.

The scale for the following error of 10 µm per square.

- Proper adjustment with Feed forward.



- Wrong adjustment with Feed forward.



MANFFGAIN
Percentage of Feed-Forward in JOG.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drives and simulated drives.
Possible values: from0 to 120 %.
Default value: 0.
Associated variable: (V.)[ch].MPA.MANFFGAIN[set].xn

The CNC only takes this parameter into account when working with feed forward; parameter FFWTYPE if it is an analog or simulated drive and OPMODEP if it is a Sercos drive.

Sometimes, the Feed Forward selected for the automatic mode may be too high for the Jog mode, when there is no need to control the following error that closely. In those cases, parameter MANFFGAIN helps adapt the Feed-Forward applied to the jog mode.

An axis that is going to work as an independent axis should have the same feed forward value in automatic and in jog modes; in other words, parameters FFGAIN and MANFFGAIN should be the same.

ACFWFACTOR
Acceleration time constant.

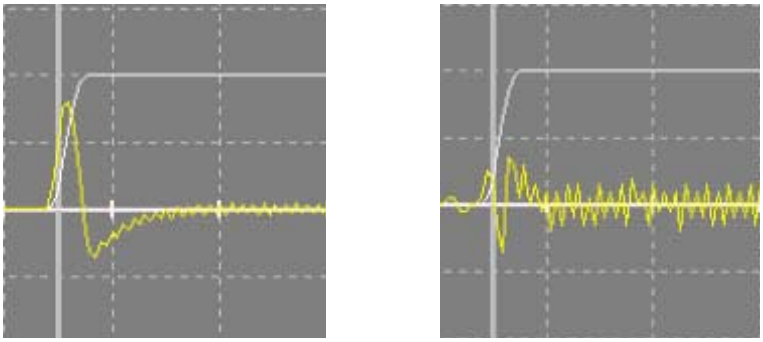
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drives, velocity-Sercos drives and simulated drives.
Possible values: From 0.001 to 1000000.0000 ms.
Default value: 1000.0000 ms.
Associated variable: (V.)[ch].MPA.ACFWFACTOR[set].xn

The CNC only takes this parameter into account when working with AC-forward; parameter FFWTYPE if it is an analog or simulated drive and OPMODEP if it is a velocity-Sercos drive.

It is recommended to assign to this parameter a value close to the system's response time. Since the system response time is usually an unknown value that depends on the inertia of the machine and on the drive adjustment, it is recommended to try with several values.

The best adjustment is achieved when minimizing the following error as much as possible but without inverting the peaks. The peaks of the right graph are inverted. Bad adjustment.

The graph on the left shows the system response without AC-Forward gain (10 µm per square) and the one on the right with AC-Forward (1 µm per square).



ACFGAIN
Percentage of AC-Forward in automatic.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: from0 to 120 %.
Default value: 0.
Associated variable: (V.)[ch].MPA.ACFGAIN[set].xn

See machine parameter MANACFGAIN.

MANACFGAIN
Percentage of AC-Forward in jog mode.

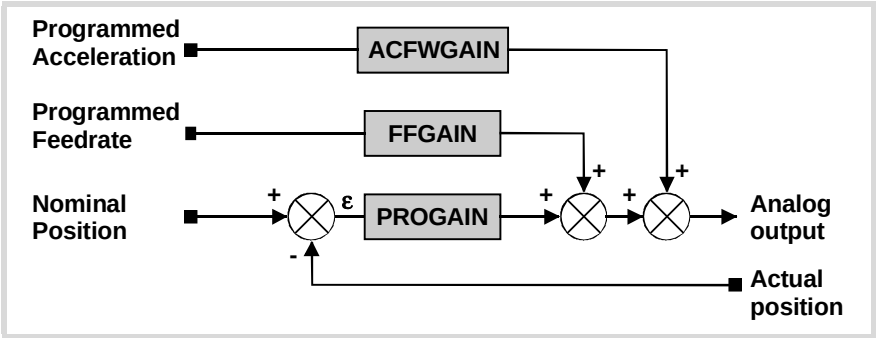
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drives and simulated drives.
Possible values: from0 to 120 %.
Default value: 0.
Associated variable: (V.)[ch].MPA.MANACFGAIN[set].xn

The CNC takes this parameter into account only when working with AC-Forward; parameter ACFWFACTOR.

An axis that is going to work as an independent axis should have the same AC-forward value in automatic and in jog modes; in other words, parameters ACFGAIN and MANACFGAIN should be the same.

These parameters are similar to parameters FFGAIN and MANFFGAIN; but they affect the AC-Forward. Both parameters improve system response in acceleration changes and

minimize the amount of following error "ε" when starting up, braking and reversing the moving direction.



LINEAR ACCELERATION.

LACC1
Acceleration of the first section.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 1.0000 to 1000000.0000 mm/s² or degrees/s² / from 0.03937 to 3937.00787 inch/s².
Default value: 1000.0000 mm/s² or degrees/s² / 39.37008 inch/s².
Associated variable: (V.)[ch].MPA.LACC1[set].xn

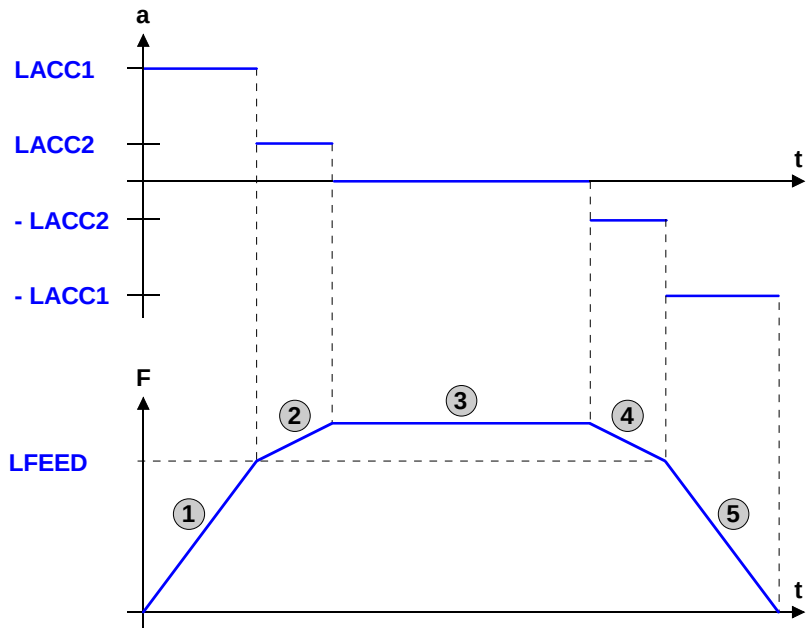
See parameter LACC2.

LACC2
Acceleration of the second section.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 1.0000 to 1000000.0000 mm/s² or degrees/s² / from 0.03937 to 3937.00787 inch/s².
Default value: 1000.0000 mm/s² or degrees/s² / 39.37008 inch/s².
Associated variable: (V.)[ch].MPA.LACC2[set].xn

Parameters LACC1 and LACC2 set the acceleration values when it is linear (parameter SLOPETYPE) or when working with HSC in FAST mode. The CNC applies parameters LACC1 and LACC2 to the movements in G1, G2 and G3; movements in G0 depend on parameter G0ACDCJERK.

G0ACDCJERK	Meaning.
Yes.	Movements in G0 use parameters LACC1G0, LACC2G0 and LFEEDG0.
No.	Movements in G0 use parameters LACC1, LACC2 and LFEED.



1 Acceleration according to parameter LACC1 until reaching the feedrate indicated in LFEED.

2 Acceleration according to parameter LACC2 until reaching the programmed feedrate.

3 Movement at programmed feedrate (without acceleration).

4 Deceleration according to parameter LACC2 until reaching the feedrate indicated in LFEED.

5 Deceleration according to parameter LACC1.

LFEED
Acceleration changing speed.
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 200000.0000 mm/min or degrees/min / from 0 to 7873.992 inch/min / from 0 to 100000.0000 rpm.
Default value: 1000.0000 mm/min, degrees/min or rpm / 39.37008 inch/min.
Associated variable: (V.)[ch].MPA.LFEED[set].xn

While accelerating, when reaching the feedrate defined in this parameter, it changes the acceleration from LACC1 to LACC2. While decelerating, when reaching the feedrate defined in this parameter, it changes the acceleration from LACC2 to LACC1.

TRAPEZOIDAL AND SQUARE SINE ACCELERATION.

ACCEL
Acceleration.
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0.001 to 100000.0000 mm/s² or degrees/s² / from 0.0003937 to 3937.00787 inch/s².
Default value: 1000.0000 mm/s² or degrees/s² / 39.37008 inch/s².
Associated variable: (V.)[ch].MPA.ACCEL[set].xn

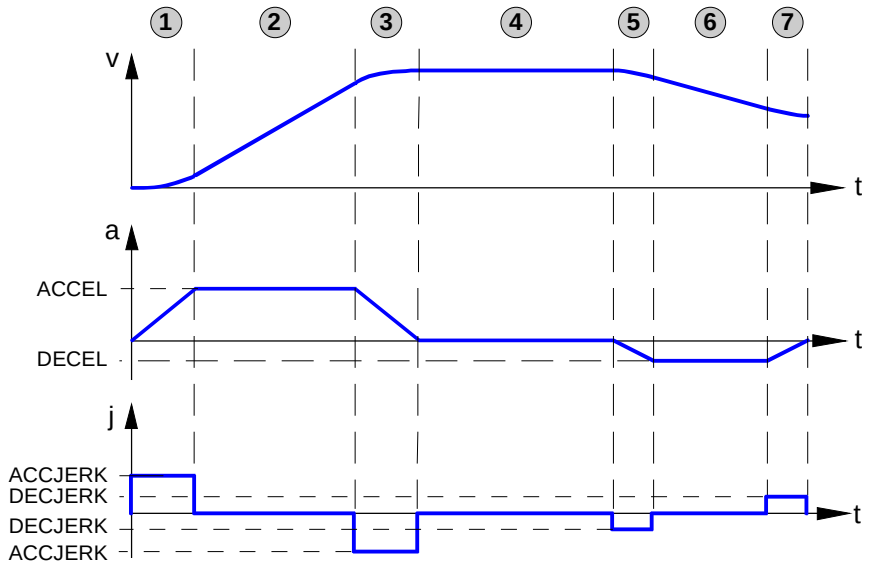
See machine parameter DECEL.

DECEL
Deceleration.
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0.001 to 100000.0000 mm/s² or degrees/s² / from 0.0003937 to 3937.00787 inch/s².

Default value: 1000.0000 mm/s² or degrees/s² / 39.37008 inch/s².
Associated variable: (V.)[ch].MPA.DECEL[set].xn

Parameters ACCEL and DECEL set the acceleration values when it is trapezoidal or square-sine (bell shaped) (parameter SLOPETYPE). The CNC applies parameters ACCEL and DECEL to the movements in G1, G2 and G3; movements in G0 depend on parameter G0ACDCJERK.

G0ACDCJERK	Meaning.
Yes.	Movements in G0 use parameters ACCELG0 and DECELG0.
No.	Movements in G0 use parameters ACCEL and DECEL.



1 The axis starts moving with a uniformly increasing acceleration, with a slope limited by ACCJERK, until reaching the acceleration indicated in ACCEL.

2 The acceleration becomes constant at the ACCEL value.

3 Before reaching the programmed feedrate, there is a steady deceleration with a slope limited by ACCJERK.

4 It goes on at the programmed feedrate and with no acceleration.

5 To slow down or stop the axis, a deceleration is applied with a slope limited by DECJERK.

6 The deceleration becomes constant at the DECEL value.

7 Before reaching the programmed feedrate, or stopping, there is a deceleration limited by DECJERK.

ACCJERK
Acceleration Jerk.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 1.0000 a 1000000000.0000 mm/s³ or degrees/s³ / from 0.03937 to 39370078.74016 inch/s³.
Default value: 10000.000 mm/s³ or degrees/s³ / 393.70087 inch/s³.
Associated variable: (V.)[ch].MPA.ACCJERK[set].xn

See machine parameter DECJERK.

DECJERK
Deceleration Jerk.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 1.0000 a 1000000000.0000 mm/s³ or degrees/s³ / from 0.03937 to 39370078.74016 inch/s³.
Default value: 10000.000 mm/s³ or degrees/s³ / 393.70087 inch/s³.
Associated variable: (V.)[ch].MPA.DECJERK[set].xn

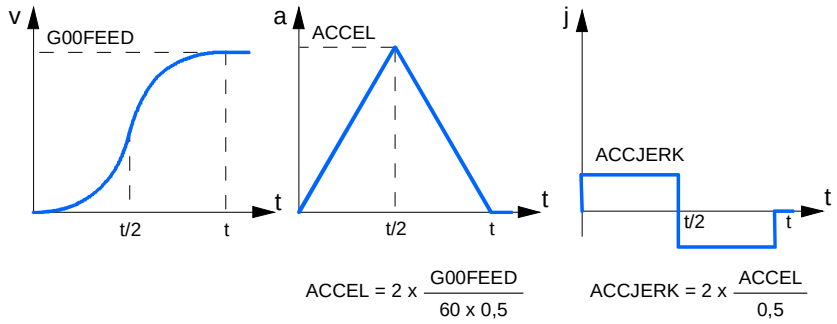
Parameters ACCJERK and DECJERK set the slope of the acceleration and deceleration. Both parameters help limit the acceleration changes so the machine runs more smoothly in small speed increments or decrements and with FFGAIN values close to 100%. The lower the value assigned to these parameters, the smoother the machine response, but the acc/dec time will increase.

The CNC applies parameters ACCEL and DECEL to the movements in G1, G2 and G3; movements in G0 depend on parameter G0ACDCJERK. The CNC ignores these parameters in threading movements (G33) and with HSC FAST.

G0ACDCJERK	Meaning.
Yes.	Movements in G0 use parameters ACCJERKG0 and DECJERKG0.
No.	Movements in G0 use parameters ACCJERK and DECJERK.

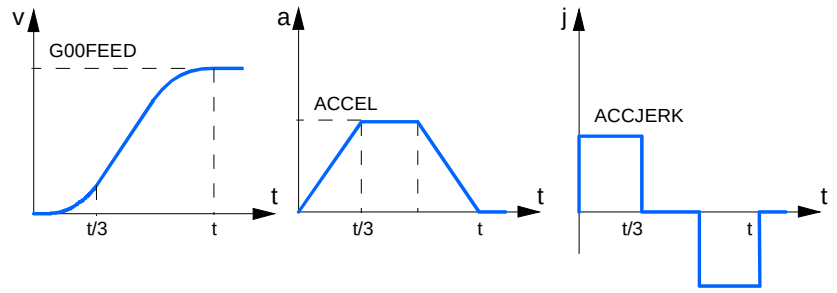
Example. Square sine acceleration.
Here are two examples so a stopped axis reaches G00FEED at maximum acceleration in a particular amount of time (0.5 seconds).

- The axis parameters may be set with maximum acceleration and minimum jerk.



$$ACCEL = 2 \times \frac{G00FEED}{60 \times 0,5} \qquad ACCJERK = 2 \times \frac{ACCEL}{0,5}$$

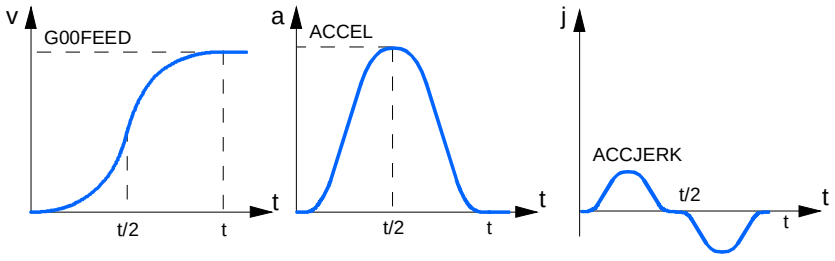
- The axis parameters may be set with less acceleration and greater jerk so the axis can reach the maximum acceleration faster and it stays longer at maximum acceleration.



$$ACCEL = \frac{3}{2} \times \frac{G00FEED}{60 \times 0,5} \qquad ACCJERK = 3 \times \frac{ACCEL}{0,5}$$

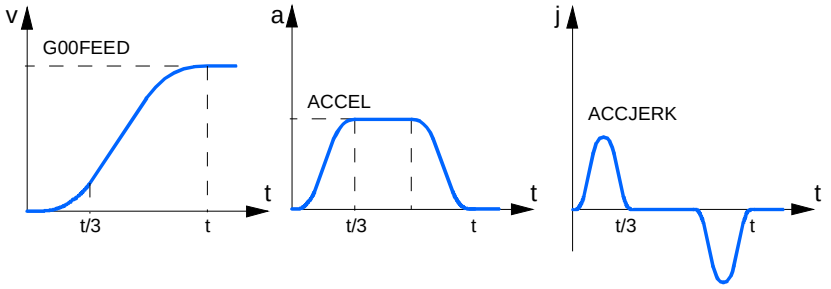
Example. Square sine acceleration.
Here are two examples so a stopped axis reaches G00FEED at maximum acceleration in a particular amount of time (0.5 seconds).

- The axis parameters may be set with maximum acceleration and minimum jerk.



$$ACCEL = 2 \times \frac{G00FEED}{60 \times 0,5} \qquad ACCJERK = 2 \times \frac{ACCEL}{0,5}$$

- The axis parameters may be set with less acceleration and greater jerk so the axis can reach the maximum acceleration faster and it stays longer at maximum acceleration.



$$ACCEL = \frac{3}{2} \times \frac{G00FEED}{60 \times 0,5} \qquad ACCJERK = 3 \times \frac{ACCEL}{0,5}$$

ENABLE SPECIFIC ACCELERATION VALUES FOR MOVEMENTS IN G0.

G0ACDCJERK
More parameters if G0.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Yes / No.
Default value: No.
Associated variable: (V.)[ch].MPA.G0ACDCJERK[set].xn

This parameter may be used to set some specific acceleration and jerk values for movements in G0. Otherwise, movements in G0 use the same acceleration and jerk values as movements in G1, G2 and G3.

Parameters to be used in G0 movements.	**G0ACDCJERK**	
	No	**Yes**
Linear acceleration (G0 movements).		
• Acceleration of the first section.	LACC1	LACC1G0
• Acceleration of the second section.	LACC2	LACC2G0
• Acceleration changing speed.	LFEED	LFEEDG0
Trapezoidal and square sine acceleration (G0 movements).		
• Acceleration.	ACCEL	ACCELG0
• Deceleration.	DECEL	DECELG0
• Acceleration Jerk.	ACCJERK	ACCJERKG0
• Deceleration Jerk.	DECJERK	DECJERKG0

LINEAR ACCELERATION (G0 MOVEMENTS).

LACC1G0
Acceleration of the first section (G0 movements).
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 1.0000 to 1000000.0000 mm/s² or degrees/s² / from 0.03937 to 3937.00787 inch/s².
Default value: 1000.0000 mm/s² or degrees/s² / 39.37008 inch/s².
Associated variable: (V.)[ch].MPA.LACC1G0[set].xn

See parameter LACC2G0.

LACC2G0
Acceleration of the second section (G0 movements).
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 1.0000 to 1000000.0000 mm/s² or degrees/s² / from 0.03937 to 3937.00787 inch/s².
Default value: 1000.0000 mm/s² or degrees/s² / 39.37008 inch/s².
Associated variable: (V.)[ch].MPA.LACC2G0[set].xn

Parameters LACC1G0 and LACC2G0 set the acceleration values when it is linear (parameter SLOPETYPE) or when working with HSC in FAST mode. The CNC applies parameters LACC1G0 and LACC2G0 to G0 movements.

LFEEDG0
Acceleration changing speed (G0 movements).
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 200000.0000 mm/min or degrees/min / from 0 to 7873.992 inch/min / from 0 to 100000.0000 rpm.
Default value: 1000.0000 mm/min, degrees/min or rpm / 39.37008 inch/min.
Associated variable: (V.)[ch].MPA.LFEEDG0[set].xn

While accelerating, when reaching the feedrate defined in this parameter, it changes the acceleration from LACC1G0 to LACC2G0. While decelerating, when reaching the feedrate defined in this parameter, it changes the acceleration from LACC2G0 to LACC1G0.

TRAPEZOIDAL AND SQUARE SINE ACCELERATION (G0 MOVEMENTS).

ACCELG0
Acceleration (G0 movements).
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0.001 to 100000.0000 mm/s² or degrees/s² / from 0.0003937 to 3937.00787 inch/s².
Default value: 1000.0000 mm/s² or degrees/s² / 39.37008 inch/s².
Associated variable: (V.)[ch].MPA.ACCELG0[set].xn

See machine parameter DECELG0.

DECELG0
Deceleration (G0 movements).
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0.001 to 100000.0000 mm/s² or degrees/s² / from 0.0003937 to 3937.00787 inch/s².
Default value: 1000.0000 mm/s² or degrees/s² / 39.37008 inch/s².
Associated variable: (V.)[ch].MPA.DECELG0[set].xn

Parameters ACCEL and DECEL set the acceleration values when it is trapezoidal or square-sine (bell shaped) (parameter SLOPETYPE). The CNC applies parameters ACCEL and

DECEL to the movements in G1, G2 and G3; movements in G0 depend on parameter G0ACDCJERK.

G0ACDCJERK	Meaning.
Yes.	Movements in G0 use parameters ACCELG0 and DECELG0.
No.	Movements in G0 use parameters ACCEL and DECEL.

ACCJERKG0
Acceleration jerk (G0 movements).

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 1.0000 a 1000000000.0000 mm/s³ or degrees/s³ / from 0.03937 to 39370078.74016 inch/s³.
Default value: 10000.000 mm/s³ or degrees/s³ / 393.70087 inch/s³.
Associated variable: (V.)[ch].MPA.ACCJERKG0[set].xn

See machine parameter DECJERKG0.

DECJERKG0
Deceleration jerk (G0 movements).

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 1.0000 a 1000000000.0000 mm/s³ or degrees/s³ / from 0.03937 to 39370078.74016 inch/s³.
Default value: 10000.000 mm/s³ or degrees/s³ / 393.70087 inch/s³.
Associated variable: (V.)[ch].MPA.DECJERKG0[set].xn

Parameters ACCJERKG0 and DECJERKG0 set the slope of the acceleration and deceleration. Both parameters help limit the acceleration changes so the machine runs more smoothly in small speed increments or decrements and with FFGAIN values close to 100%. The lower the value assigned to these parameters, the smoother the machine response, but the acc/dec time will increase.

The CNC applies parameters ACCEL and DECEL to the movements in G1, G2 and G3; movements in G0 depend on parameter G0ACDCJERK. The CNC ignores these parameters in threading movements (G33) and with HSC FAST.

G0ACDCJERK	Meaning.
Yes.	Movements in G0 use parameters ACCJERKG0 and DECJERKG0.
No.	Movements in G0 use parameters ACCJERK and DECJERK.

CONFIGURATION OF THE HSC MODE.

HSC
Parameter table of the HSC mode..

This parameter shows the table to define the HSC work mode.

HSC	
CORNERACC	Maximum acceleration permitted at the corners.
CURVACC	Maximum contouring acceleration permitted.
CORNERJERK	Maximum Jerk permitted at the corners.
CURVJERK	Maximum contouring Jerk permitted.
FASTACC	Maximum acceleration permitted (FAST mode).
MAXERROR	Position error.
CONTERROR	Tolerance per axis.

These parameters help increase or decrease the acceleration and jerk limits on curved path, due to the centripetal acceleration and at path joints, without affecting the way the axis accelerates or decelerates.

CORNERACC
Maximum acceleration permitted at the corners.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 100000.0000 mm/s² or degrees/s² / from 0 to 3937.00787 inch/s².
Default value: 0.
Associated variable: (V.)[ch].MPA.CORNERACC[set].xn

This parameter sets the maximum acceleration allowed for the axis during block transition. If this parameter is set with a ·0· value, the maximum acceleration of the axis is respected.

CURVACC
Maximum contouring acceleration permitted.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 100000.0000 mm/s² or degrees/s² / from 0 to 3937.00787 inch/s².
Default value: 0.
Associated variable: (V.)[ch].MPA.CURVAACC[set].xn

If this parameter is set with a ·0· value, the maximum acceleration of the axis is respected.

CORNERJERK
Maximum Jerk permitted at the corners.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 a 1000000000.0000 mm/s³ or degrees/s³ / from 0 to 39370078.74016 inch/s³.
Default value: 0.
Associated variable: (V.)[ch].MPA.CORNERJERK[set].xn

If this parameter is set with a ·0· value, the maximum jerk of the axis is respected.

CURVJERK
Maximum contouring Jerk permitted.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 a 1000000000.0000 mm/s³ or degrees/s³ / from 0 to 39370078.74016 inch/s³.
Default value: 0.
Associated variable: (V.)[ch].MPA.CURVJERK[set].xn

If this parameter is set with a ·0· value, the maximum jerk of the axis is respected.

FASTACC
Maximum acceleration permitted (FAST mode).

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 100000.0000 mm/s² or degrees/s² / from 0 to 3937.00787 inch/s².
Default value: 0.
Associated variable: (V.)[ch].MPA.FASTACC[set].xn

If this parameter is set with a ·0· value, the maximum acceleration of the axis is respected.

MAXERROR
Position error.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 9999.99999 mm or degrees / from 0 to 3937.00787 inch.
Default value: 0.1000 mm / 0.00394 inch.
Associated variable: (V.)[ch].MPA.MAXERROR[set].xn

Maximum axis position error in HSC when working outside the plane/trihedron.

CONTERROR
Tolerance per axis for smoothing the generated n-dimensional path.
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Default value: 0.1000 mm/º.
Associated variable: (V.)[ch].MPA.CONTERROR[set].xn

For 5-axis machining, this parameter may be used to set the tolerance per axis for smoothing the generated n-dimensional path. This parameter does not affect the 3D error when working with RTCP. We recommend setting this parameter on rotary axes.

HOME SEARCH.

The home search process depends on the characteristics of the machine.

I0TYPE
Reference mark (I0) type.
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Regular (non-distance-coded) / increasing distance-coded / decreasing distance-coded.
Default value: Incremental (Not distance-coded)
Associated variable: (V.)[ch].MPA.I0TYPE[set].xn

This parameter indicates how the reference marks of the linear encoder are read with respect to the moving direction during the home search.

REFVALUE
Home position.
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Within ±99999.9999 mm or degrees / within ±3937.00787 inch.
Default value: 0.
Associated variable: (V.)[ch].MPA.REFVALUE[set].xn

The machine reference point must be defined in the following cases:

- The feedback system does not have distance-coded marks.
- The feedback system has distance-coded marks and leadscrew error compensation is being applied on that axis.

Set the home position referred to Machine Reference Zero.

REFSHIFT
Offset of the reference point (home).
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Within ±99999.9999 mm or degrees / within ±3937.00787 inch.
Default value: 0.
Associated variable: (V.)[ch].MPA.REFSHIFT[set].xn

Sometimes, to readjust the machine, it is necessary to take down the feedback device, thus when putting back up, the new home point might no coincide with the previous one. Since the home point must still be the same, the difference between the new point and the old point must be assigned to parameter REFSHIFT . This way, when the axis finds the I0, it moves the distance indicated in REFSHIFT and at that point updates its coordinate to the value of REFVALUE.

The value to be set in REFSHIFT must be measured without leadscrew compensation being active because it is considered to be a correction to the encoder position value.

When the axes are position-Sercos; during parameter matching, the CNC sends the value of parameter REFSHIFT to the drive so it takes it into account; this way the CNC coordinate and that of the drive will be the same.

REFFEED1
Fast home searching feedrate.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 200000.0000 mm/min or degrees/min / from 0 to 7873.992 inch/min / from 0 to 100000.0000 rpm.
Default value: 1000.0000 mm/min or degrees/min / 39.37001 inch/min / 100.0000 rpm.
Associated variable: (V.)[ch].MPA.REFFEED1[set].xn

See machine parameter REFFEED2.

REFFEED2
Slow home searching feedrate.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 200000.0000 mm/min or degrees/min / from 0 to 7873.992 inch/min / from 0 to 100000.0000 rpm.
Default value: 100.0000 mm/min or degrees/min / 3.93700 inch/min / 10.0000 rpm.
Associated variable: (V.)[ch].MPA.REFFEED2[set].xn

When the feedback system does not have distance-coded reference marks (I0, the home search is carried out at the feedrate indicated by "REFFEED1" until the home switch is reached. It then reverses its movement at the feedrate indicated by "REFFEED2" and it goes on until the CNC receives the reference marker pulse from the feedback device.

REFPULSE
Type of I0 pulse.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Positive / Negative.
Default value: Positive.
Associated variable: (V.)[ch].MPA.REFPULSE[set].xn

This parameter indicates the type of flank of the I0 signal that is used for home search.

POSINREF
Home search moving the axis to the reference point.

Parameter valid for linear and rotary axes.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Yes / No.
Default value: No.
Associated variable: (V.)[ch].MPA.POSINREF[set].xn

This parameter indicates whether the axis moves or not to the machine reference point (parameter REFVALUE) while homing. This movement is not necessary when the axis has internal absolute feedback (motor feedback).

On axes with absolute motor feedback, the CNC knows at all times the relative axis position per motor revolution. In these cases, while homing the axis, the CNC knows its position as soon as the home switch is pressed; hence not being necessary to move up to the reference point.

ABSOFF
Offset with respect to the distance-coded ref. mark.

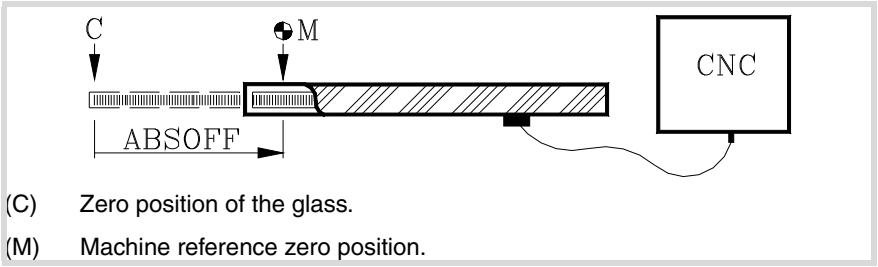
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Within ±99999.9999 mm or degrees / within ±3937.00787 inch.
Default value: 0.
Associated variable: (V.)[ch].MPA.ABSOFF[set].xn

The CNC takes this parameter into account if I0TYPE = Distance coded or ABSFEEDBACK = Yes.

With linear encoders with distance-coded (I0 reference marks, it is possible to know the position of the machine by simply moving the axis a distance of 20 or 100 mm. After reading two consecutive distance-coded reference marks (20 or 100 mm away from each other), it is possible to know the axis position with respect to the zero point of the graduated glass (C).

In order for the CNC to show the position with respect to Machine Zero (M), this parameter must be assigned the position of the machine zero (M) with respect to the glass zero point (C).

The zero point of the glass (beginning of the distance code) may be in or out the scale's measuring length.



(C) Zero position of the glass.

(M) Machine reference zero position.

EXTMULT
External factor for distance-coded reference marks.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drives and simulated drives.
Possible values: From 0 to 65535.
Default value: 0.
Associated variable: (V.)[ch].MPA.EXTMULT[set].xn

The CNC takes this parameter into account if I0TYPE = Distance coded.

This parameter indicates the relationship between the mechanical period (of the graduation on the glass) and the electrical period (of the feedback signal) being applied to the CNC.

Example with a Fagor linear encoder.

A "FOX" type Fagor linear encoder has a graduation period of 100 µm (gap between lines) and an electrical signal period of 4 µm.

EXTMULT = 100 / 4 = 25

Values to be assigned for Fagor encoders with distance-coded I0.

Rotary encoder.			**EXTMULT**
HO	SO	90,000 pulses.	5
HO	SO	180.000 pulses.	10
HOP	SOP	18.000 pulses.	1

Linear encoder.						**EXTMULT**
SOP SVOP	GOP	MOP MOC MOT MOVP	COP COC COT COVP		FOP	1
SOX SVOX	GOX	MOX MOVX	COX COVX		FOT	5
		MOY MOVY	COY			10
				LOP		1
				LOX		10
					FOX	25

I0CODDI1
Pitch between 2 fixed distance-coded reference marks.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 65535.
Default value: 1000.
Associated variable: (V.)[ch].MPA.I0CODDI1[set].xn

See machine parameter I0CODDI2.

I0CODDI2
Pitch between 2 variable distance-coded reference marks.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 65535.
Default value: 1001.
Associated variable: (V.)[ch].MPA.I0CODDI2[set].xn

The CNC takes this parameter into account if I0TYPE = Distance coded.

The pitch between reference marks is defined in number of waves.

Example with a Fagor linear encoder.

Pitch between 2 fixed reference marks.	20.000 mm
Pitch between 2 variable reference marks.	20.020 mm
Period of the sinusoidal signal	20 µm
Number of waves between fixed reference marks	20000/(20 x EXTMULT) = 1000
Number of waves between variable reference marks	20020/(20 x EXTMULT) = 1001

Values to be assigned for Fagor encoders with distance-coded I0.

Rotary encoder.			I0CODDI1	I0CODDI2
HO	SO	90,000 pulses.	1000	1001
HO	SO	180.000 pulses.	1000	1001
HOP	SOP	18.000 pulses.	1000	1001

Linear encoder.						I0CODDI1	I0CODDI2
SOP SVOP	GOP	MOP MOC MOT MOVP	COP COC COT COVP		FOP	1000	1001
SOX SVOX	GOX	MOX MOVX	COX COVX		FOT	1000	1001
		MOY MOVY	COY			1000	1001
				LOP		2000	2001
				LOX		2000	2001
					FOX	1000	1001

FOLLOWING ERROR.

Following error is the difference between the theoretical position and the actual (real) position of the axis. The following error decreases when increasing the axis gain.

The more similar (identical) the following errors of interpolating axes are, the better the machining of curved sections will be on circular interpolations.

FLWEMONITOR
Type of monitoring of the following error (axis lag).

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: OFF / Standard / Linear.
Default value: Standard.
Associated variable: (V.)[ch].MPA.FLWEMONITOR[set].xn

This parameter determines how the following error will be monitored.

- If set to "OFF", the following error is not monitored, thus no error message will be issued.

- The "standard" monitoring monitors the following error all the time and an error message will be issued when it exceeds the value of parameter MAXFLWE if the axis is moving or MINFLWE if the axis is stopped.
- Linear monitoring performs a dynamic supervision that allows a percentage of following error. This percentage is set by parameter FEDYNFAC.

If this watch is canceled, the CNC shows on power-up a message indicating that this safety is disabled. This situation can only be allowed during setup; once setup is completed, this watch must be enabled.

MINFLWE
Maximum following error when stopped.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drives and Sercos drives.
Possible values: From 0.0001 to 99999.9999 mm or degrees / from 0 to 3937.00787 inch.
Default value: 1.0000 mm or degrees / 0.03937 inch.
Associated variable: (V.)[ch].MPA.MINFLWE[set].xn

The CNC takes this parameter into account only if FLWEMONITOR other than OFF.

This parameter indicates the maximum amount of following error allowed when the axis is stopped. The MINFLWE value cannot be greater than 1/4 of the total axis travel (LIMIT+ to LIMIT-).

MAXFLWE
Maximum following error when moving.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drives and Sercos drives.
Possible values: From 0.0001 to 99999.9999 mm or degrees / from 0 to 3937.00787 inch.
Default value: 30.0000 mm or degrees / 1.18110 inch.
Associated variable: (V.)[ch].MPA.MAXFLWE[set].xn

The CNC takes this parameter into account only if FLWEMONITOR other than OFF.

The meaning of this parameter depends on the value of FLWEMONITOR.

- With FLWEMONITOR = Standard, MAXFLWE indicates the maximum amount of following error allowed when the axis is moving.
- With FLWEMONITOR = Linear, MAXFLWE indicates the value starting at which the following error is dynamically monitored.

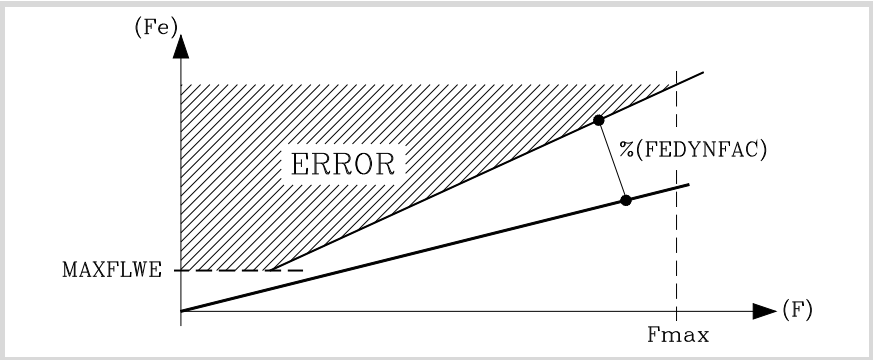
FEDYNFAC
Percentage of following error deviation allowed.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drives and Sercos drives.
Possible values: from0 to 100 %.
Default value: 50.
Associated variable: (V.)[ch].MPA.FEDYNFAC[set].xn

The CNC takes this parameter into account only if FLWEMONITOR other than OFF.

This parameter indicates the permitted percentage error; in other words the deviation of the real following error with respect to the theoretical one.

The CNC calculates the maximum and minimum following error (Fe) at all times depending on feedrate (F). If is not within the permitted zone (shaded area of the figure) , the CNC will issue the relevant error message.



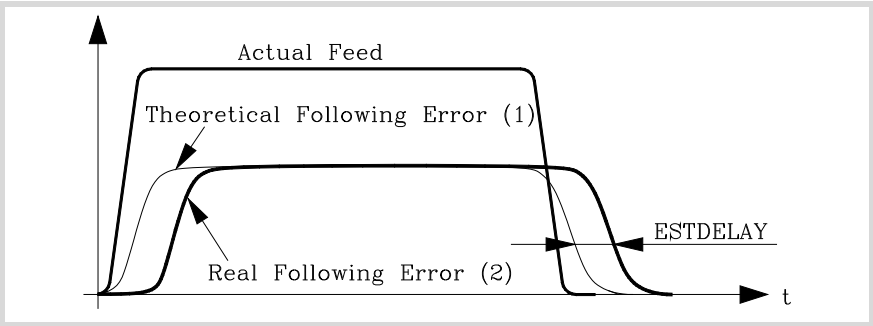
Parameter MAXFLWE indicates the value starting at which the following error will be dynamically monitored.

ESTDELAY
Following error delay.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drives and Sercos drives.
Possible values: From 0 to 1000000 ms.
Default value: 0.
Associated variable: (V.)[ch].MPA.ESTDELAY[set].xn

The CNC takes this parameter into account if FLWEMONITOR=Linear.

This parameter is used to define a delay applied when estimating the following error so the theoretical value (1) comes closer to the real one (2) thus avoiding undesired following error messages. This parameter is also used as the time required to issue the tendency error (run-away protection).



Setting parameter ESTDELAY requires activating the estimate of linear following error and using the oscilloscope to check variables (V.)A.FLWE.xn and (V.)A.FLWEST.xn. The value of parameter ESTDELAY must be slightly higher than the maximum delay between the two signals. When calculating this parameter, bear in mind that the worst scenario is when reversing the movement in G00.

INPOMAX
Time to get in position.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 1000000 ms.
Default value: 0.
Associated variable: (V.)[ch].MPA.INPOMAX[set].xn

Parameter INPOMAX limits (maximum time) the time the axis needs to get in position. This parameter provides the possibility to watch the positioning of the axis ensuring that it gets in position in a specific amount of time and issuing an error if otherwise.

INPOTIME
Minimum time to stay in position.
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 1000000 ms.
Default value: 0.
Associated variable: (V.)[ch].MPA.MPA.INPOTIME[set].xn

Parameter INPOTIME sets the time the axis must stay in the in-position zone so the CNC considers it to be "in position".

Parameters INPOMAX and INPOTIME ensure that when working with dead axes (axes only controlled while moving), the movement will be completed when they are in position.

AXIS LUBRICATION.

DISTLUBRI
Distance to move to lubricate the axis.
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 2000000000 mm or degrees / from 0 to 78739920 inch.
Default value: 0 (no lubrication).
Associated variable: (V.)[ch].MPA.DISTLUBRI[set].xn

The PLC reads this parameter in mm instead of doing it in tenths of a micron (0.0001 mm).

The lubrication signal is activated after travelling the distance indicated in this parameter. The CNC logic inputs and outputs: LUBR(axis), LUBRENA(axis) and LUBROK(axis) must be used in order for the PLC lubricates the axes and gears.

1. The LUBRENA(axis) mark indicates whether this feature is to be used or not.
2. When the axis has traveled the distance set by parameter DISTLUBRI, the CNC turns the LUBR(axis) mark on to let the PLC know that it must lubricate the axis.
3. After lubricating the axis, the PLC sets the LUBROK(axis) mark high (=1) to let the CNC know that the axis has been lubricated.
4. The CNC sets the LUBR(axis) mark low (=0) and resets its count to "0".

MODULE CONFIGURATION (ROTARY AXES AND SPINDLE).

MODUPLIM
Upper limit of the module.
Parameter valid for rotary axes and spindles.
Parameter valid for analog drives, velocity-Sercos drives and simulated drives.
Possible values: Within ±99999.9999º.
Default value: 360.
Associated variable: (V.)[ch].MPA.MODUPLIM[set].xn

See machine parameter MODLOWLIM.

MODLOWLIM
Lower limit of the module.
Parameter valid for rotary axes and spindles.
Parameter valid for analog drives, velocity-Sercos drives and simulated drives.
Possible values: Within ±99999.9999º.
Default value: 0.
Associated variable: (V.)[ch].MPA.MODLOWLIM[set].xn

The CNC takes these parameters into consideration only when AXISMODE = Module.

For a reading within ±180°, set MODUPLIM = 180º and MODLOWLIM = -180º.

MODNROT
Module error. Number of turns.
Parameter valid for rotary axes and spindles.
Parameter valid for analog drives, velocity-Sercos drives and simulated drives.
Possible values: From 1 to 32767 turns.
Default value: 1.
Associated variable: (V.)[ch].MPA.MODNROT[set].xn

See machine parameter MODERR.

MODERR
Module error. Number of increments.
Parameter valid for rotary axes and spindles.
Parameter valid for analog drives, velocity-Sercos drives and simulated drives.
Possible values: Within ±32767.
Default value: 0.
Associated variable: (V.)[ch].MPA.MODERR[set].xn

The CNC takes these parameters into consideration only when AXISMODE = Module and MODCOMP = Yes. These parameters indicate the compensation to be applied to compensate for an inexact axis resolution. The CNC applies module compensation throughout the entire revolution.

Parameter MODERR indicates the amount of error to be compensated when the axis has rotated the revolutions indicated in parameter MODNROT. This correction is necessary, for example, when using a 1024 line encoder whose parameter has been set to 1000 pulses.

SPINDLE SPEED.

SZERO
Speed considered "0 rpm".
Parameter valid for spindles.
Parameter valid for analog drives and simulated drives.
Possible values: From 0 to 100000 rpm.
Default value: 1.
Associated variable: (V.)[ch].MPA.SZERO[set].xn

This parameter indicates the rpm value below which the spindle is considered to be stopped. The CNC uses this parameter to switch the spindle to C axis mode and also to accept the velocity command programmed with SANALOG when the spindle is controlled from the PLC. This parameter can also be used from the PLC to permit opening the machine doors.

POLARM3
Sign of the velocity command for M3.
Parameter valid for spindles.
Parameter valid for analog drives, velocity-Sercos drives and simulated drives.
Possible values: Positive / Negative.
Default value: Positive.
Associated variable: (V.)[ch].MPA.POLARM3[set].xn

See machine parameter POLARM3.

POLARM4
Sign of the velocity command for M4.
Parameter valid for spindles.
Parameter valid for analog drives, velocity-Sercos drives and simulated drives.
Possible values: Positive / Negative.
Default value: Negative.
Associated variable: (V.)[ch].MPA.POLARM4[set].xn

Parameters POLARM3 and POLARM indicate the direction of the velocity command and, therefore, the spindle turning direction for functions M3 and M4 respectively.

The CNC offers the following variable to reverse, via part-program or PLC, the direction of the velocity command set in this parameter.

Variable.	Meaning.
(V.)[ch].A.POLARITY.sn (V.)[ch].SP.POLARITY.sn	This variable may be used to reverse the meaning of parameters POLARM3 and POLARM4. The variable does not modify the values of the machine parameters.

ANALOG COMMAND SETTING.

SERVOOFF
Offset compensation.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for an analog drive.
Possible values: Within ±32767.
Default value: 0.
Associated variable: (V.)[ch].MPA.SERVOOFF[set].xn

Velocity command applied as offset to the drive. The velocity command is given in D/A converter units that could be any integer within ±32767; a value of ±32767 corresponds to an analog velocity command of ±10V.

SERVOOFF	1	3277	32767
Velocity command.	0,3 mV	1 V	10 V

MINANOUT
Minimum velocity command.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for an analog drive.
Possible values: From 0 to 32767.
Default value: 0.
Associated variable: (V.)[ch].MPA.MINANOUT[set].xn

The velocity command is given in D/A converter units that could be any integer between 0 and 32767; a value of ±32767 corresponds to an analog velocity command of 10V.

SERVOOFF	1	3277	32767
Velocity command.	0,3 mV	1 V	10 V

NUMBER OF THE ANALOG OUTPUT AND OF THE FEEDBACK INPUT ASSOCIATED WITH THE AXIS.

ANAOUTTYPE
Type of analog output associated with the axis.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for an analog drive.
Possible values: Remote CAN / Drive / Sercos Counter.
Default value: Remote CAN.
Associated variable: (V.)[ch].MPA.ANAOUTTYPE[set].xn

This parameter is only valid for analog axes and indicates where the analog output is located; in the remote CAN modules, in a Sercos drive or in an RCS-S module (Sercos counter).

ANAOUTID
Number of the analog output associated with the axis.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for an analog drive.
Possible values: From 1 to 16 for remote modules / Sercos drives, from 101 to 132 or from 201 to 232 / Sercos counter.

Default value: 0.
Associated variable: (V.)[ch].MPA.ANAOUTID[set].xn

The velocity command for an analog axis may be taken from an analog output of the remote CAN modules or of a Sercos drive. or of an RCS-S module (Sercos counter). In either case, this parameter indicates the analog output used for the command.

When two axes will not be in the configuration at the same time, the CNC allows using the same analog output for both axes.

Managing the analog command from CANfagor remote modules.

In this case, this parameter indicates the number of the analog output used for the command. The analog output modules are numbered following the logic order of the remote groups (rotary switch of the power supply). If there are several analog-output modules in each group, the order is from top to bottom and from left to right. The analog outputs of the first module will be 1 through 4, those of the second module 5 through 8 and so on.

Managing the analog command from Sercos drives.

In this case, this parameter indicates the number of the analog output and the drive to be used. The parameter is set with a three-digit number; the first digit indicates the number of the analog output to be used (1 or 2) and the other two digits indicate the logic address of the drive (1 through 32). The "address" rotary selector switch sets the logic address.

For example, if the parameter is with a value of ·107·, it indicates that the CNC uses analog input ·1· of the drive whose logic address is ·7·

Managing the analog command from the RCS-S module.

In this case, this parameter indicates the number of the analog output used for the command. The RCS-S modules are numbered according to their logic order (parameter SERCOUNTID). The analog outputs of the first module will be 1 through 4, those of the second module 5 through 8 and so on.

COUNTERTYPE
Type of feedback input for the axis.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drives and velocity-Sercos drives.
Possible values: Remote CAN / Drive / Sercos Counter / Local.
Default value: Remote.
Associated variable: (V.)[ch].MPA.COUNTERTYPE[set].xn

Meaning for an analog axis.

This parameter indicates the location of the axis feedback input; at the remote modules, at the free feedback of a Sercos drive, in an RCS-S module (Sercos counter) or at the local feedback inputs of the central unit. When the feedback input is at a drive, the CNC does not allow homing the axis.



When the axis feedback is at a Sercos drive, at this drive set parameter PP5=-0,0001.

Meaning for a velocity-Sercos axis.

This parameter indicates the location of the axis external feedback (direct) input; at the remote modules, at the free feedback of a Sercos drive or at the local feedback inputs of the central unit.

The axis can use the second free feedback input of that drive or of another one. When the axis uses the second feedback input of another drive, the CNC does not allow homing the axis.



When the internal (motor) feedback and the external (direct) feedback of an axis are at different drives, set parameter PP5=-0,0001 at the drive that receives the external feedback.

COUNTERID
Number of the feedback input for the axis.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drives and velocity-Sercos drives.
Possible values: Remote modules, from 1 to 40 / Sercos drive, from 1 to 32 / Local inputs, from 1 to 2.
Default value: 0.
Associated variable: (V.)[ch].MPA.COUNTERID[set].xn

The feedback input of an analog axis and the input for the external (direct) feedback of a Sercos axis may be located at the remote modules, at the free feedback input of a Sercos drive, in an RCS-S module (Sercos counter) or at the local feedback inputs of the central unit. Depending on the location of the feedback input, this parameter will indicate which feedback input will be used (either local or remote) or the drive/counter number. When two axes will not be in the configuration at the same time, the CNC allows using the same feedback input for both axes.

Feedback input in CANfagor remote modules.

A counter module is used as feedback input in the remote modules. This parameter indicates the number of the feedback input being used.

The counter modules are numbered following the logic order of the remote groups (rotary switch of the power supply). If there are several Counter modules in each group, the order is from top to bottom and from left to right. The counters of the first module will be 1 through 4, those of the second module 5 through 8 and so on.

Feedback input in Sercos drives.

In this case, the second feedback input of the drive will always be used. The parameter indicates the logic address of the drive (from 1 to 32). The "address" rotary selector switch sets the logic address.

Feedback input of an RCS-S module.

This parameter indicates the number of the feedback input being used.

The RCS-S modules are numbered according to their logic order (parameter SERCOUNTID). The feedback inputs of the first module will be 1 through 4, those of the second module 5 through 8 and so on.

Local feedback input.

The central unit has two feedback inputs. This parameter indicates the feedback input being used.

SET THE DRIVE ASSOCIATED WITH THE AXES OF A MULTI-AXIS GROUP.

DRIVESET
Set of the drive to be used.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for a Sercos drive.
Possible values: From 0 to 8.
Default value: 0.
Associated variable: (V.)[ch].MPA.DRIVESET[set].xn

At the axes that belong to a multi-axis group, parameter DRIVESET indicates the set to be activated at the drive after a set or gear change at the CNC (G112 and M41 to M44). The CNC sends the changes of Kv, feed-forward (parameter FFGAIN) and AC-forward (parameter ACFGAIN) to the set indicated in this parameter.

Being parameter matching active, the CNC sends the parameters of set ·1· of the master axis to the set ·0· and to the gear ·1· (first) of the drive. For the rest of the sets of the master axis, the CNC only sends the gear data to the drive; set ·2· of the CNC to the gear 2 of the drive and so on. At the master axis of multi-axis group, in order for parameter matching to work properly, this parameter must coincide with the set number where it is; i.e. DRIVESET=1 in set ·1·, DRIVESET=2 in ·2· and so on. At the slave axis of a multi-axis group, this parameter indicates the gear to be used at the drive. If the axis belongs to a multi-axis group, the parameter does not admit a ·0· (zero) value.

FEEDBACK TYPE OF THE RCS-S MODULE.

FEEDBACKTYPE
Feedback type of the RCS-S module.

Possible values: TTL / differential TTL / Vpp / SSI.
Default value: Differential TTL.
Associated variable: (V.)[ch].MPA.FEEDBACKTYPE[set].xn

When the feedback input of the axis is located in an RCS-S module (Sercos counter), this parameter sets the type of signal.

SSITYPE
SSI feedback type connected to the corresponding feedback.

Possible values: Fagor SSITYPE / G user / User.
Default value: User.
Associated variable: (V.)[ch].MPA.SSITYPE[set].xn

When FEEDBACKTYPE=SSI, this parameter is used to set the format of the SSI transmission. The parameter offers a number of preset types for connecting the most common linear and rotary encoders. If the encoder is not listed, the user option may be selected to set the properties of the SSI communication.

Value.	Meaning.
Fagor LA	Fagor linear encoder.
Fagor GA SA SVA	Fagor linear encoder.
Fagor HA-27-D200	Fagor encoder.
Fagor HA-23-D90 SA-23-D90 SA-23-D170	Fagor encoder.
ABSIND (inductosyn LIN+ABS)	Absolute linear feedback.
ABSIND (inductosyn ROT+ABS)	Absolute rotary feedback in one encoder revolution.
ABSIND (resolver)	Absolute rotary feedback in one revolution.
ABSIND (inductosyn LIN)	Absolute linear feedback in the pitch. Homing the linear encoder required.
ABSIND (inductosyn ROT)	Absolute rotary feedback in the pitch. Homing the linear encoder required.
G user	User configuration.
User	User configuration.

SSI
SSI (Synchronous Serial Interface) parameter table.

This parameter shows the parameter table to set the SSI transmission format. Depending on the type of linear encoder selected, (parameter SSITYPE), some of these values will be preset and cannot be changed. All values may be used for the "user" type linear encoder.

Example for a Fagor absolute linear encoder (Fagor LA).
Example of parameter setting for connecting the Fagor LA absolute linear encoder, set with default parameters (factory set).

SSICLOCKFRECUENCY = 1 / T = 400.	400 kHz.
SSIDATALENGTH = n	Bit length of the position value (coordinate).
SSICRCBITS = 0	There are no CRC bits.
SSISTARTBITS = 0	There are no start bits.
SSIPACKFORMAT = 3	Data + Alarm.
SSIALARMBITS = 1	1 alarm/acknowledge bit.
SSIDATAMODE = 1	MSB First.
SSIALARMLEVEL = 1	The acknowledge is indicated with data at 0; therefore, an alarm signal is produced when the bit is at 1.

SSICLKFREQ
SSI communication frequency.

Possible values: From 59 to 7500 kHz.
Default value: 150 kHz.
Associated variable: (V.)[ch].MPA.SSICLKFREQ[set].xn

Frequency of the clock for SSI communication.

SSIDATAFORMAT
Parameters to set the format of the SSI transmission reading data.

The position read via SSI transmission may be received in different formats.

- The following parameters set the amount of SSI data that will be read.

Parameter.	Meaning.
SSIDATALENGTH	Number of bits of the SSI transmission that make up the position value (coordinate).
SSICRCBITS	Number of bits, if any, for some kind of transmission validity check (CRC, checksum, parity).
SSIALARMBITS	Number of bits, if any, for transmission OK check. The feedback device can send one or more bits indicating an alarm condition if something goes wrong or one or more acknowledgment bits if the transmission is successful.
SSISTARTBITS	Number of bits, if any, to wait for before receiving position data.

- The following parameters set the order and format used for the system to send the data.

Parameter.	Meaning.
SSIPACKFORMAT	Order in which the encoder sends the data. Whether the position data, the CRC or the alarm is received first and in which order. If start bits have been programmed, it is assumed that they will be received first.
SSIDATAMODE	Order in which the most significant bit is transmitted • SSIDATAMODE = 0; the LSB (Least Significant Bit) is the first one. • SSIDATAMODE = 1; the MSB (Most Significant Bit) is the first one.

- The following parameters set how to interpret the bits that determine that the SSI transmission has been successful.

Parameter.	Meaning.
SSIALARMLEVEL	Values that may make the encoder transmit one or more bits indicating some kind of error.
SSICRCTYPE	Type of calculation to do when the encoder transmits extra data besides the position value in order to run a data coherence check.

SSIDATALENGTH
Number of bits of the SSI transmission that make up the position value (coordinate).

Possible values: From 0 to 255.
Default value: 0.
Associated variable: (V.)[ch].MPA.SSIDATALENGTH[set].xn

Number of bits of the SSI transmission that make up the position value (coordinate). For example, 32 for Fagor absolute linear encoders with the default parameters.

SSIPACKFORMAT
SSI package format.

Possible values: Data / Data-CRC / CRC-Data / Data-Alarm / Alarm-Data / Data-CRC-Alarm / Alarm-Data-CRC / Data-Alarm-CRC / Alarm-CRC-Data / CRC-Data-Alarm / CRC-Alarm-Data.
Default value: Data.
Associated variable: (V.)[ch].MPA.SSIPACKFORMAT[set].xn

This parameter indicates the order in which the different data types of the SSI transmission will be received. Depending on parameters SSICRCBITS and SSIALARMBITS, only the possible options will be displayed. The order indicated by the parameter is the one that

corresponds to the SSI transmission sequence. If start bits have been programmed, the CNC assumes that they will be received first.

Example:
SSICRCBITS = 5
SSIALARMBITS = 1
SSIDATABITS = 23
SSIPACKFORMAT = Alarm - Data - CRC
The CNC expects the SSI transmission to be a sequence of bits where the first one is the alarm bit followed by 23 position bits or data bits and finally the 5 CNC bits.

SSICRCTYPE
CRC type.

Possible values: Do not calculate CRC / Fagor Checksum / INDUCTOSYN Checksum.
Default value: Do not calculate CRC
Associated variable: (V.)[ch].MPA.SSICRCTYPE[set].xn

When the encoder transmits extra data besides the position data, this parameter sets the type of calculation to do to check data coherence. This parameter only makes sense when SSICRCBITS is other than 0.

Value.	Meaning.
Do not calculate CRC.	Even if the CNC receives the CRC bits, they are not processed at all and, therefore, no errors are reported when the transmission is not correct. It is not a recommendable option and it must only be used during setup.
Fagor checksum.	Data checking algorithm used in Fagor Automation feedback systems.
INDUCTOSYN checksum.	Data checking algorithm used when connecting to an INDUCTOSYN module.

SSICRCBITS
Number of CRC bits of the SSI transmission to check that the transmission is valid.

Possible values: From 0 to 31.
Default value: 0.
Associated variable: (V.)[ch].MPA.SSICRCBITS[set].xn

Number of bits, if any, for the transmission validity check (CRC, checksum, parity).

For example, Fagor absolute linear encoders may be configured to transmit 5 CNC bits together with the position value. If a linear encoder is configured in this mode, program SSIDATALENGTH=27 and SSICRCBITS =5.

SSISTARTBITS
Number of start bits.

Possible values: From 0 to 3.
Default value: 0.
Associated variable: (V.)[ch].MPA.SSISTARTBITS[set].xn

Number of bits, if any, to wait for before receiving position data.

SSIALARMBITS
Number of alarm bits

Possible values: From 0 to 2.
Default value: 0.
Associated variable: (V.)[ch].MPA.SSIALARMBITS[set].xn

The feedback device can send one or more bits indicating an alarm condition if something goes wrong or one or more acknowledgment bits if the transmission is successful.

SSIALARMLEVEL
Level of the alarm bits to issue an error.

Possible values: From 0 to 3.

Default value: 0.
Associated variable: (V.)[ch].MPA.SSIALARMLEVEL[set].xn

Value that the alarm/acknowledge bits for an error condition to occur. This parameter only makes sense when SSIALARMBITS is other than 0.

Example:
If an encoder indicates an error condition with 2 bits where the first one is a 1 and the second a 0
SSIALARMBITS = 2
SSIALARMLEVEL = 2.

Example:
If an encoder has a transmission-OK acknowledgment bit and it indicates it with a logic 0.
SSIALARMBITS = 1
SSIALARMLEVEL = 1. Because a 0 means that everything is OK.

SSIDATAMODE
Transmission mode.

Possible values: From 0 to 1.
Default value: 0.
Associated variable: (V.)[ch].MPA.SSIDATAMODE[set].xn

Order in which the most significant bit is transmitted.

Value.	Meaning.
0	The LSB (Least Significant Bit) is the first one.
1	The MSB (Most Significant Bit) is the first one.

STARTDELAY
Number of clocks to wait between the first down flank and the first up flank.

Possible values: From 0 to 255.
Default value: 0.
Associated variable: (V.)[ch].MPA.STARTDELAY[set].xn

This parameter sets the number of SSI clocks to wait for since the first SSI down flank until the first up flank is generated. This value is needed to implement the wait time to convert the signal in some encoders.

Example:
CLOCKFREQ = 400 kHz
STARTDELAY = 3
Waiting time = (1/400*1000)*3 = 7,5 clocks.

SSIRESOL
Digital counting (feedback) resolution.

Possible values: From 1 to 999999999.
Default value: 10000.
Associated variable: (V.)[ch].MPA.SSIRESOL[set].xn

Number of digital counting units contained in a pitch.

Example:
Fagor absolute linear encoder with a pitch of 20 microns and a digital resolution of 1 dμ.
SSIRESOL = 20 μ / 0.1 μ = 200.

Example:
For SSITYPE = INDUCTOSYN absolute rotary, for a 2 or 4-degree turn and depending on whether it is a high resolution or low resolution encoder, the counting increment is 10000 units.
PITCH = 2º or 4º
SSIRESOL = 10000
The resolution of the axis will be; 2º or 4º / 10000 = 0.0002 or 0.0004 degrees.

DELAY ESTIMATE AT THE DRIVE.

AXDELAY
Parameter value at the drive.

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Standard / From 0 to 127 cycles.
Default value: Standard.
Associated variable: (V.)[ch].MPA.AXDELAY[set].xn

Parameter AXDELAY is an estimate of the drive's delay when applying the velocity command sent by the CNC. The CNC uses this parameter to compensate for the delay difference between the axes of a channel so the command reaches all the axes at the same time and the movement of the axes starts and ends at the same time. Depending on the delay set, the CNC calculates how far in advance it must send the velocity command to the drive.

AXDELAY	Meaning.
Standard.	For Fagor drives, that do not require any delay compensation.
0.	The CNC does not apply delay compensation at the drive.
1 to 127.	Delay cycles at the drive. The CNC automatically compensates for the delay differences between the axes of the channel.

The CNC also takes this delay into account when calculating the anticipation time (ADVINPOS mark), in thread repair and when calculating the braking distance for the probe.

CORRECTION OF BUS AND DRIVE DELAY.

TRANSDELAY
Correction of the delay inserted by the bus and the drive.

Parameter valid for linear and rotary axes.
Parameter valid for analog drive, Sercos and simulated.
*Possible values: From 0 to 4*LOOPTIME µs.*
Default value: 0.
Associated variable: (V.)[ch].MPA.TRANSDELAY[set].xn

Parameter TRANSDELAY may be used to compensate for the delay inserted by the bus itself and by the drive. This happens especially in the following cases.

- Analog axes (feedback through Can). About 2.5 sampling periods.
- Velocity Sercos axes. About 1.2 sampling periods.

This parameter affects the adjustment of the ACFWFACTOR y ACFGAIN and may be used to obtain better results in speed variations with less following error (axis lag).

2.7 Machine parameters for JOG mode.

HANDWHEEL CONFIGURATION.

NMPG
Number of handwheels connected to the CNC.

Possible values: From 0 to 12.
Default value: 0 (there are no handwheels).
Associated variable: (V.)MPMAN.NMPG

Electronic handwheels may be used to move the axes. Depending on the type of handwheel, The CNC may have general handwheels to move any axis or individual handwheels that will only move their associated axes. Several individual and general handwheels may be used at the same time, up to a total of 12 handwheels. The portable operator panel HBLS does not count as handwheel when setting this parameter.

To set the resolution of each handwheel, the distance it moves at each switch position, set the parameter MPGRESOL.

MANPG
Table of handwheels.

This parameter shows the list of available handwheels.

MANPG n
Handwheel configuration.

This parameter shows the table to configure the handwheels. Each table shows the following parameters to configure it.

MANPG n	
COUNTERTYPE	Handwheel feedback type.
COUNTERID	Feedback input for the handwheel.
MPGAXIS	Name of the axis associated with the handwheel.

COUNTERTYPE
Type of feedback input for the handwheel.

Possible values: Remote / Keyboard / Local / Sercos Counter.
Default value: Remote.
Associated variable: (V.)MPMAN.COUNTERTYPE[hw]
Parameter included in the MANPG table.

This parameter indicates the location of the handwheel feedback input; at the counter modules of the remote CAN groups, at the RCS-S modules (Sercos counter) at the keyboard or at the local feedback inputs of the central unit.

COUNTERID
Feedback input associated with the handwheel.

Possible values: From 1 to 40 if remote CAN feedback / from -1 to -9 if feedback at the keyboard / from 1 to 2 if local feedback / from 1 to 32 if feedback at RCS-S module.
Default value: 0.
Associated variable: (V.)MPMAN.COUNTERID[hw]
Parameter included in the MANPG table.

The handwheels may be connected through the keyboards (3 per keyboard) , and through the counters modules of the remote CAN groups (4 per module), through the RCS-S counter modules (4 per module) or through the local feedback inputs.

Handwheels connected through the keyboard.

Handwheels are numbered from -9 to -1. Handwheels are numbered following the logic order of the keyboards (rotary switch of the Power Supply element).

Keyboard order.	Numbering of the handwheels.
First keyboard in the bus.	-1 -2 -3
Second keyboard in the bus.	-4 -5 -6
Third keyboard in the bus.	-7 -8 -9

Handwheels connected through the remote CAN groups.

Handwheels are numbered from 1 to 40. The handwheels are numbered following the logic order of the remote groups (rotary switch of the Power Supply element). If there are several Counter modules in each group, the order is from top to bottom and from left to right.

Handwheels connected through the local feedback inputs.

Handwheels are numbered from 1 to 2. The central unit has two feedback inputs and this parameter indicates which one of them is used.

Handwheels connected through the RCS-S modules (Sercos counter).

Handwheels are numbered from 1 to 32. Handwheels are numbered according to the logical order of the RCS-.S modules (parameter SERCOUNTID). The feedback inputs of the first module will be 1 through 4, those of the second module 5 through 8 and so on.

MPGAXIS
Name of the axis associated with the handwheel.

Possible values: Any axes defined in AXISNAME.
Associated variable: (V.)MPMAN.MPGAXIS[hw]
Parameter included in the MANPG table.

This parameter sets whether the handwheel is associated with a particular axis (individual handwheel) or it can move any axis of the machine (general handwheel). To set an individual handwheel, define the name of the axis it is associated with. To set a general handwheel, do not assign any value to this parameter, leave it blank.

An axis may be moved indistinctly with its individual handwheel or with a general handwheel. To move an axis with a general handwheel, use the jog keypad to select the axis to be moved. To move an axis with an individual handwheel, no prior axis selection is needed.

- If there are several axes selected in handwheel mode, the general handwheel will move all of them.
- If an axis has been selected which has an individual handwheel selected with it, this axis may be moved with the general handwheel, with the individual one or with both at the same time. When using both handwheels simultaneously, the CNC will add or subtract the pulses provided by both handwheels depending on which direction they are turned.
- If the CNC has several general handwheels, any of them can move the axes selected in handwheel mode. When using several handwheels simultaneously, each axis involved will be applied the sum of the increments of all the handwheels.

CONFIGURE THE JOG KEYS.

JOGKEYDEF
JOGKEYBD2DEF
..
JOGKEYBD8DEF
Tables to configure the jog keys.

These parameters may be used to configure the jog keypad of each panel. Parameter JOGKEYDEF corresponds to the first jog panel, JOGKEYBD2DEF to the second one and so on.

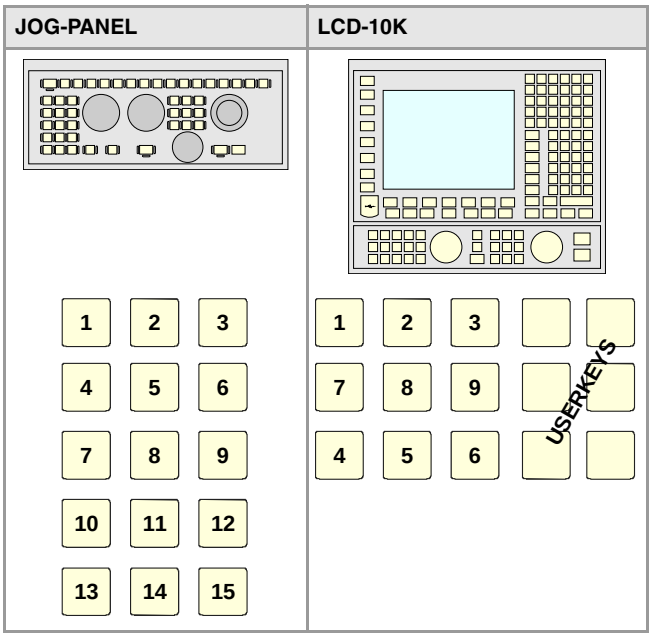
The CNC numbers the operator panels following the order (sequence) that they occupy in the CAN bus (Address switch). The first jog panel will be the one with the lowest number and so on.

JOGKEYDEF n
JOGKEYBD2DEF n
..
JOGKEYBD8DEF n
Axis and moving direction of each jog key.

Associated variable: (V.)MPMAN.JOGKEYDEF[jk] / (V.)MPMAN.JOGKEYkbDEF[jk]
Parameter included in the JOGKEYDEF table.

There is one table for each jog keypad. Each one of these parameter sets the function of each JOG key. The CNC always offers 15 parameters; if the jog keypad has fewer keys, the parameters that do not have any keys associated with them will be ignored.

The relationship between these parameters and the jog keys is the following.



The JOG keypad may consist of the following types of keys. Both types of keys may be defined at the same JOG keypad. To define the behavior of each key, assign to them one of the following values.

Keys.	Meaning.
X+ 7+	Keys to define the axis and the jogging direction. The parameter is set with a value between -1 and +16 (signed). The sign indicates the positive direction (+) or the negative direction (-) and the number corresponds to the logic axis, according to parameter AXISNAME.
X 7	Keys to define the axis to be jogged. The parameter is set with a value between 1 and 16 (unsigned) that corresponds to the logic axis according to parameter AXISNAME.
+ -	Keys to define the direction of the movement. The parameter is set with one of the "+" and "-" values to indicate the moving direction.
	Rapid key. The parameter is set with the "R" value.

JOGTYPE
Behavior of the jog keys.
Possible values: Pressed axis / Maintained axis.
Default value: Pressed axis.
Associated variable: (V.)MPMAN.JOGTYPE

This parameter is applied when the jog keyboard has different keys for selecting the axis and the jogging direction. In this case, jogging an axis requires activating both the axis key and the moving direction.

There are two options, depending on how the jog keyboard has been configured.

- With the "pressed axis" option, the axis will move while both keys are pressed, the axis key and the direction key.
- With the "maintained axis" option, pressing the axis key will select it. The axis will move while the direction key is kept pressed. To de-select the axis, press [ESC] or [STOP].

CONFIGURE THE USER KEYS AS JOG KEYS

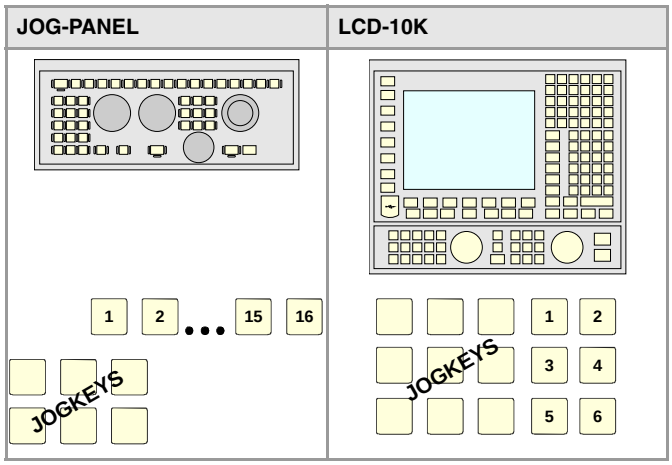
USERKEYDEF
USERKEYBD2DEF
..
USERKEYBD8DEF
Tables to configure the user keys as jog keys.

These parameters may be used to configure the user keys of each panel as jog keys. Parameter USERKEYDEF corresponds to the first jog panel, USERKEYBD2DEF to the second one and so on.

USERKEYDEF n
USERKEYBD2DEF n
..
USERKEYBD8DEF n
Axis and moving direction of each user key.
Associated variable: (V.)MPMAN.USERKEYDEF[uk] / (V.)MPMAN.USERKEYkbDEF[uk]
Parameter included in the USERKEYDEF table.

There is one table for each jog keypad. Each one of these parameter sets the function of each user key. The CNC always offers 16 parameters; if the jog keypad has fewer keys, the parameters that do not have any keys associated with them will be ignored.

The relationship between these parameters and the user keys is the following.



The meaning of this parameter is similar to that of machine parameter JOGKEYBDkbDEF. To define the behavior of each key, assign to them one of the following values:

- They are defined with a value between 1 and 16 (signed) for the keys that define the axis and the direction. The sign indicates the positive direction (+) or the negative direction (-) and the number corresponds to the logic axis, according to parameter AXISNAME.
- They are defined with a value between 1 and 16 (unsigned) for the keys that only define the axis.

- They are defined with the "+" and "-" values for the keys that only define the moving direction.
- They are defined with the "R" value for the rapid key.

The user keys defined this way behave like the jog keys whether they've been defined signed or unsigned and also respect the setting of machine parameter JOGTYPE.

HBLS PORTABLE OPERATOR PANEL.

HBLS
Activate the portable operator panel HBLS.
Possible values: Yes / No.
Default value: No.
Associated variable: (V.)MPMAN.HBLS

This parameter indicates whether there is a portable panel HBLS connected to the CNC through the serial line or not. In order to be able to use the portable operator panel HBLS, the serial line must be configured as RS422 (parameter RSTYPE). The portable operator panel HBLS must not be considered when counting the handwheels of the system (parameter NMPG).

2.7.1 Example of how to set the handwheels and jog keys.

Handwheel setting.

On a machine with X, Y, Z and A axes, we would like an individual handwheel for the X axis, an individual handwheel for the Y and a general handwheel for the Z and A axes.

- Individual handwheel for the X axis.

  Feedback input: keyboard (MPG1).

  Disk graduated with 100 lines

  Pulses/ turn: 100

  Resolutions (X): 0.001, 0.01, 0.1
- Individual handwheel for the Y axis.

  Feedback input: keyboard (MPG2).

  Disk graduated with 100 lines

  Pulses/ turn: 200

  Resolutions (Y): 0.001, 0.01, 0.1
- General handwheel for the rest of the axes (Z, A).

  Feedback input: Counter (X1).

  Disk graduated with 100 lines

  Pulses/ turn: 100

  Resolutions (Z): 0.001, 0.01, 0.1

  Resolutions (A): 0.01, 0.1, 1

Parameter setting:

Machine parameter.	Value.	Meaning.
NMPG	3	3-handwheel system.

X axis handwheel (MANPG 1).

Machine parameter.	Value.	Meaning.
COUNTERTYPE	Keyboard.	Handwheel connected to the keyboard.
COUNTERID	-1	MPG1 connector.
MPGAXIS	X	Associated axis.
MPGRESOL 1	0.001	Resolution 0.001
MPGRESOL 10	0.01	Resolution 0.01
MPGRESOL 100	0.1	Resolution 0.1

Y axis handwheel (MANPG 2).

Machine parameter.	Value.	Meaning.
COUNTERTYPE	Keyboard.	Handwheel connected to the keyboard.
COUNTERID	-2	MPG2 connector.
MPGAXIS	Y	Associated axis.
MPGRESOL 1	0.0005	Resolution 0.001
MPGRESOL 10	0.005	Resolution 0.01
MPGRESOL 100	0.05	Resolution 0.1

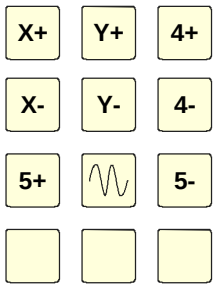
Handwheel associated with the rest of the axes (Z + A).

Machine parameter.	Value.	Meaning.
COUNTERTYPE	Remote.	Handwheel connected to the remote modules.
COUNTERID	1	Feedback input X1.
MPGAXIS		Handwheel associated with the rest of the axes.
Z- MPGRESOL 1	0.001	Z axis. Resolution 0.001
Z- MPGRESOL 10	0.01	Z axis. Resolution 0.01
Z- MPGRESOL 100	0.1	Z axis. Resolution 0.1
A- MPGRESOL 1	0.01	A axis. Resolution 0.01
A- MPGRESOL 10	0.1	A axis. Resolution 0.1
A- MPGRESOL 100	1	A axis. Resolution 1

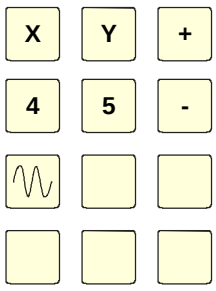
Setting the jog keys.

On a machine with the X, Y, U, V axes defined as AXISNAME 1, 2, 3, 4, we would like to jog the X and Y axes with the keys that have the same names, the U axis with the keys for the 4th axis and the V axis with those of the 5th axis.

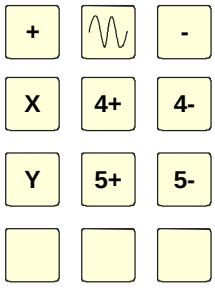
Example 1: JOG-PANEL keyboard.

Jog keyboard.	JOGKEYDEF	Key.	Value.
	1	[X+]	1+
	2	[Y+]	2+
	3	[4+]	3+
	4	[X-]	1-
	5	[Y-]	2-
	6	[4-]	3-
	7	[5+]	4+
	8	[R]	R
	9	[5-]	4-
	10 - 15	- - -	

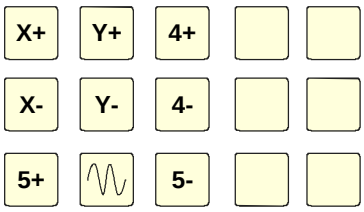
Example 2: JOG-PANEL keyboard.

Jog keyboard.	JOGKEYDEF	Key.	Value.
	1	[X]	1
	2	[Y]	2
	3	[+]	+
	4	[4]	3
	5	[5]	4
	6	[-]	-
	7	[R]	R
	8 - 15	- - -	

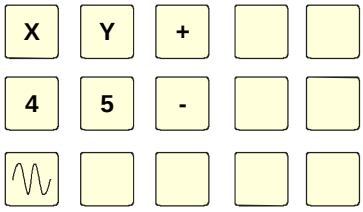
Example 3: JOG-PANEL keyboard.

Jog keyboard.	JOGKEYDEF	Key.	Value.
	1	[+]	+
	2	[R]	R
	3	[-]	-
	4	[X]	1
	5	[4+]	3+
	6	[4-]	3-
	7	[Y]	2
	8	[5+]	4+
	9	[5-]	4-
	10 - 15	- - -	

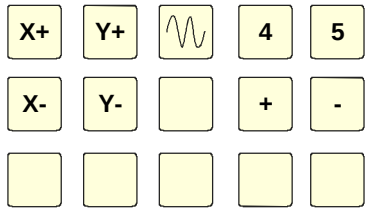
Example 4: LCD-10K keyboard.

Jog keyboard.	JOGKEYDEF	Key.	Value.
	1	[X+]	1+
	2	[Y+]	2+
	3	[4+]	3+
	4	[5+]	4+
	5	[R]	R
	6	[5-]	4-
	7	[X-]	1-
	8	[Y-]	2-
	9	[4-]	3-

Example 5: LCD-10K keyboard.

Jog keyboard.	JOGKEYDEF	Key.	Value.
	1	[X]	1
	2	[Y]	2
	3	[+]	+
	4	[R]	R
	5	- - -	
	6	- - -	
	7	[4]	3
	8	[5]	4
	9	[-]	-

Example 6: LCD-10K keyboard.

Jog keyboard.	JOGKEYDEF	Key.	Value.
JOGKEYS USERKEYS X+ Y+ 4 5 X- Y- + - 	1	[X+]	1+
	2	[Y+]	2+
	3	[R]	R
	4 - 6	- - -	
	7	[X-]	1-
	8	[Y-]	2-
	9	- - -	
	USERKEYDEF	**Key.**	**Value.**
	1	[4]	4
	2	[5]	5
	3	[+]	+
	4	[-]	-
	5 - 6	- - -	

2.8 Machine parameters for the M function table.

M FUNCTION TABLE.

MTABLESIZE
Number of table elements.

Possible values: from 0 to 200.
Default value: 50.
Associated variable: (V.)MPM.MTABLESIZE

This parameter indicates the number of M functions to be defined in the table. Each function may be assigned a subroutine and defined the type of synchronization.

It must be borne in mind that some auxiliary functions, besides what is indicated in this table, have a specific meaning when used in a CNC program. These functions are M00, M01, M02, M03, M04, M05, M06, M08, M09, M19, M30, M41, M42, M43 and M44.

DATA
M function table.

It shows the "M" function setting table. The following fields must be defined for each M function.

DATA	
MNUM	M function number.
SYNCHTYPE	Type of synchronism.
MPROGNAME	Name of subroutine associated with M function.
MTIME	Estimated time for an M function.
MPLC	Send the M function to the PLC during block search.
COMMENT	Description of the M function.

MNUM
M function number.

Possible values: from 0 to 65535.
Associated variable: (V.)MPM.MNUM[pos]
Parameter included in the DATA table.

This parameter indicates the number of the M function.

SYNCHTYPE
Type of synchronization.

Possible values: Without synchronization / Before-Before / Before-After / After-After.
Default value: Before - Before.
Associated variable: (V.)MPM.SYNCHTYPE[pos]
Parameter included in the DATA table.

Since the M functions may be programmed together with the movement of the axes, in the same block, it must be indicated when the function is to be sent out to the PLC and when it will be checked that it has been already executed (synchronization).

Type of synchronism.	Meaning.
Without synchronization.	M function not synchronized.
Before - Before.	The M function is sent to the PLC and synchronized before the movement.
Before - After.	The M function is sent to the PLC before the movement and synchronized after the movement.
After - After.	The M function is sent to the PLC and synchronized after the movement.

The M functions may be sent and/or synchronized before or after the movement.

- If an M function is used to turn on a lamp, it will be set without synchronization because there is no need to check that the lamp has indeed turned on.
- Functions M03 and M04 to start up the spindle, they should be executed and synchronized before the movement.
- Function M5 to stop the spindle, should be executed and synchronized after the movement.

MPROGNAME
Name of subroutine associated with M function.

Possible values: any text with up to 64 characters.
Default value: Without subroutine.
Associated variable: (V.)MPM.MPROGNAME[pos]
Parameter included in the DATA table.

The subroutines associated with the M functions must be located in the "C:\CNC8070 \MTB \SUB" folder.

When associating a subroutine with an M function, the execution of the M function executes the associated subroutine without executing the M function itself. To send the M function to the PLC, it must be programmed in the subroutine.

The type of synchronization of the M functions that have an associated subroutine must be "without synchronization" or "After-After". The CNC executes the associated subroutine after executing the programmed movement (if any).

To have different procedures in the subroutines associated with certain M functions, the code of each channel may be differentiated within the subroutine using the variable (V.)G.CNCHANNEL..

MTIME
Estimated time for an M function.

Possible values: From 0 to 1000000 ms.
Default value: 0 ms.
Associated variable: (V.)MPM.MTIME[pos]
Parameter included in the DATA table.

In EDISIMU mode, there is an option that allows calculating the time required to execute a part with the machining conditions established in the program. This parameter may be set to fine tune that calculation.

MPLC
Send the M function to the PLC during block search.

Possible values: Yes / No.
Default value: Yes.
Associated variable: (V.)MPM.MPLC[pos]
Parameter included in the DATA table.

This field indicates whether the M function must be sent to the PLC during block search.

COMMENT
Description of the M function.

Associated variable: (V.)MPM.COMMENT[pos]
Parameter included in the DATA table.

This field offers the possibility to associate a short description with the M function. This field is for information only; it is not used by the CNC.

The comments are saved in the file MComments.txt and it is possible to have one file per language. These files are saved in the folder C:\CNC8070 \MTB \data \Lang.

2.9 Machine parameters for kinetics.

2.9.1 Kinematics configuration.

Up to 6 different kinetics may be set for a machine. The type of kinetics to be applied must be defined for each of them. Channel machine parameter KINID indicates the kinematics number (not type) assumed by the CNC in that channel on power-up. To select a kinematics from the part-program, use the #KIN ID instruction.

Types of kinematics (predefined by Fagor or integrated by the OEM).

The CNC offers a number of pre-defined kinematics that may be easily configured by machine parameters. Besides these kinematics, the OEM can integrate 6 additional kinematics.

The OEM kinematics are integrated through a generic API and are then configured using these machine parameters. To include the kinematics for your machine, contact Fagor Automation.

Axes of the kinematics.

One kinematics may be active per channel. A kinematics may be configured by between 3 and 8 axes. All the axes making up the kinematics must belong to the same channel and must occupy the first positions in the following order.

Axis order.	Meaning.
1st axis	First main axis of the plane (abscissa).
2nd axis	2nd main axis of the plane (ordinate).
3rd axis	Longitudinal axis.
4th axis	Four axes of the kinematics.
5th axis	Fifth axis of the kinematics.
6th axis	Sixth axis of the kinematics.
7th axis	Seventh axis of the kinematics.
8th axis	Eighth axis of the kinematics.
9th axis and the next ones.	Rest of the axes.

The first 3 axes must be linear; spindle compensation will be applied on to them. The rest of the axes may be either rotary or linear, depending on the type of kinematics.

KINEMATICS CONFIGURATION.

NKIN
Number of kinematics defined.
Possible values: From 0 to 6.
Default value: 0.
Associated variable: (V.)MPK.NKIN

Up to 6 different kinetics may be set for a machine. The type of kinetics to be applied must be defined for each of them. Channel machine parameter KINID indicates the kinematics number (not type) assumed by the CNC in that channel on power-up. To select a kinematics from the part-program, use the #KIN ID instruction.

KINEMATIC n
Kinematics table.

It shows the table to define the kinematics of the system. The kinematics preset by Fagor have the following machine parameters to configure them.

DATA	
TYPE	Kinematics type.
TDATA n	Numeric parameter in decimal format.
TDATA_I n	Numeric parameter in integer format.

The OEM kinematics have the following machine parameters to configure them.

DATA	
TYPE	Kinematics type.
NKINAX	Number of axes of the kinematics.
PARAM_D_SIZE	Number of numeric parameters in decimal format.
TDATA n	Numeric parameter of the kinematics in decimal format.
PARAM_I_SIZE	Number of parameters of the kinematics in integer format.
TDATA_I n	Numeric parameter of the kinematics in integer format.
AUXCTE_SIZE	Size of the area of auxiliary variables.
KINDATA_SIZE	Size of the area for general purpose data.

TYPE
Kinematics type.

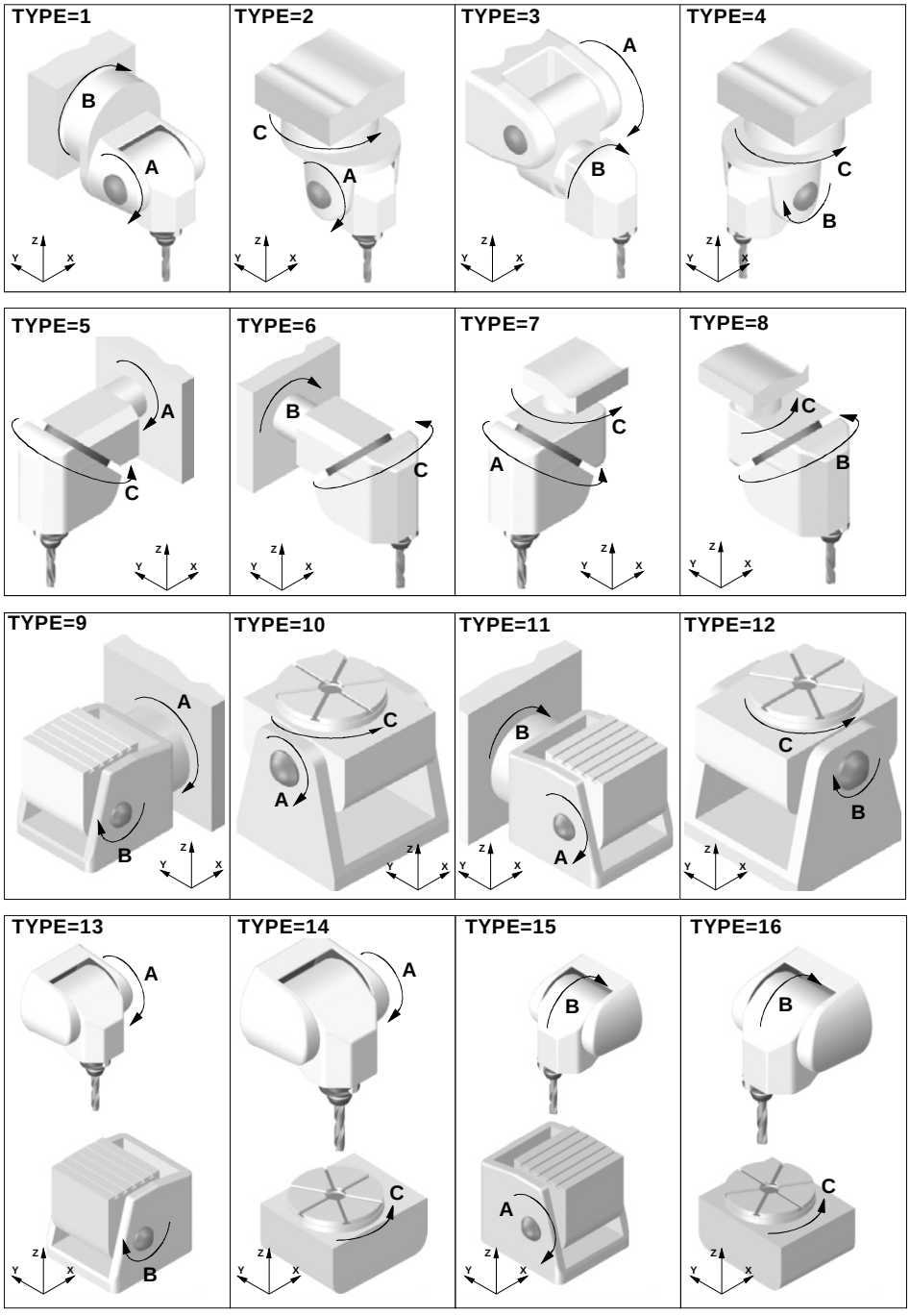
Possible values: From 0 to 99 (kinematics predefined by Fagor) / from 100 to 105 (OEM kinematics).
Default value: 0.
Associated variable: (V.)MPK.TYPE

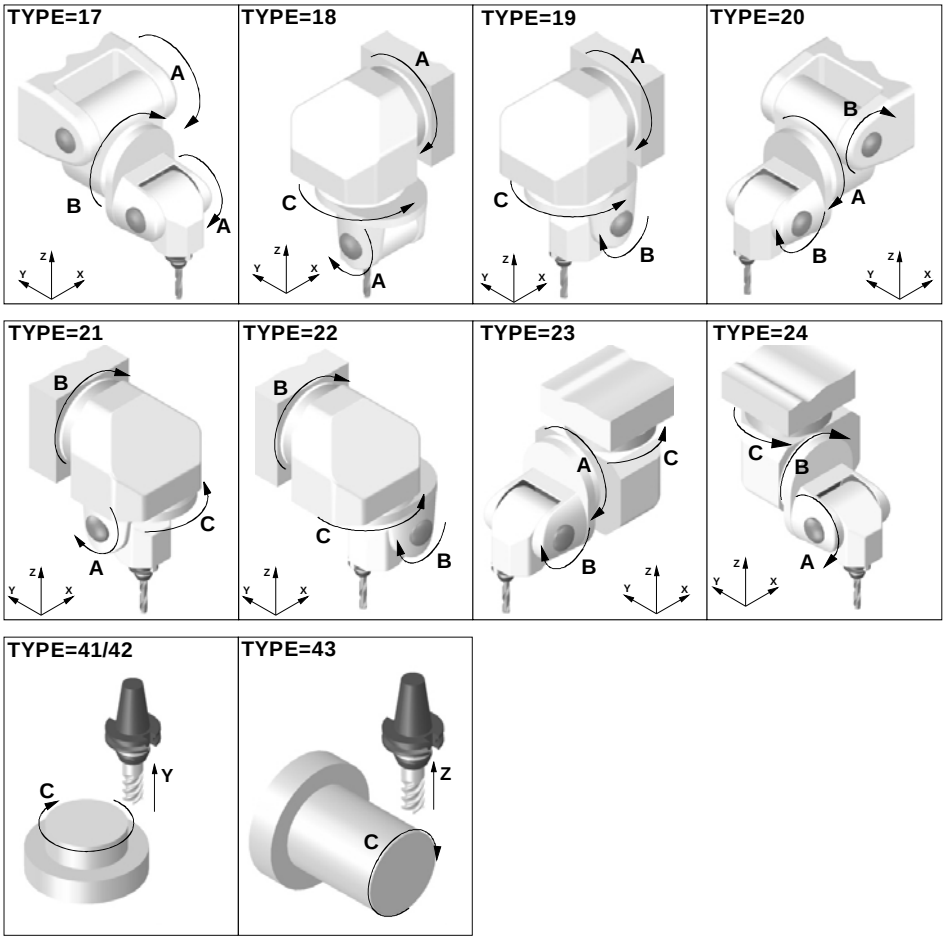
The first 100 of the 106 possible kinematics are predefined by Fagor and the other 6 may be integrated into the CNC by the manufacturer (OEM).

Kinematics predefined by Fagor.

TYPE	Kinematics type.
TYPE = 1	Orthogonal or spherical spindle head YX.
TYPE = 2	Orthogonal or spherical spindle head ZX.
TYPE = 3	Orthogonal or spherical spindle head XY.
TYPE = 4	Orthogonal or spherical spindle head ZY.
TYPE = 5	Swivel (angular) spindle XZ.
TYPE = 6	Swivel (angular) spindle YZ.
TYPE = 7	Swivel (angular) spindle ZX.
TYPE = 8	Swivel (angular) spindle ZY.
TYPE = 9	Rotary table AB.
TYPE = 10	Rotary table AC.
TYPE = 11	Rotary table BA.
TYPE = 12	Rotary table BC.
TYPE = 13	Spindle - table AB.
TYPE = 14	Spindle - table AC.
TYPE = 15	Spindle - table BA.
TYPE = 16	Spindle - table BC.
TYPE = 17	Orthogonal spindle with three rotary axes ABA.
TYPE = 18	Orthogonal spindle with three rotary axes ACA.
TYPE = 19	Orthogonal spindle with three rotary axes ACB.
TYPE = 20	Orthogonal spindle with three rotary axes BAB.

TYPE	Kinematics type.
TYPE = 21	Orthogonal spindle with three rotary axes BCA.
TYPE = 22	Orthogonal spindle with three rotary axes BCB.
TYPE = 23	Orthogonal spindle with three rotary axes CAB.
TYPE = 24	Orthogonal spindle with three rotary axes CBA.
TYPE = 41	"C" axis. Machining on the face of the part when ALIGNC = YES.
TYPE = 42	"C" axis. Machining on the face of the part when ALIGNC = NO.
TYPE = 43	"C" axis. Machining of the turning side of the part.



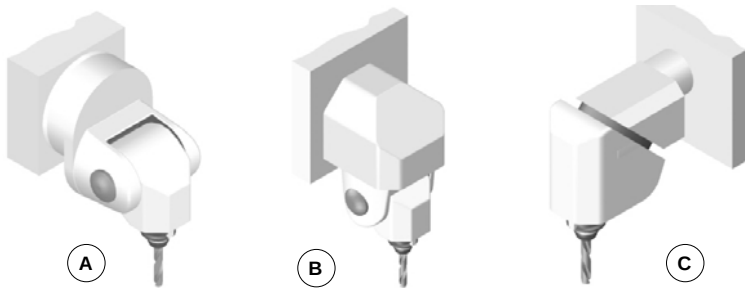


TDATA n
Numeric parameter in decimal format.
Possible values: Within ±999999999.
Default value: 0.
Associated variable: (V.)MPK.TDATAkin[nb]

TDATA_I n
Numeric parameter in integer format.
Possible values: Within ±2147483647.
Default value: 0.
Associated variable: (V.)MPK.TDATA_Ikin[nb]

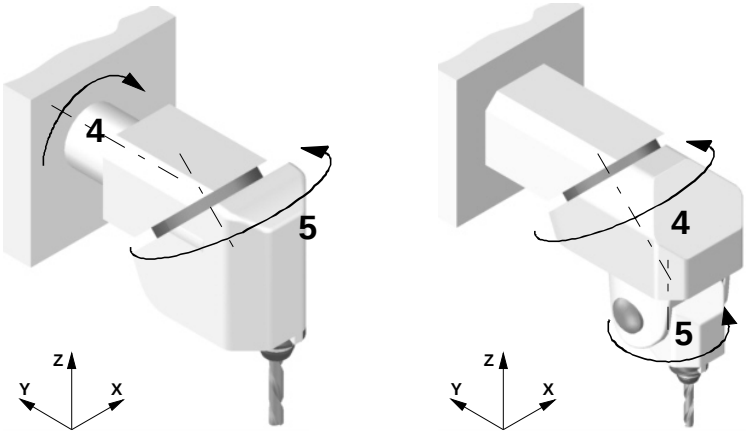
2.9.2 Definition of the spindle kinetics (types 1 through 8)

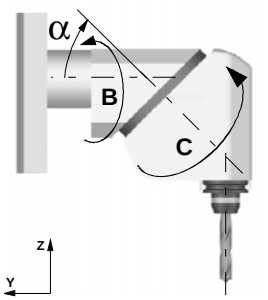
It is possible to control spheric spindle heads orthogonal and angular.



A Spherical spindle. B Orthogonal spindle. C Swivel (angular) spindle.

When having an angular spindle head, the main rotary axis (4) must rotate around one of the main axes (X, Y, Z) and the secondary or dragged axis (5) will form a particular angle. The left figure meets this condition, whereas in the right one the main rotary axis (4) does not rotate around the Y axis (it forms an angle with it).





From now on, all the explanations will assume that the main axes are X Y and Z and their associated rotary axes A, B and C.

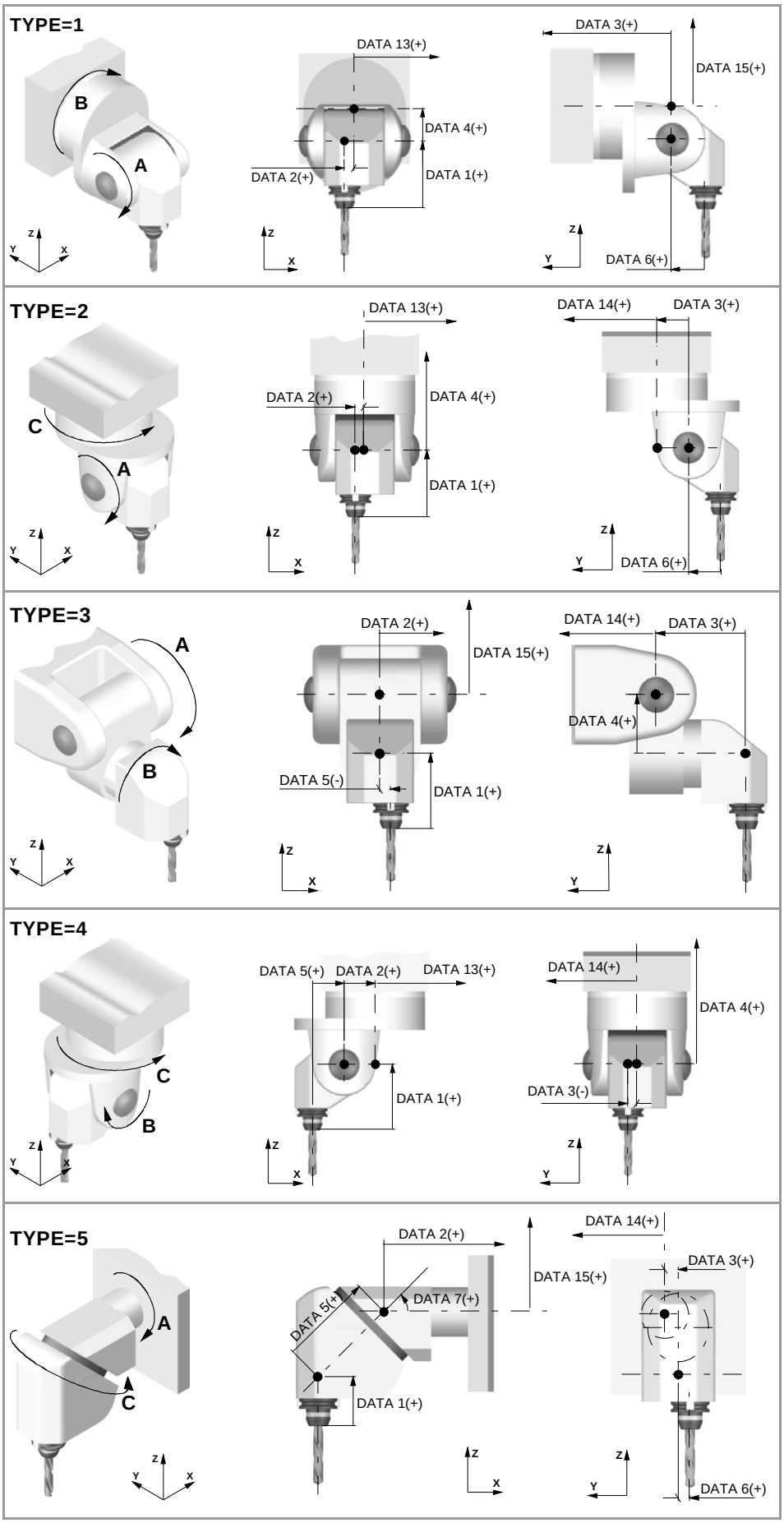
TDATA1··TDATA7 Spindle dimensions.

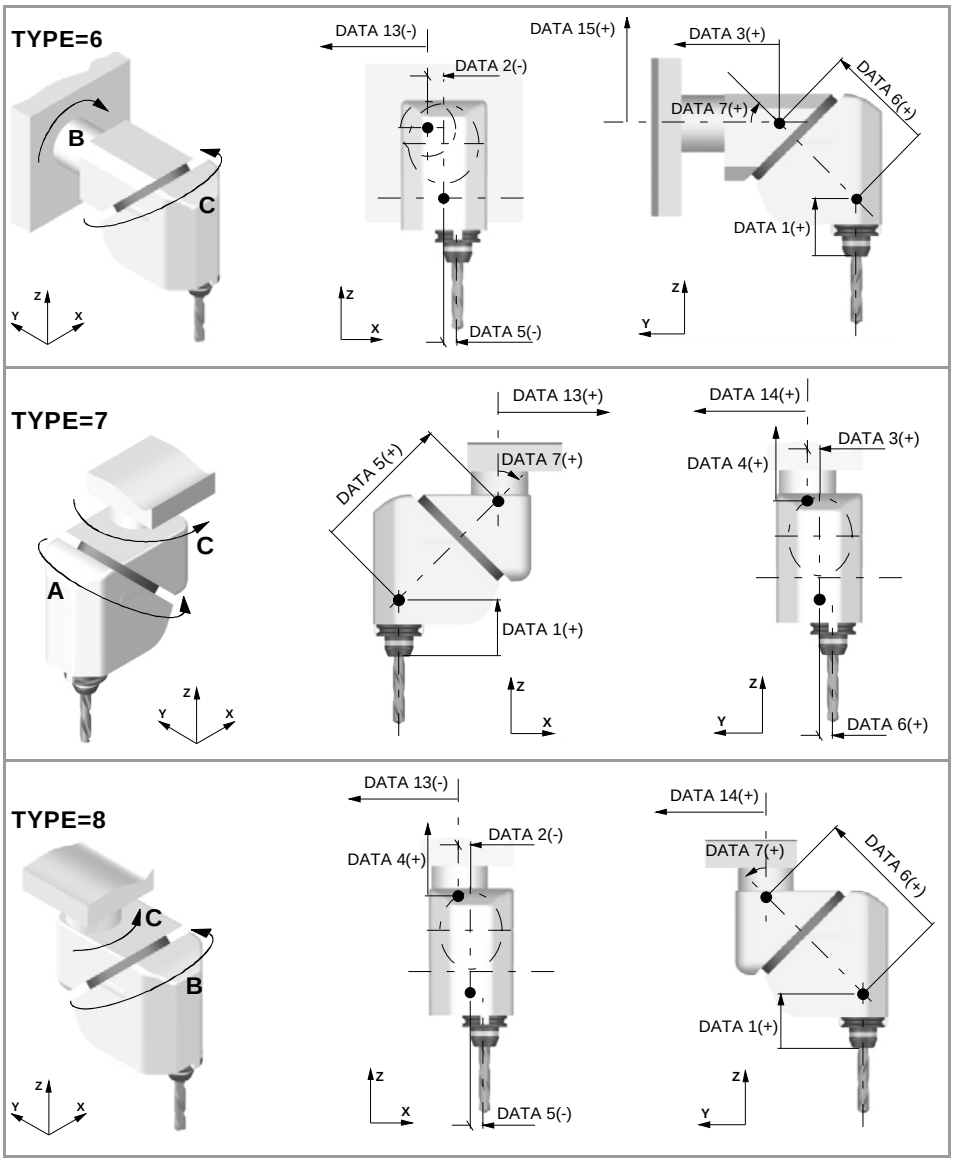
All the parameters need not be set. The parameters to be defined for each model and their meanings are listed here below. All the parameters may be set with a positive or negative value. The (+) sign in the illustrations indicates that the direction is assumed as positive.

TDATA1	It indicates the distance between the tip of the quill and the secondary rotary axis along the Z axis.
TDATA2	It indicates the distance between the secondary rotary axis and the main axis along the X axis.
TDATA3	It indicates the distance between the secondary rotary axis and the main axis along the Y axis.
TDATA4	It indicates the distance between the secondary rotary axis and the main axis along the Z axis.
TDATA5	It indicates the distance between the tool axis and the secondary rotary axis along the X axis.

TDATA6 It indicates the distance between the tool axis and the secondary rotary axis along the Y axis.

TDATA7 It indicates the angle between the main and secondary rotary axes on rotary spindles.





TDATA8
Rest position of the main rotary axis.

Possible values: Within ±99999.9999º.
Default value: 0.

See parameter TDATA9.

TDATA9
Rest position of the secondary rotary axis.

Possible values: Within ±99999.9999º.
Default value: 0.

The rest position is when the tool is perpendicular to the work plane (parallel to the longitudinal axis).

TDATA10
Turning direction of the main rotary axis.

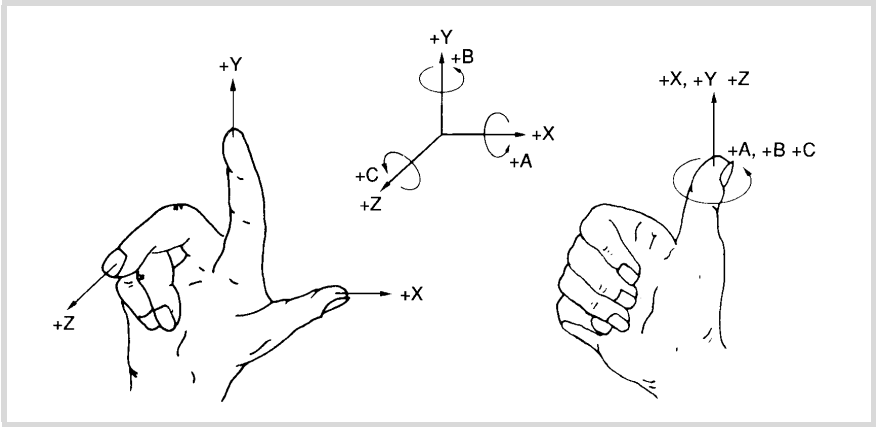
Possible values: 0 (by the standard DIN 66217) / 1 / opposite to the standard DIN 66217).
Default value: 0.

See parameter TDATA11.

TDATA11
Turning direction of the secondary rotary axis.

Possible values: 0 (by the standard DIN 66217) / 1 / opposite to the standard DIN 66217).
Default value: 0.

The direction of the XYZ axes, according to the DIN 66217 standard is easy to remember using the rule of the right hand. On rotary axes, the turning direction is established when bending your fingers (closing your hand) around the associated linear axis while your thumb is pointing in the positive direction of the linear axis.



TDATA12
Manual rotary axes or servo-controlled.

Possible values: 0 to 3.
Default value: 0.

This parameter indicates whether the rotary axes are manual or servo-controlled.

TDATA12	Meaning.
0	The two axes are servo-controlled.
1	Main axis manual and secondary axis servo-controlled.
2	Main axis servo-controlled and secondary axis manual.
3	The two axes are manuals.

TDATA 13··TDATA15
Spindle position referred to the machine reference point.

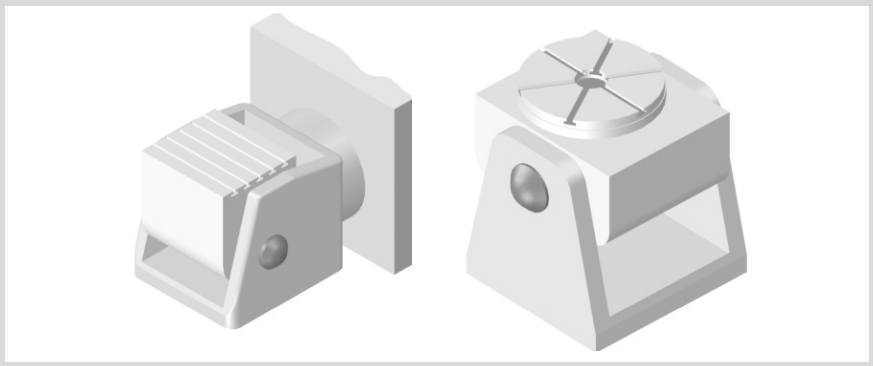
TDATA13 Distance between the main rotary axis and the reference point, along the X axis.

TDATA14 Distance between the main rotary axis and the reference point, along the Y axis.

TDATA15 Distance between the main rotary axis and the reference point, along the Z axis.

2.9.3 Definition of the table kinetics (Types 9 through 12).

The following types of rotary tables may be controlled.



TDATA2··TDATA5
Table dimensions.

The dimensions may be defined with a positive or negative value. The (+) sign in the illustrations indicates that the direction is assumed as positive.

TDATA2	It indicates the position of the secondary rotary axis or the intersection with the primary axis along the X axis.
TDATA3	It indicates the position of the secondary rotary axis or the intersection with the primary axis along the Y axis.
TDATA4	It indicates the position of the secondary rotary axis or the intersection with the primary axis along the Z axis.
TDATA5	It indicates the distance between the secondary and the main rotary tables.

TDATA8
Rest position of the main rotary axis.
Possible values: Within ±99999.9999º.
Default value: 0.

See parameter TDATA9.

TDATA9
Rest position of the secondary rotary axis.
Possible values: Within ±99999.9999º.
Default value: 0.

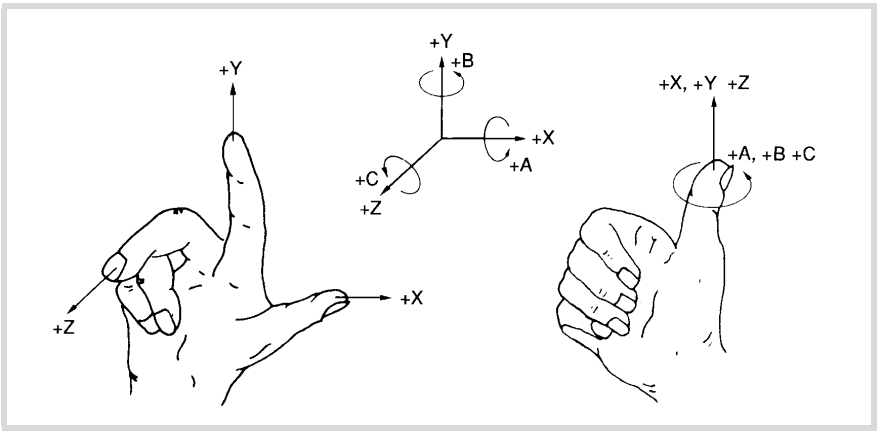
The rest position is when the tool is perpendicular to the work plane (parallel to the longitudinal axis).

TDATA10
Turning direction of the main rotary axis.
Possible values: 0 (by the standard DIN 66217) / 1 / opposite to the standard DIN 66217).
Default value: 0.

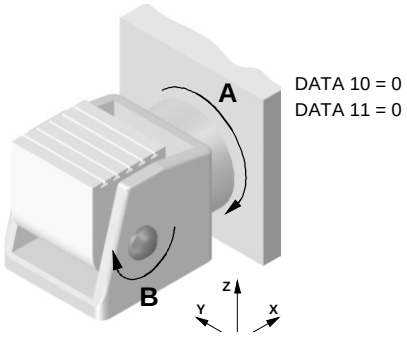
See parameter TDATA11.

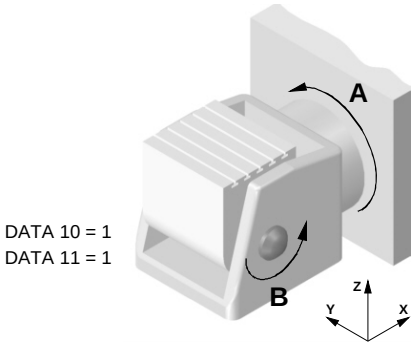
TDATA11
Turning direction of the secondary rotary axis.
Possible values: 0 (by the standard DIN 66217) / 1 / opposite to the standard DIN 66217).
Default value: 1.

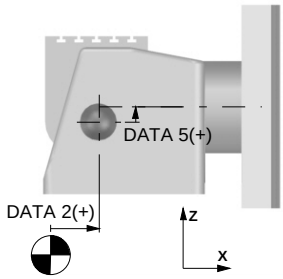
The direction of the XYZ axes, according to the DIN 66217 standard is easy to remember using the rule of the right hand. On rotary axes, the turning direction is established when bending your fingers (closing your hand) around the associated linear axis while your thumb is pointing in the positive direction of the linear axis. On kinematics with rotary axes on the table, the turning direction is set from the point of view of the tool; therefore, TDATA10 and TDATA11 must be set to 1.

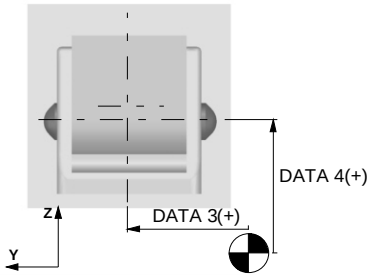


TYPE=9

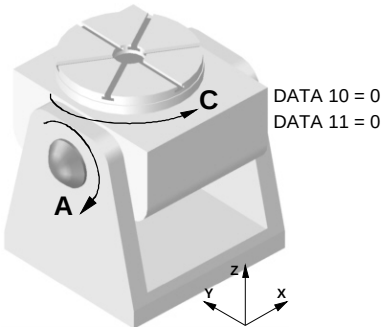


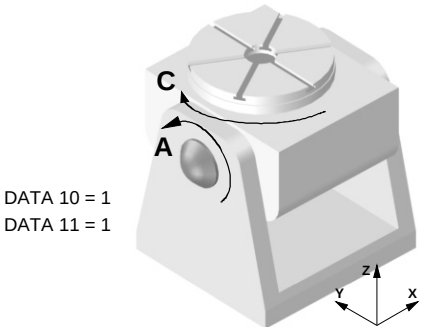


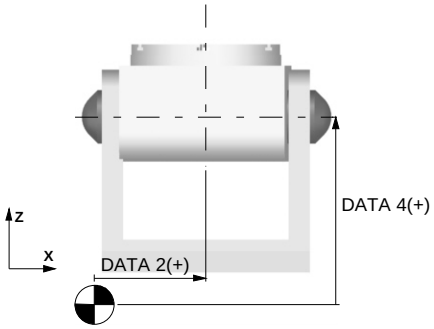


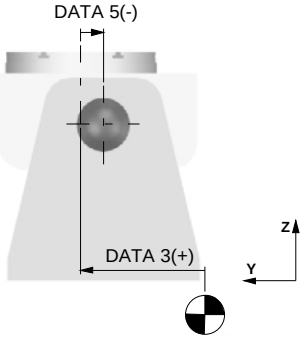


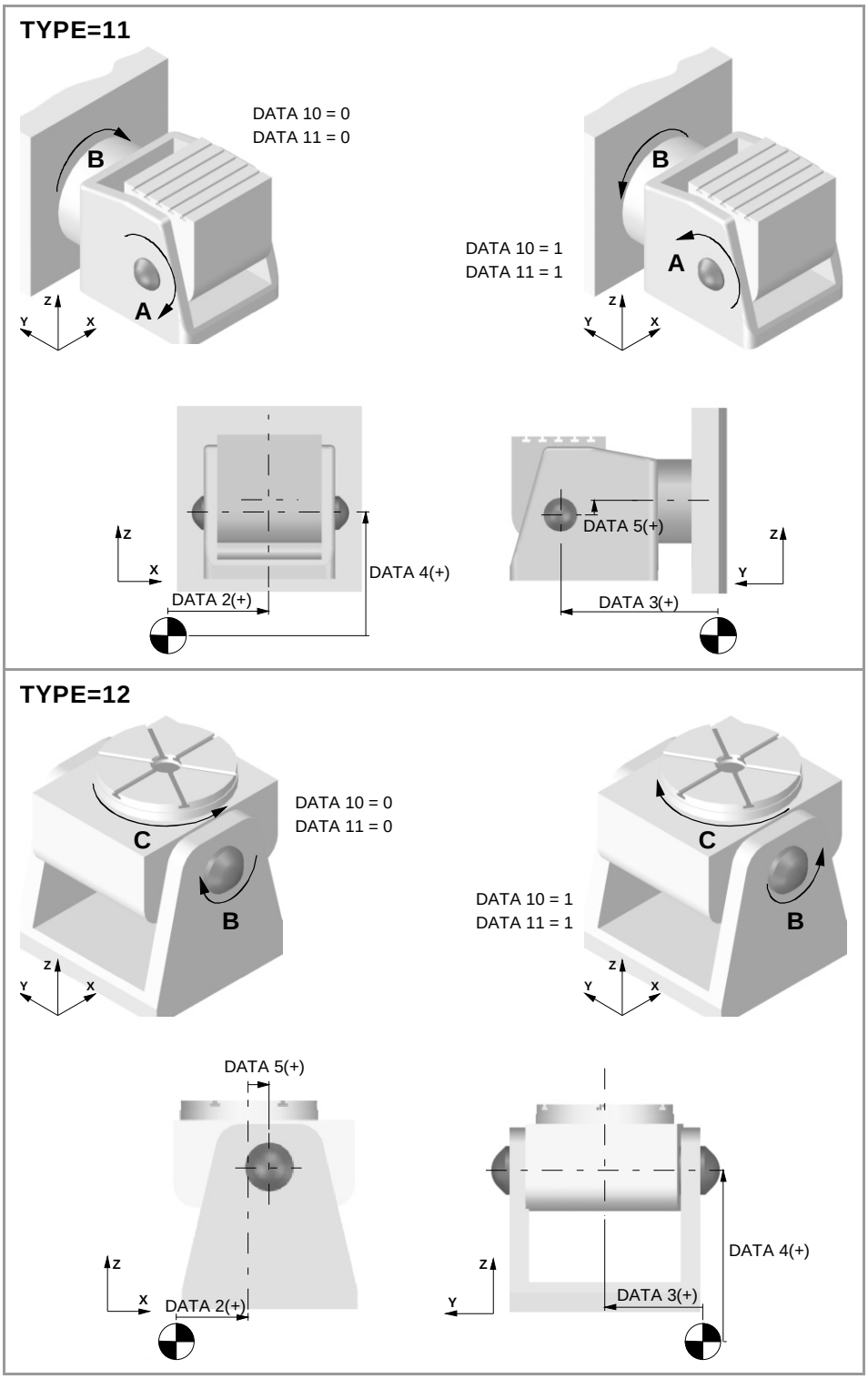
TYPE=10











TDATA12
Manual rotary axes or servo-controlled.

Possible values: 0 to 3.

Default value: 0.

This parameter indicates whether the rotary axes are manual or servo-controlled.

TDATA12	Meaning.
0	The two axes are servo-controlled.
1	Main axis manual and secondary axis servo-controlled.
2	Main axis servo-controlled and secondary axis manual.
3	The two axes are manuals.

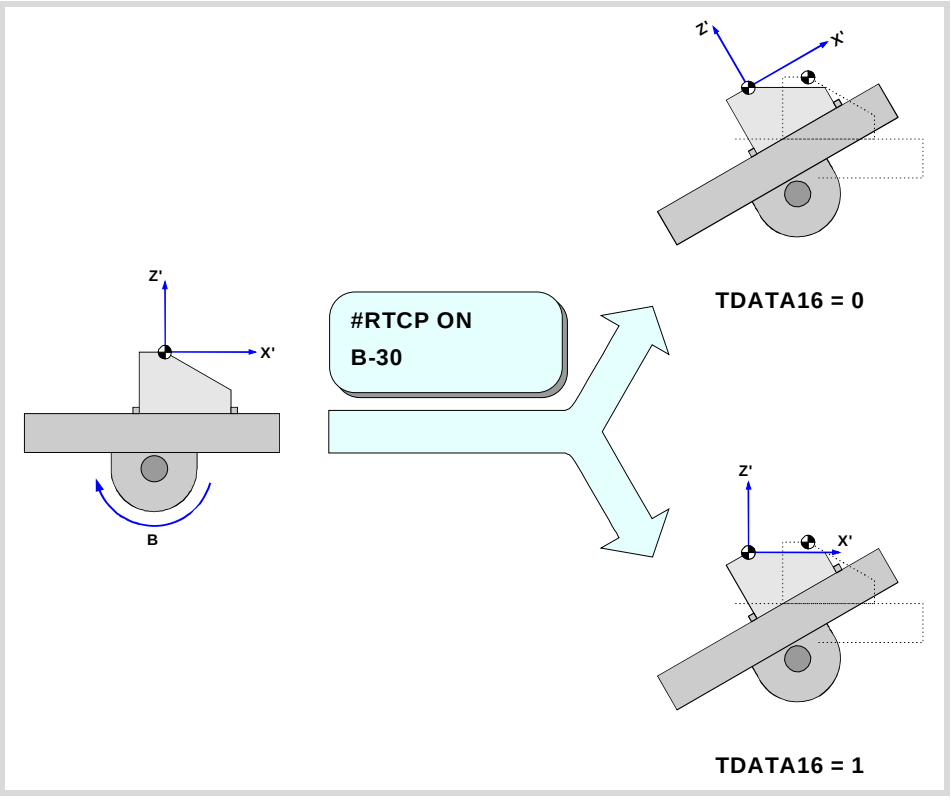
TDATA16

Rotate the part coordinate system when rotating the table.

Possible values: 0 / 1.
Default value: 0.

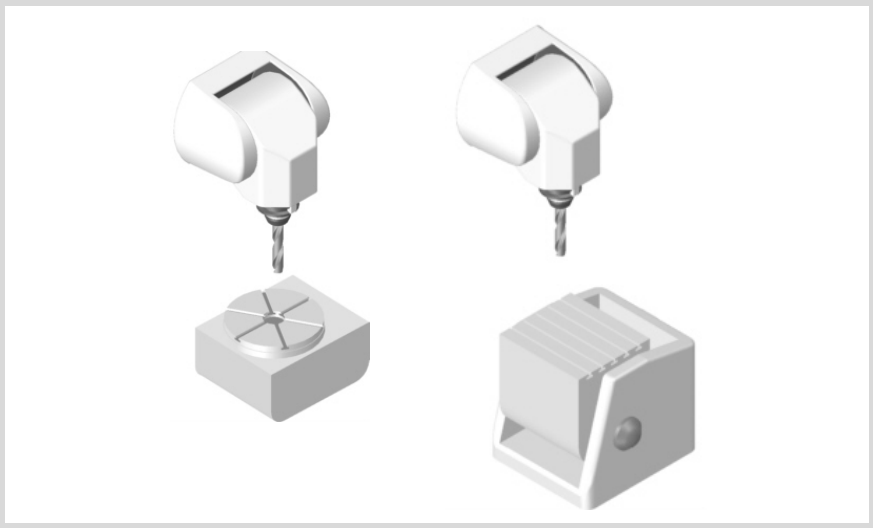
Having the RTCP mode active, this parameter determines whether the part coordinate system is fixed on the part and rotates with it or it stays parallel to the machine coordinate system.

TDATA16	Meaning.
0	When changing the part orientation, the position occupied by the tool tip on the part does not change. The coordinate system is fixed on the part and rotates with it.
1	When changing the part orientation, the position occupied by the point set as part zero does not change. The coordinate system stays parallel to the machine coordinate system.



2.9.4 Definition of the kinematics of the spindle - table (Types 13 through 16).

In this type of kinematics, one rotating axis is at the spindle and the other one at the table. The one at the spindle orients the tool and that of the work table orients the part.



The order of the axes in the channel where the kinematics is applied is:

- The first two axes correspond to the work plane.
- The third axis corresponds to the tool axis.
- The fourth axis corresponds to the rotary axis of the spindle.
- The fifth axis corresponds to the rotary axis of the table.

The type of kinematics is defined being the tool parallel to the third axis of the channel and the work plane perpendicular to the tool.

TDATA1··TDATA6
Spindle dimensions and table placement.

All of them need not be defined. The parameters to be defined for each kinematics and their meanings are listed here below. They may be defined with a positive or negative value. The (+) sign in the illustrations indicates the direction it assumes as positive.

TDATA1	It indicates the distance between the tip of the quill and the spindle rotary axis along the Z axis.
TDATA2	It indicates the distance between the tool axis and the spindle rotary axis along the X axis.
TDATA3	It indicates the distance between the tool axis and the spindle rotary axis along the Y axis.
TDATA4	It indicates the position of the table's rotary axis along the X axis.
TDATA5	It indicates the position of the table's rotary axis along the Y axis.
TDATA6	It indicates the position of the table's rotary axis along the Z axis.

TDATA8
Rest position of the main rotary axis.

Possible values: Within ±99999.9999º.
Default value: 0.

See parameter TDATA9.

TDATA9
Rest position of the secondary rotary axis.

Possible values: Within ±99999.9999º.
Default value: 0.

The rest position is when the tool is perpendicular to the work plane (parallel to the longitudinal axis).

TDATA10
Turning direction of the main rotary axis.

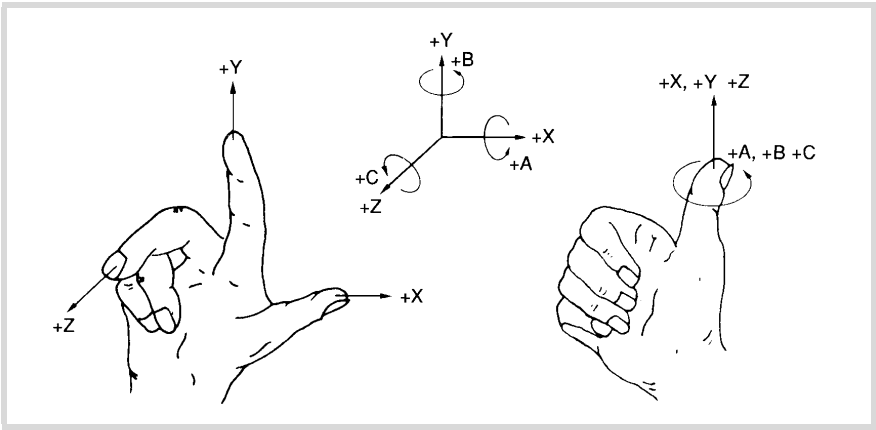
Possible values: 0 (by the standard DIN 66217) / 1 / opposite to the standard DIN 66217).
Default value: 0.

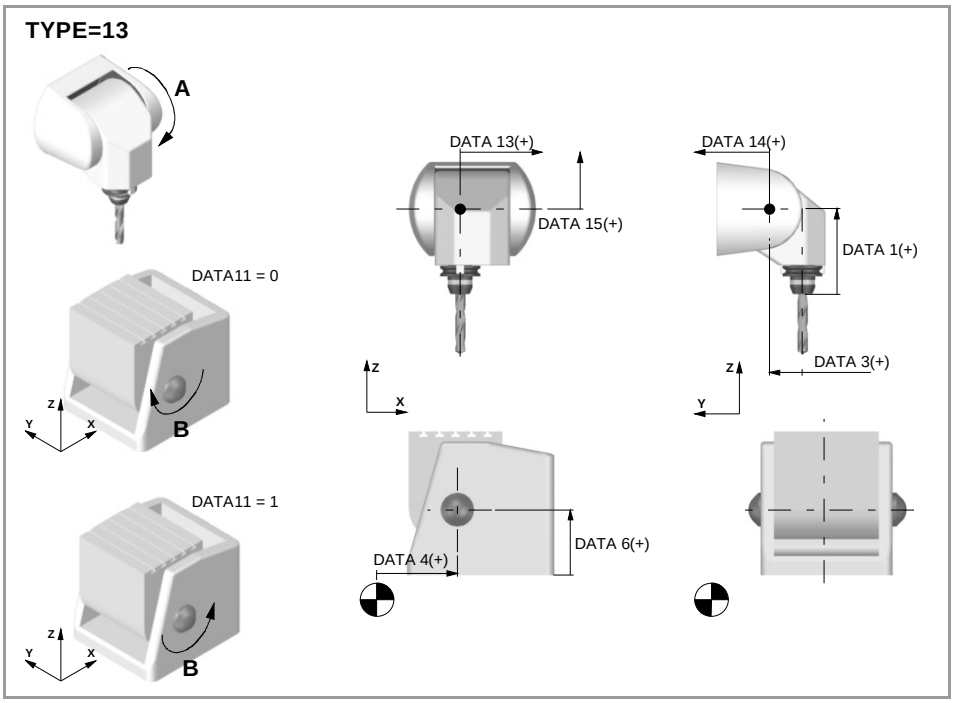
See parameter TDATA11.

TDATA11
Turning direction of the secondary rotary axis.

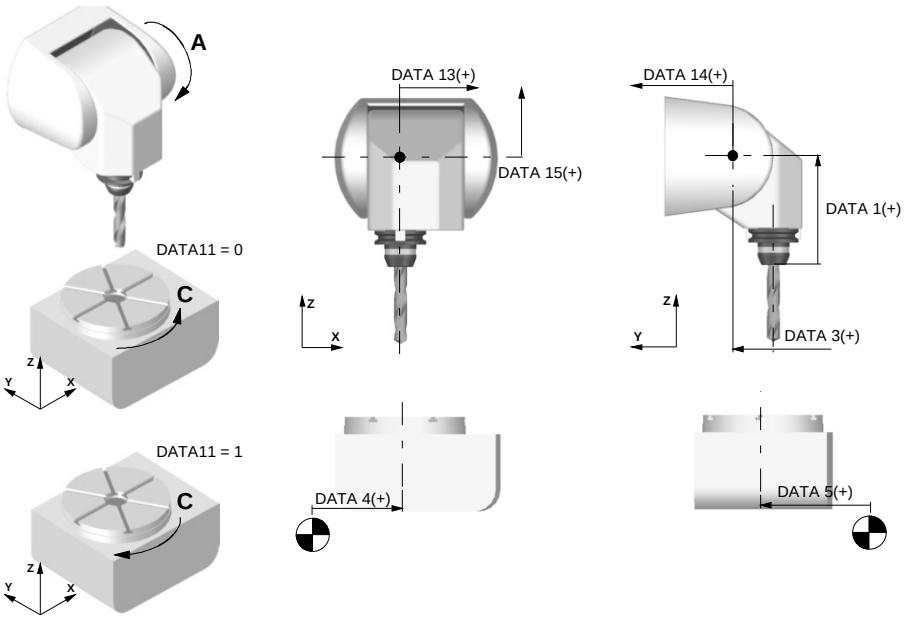
Possible values: 0 (by the standard DIN 66217) / 1 / opposite to the standard DIN 66217).
Default value: 0.

The direction of the XYZ axes, according to the DIN 66217 standard is easy to remember using the rule of the right hand. On rotary axes, the turning direction is established when bending your fingers (closing your hand) around the associated linear axis while your thumb is pointing in the positive direction of the linear axis. On kinematics with rotary axes on the table, the turning direction is set from the point of view of the tool; therefore, TDATA10 and TDATA11 must be set to 1.

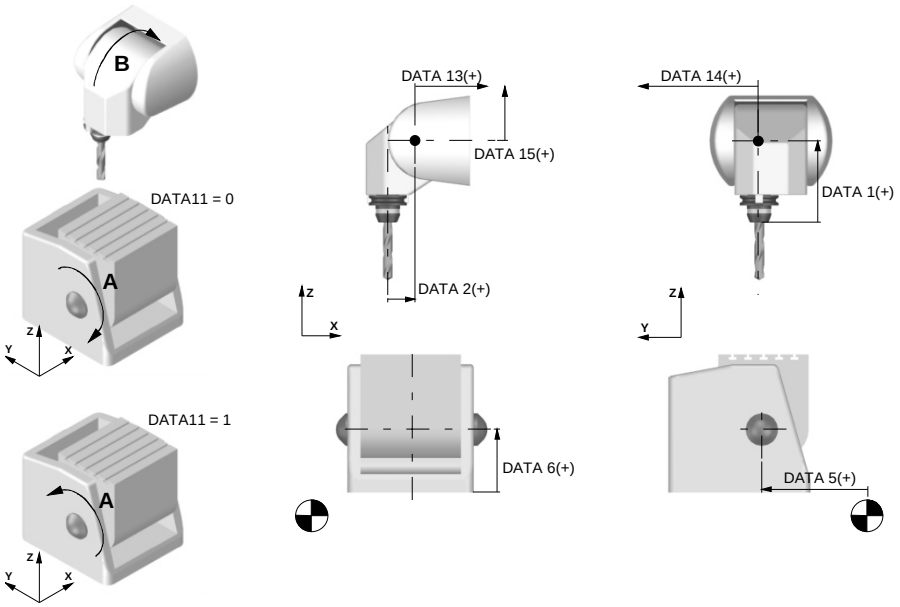




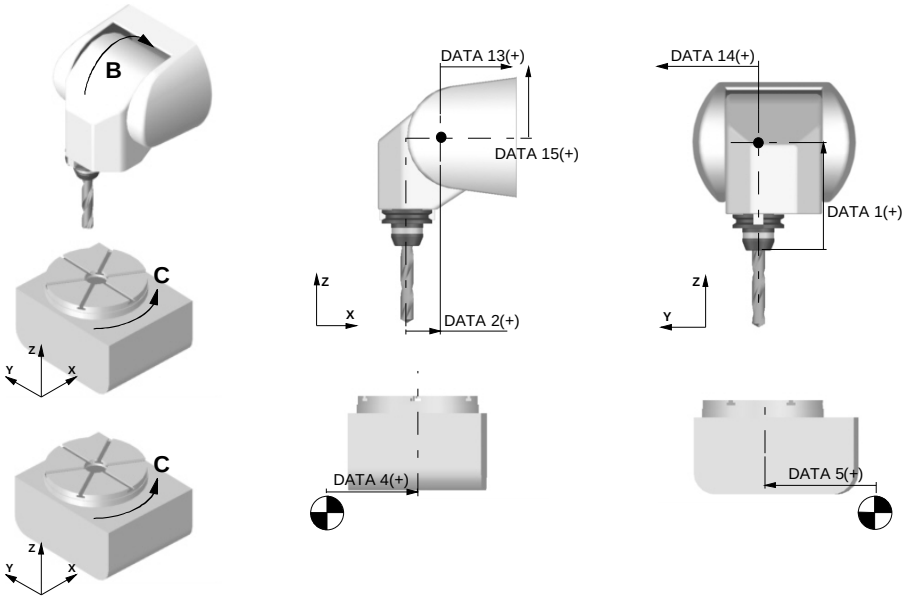
TYPE=14



TYPE=15



TYPE=16



FAGOR

CNC 8060
CNC 8065

(REF: 1402)

TDATA12
Manual rotary axes or servo-controlled.

Possible values: 0 to 3.
Default value: 0.

This parameter indicates whether the rotary axes are manual or servo-controlled.

TDATA12	Meaning.
0	The two axes are servo-controlled.
1	Main axis manual and secondary axis servo-controlled.
2	Main axis servo-controlled and secondary axis manual.
3	The two axes are manuals.

TDATA13··TDATA15
Spindle placement.

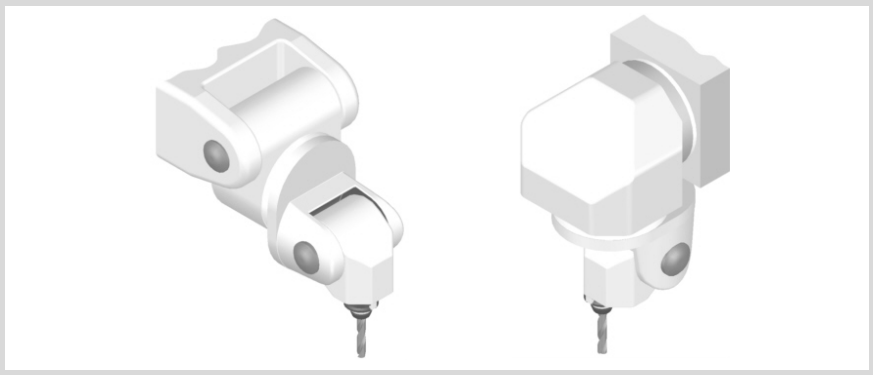
TDATA13 Distance defining the spindle placement, from the rotary axis, along the X axis.

TDATA14 Distance defining the spindle placement, from the rotary axis, along the Y axis.

TDATA15 Distance defining the spindle placement, from the rotary axis, along the Z axis.

2.9.5 Definition of the spindle kinetics (types 17 through 24)

It is possible to control orthogonal spindles with three rotary axes.



TDATA1··TDATA12
Spindle dimensions.

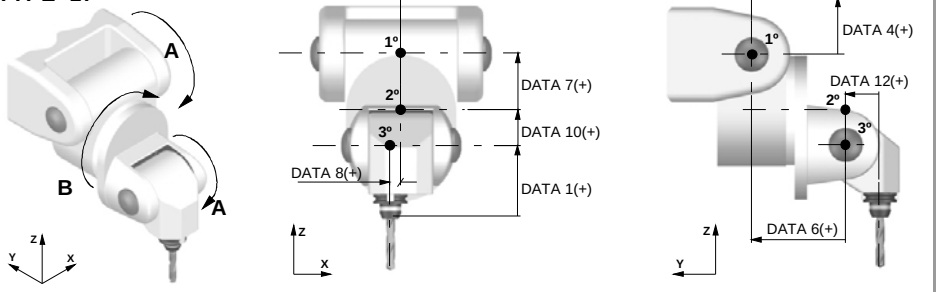
All the parameters need not be set. The parameters to be defined for each spindle model and their meanings are listed here below.

They may be defined with a positive or negative value. The (+) sign in the illustrations indicates that the direction is assumed as positive, numbers 1st, 2nd and 3rd indicate the rotation centers.

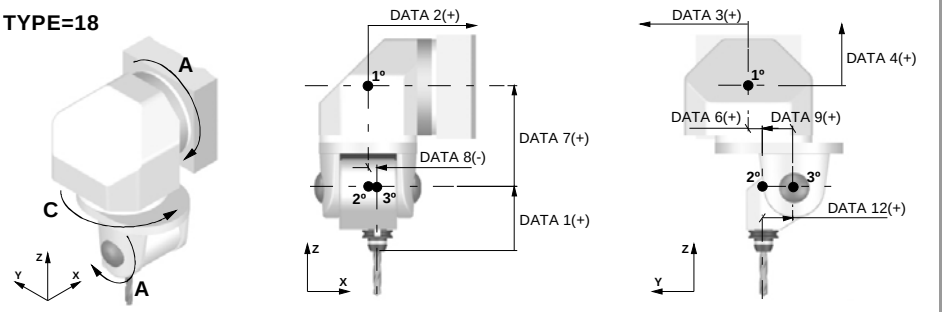
TDATA1	It indicates the distance between the tip of the quill and the third rotation center, the third rotary axis along the Z axis.
TDATA2	It indicates the distance of the spindle from the 1st rotation center to its fixture on the machine along the X axis.
TDATA3	It indicates the distance of the spindle from the 1st rotation center to its fixture on the machine along the Y axis.
TDATA4	It indicates the distance of the spindle from the 1st rotation center to its fixture on the machine along the Z axis.
TDATA5	It indicates the distance from the 2nd rotation center to the 1st rotation center along the X axis.
TDATA6	It indicates the distance from the 2nd rotation center to the 1st rotation center along the Y axis.
TDATA7	It indicates the distance from the 2nd rotation center to the 1st rotation center along the Z axis.
TDATA8	It indicates the distance from the 3rd rotation center to the 2nd rotation center along the X axis.
TDATA9	It indicates the distance from the 3rd rotation center to the 2nd rotation center along the Y axis.
TDATA10	It indicates the distance from the 3rd rotation center to the 2nd rotation center along the Z axis.
TDATA11	It indicates the distance from the shaft of the tool to the 3rd rotation center along the X axis.
TDATA12	It indicates the distance from the shaft of the tool to the 3rd rotation center along the Y axis.

The main rotary axis is the rotary axis of the spindle that when it rotates it drags the other two rotation axes. The secondary rotary axis is the rotary axis of the spindle that when it rotates it drags one rotation axis. The third rotary axis is the rotary axis of the spindle that when it rotates it does not drag any other rotation axis, only the tool.

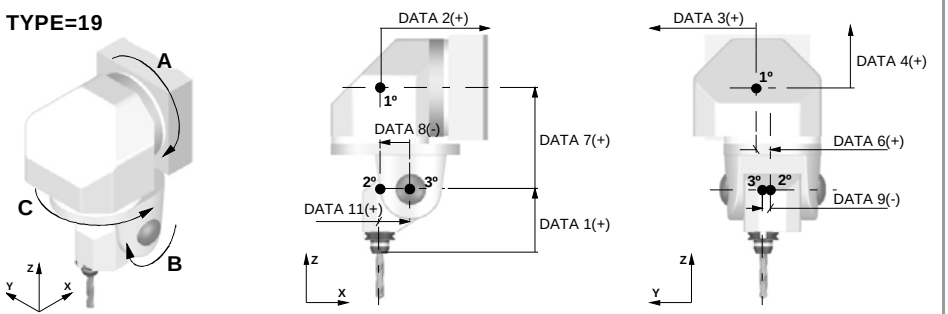
TYPE=17



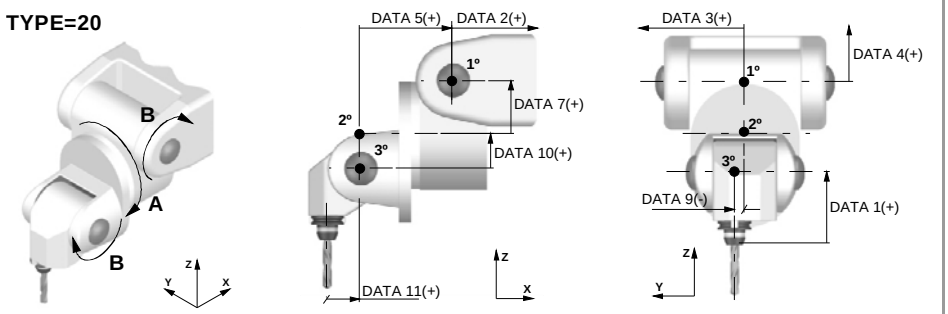
TYPE=18



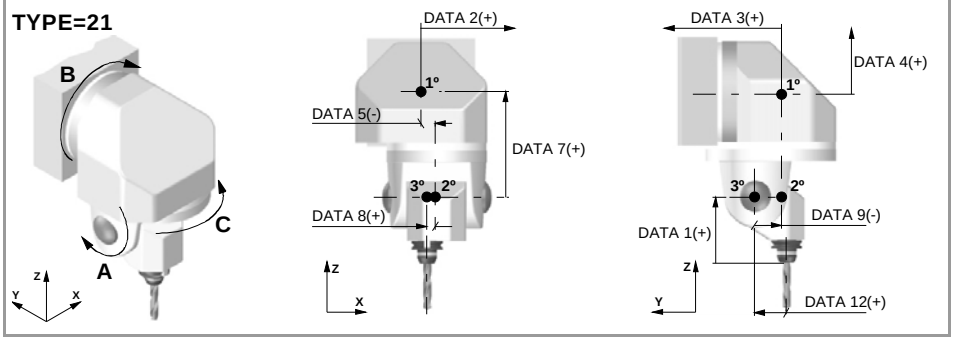
TYPE=19



TYPE=20



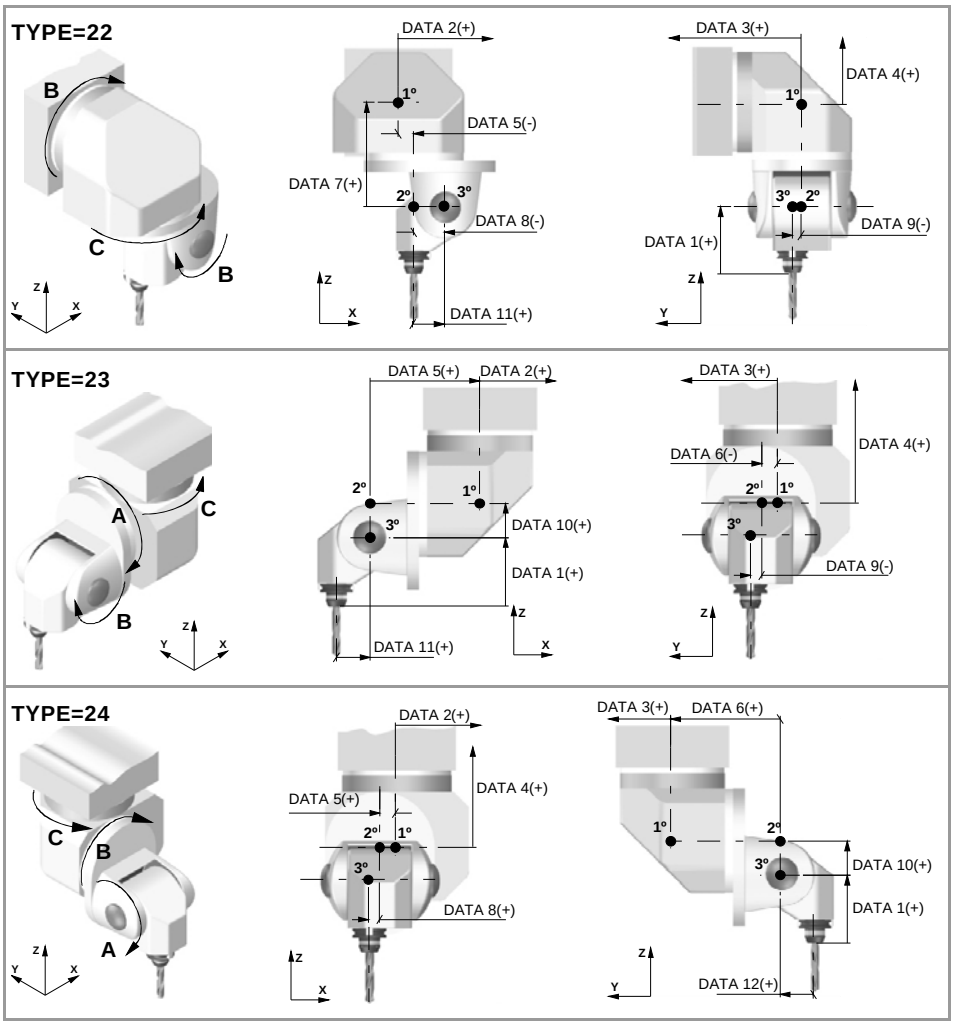
TYPE=21



FAGOR

CNC 8060
CNC 8065

(REF: 1402)



TDATA13
Rest position of the main rotary axis.

Possible values: Within ±999999999.0000º.
By default: 0º.

See parameter TDATA13.

TDATA14
Rest position of the secondary rotary axis.

Possible values: Within ±999999999.0000º.
By default: 0º.

See parameter TDATA14.

TDATA15
Rest position of the third rotary axis.

Possible values: Within ±999999999.0000º.
By default: 0º.

The rest position is when the tool is perpendicular to the work plane (parallel to the longitudinal axis).

TDATA16
Turning direction of the main rotary axis.

Possible values: 0 (by the standard DIN 66217) / 1 / opposite to the standard DIN 66217).
Default value: 0.

See parameter TDATA18.

TDATA17
Turning direction of the main rotary axis.

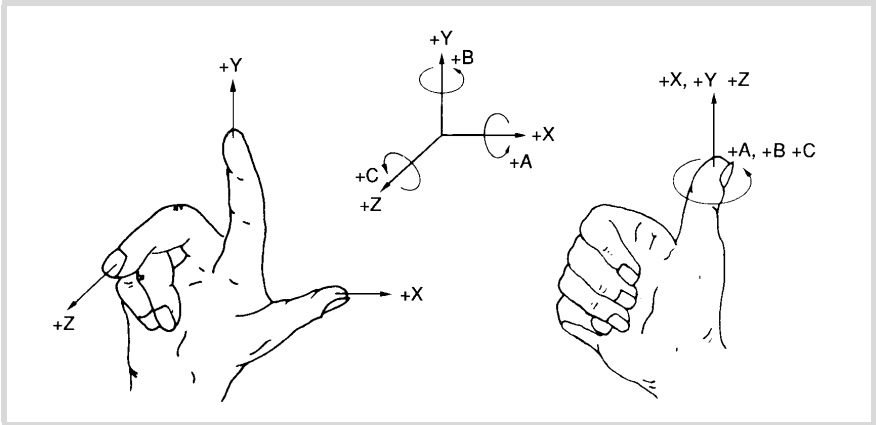
Possible values: 0 (by the standard DIN 66217) / 1 / opposite to the standard DIN 66217).
Default value: 0.

See parameter TDATA18.

TDATA18
Turning direction of the secondary rotary axis.

Possible values: 0 (by the standard DIN 66217) / 1 / opposite to the standard DIN 66217).
Default value: 0.

The direction of the XYZ axes, according to the DIN 66217 standard is easy to remember using the rule of the right hand. On rotary axes, the turning direction is established when bending your fingers (closing your hand) around the associated linear axis while your thumb is pointing in the positive direction of the linear axis.



TDATA19
Manual rotary axes or servo-controlled.

Possible values: 0 to 7.
Default value: 0.

This parameter indicates whether the rotary axes are manual or servo-controlled.

TDATA12	Meaning.
0	The three axes are servo-controlled.
1	Main axis manual and the rest servo-controlled.
2	Secondary axis manual and the rest servo-controlled.
3	Main axis and secondary axis manual, the third axis servo-controlled.
4	Third axis manual and the rest servo-controlled.
5	Main axis and third axis manual, the secondary axis servo-controlled.
6	Secondary axis and third axis manual, the main axis servo-controlled.
7	All the axes are manual.

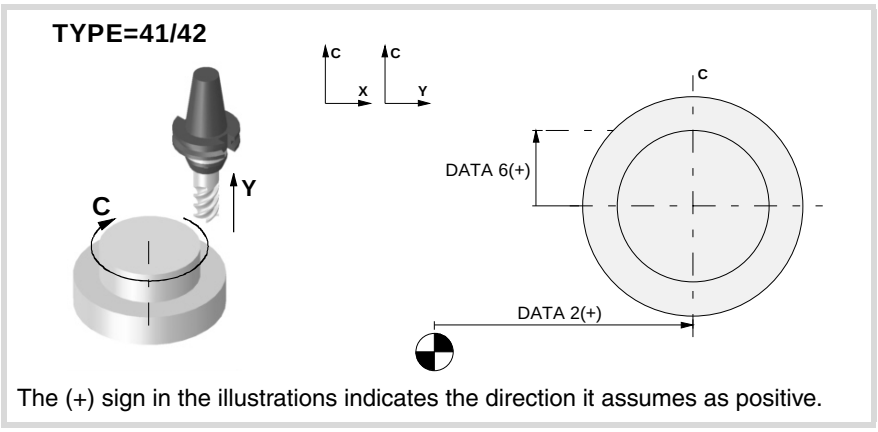
2.9.6 Definition of the –C– axis kinematics (Types 41 through 42).

In this type of kinematics, one must define the physical location of the rotary axis with respect to the linear axes. If these kinematics are defined, it assumes that the rotary axis coincides with the linear axis (e.g. the spindle of a lathe).

These kinematics are selected via part-program with function #FACE. If when executing this function the kinematics is not selected, the CNC will take it from the first kinematics type 41 or 42 defined in the table.

TDATA2
Rotary axis position.
Possible values: Within ±999999999.0000
Default value: 0.

It indicates the distance from the rotary axis to the linear axis on which it unrolls. When defined with a ·0· value, it assumes that the rotary axis coincides with the linear axis (e.g. the spindle of a lathe).



The (+) sign in the illustrations indicates the direction it assumes as positive.

TDATA4
Angular offset of the rotary axis.
Possible values: Within ±999999999.0000º.
Default value: 0.

TDATA5
Position of the rotation axis.
Possible values: 0 (the rotation axis is in part-zero) / 1 (the position of the rotating axis is given by DATA2).
Default value: 0.

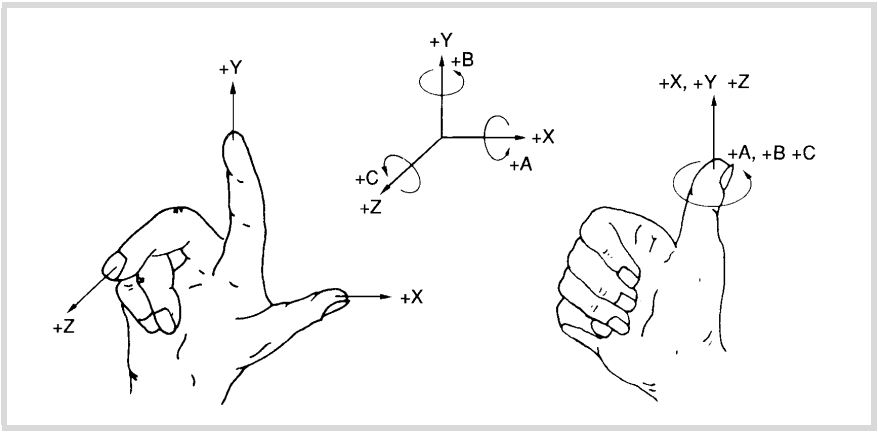
TDATA6
Tool misalignment with the C axis.
Possible values: Within ±999999999.0000
Default value: 0.

This parameter may be used to correct the misalignment of the tool. A tool is out of alignment when its zero position does not coincide with the rotation axis of the axis. The misalignment of the tool results in a circular area with a TDATA6 radius that cannot be machined.

TDATA10
Turning direction of the rotation axis.

Possible values: 0 (by the standard DIN 66217) / 1 / opposite to the standard DIN 66217).
Default value: 0.

On rotary axes, the turning direction is established when bending your fingers (closing your hand) around the associated linear axis while your thumb is pointing in the positive direction of the linear axis.



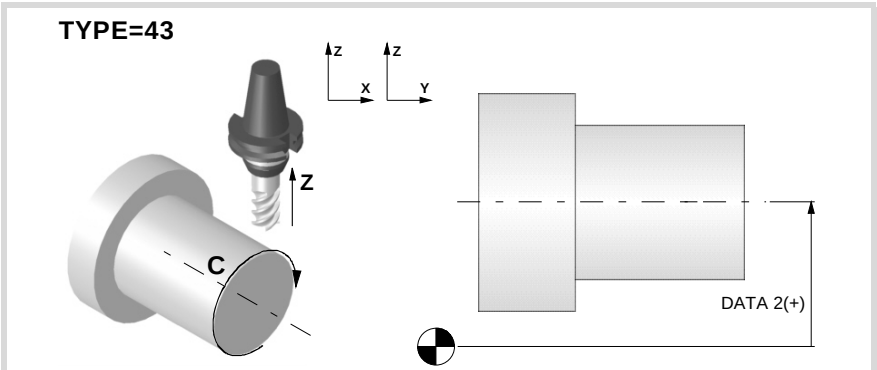
2.9.7 Definition of the –C– axis kinematics (Type 43) .

In this type of kinematics, one must define the physical location of the rotary axis with respect to the linear axes. If these kinematics are defined, it assumes that the rotary axis coincides with the linear axis (e.g. the spindle of a lathe).

These kinematics are selected via part-program with function #CYL. If when executing this function the kinematics is not selected, the CNC will take it from the first kinematics type 43 defined in the table.

TDATA2
Rotary axis position.
Possible values: Within ±999999999.0000
Default value: 0.

It indicates the position of the rotary axis along the tool axis. When defined with a ·0· value, it assumes that the rotary axis coincides with the linear axis (e.g. the spindle of a lathe).

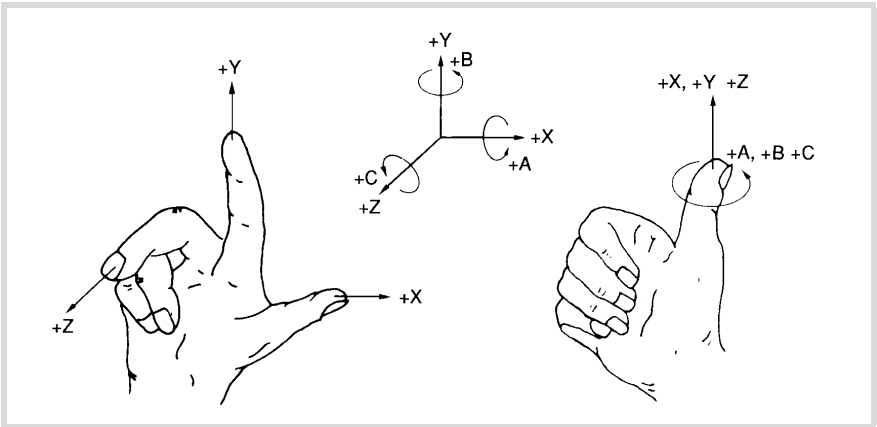


The (+) sign in the illustrations indicates the direction it assumes as positive.

TDATA4
Angular offset for the rotary axis.
Possible values: Within ±999999999.0000º.
Default value: 0.

TDATA10
Turning direction of the rotation axis.
Possible values: 0 (by the standard DIN 66217) / 1 / opposite to the standard DIN 66217).
Default value: 0.

On rotary axes, the turning direction is established when bending your fingers (closing your hand) around the associated linear axis while your thumb is pointing in the positive direction of the linear axis.



2.9.8 Definition of the OEM kinematics (Types 100 through 105).

In OEM kinematics, you must indicate the number of parameters, auxiliary variables and general purpose data being used.

NKINAX
Number of axes of the kinematics.
Possible values: From 0 to 8.
Default value: 0.
Associated variable: (V.)MPK.NKINAX[kin]

PARAM_D_SIZE
Number of parameters in decimal format.
Possible values: From 0 to 100.
Default value: 0.
Associated variable: (V.)MPK.PARAM_D_SIZE[kin]

TDATA1··100
Parameters for configuring the kinematics (decimal format).

There are 100 kinematics predefined by Fagor. For OEM kinematics, the number of parameters may be configured.

PARAM_I_SIZE
Number of parameters in integers.
Possible values: From 0 to 100.
Default value: 0.
Associated variable: (V.)MPK.PARAM_I_SIZE[kin]

TDATA_I1··100
Parameters for configuring the kinematics (integer format).

There are 100 kinematics predefined by Fagor. For OEM kinematics, the number of parameters may be configured.

AUXCTE_SIZE
Size of the area of auxiliary variables.
Possible values: From 0 to 1000 bytes.
Default value: 0.
Associated variable: (V.)MPK.AUXCTE_SIZE[kin]

KINDATA_SIZE
Size of the area for general purpose data.
Possible values: From 0 to 100000 bytes.
Default value: 0.
Associated variable: (V.)MPK.KINDATA_SIZE[kin]

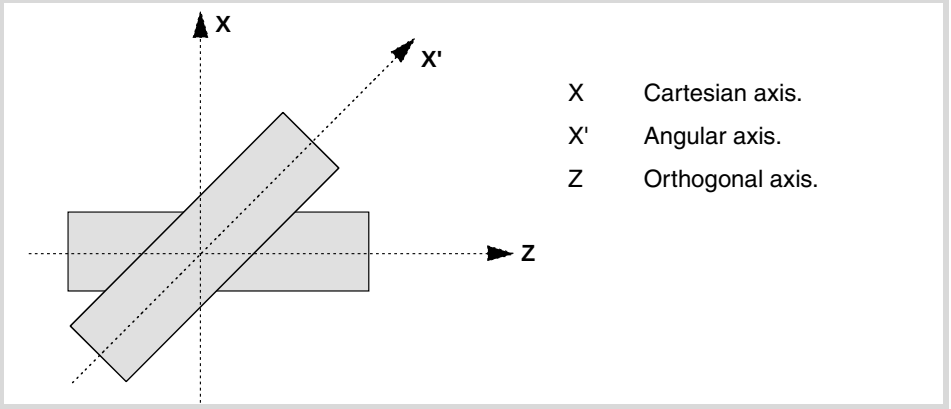
2.9.9 Configuration of angular transformations.

Up to 14 different angular transformations may be set for the same machine. The CNC assumes no transformation on power-up; the angular transformations are activated via part-program using the instruction #ANGAX ON. The angular transformation of an incline axis is kept active after a RESET or an M30.

What is an angular transformation of an incline axis?

The angular transformation of an incline axis may be used to program movements in Cartesian coordinates and execute them in a non-Cartesian plane; in other words, in a plane where the axes are not at 90º from each other.

On certain machines, the axes are configured in a Cartesian way, they are not perpendicular to each other. A typical case is the X axis of a lathe that due to sturdiness reasons is not perpendicular to the Z axis.



Programming in the Cartesian system (Z-X) requires activating an angular transformation of an inclined plane that converts the movements of the real (non-perpendicular) axes (Z-X'). This way, a movement programmed on the X axis is transformed into movements on the Z-X' axes; i.e. it then moves along the Z axis and the angular X' axis.

Considerations for the angular transformation of an incline axis.

The axes that make up the angular transformation must meet the following requirements:

- The angular transformation may be defined with any pair of axes of the system, but both axes must belong to the same channel in order to activate the transformation.
- Both axes must be linear.
- Both axes may be masters in a pair of slaved (coupled) axes or gantry axes.

Home search is not possible when the angular transformation is active.

If the angular transformation is active, the coordinates displayed will be those of the Cartesian system. Otherwise, it will display the coordinates of the real axes.

CONFIGURATION OF ANGULAR TRANSFORMATIONS.

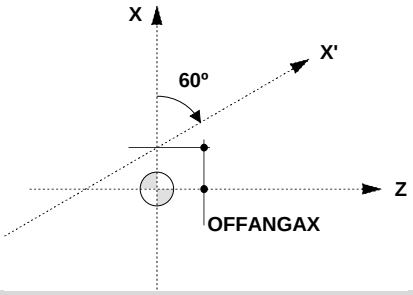
NANG
Number of angular transformations defined.
Possible values: From 0 to 14.
Default value: 0.
Associated variable: (V.)MPK.NANG

Up to 14 different angular transformations may be set for the same machine. The CNC assumes no transformation on power-up; the angular transformations are activated via part-program using the instruction #ANGAX ON. The angular transformation of an incline axis is kept active after a RESET or an M30.

ANGTR
Number of angular transformations.

It shows the configuration tables for angular transformations. The following parameters must be defined for each kinematics:

DATA	
ANGAXNA	Name of the angular axis (incline axis).
ORTAXNA	Name of the orthogonal axis.
ANGANTR	Angle between the Cartesian axis and the incline axis.
OFFANGAX	Offset of the origin of the angular transformation.



ANGAXNA X
ORTAXNA Z
ANGANTR 60º

ANGAXNA
Name of the angular axis (incline axis).

Associated variable: (V.)MPK.ANGAXNA[ang]

Any axis defined in the parameter "AXISNAME".

ORTAXNA
Name of the orthogonal axis.

Associated variable: (V.)MPK.ORTAXNA[ang]

Name of the axis perpendicular to the Cartesian axis, used in the angular transformation.

Any axis defined in the parameter "AXISNAME".

ANGANTR
Angle between the Cartesian axis and the incline axis.

Possible values: Within ±360.0000º.
Default value: 30 degrees.
Associated variable: (V.)MPK.ANGANTR[ang]

Angle between the Cartesian angle and the angular axis it is associated with. If its value is 0º, there is no need to do an angular transformation.

Positive angle when the angular axis has been rotated clockwise and negative if counterclockwise.

OFFANGAX
Offset of the origin of the angular transformation.

Possible values: Within ±99999.9999 mm / within ±3937.00787 inch.
Default value: 0.
Associated variable: (V.)MPK.OFFANGAX[ang]

Distance between machine zero and the origin of the coordinate system of the incline axis.

2.10 Machine parameters for the magazine.

TOOL MAGAZINE CONFIGURATION.

NTOOLMZ
Number of tool magazines.
Possible values: From 0 to 4.
Default value: 1.
Associated variable: (V.)TM.NTOOLMZ

Number of system's tool magazines.

Although each channel has its own tool management, the tool magazines are not associated with any particular channel. They are not associated with any particular spindle either.

GROUND
Ground tools are permitted (manual load).
Possible values: Yes / No.
Default value: No.
Associated variable: (V.)TM.MZGROUND[mz]

They are the ones not located in the magazine. When programming them, the CNC requests them to be inserted in the spindle.

MAGAZINE n
Tool magazine table.

This parameter shows the tables to define the magazine data. Each table shows the following machine parameters to configure it.

MAGAZINE n	
STORAGE	Parameters related to storage.
MANAGEMENT	Magazine management related parameters.

MAGAZINE DATA.

STORAGE
Parameters related to magazine.

This parameter shows or hides the magazine data. Each table shows the following machine parameters to configure it.

MAGAZINE n	
SIZE	Size of the tool magazine (number of pockets).
RANDOM	Random magazine.

SIZE
Size of the magazine (number of pockets).
Possible values: From 0 to 1000.
Default value: 20.
Associated variable: (V.)TM.MZSIZE[mz]
Parameter included in the MAGAZINE // STORAGE table.

This parameter indicates the capacity of the magazine without including the ground tools.

RANDOM
Random magazine.

Possible values: Yes / No.
Default value: No.
Associated variable: (V.)TM.MZRANDOM[mz]
Parameter included in the MAGAZINE // STORAGE table.

This parameter indicates whether the tools must always occupy the same position (non-random) or they may occupy any position (random).

TOOL MAGAZINE MANAGEMENT.

MANAGEMENT
Magazine management related parameters.

This parameter shows the data to configure the magazine management. Each table shows the following machine parameters to configure it.

MANAGEMENT	
TYPE	Type of magazine.
CYCLIC	Cyclic tool changer.
OPTIMIZE	Tool management.
M6ALONE	Action when executing an M06 without selecting a T.

TYPE
Magazine type.

Possible values: Asynchronous / Synchronous / Turret / Synchronous + 2 Arms / Synchronous + 1 Arm.
Default value: Synchronous.
Associated variable: (V.)TM.MZTYPE[mz]
Parameter included in the MAGAZINE // MANAGEMENT table.

This parameter indicates the type of magazine. See *"7.1 Types of tool magazine."* on page 343.

CYCLIC
Cyclic tool changer.

Possible values: Yes / No.
Default value: Yes.
Associated variable: (V.)TM.MZCYCLIC[mz]
Parameter included in the MAGAZINE // MANAGEMENT table.

A cyclic tool changer requires a tool change command (function M06) after searching a tool and before searching the next one. With a non-cyclic tool changer, it is possible to search for several tools in a row without necessarily having to make the actual tool change (M06 function).

OPTIMIZE
Tool management..

Possible values: Yes / No.
Default value: Yes.
Associated variable: (V.)TM.MZOPTIMIZED[mz]
Parameter included in the MAGAZINE // MANAGEMENT table.

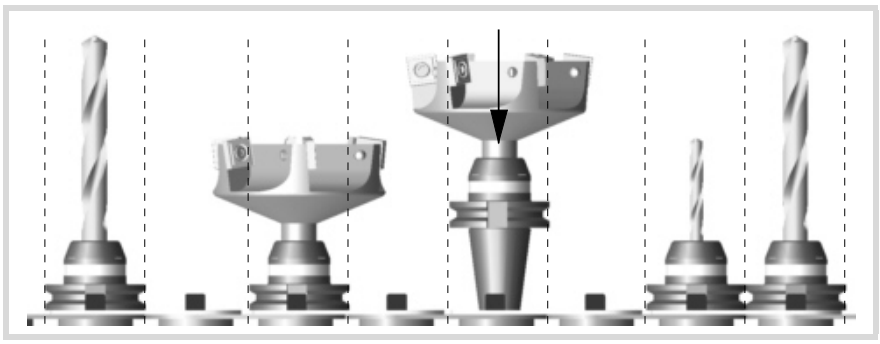
When programming several T's in a row without an M6, this parameter indicates whether all the programmed tools are selected (OPTIMIZE = No) or just the ones involving a tool change (OPTIMIZE = Yes). Optimizing only works when executing a program. In MDI mode, this parameter is ignored and all the blocks are executed.

T2	It looks for the tool if OPTMIZE = No.
T3 M6	It always looks for the tool. The M6 implies a tool change.
T5	It always looks for the tool. M6 comes next.
M6	

RESPECTSIZES
In random magazine, look for pockets of the same size.

Possible values: Yes / No.
Default value: No.
Associated variable: (V.)TM.RESPECTSIZES
Parameter included in the MAGAZINE // MANAGEMENT table.

Depending on the size of the tool, it may occupy more than one magazine position. In random magazines, this parameter indicates that the tools must always be saved in positions of the same size.



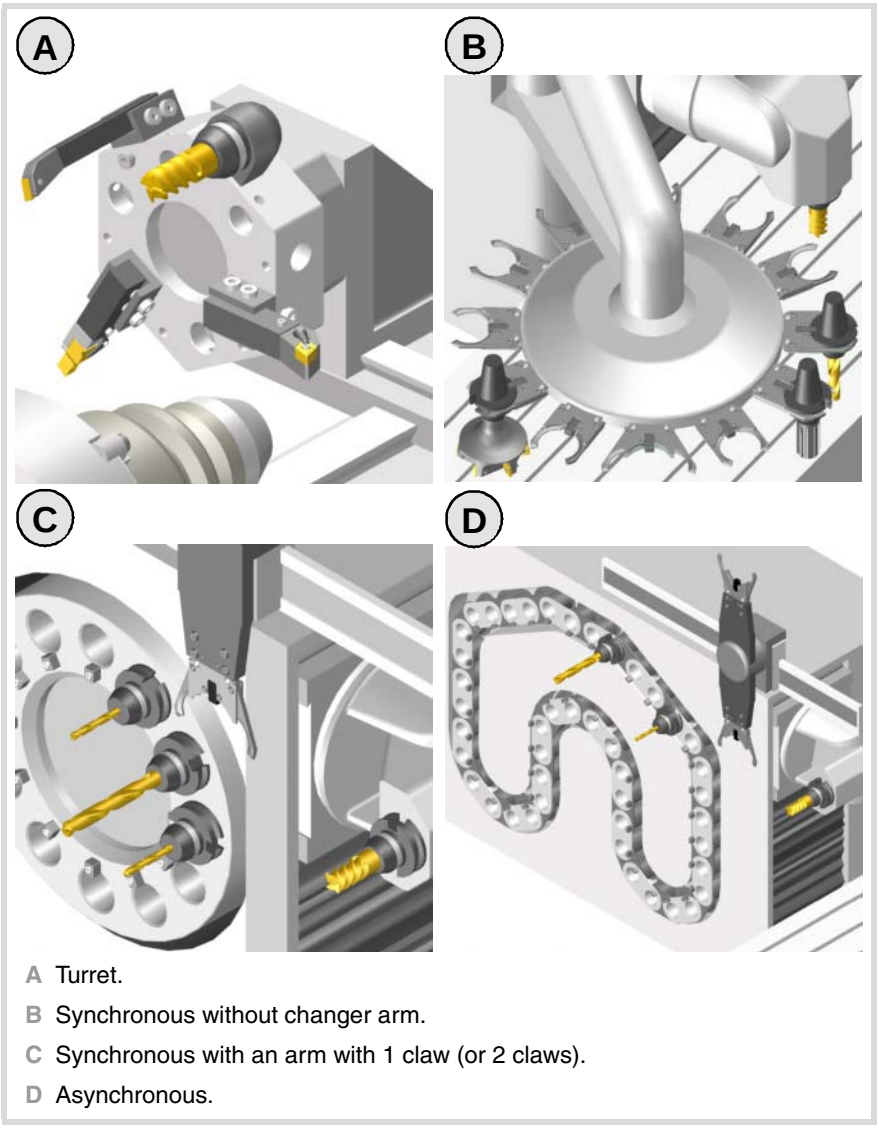
M6ALONE
Action when executing an M06 without selecting a tool.

Possible values: Nothing / Issue a warning / Issue an error.
Default value: Show an error.
Associated variable: (V.)TM.MZM6ALONE[mz]
Parameter included in the MAGAZINE // MANAGEMENT table.

The M06 function implies a tool change. This parameter indicates what happens when executing an M06 without having selected a tool first.

2.10.1 Types of tool magazine.

The CNC can manage the following types of magazine:



A Turret.
B Synchronous without changer arm.
C Synchronous with an arm with 1 claw (or 2 claws).
D Asynchronous.

Turret type.

It is a typical magazine for lathes. The tool cannot be changed while the part is being machined.

Synchronous magazine without changer arm.

In a synchronous tool changer without arm, the magazine must move up to the spindle to change the tool. The tool cannot be changed while the part is being machined.

The tool change is carried out as follows:

1 It ends the movement of the axes.
2 The magazine approaches the spindle to grab the tool.
3 It selects the new tool and puts it in the spindle.
4 The magazine retracts.
5 The CNC resumes program execution.

Asynchronous magazines with changer arm (1 or 2 claws).

Synchronous magazines with tool changer arm (1 or 2 holders) have the magazine close to the spindle. The tool cannot be changed while machining the part because the arm would collide.

The tool change is carried out as follows: Example with 2 holders:

1 The new tool is selected in the magazine.

2 It ends the movement of the axes.

3 The arm takes a tool on each holder (from the magazine and from the spindle) and swaps them.

4 The arm retracts.

5 The CNC resumes program execution.

Asynchronous magazines.

An asynchronous magazine is located away from the spindle. Most of the movements may be carried out while machining the part, thus minimizing machining time.

The tool change is carried out as follows:

1 While machining, a new tool is selected at the magazine, the changer arm picks it up and brings it closer to the spindle.

2 It ends the movement of the axes.

3 The other holder picks up the tool that is in the spindle and makes the change.

4 The program resumes execution and the changer arm returns to the magazine to leave the tool.

2.11 Machine parameters for HMI (Interface).

These parameters are used to define the communication environment (interface) between the operator and the CNC. To validate the changes made to these parameters, the CNC must be reset.

INTERFACE SETTING.

WINDOW
Dimensions of the main window.

This parameter shows the data to configure the main window. The table has the following machine parameters.

WINDOW	
POSX	X coordinate of the top left corner.
POSY	Y coordinate of the top left corner.
WIDTH	Width of the window.
HEIGHT	Height of the window.

POSX
X coordinate of the top left corner.
Default value: 0.
Parameter included in the WINDOW table.

This parameter cannot be modified.

POSY
Y coordinate of the top left corner.
Default value: 0.
Parameter included in the WINDOW table.

This parameter cannot be modified.

WIDTH
Width of the window.
Possible values: From 800 to 1024.
Default value: 800.
Parameter included in the WINDOW table.

The size of the window is given in pixels. We recommend no to change this values at the CNC.

HEIGHT
Height of the window.
Possible values: From 600 to 768.
Default value: 600.
Parameter included in the WINDOW table.

The size of the window is given in pixels. We recommend no to change this values at the CNC.

HMITYPE
Interface type of the CNC.
Possible values: Standard / Classic / Advanced.
Default value: Standard

This parameter sets how to operate with the CNC, through the interface.

Value.	Meaning.
Standard.	Default way to operate at the CNC that may be classic or advanced depending on the CNC model.
Classic.	Hotkey-based method to operate to select the work mode (automatic, jog, editor, etc.).
Advanced.	Default operating way for the 8065. Operating method based on the [Main-Menu] key that shows a main menu to access the various work modes (automatic, jog, editor, etc.) through softkeys. Hotkeys for direct access can also be used. This is the standard way of the 8065.

SFTYPE
(Reserved)

VMENU
Position of the vertical softkey-menu.
Possible values: Left / Right.
Default value: Right.

Depending on hardware, the vertical softkeys F8 through F12 appear on the left or on the right of the monitor. This parameter may be used to put the vertical softkey menu on the proper side.

LANGUAGE
Work language.
Default value: English.

Select a language among those available.

ENGLISH	SPANISH	ITALIAN	GERMAN
FRENCH	BASQUE	PORTUGUESE	CHINESE
RUSSIAN	CZECK	KOREAN	

FFORMAT
Display format for the programmed F.
Default value: 5.2.

This parameter sets the numerical format (integers and decimals) for displaying the feedrate value. If the format is 0.0, the interface assumes the format defined by means of the FGUIM application.

SFORMAT
Display format for the programmed S.
Default value: 5.1.

This parameter sets the numerical format (integers and decimals) for displaying the spindle peed value. If the format is 0.0, the interface assumes the format defined by means of the FGUIM application.

KEYBOARD CONFIGURATION.

USERKEY
Customizing the user key.

This table shows the parameters to configure the user key. The table has the following machine parameters.

USERKEY	
FUNCTION	Function of the user key.
COMPONENT	Access a component of the CNC.
APPLICATION	Execute a PC application.

FUNCTION
Function of the user key.

Possible values: Windows / Component / Application / CNC OFF / Nothing.
Default value: Nothing.
Parameter included in the USERKEY table.

Depending on the function selected, it is possible to carry out one of the following tasks.

Value.	Meaning.
Windows	Minimize the CNC and show the Windows screen.
Component	Access a component (work mode) without CNC hotkey.
Appplication	Execute an external application, for example the FGUIM.
CNC OFF	It turns the CNC application off.
Nothing.	Disable the key.

COMPONENT
Access a component of the CNC.

Possible values: Diagnosis / PLC / Machine parameters / DDSSETUP / TUNING
Parameter included in the USERKEY table.

This parameter shows the list of components (work modes) of the CNC that have no preset key on the operator panel. Besides these components, it will also display the components created with the tool FGUIM.

APPLICATION
Execute a PC application.

Parameter included in the USERKEY table.

When setting parameter USERKEY=Application, this parameter allows to select the application The whole application path must be indicated; for example C:\CNC8070 \FAGOR \ RELEASE \FGUIM.EXE.

CHANGEKEY
Customizing the change key.

This table shows the parameters to configure the change key. The table has the following machine parameters.

CHANGEKEY	
FUNCTION	Function of the change key.
SYSMENUMODE	Behavior of the system menu.
SYSHMENU	Horizontal system menu.
SYSVMENU	Vertical system menu.

FUNCTION
Function of the change key.

Possible values: Next page / Next channel / Menu
Default value: Next page.
Parameter included in the CHANGEKEY table.

Depending on the function selected, it is possible to carry out one of the following tasks.

Value.	Meaning.
Next page.	The key selects the next page of the active work mode.
Next Channel.	The key selects the next channel.
Menu.	The key shows the list of channels and pages on the softkey menus.

SYSMENUMODE
Behavior of the system menu.

Possible values: Volatile / Fixed.
Default value: Volatile.
Parameter included in the CHANGEKEY table.

This parameter determines when the system menu is disabled.

Value.	Meaning.
Volatile.	The softkey menu is disabled when selecting a menu option or when changing the active component.
Fixed.	The softkey menu remains until the change key is pressed again.

SYSHMENU
Horizontal system-menu.

Possible values: Disabled / Pages / Channels / Components.
Default value: Disabled.
Parameter included in the CHANGEKEY table.

See parameter SYSHMENU.

SYSVMENU
Vertical system-menu.

Possible values: Disabled / Pages / Channels / Components.
Default value: Disabled.
Parameter included in the CHANGEKEY table.

This parameter sets the options that will appear on each softkey-menu.

Value.	Meaning.
Disabled.	The menu will be disabled.
Screens.	The menu shows the various pages or screens of the active work mode.
Channels.	The menu shows the available channels.
Components.	The menu shows the components or work modes of the CNC.

ESCAPEKEY
Customizing the escape key.

This table shows the parameters to configure the escape key. The table has the following machine parameters.

ESCAPEKEY	
FUNCTION	Function associated with the escape key.
NPREVIOUS	Maximum number of previous components stored.

FUNCTION
Function associated with the escape key.
Possible values: Previous menu / Previous component / Previous menu-component.
Default value: Previous menu.
Parameter included in the ESCAPEKEY table.

In this parameter is possible to choose between showing the previous softkey-menu, the previous work mode or both. If "both" is selected, every time this key is pressed, it will show the previous softkey menu until reaching the main menu. From then on, the work mode will change.

NPREVIOUS
Maximum number of previous components stored.
Possible values: From 1 to 5.
Default value: 1.
Parameter included in the ESCAPEKEY table.

Number of components stored by the CNC to be displayed when pressing the escape key.

SIMULATED JOG KEYBOARD.

SIMJOGPANEL
Simulated JOG panel.
Possible values: Yes / No.
Default value: No.

This parameter indicates whether the simulated panel is available or not. To select or deselect it, press [CTRL] + [J].

A simulated jog keyboard is a window that lays over the CNC screen and may be used to simulated the jog keys and the keys for accessing the work modes. It may have to be used when working with telediagnosis (remote control of the CNC).

CNC SHUT DOWN.

WINEXIT
Exit Windows when closing the CNC.
Possible values: Yes / No.
Default value: No (Windows is not closed).

This parameter indicates whether Windows is closed or not when exiting the CNC using [ALT] + [F4].

GRAPHICS CONFIGURATION.

GRAPHTYPE
List of graphics in each channel.

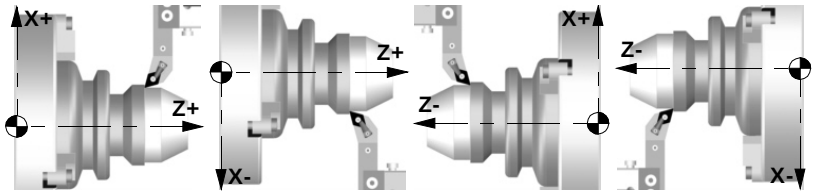
On the lathe model, this parameter shows the table to define the configuration of the graphics in each channel.

GRAPHTYPE	
GRAPHTYPECH n	Type of graphics in each channel.

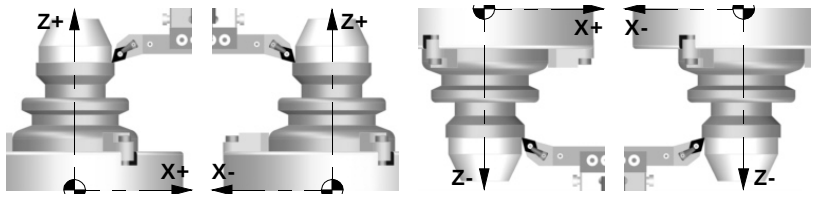
GRAPHTYPECH n
Type of graphics in each channel.
Possible values: Horizaontal or vertical lathe; X+ Z+ / X- Z+ / X+ Z- / X- Z-.
Default value: Horizontal X+ Z+.
Parameter included in the GRAPHTYPE table.

In the lathe model, this parameter sets the configuration of the graphics of the channel.

Horizontal lathe.



Vertical lathe.



It is also possible to display the graphics of a dual-turret lathe (TT lathe). To do that, use the FGUIM application to change the Channel1 and Channel2 properties of the graphics window to display the execution of both channels on a single graphics.

DIAGPSW
(Reserved)

2.12 OEM machine parameters.

READING DRIVE (SERCOS OR MECHATROLINK) VARIABLES.

DRIVEVAR
Drive variables table.

Parameter DRIVEVAR may be used to define the list of DRV variables that grant access to the variables and/or parameters of the digital drives via part-program, MDI/MDA mode, PLC or interface.

- Sercos. DRV variables can access the variables of the drive. These variables may be read-only or read and written.
- Mechatrolink. DRV variables can access the variables and/or parameters of the drive. The variables of the drive can only be read, whereas the parameters may be read or written.

The table has the following machine parameters.

DRIVEVAR	
SIZE	Number of elements of the drive variables table
DATA	Variables of the drives.

Characteristics and limitations of Mechatrolink devices.

The access to Mechatrolink devices using DRV variables has the following characteristics and limitations.

- Access to variables and/or parameters of the devices is only available for the Mlink-II mode with 32 bytes.
- The variables can only be read, whereas the parameters may be read or written.
- Access to the variables of the drive is asynchronous (managed through the service channel) whereas access to the parameters is synchronous (managed through the cyclic channel).
- Either the parameters or the variables may be defined for the same axis, not both.
- If variables have been defined, it is not possible to access the DDSSetup or the setup assistance.
- The maximum number of variables per axis or spindle is 2. There is no limit for parameters per axis.

SIZE
Number of variables to be consulted at the drive.

Possible values: From 0 to 99.
Default value: 0.
Associated variable: (V.)DRV.SIZE
Parameter included in the DRIVEVAR table.

This parameter sets the number of variables to be consulted at the drive.

DATA
Variables of the drives.

This parameter shows the table to define the list of DRV variables.

For Sercos drives, the following fields must be defined for each DRV variable.

Field.	Meaning.
MNEMONIC	Mnemonic assigned to the DRV variable. The access to the variable from the CNC will be as follows: (V.)DRV.{mnemonic}.{axis} (V.)DRV.{mnemonic}.{spindle}
AXIS	Name of the axis or spindle being accessed by the variable; defining it as "*" means all axes and spindles.
ID	ID Sercos identifier of the variable to be consulted at the drive.
TYPE	Access type, synchronous or asynchronous. Variables of synchronous access are managed through the cyclic channel. Variables of asynchronous access are managed through the service channel, even if the default option at the drive is the opposite. Not all the variables may be defined with synchronous access; only the ones that can use the Sercos cyclic channel. Refer to the drive manual. If possible, we recommend to define the variables with synchronous access because the read-write requests of these variables through the bus are faster and reduces traffic through the bus.
MODE	Access mode. These variables may be read-only or read and written.

For Mechatrolink devices, the following fields must be defined for each DRV variable.

Field.	Meaning.
MNEMONIC	Mnemonic assigned to the DRV variable. The access to the variable from the CNC will be as follows: (V.)DRV.{mnemonic}.{axis} (V.)DRV.{mnemonic}.{spindle}
AXIS	Name of the axis or spindle being accessed by the variable; defining it as "*" means all axes and spindles.
ID	Identifier of the parameter (0000-FFFF) or of the variable (0-F) to be consulted at the device.
TYPE	Type of consultation; variable or parameter.
MODE	Access mode. The variables of the drive can only be read, whereas the parameters may be read or written.

GENERIC OEM PARAMETERS.

The values of the table may be modified at any time. The new values are assumed immediately without having to validate the parameters. This means that the values and permissions of the table may be modified while executing a program.

The simulation environment has a copy of this table. On CNC power-up, the values of the parameters of the real table are copied into the simulation table and from there on, they become different in the writing of the variables of both tables.

In the simulation table, only the parameter values may be modified, not the rest of the permissions. The values of the simulation table can only be read or modified through their variable.

MTBPAR
OEM-parameter table.

This table offers 1000 generic parameters that the OEM can use as machine parameters. The table has the following machine parameters.

MTBPAR	
SIZE	Number of OEM parameters.
DATA	List of OEM parameters.

SIZE
Number of OEM parameters.

Possible values: From 0 to 1000.
Default value: 0.
Associated variable: (V.)MTB.SIZE

This parameter sets the number of OEM parameters that will be used.

DATA
List of OEM parameters.

This parameter shows the table of OEM parameters. The following fields must be defined for each parameter.

DATA	
VALUE	Value of the parameter.
MODE	Mode for accessing the parameter from its variable.
INCHES	The parameter is affected by the change of units.
COMMENT	OEM parameter description.

VALUE
Parameter value.

Possible values: Within ±99999.9999.
Default value: 0.

If the parameter is affected by the change of units (INCHES field), the value is entered in the table in the units selected by machine parameter INCHES.

Each parameter has its own variables for reading or modifying (if it has a write permission) their value from the part-program, PLC or interface. The access to these parameters using variables will be as follows.

(V.)MTB.P[i] Value of the OEM parameter [i].

(V.)MTB.PF[i] Value of the OEM parameter [i]. Value per 10000.

Mnemoni.	Value.	V.MTB.P[i]	V.MTB.PF[i]
P0	7	V.MTB.P[0] = 7	V.MTB.PF[0] = 70000
P8	12,5	V.MTB.P[8] = 12,5	V.MTB.PF[8] = 125000

It must be borne in mind that reading and writing these variables interrupts block preparation affecting program execution time. If the value of the parameter is not going to be changed during execution, it is recommended to read the MTB variables at the beginning of the program using arithmetic parameters (local or global) and use the global ones throughout the program.

MODE
Mode for accessing the parameter from its variable.

Possible values: Read / Write.
Default value: Read.

The access to the parameters from the variables may be read-only or read-write. Both reading and writing interrupt block preparation.

If a read-only access has been defined, the parameter cannot be modified from its variable. If a write access has been defined, the parameter can be modified through its variable.

The value may always be written directly into the table, regardless of the value assigned to this field.

INCHES
The parameter is affected by the change of units.

Possible values: Yes / No.
Default value: Yes.

This field indicates whether the value of the parameter is affected by the change of units, mm or inches. For example, when the parameter represents a coordinate.

COMMENT
OEM parameter description.

This field offers the possibility to associate a short description with the parameter. This field is for information only; it is not used by the CNC.

The comments are saved in the file MTBComments.txt and it is possible to have one file per language. These files are saved in the folder C:\CNC8070 \MTB \data \Lang".

CAM EDITOR.

The cam editor is an graphic assistance element for designing cams. The user must make sure that the design is coherent with the required specifications.

CAMTABLE
Table of electronic cams.

This table shows the parameters to define the electronic cams. The table has the following machine parameters.

CAMTABLE	
SIZE	Number of electronic cams.
DATA	List of cams.
CAM n	Access the cam editor.

SIZE
Number of electronic cams.

Possible values: From 0 to 16.
Default value: 0.
Parameter included in the CAMTABLE table.

This parameter sets the number of electronic cams of the system.

DATA
List of electronic cams.

This parameter shows the list of available cams.

CAM n
Electronic-cam editor.

Cam editor with friendly assistance to analyze the behavior of the cam projected through graphically assisted data entry for speed, acceleration and jerk.



This function has a specific manual.
Refer to the documentation included in the CD-ROM that comes with the product for further information on the requirements and operation of the electronic cam.

2.

MACHINE PARAMETERS.
OEM machine parameters.

FAGOR

CNC 8060
CNC 8065

(REF: 1402)

INTRODUCTION TO THE PLC.

3

The PLC program may be either edited at the front panel or copied from a peripheral or PC. The PLC program has a modular structure and may combine files in "C" language, contact (ladder) language or mnemonic language.

For the program to be executed, its object file (executable) must be generated. On power-up, the CNC executes the executable PLC program stored in memory; if there isn't one, the CNC will show the relevant error code.

Data exchange takes place in automatic mode between the CNC and the PLC. The following is possible from the PLC:

- Control physical inputs and outputs (remote modules).
- Consult and/or modify CNC-PLC exchange variables.
- Consult and/or modify the internal CNC variables.
- Display messages or errors at the CNC.

The following is possible from the CNC:

- Transfer auxiliary functions M, H and S.
- Access PLC resources from any part-program.

Abbreviations used in this chapter.

(=0)	Low logic level.
(=1)	High logic level.
(g.m.p.)	General machine parameter.

3.1 PLC program.

The PLC program may combine several files in mnemonic language (extension: "plc"), several files in "C" language (extension: "c") and a file in contact (ladder) language (extension "ld"). All the files making up the PLC program must be in the folder \MTB \PLC \PROJECT.

It is recommended to use the mnemonic language files or the contact (ladder) language files as the main PLC program and the "C" language files for auxiliary tasks (for example, temperature compensation).

Subroutines in the PLC program.

The subroutines of the program in mnemonic or contact (ladder) language must be defined outside of the modules, for example at the end of the program after the END instruction.

The subroutines of the "C" language file must be defined as external at the beginning of the mnemonic language program (extension: "plc") or contact (ladder) language program (extension "ld"). The name of the subroutine must be written in capital letters in both files.

"C" language programming offers a math library (trigonometric, logarithmic, etc.) that allows performing operations with tables, arrays, float type variables, etc.

PLC program with mnemonic language file.

Mnemonic.plc

```
PRG
()= MOV 1234 R201 = MOV 2345 R202
()= CAL ADD
…
END

SUB SUMA
()= ADS R201 R202 R203
END
```

PLC program with a mnemonic language file and a "C" language file.

Mnemonic.plc

```
EXTERN SUMA
PRG
()= MOV 1234 R201 = MOV 2345 R202
()= CAL ADD
…
END

Languagec.c
#include "plclib.h"
void ADD(void)
(
R203=R201+R202
)
```

3.2 Modular structure of the PLC program.

The PLC program may comprise the following modules. Every module must begin with the directive instruction that defines it (CY1, PRG, PE) and end with the directing instruction END.

- First Cycle module (CY1).
- Main module (PRG).
- Periodic module (PE).

First Cycle module.

It is an optional module. The module begins with the directing instruction CY1 and ends with the directive instruction END.

The PLC executes the first cycle module onlyonce when starting up the PLC program. This modules is used to initialized the various resources and variables before executing the main program.

Main module.

The main module begins with the directing instruction PRG and ends with the directive instruction END.

The PLC executes the main module cyclically with the frequency set by parameter PRGFREQ. This module is in charge of analyzing and modifying the CNC inputs, outputs and variables. The PRG execution takes about 100 µs.

Parameter PRGFREQ indicates how often (every how many CNC cycles) the PLC executes a full cycle of the main module. Hence, with a sampling period of 4 ms (LOOPTIME=4) and a frequency of 2 cycles (PRGFREQ=2), the PLC executes the main module every 4 x 2 = 8ms.

Periodic module.

It is an optional module. The periodic module begins with the directing instruction PE and ends with the directive instruction END.

The PLC executes the periodic module cyclically with the frequency set by the directive instruction PE, between 1 and 2147483647 ms and never less than the loop time (parameter LOOPTIME). This module could be used to execute tasks that do not need to be executed at every PLC cycle.

For example, a task to be performed every 30 seconds could be defined in a periodic module using the instruction (PE 30000).

3.3 PLC program execution.

Main module (PRG)

The main module is processed as follows:

1. The PLC assigns the current value of the physical inputs (remote modules) to the I resources.
2. The PLC assumes the current values of the internal CNC variables (CNCREADY, START, FHOUT, ...).
3. The PLC executes the main program (PRG).
4. The PLC updates the internal CNC variables (EMERGEN, STOP, FEEDHOL, ...) with the current values of the associated PLC resources.
5. The PLC assigns the current value of the PLC's "O" resources to the physical outputs (remote modules).
6. The PLC concludes the cycle and is ready for the next scan.

Periodic module (PE)

The periodic module is processed as follows:

1. The PLC takes into account the current values of the physical inputs (remote modules) at the beginning of the module.
2. The PLC executes the periodic module.
3. The PLC assigns the current value of the PLC's "O" resources to the physical outputs (remote modules).
4. The PLC concludes the execution of the periodic Module.

3.4 PLC resources.

The PLC has the following resources.

- Inputs (I1-I1024) and outputs (O1-O1024).
- Local inputs (LI1-LI16) and local outputs (LO1-LO8).
- Marks (M1-M8192).
- Messages (MSG1-MSG1024).
- Errors (ERR1-ERR1024)
- Clocks (CLK).
- Registers (R1-R1024).
- Timers (T1-T512).
- Counters (C1-C256).
- Registers and marks for CNC-PLC communication.

The MSG, ERR, CLK and T resources are initialized (=0) when starting up the PLC. M, C and R resources maintain their value between CNC start-ups.

Inputs (I1-I1024) and outputs (O1-O1024).

The inputs are elements that provide information to the PLC on the signals they receive from the outside world. They are represented by the letter "I" followed by an input number between "I1" and "I1024".

The outputs are elements that let the PLC activate or deactivate the various devices of the electrical cabinet. They are represented by the letter "O" followed by an output number between "O1" and "O1024".

Numbering of the physical inputs and outputs.

There are two different ways to number the inputs and outputs. Depending on the order of the remote modules or via machine parameters. See *"3.4.1 Numbering of the physical inputs and outputs."* on page 226.

Local inputs (LI1-LI16) and local outputs (LO1-LO8).

The central unit has a set of 8 local digital inputs and 8 local digital signals that may be configured both as input and output. They are called local inputs and outputs because they are located in the central unit, not in the remote modules. The local inputs are represented with the letter LI followed by the input number, from LI1 to LI16. The local outputs are represented with the letter LO followed by the output number, from LO1 to LO8.

Local digital I/O are ignored when setting the number of remote I/O or when saving the configuration of the CAN bus. Local I/O will always be referred to in the same way regardless of the system configuration.

Local I/O will be refreshed with the same frequency as the remote I/O and also from the PE module.

Marks (M1-M8192).

They are elements capable of memorizing in a bit (like an internal relay) the value set by the user. If the mark is (=0), it will be referred to as being set low. If the mark is (=1), it will be referred to as being set high.

They are represented by the letter "M" followed by a mark number between M1 and M8192.

Messages (MSG1-MSG1024).

When activated (=1), they display a message on the CNC screen. The texts associated with the messages must be previously defined in the message and error table of the PLC.

They are represented by the letters "MSG" followed by a message number between MSG1 and MSG1024.

All of them are initialized (=0) when starting up the PLC.

Errors (ERR1-ERR1024)

When activated (=1), they cause an error, they show an error message and interrupt the execution of the CNC. The texts associated with the errors must be previously defined in the message and error table of the PLC. PLC errors may be configured so they activate the emergency signal of the PLC (_EMERGEN).

They are represented by the word "ERR" followed by an error number between ERR1 and ERR1024.

All of them are initialized (=0) when starting up the PLC.

The errors do not activate the alarm of the CNC, (_ALARM) signal.

Clocks (CLK).

They are internal clocks with different time periods to be used in the PLC program.

They are represented by the word "CLK" followed by a clock number .

All of them are initialized (=0) when starting up the PLC.

These are the clock marks available. Their half-period (or after how long their state 0/1 changes) is shown next to them.

Clock.	Period.	Clock.	Period.	Clock.	Period.
CLK1	1 ms	CLK100	100 ms	CLK1000	1 s
CLK2	2 ms	CLK200	200 ms	CLK2000	2 s
CLK4	4 ms	CLK400	400 ms	CLK4000	4 s
CLK8	8 ms	CLK800	800 ms	CLK8000	8 s
CLK16	16 ms	CLK1600	1.6 s	CLK16000	16 s
CLK32	32 ms	CLK3200	3.2 s	CLK32000	32 s
CLK64	64 ms	CLK6400	6.4 s	CLK64000	64 s
CLK128	128 ms	CLK12800	12.8 s	CLK128000	128 s

Registers (R1-R1024).

They are elements that can store a numeric variable in 32 bits. The value stored in each register is considered as a signed integer between ± 2.147.483.647. It can be processed as decimal or hexadecimal number (preceded by the "$" sign). For example:

156 (Decimal)

$9C (Hexadecimal)

They are represented by the letter "R" followed by a register number between R1 and R1024. It is also possible to refer to a register bit with the letter "B" and a bit number (0/31). The PLC takes bit 0 as the least significant bit and as bit 31 as the most significant bit.

B7R155 Refers to bit 7 of register 155.

Timers (T1-T512).

They are elements capable of maintaining their output at the same logic level (state) for a preset time period (time constant) after which their output changes states.

They are represented by the letter "T" followed by a time number between T1 and T512.

All timers are initialized (=0) when starting up the PLC.

See *"3.5 Operation of a timer."* on page 228.

Counters (C1-C256).

They are elements capable of counting up or down a preset amount of events.

They are represented by the letter "C" followed by a counter number between C1 and C256.

See *"3.6 Operation of a counter."* on page 238.

Registers and marks for CNC-PLC communication.

The PLC has access to a some internal CNC data.

The PLC can consult and/or modify certain CNC signals (marks and registers).

- Consultation signals: CNCREADY, START, FHOUT, ...
- Modifiable signals: _EMERGEN, _STOP, _FEEDHOL, ...

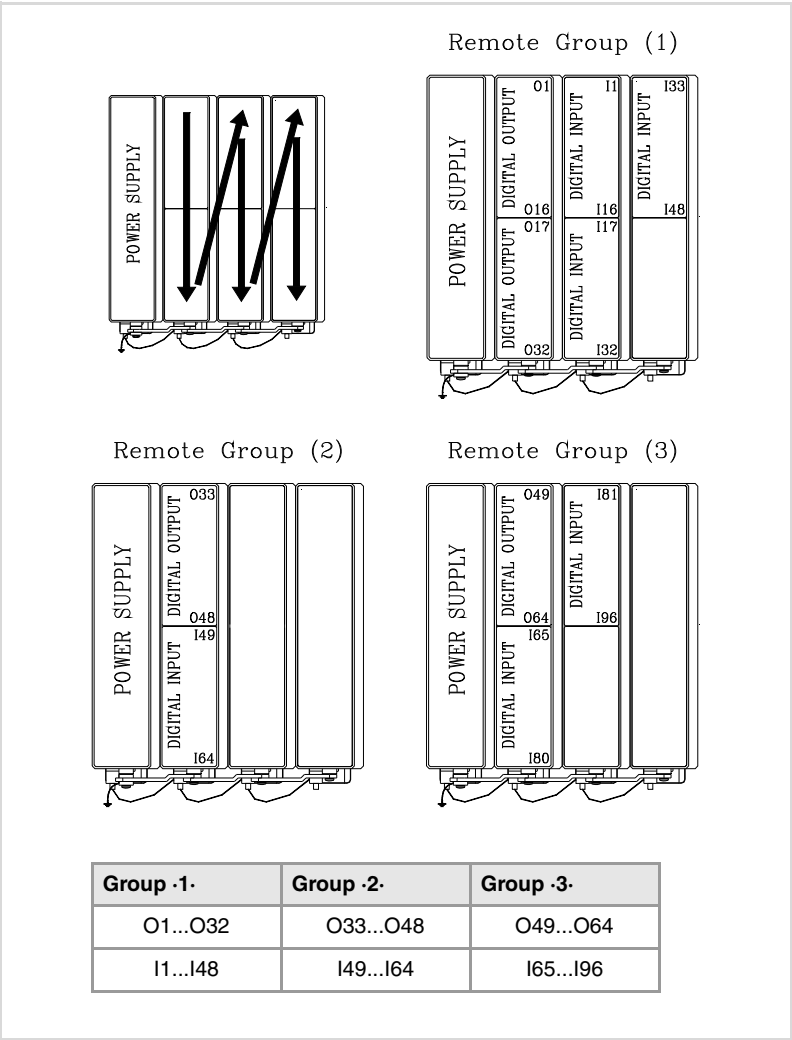
See chapter *"6 Logic CNC inputs and outputs."*.

3.4.1 Numbering of the physical inputs and outputs.

The numbering of the I/O modules may be set using the machine parameters. If these parameters are not defined, the CNC numbers the modules automatically according to the order of the remote modules.

Numbering according to the order of the remote groups.

They are numbered following the order of the remote groups (rotary switch of the Power Supply element). Within each group, they are ordered from top to bottom and from left to right.

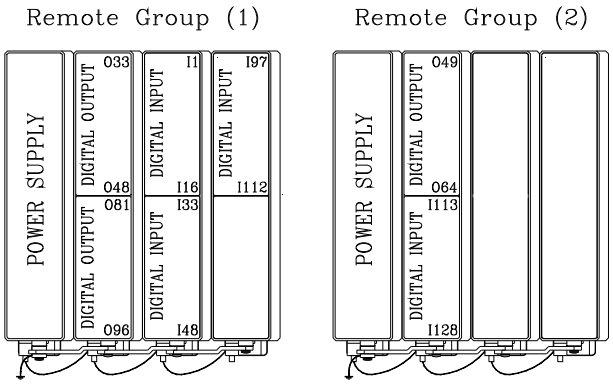


Group ·1·	Group ·2·	Group ·3·
O1...O32	O33...O48	O49...O64
I1...I48	I49...I64	I65...I96

Numbering by machine parameters.

When the numbering is set by machine parameters, each module is assigned a base index and the inputs or outputs of that module are numbered after it. The values of the base index must be multiple of 16, plus 1 (i.e. 1, 17, 33, etc.). The base indexes may follow any order, they do not have to be sequential.

When inserting a new module, the first modules will be assigned the numbering of the table and the last one will be assigned the next valid base index after the highest one assigned until then.



	Digital inputs		Digital outputs	
	Index	**Inputs**	**Index**	**Outputs**
Module 1.	1	I1...I16	33	O33...O48
Module 2.	33	I33...I48	81	O81...O96
Module 3.	97	I97...I112	49	O49...O64
Module 4.	113	I113...I128	- - -	- - -

3.5 Operation of a timer.

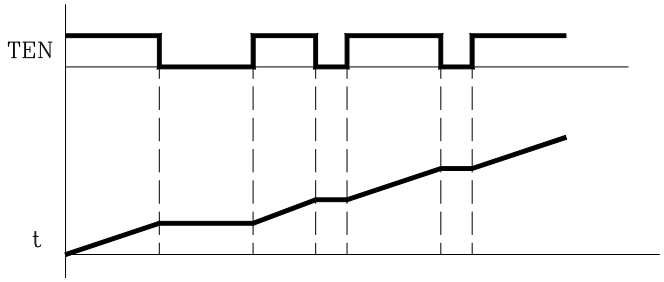
All the timers have a status output "T" and the inputs: TEN, TRS, TG1, TG2, TG3 and TG4. It is also possible to check, at any time, the elapsed time t since the timer was triggered.

When starting up the PLC, all the timers are initialized by setting their status "T" and their time count to "0".

(TEN) Enable input.

It can be used to interrupt and resume the timing. It is referred to by the words "TEN" followed by the timer number. For example TEN 1, TEN 25, TEN 102, etc.

Once the timer is triggered, if input TEN is set low (=0), the PLC stops timing; input TEN must be set back high (=1) to resume timing.



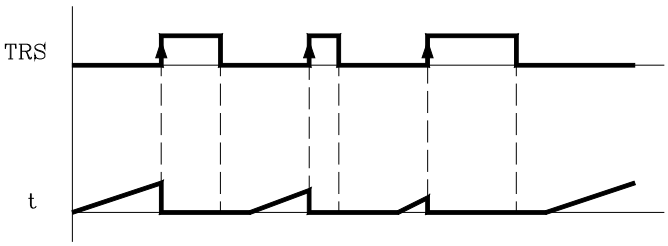
By default, every time a timer is triggered, the PLC sets this input high (=1).

```
I2 = TEN 10
```
Input I2 controls the Enable input of timer T10.

(TRS) Reset input.

It is used to initialize the timer by setting its status "T" and its timing to "0". It is referred to by the word "TRS" followed by the timer number, for example TRS 1, TRS 25, TRS 102, etc.

Once the timer has been activated, at an up-flank (0 to 1 transition) of the TRS input, the PLC resets the timer. The timer is deactivated and its trigger input must be activated to turn the timer back on.



By default and every time a timer is triggered, the PLC sets this input high (=0).

```
I3 = TRS 10
```
Input I3 controls the Reset input of timer T10.

(TG1, TG2, TG3, TG4) Trigger inputs.

They are used to activate the different work modes of the timer.

TG1	triggers the mono-stable mode.
TG2	triggers the delayed activation mode.
TG3	triggers the delayed deactivation mode.
TG4	triggers the signal limiting signal.

They are referred to by the words TG1, TG2, TG3, TG4 followed by a timer number and the initial timing value (time constant). For example TG1 1 100, TG2 25 224, TG3 102 0, etc.

Set the time constant.

The time constant is defined with a numeric value or with an internal value of a register R. Its value must be within the 0 to 4294967295 ms range, which is the equivalent of 1193 hours (almost 50 days).

```
TG1 20 100
```
Triggers timer T20 in Mono-stable mode (TG1) with a time constant of 100 ms.

```
TG2 22 R200
```
Triggers timer T22 in delayed activation mode (TG2) with the time constant stored in register R200 in ms.

Activating the timer.

The timer is activated according to the selected input number at an up-flank (0 to 1 transition) or at a down-flank (1 to 0 transition).

Later on, this same section shows how to operate in each of these modes.

(T) Status output.

It indicates the logic state of the timer. It is referred to by the letter "T" followed by the timer number. For example: T1, T25, T102, etc.

As the timer's logic state depends on the selected work mode (TG1, TG2, TG3 and TG4) it will be explained later on.

(T) Elapsed time.

It indicates the time elapsed at the timer since it was triggered. It is referred to by the letter "T" followed by the timer number. It is represented by T123 which is the same as the status output, but is used in different types of instructions.

In binary instructions, it refers to the logic state of the timer.

```
T123 = M100
```
Assigns the state (1/0) of T123 to M100.

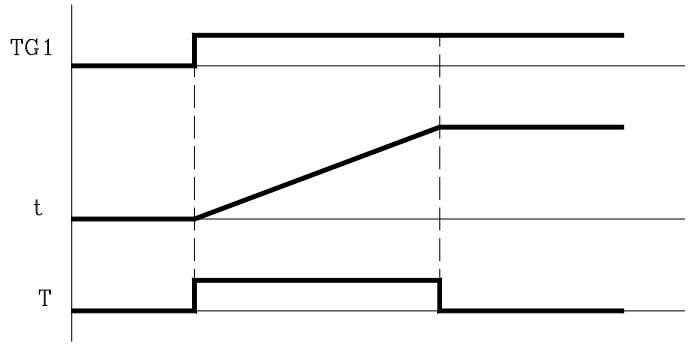
In arithmetic and comparison instructions, it refers to the elapsed time.

```
I2 = MOV T123 R200
```
Transfers the time elapsed at T123 to register R200.

```
CPS T123 GT 1000 = M100
```
Compares the time elapsed at T123 is greater than 1000. If so, it activates mark M100.

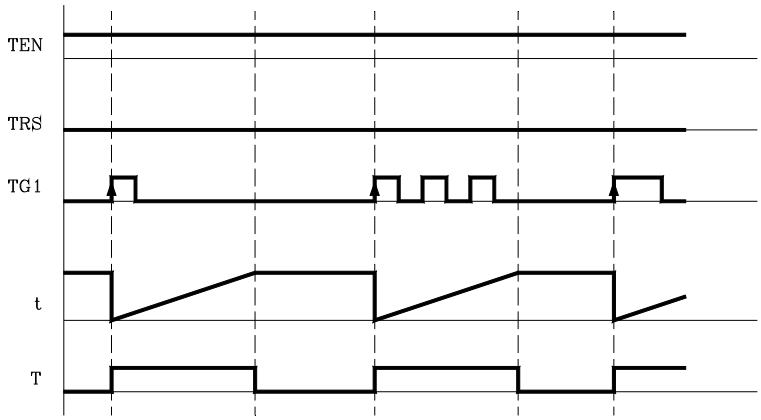
3.5.1 Monostable mode. TG1 input.

In this operating mode, the status of the timer is maintained high (T=1) from when the TG1 input is activated until the indicated time period (constant) has elapsed.



With TEN=1 and TRS=0, the timer is activated with an up-flank at trigger input TG1. At that moment, the timer status output (T) changes states (T=1) and the timing starts from ·0·.

Once the time period indicated by the time constant has elapsed, the timing is over. The status output (T) changes (T=0) and the elapsed time t is maintained.

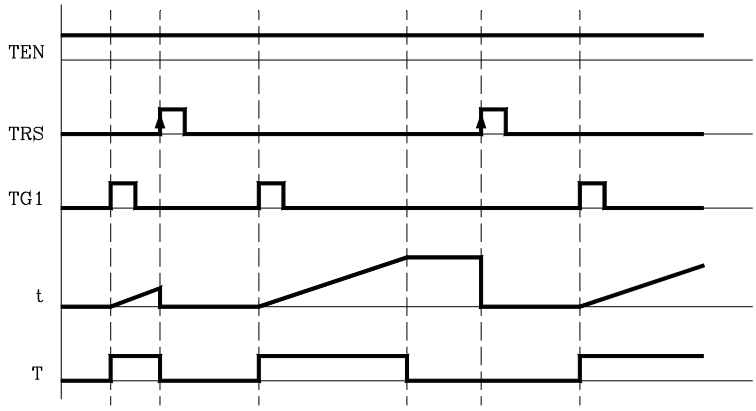


Any changes at the TG1 input (up or down-flank) while timing, has no effect.

Once the timing is over, an up-flank at trigger input TG1 is required to reactivate the timer.

Operation of the TRS input in this mode.

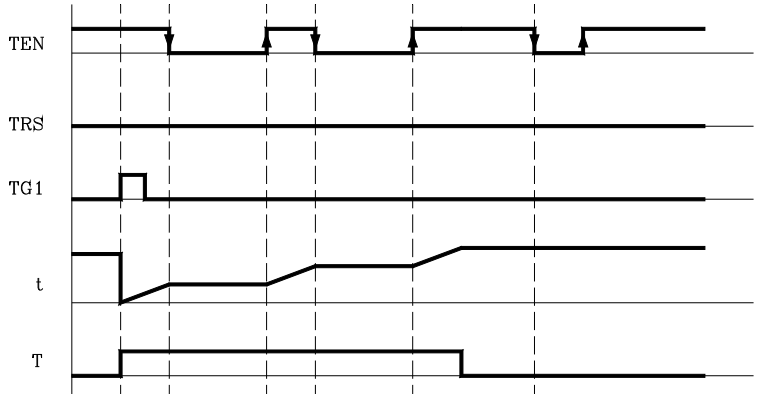
If an up-flank occurs at the TRS input while timing or after it, the PLC resets the timer setting its status output low (T=0) and resetting its timing (t=0).



Since the timer is reset, its trigger input must be activated again to turn it back on.

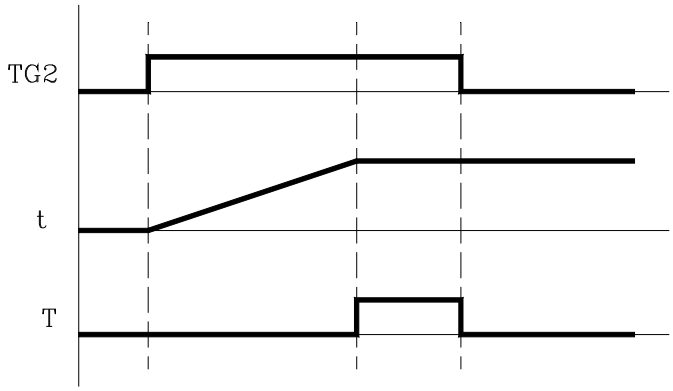
Operation of the TEN input in this mode.

If once the timer has been activated, TEN = 0, the PLC interrupts the timing and TEN must be set to "1" to resume timing.



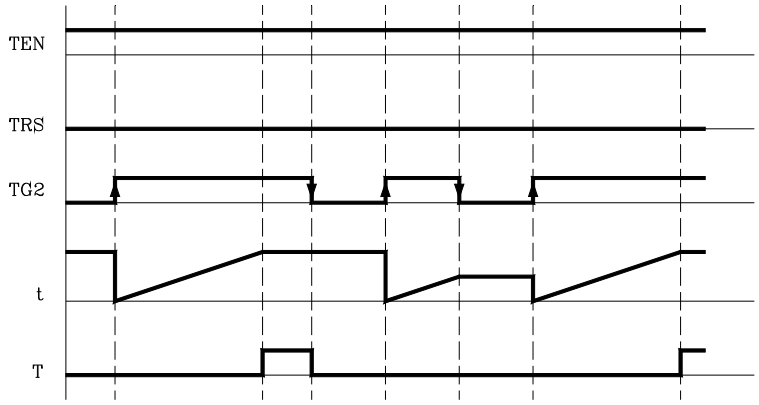
3.5.2 Delayed activation mode. TG2 input.

This mode applies a delay between the activation of the trigger input TG2 and that of the timer status output "T". The time delay is set by the time constant.



With TEN=1 and TRS=0, the timer is activated with an up-flank at the TG2 input. At that instant, the timing "t" begins from "0".

Once the time indicated with the time constant has elapsed, the timing is over, it activates the timer status output (T=1) which remains high that until a down-flank at trigger input TG2 occurs.

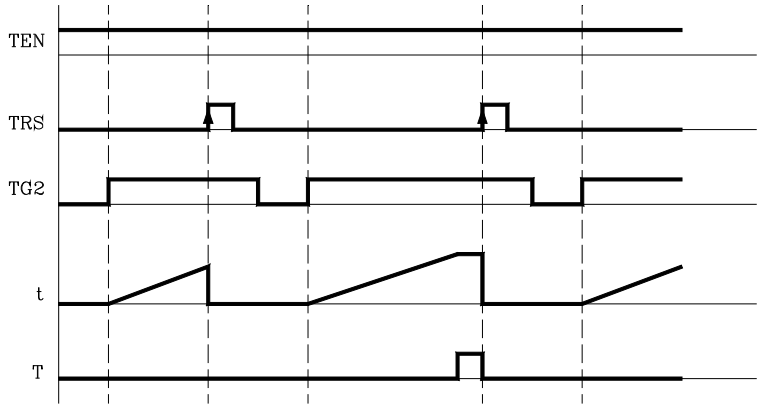


The elapsed time (t) is maintained until a new up-flank occurs at trigger input TG2.

If the down-flank at input TG2 occurs before the indicated time has elapsed, the PLC stops timing and it keeps the "t" value it has at the time.

Operation of the TRS input in this mode.

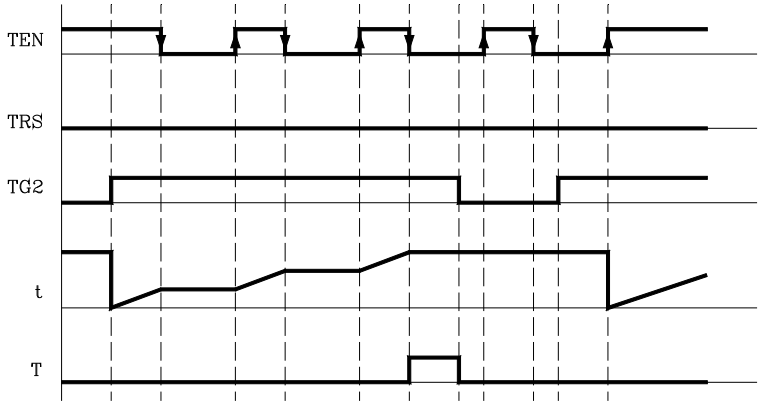
If an up-flank occurs at the TRS input while timing or after it, the PLC resets the timer setting its status output low (T=0) and resetting its timing (t=0).



Since the timer is reset, its trigger input must be activated again to turn it back on.

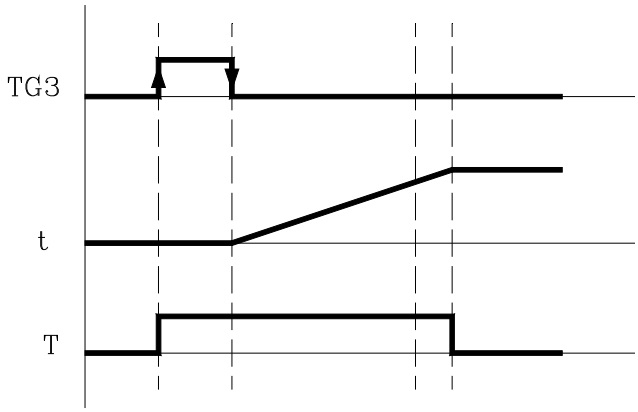
Operation of the TEN input in this mode.

If once the timer has been activated, TEN = 0, the PLC interrupts the timing and TEN must be set to "1" to resume timing.



3.5.3 Delayed deactivation mode. TG3 input.

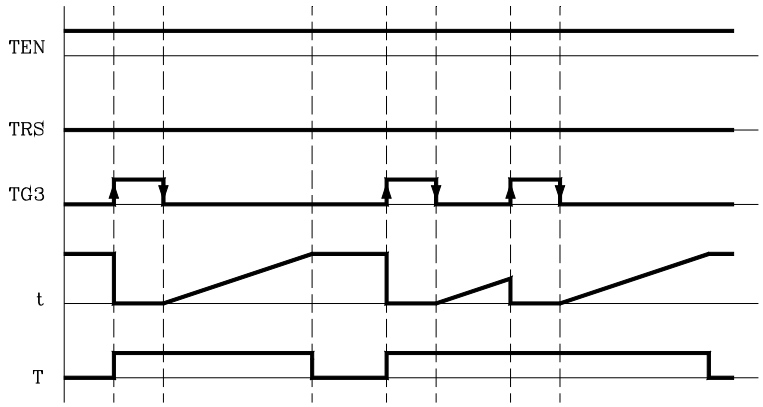
This operating mode is used to apply a delay between the deactivation of trigger input TG3 and that of the "T" output of the timer. The time delay is set by the time constant.



With TEN=1 and TRS=0, the timer is activated with an up-flank at the TG3 input. At that instant, the status output of the timer goes high (T=1).

The timer waits for a down-flank at input TG3 to start the "t" timing from "0".

Once the time indicated by the time constant has elapsed, the timing stops and the status output of the timer goes low (T=0).

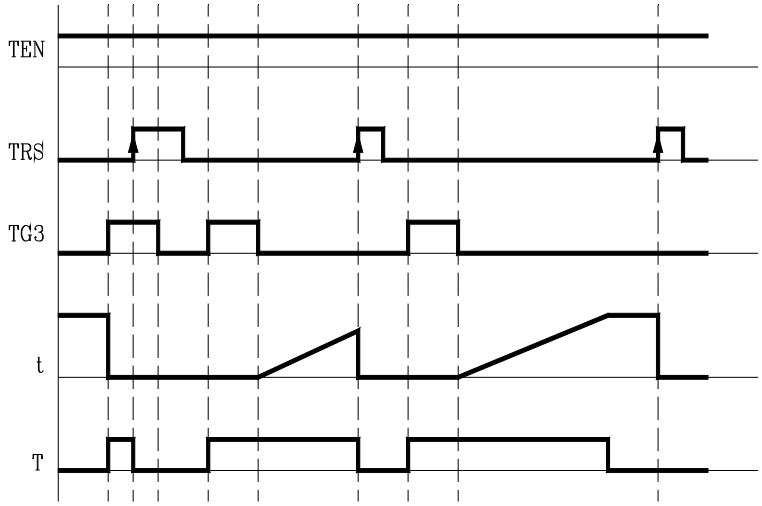


The elapsed time "t" is maintained until a new up-flank occurs at input TG3.

If the up-flank at input TG3 takes place before the indicated time has elapsed, the PLC takes it as a new trigger and sets its status output high (T=1) and starts timing again from "0".

Operation of the TRS input in this mode.

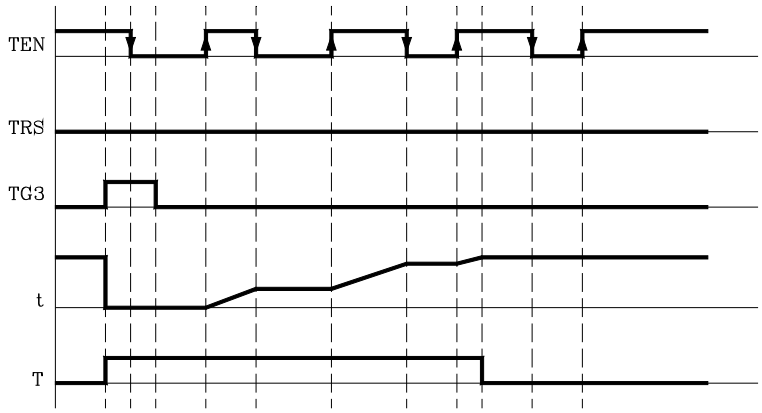
If an up-flank occurs at the TRS input while timing or after it, the PLC resets the timer setting its status output low (T=0) and resetting its timing (t=0).



Since the timer is reset, its trigger input must be activated again to turn it back on.

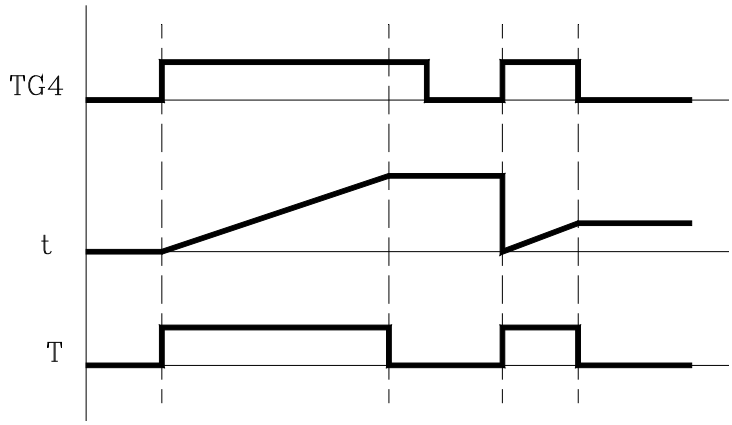
Operation of the TEN input in this mode.

If once the timer has been activated, TEN = 0, the PLC interrupts the timing and TEN must be set to "1" to resume timing.



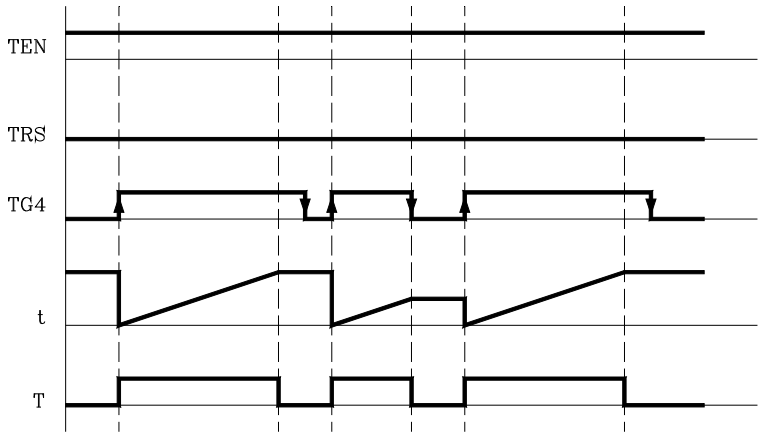
3.5.4 Signal limiting mode. TG4 Input.

In this operating mode, the timer status is kept high (T=1) from the moment the TG4 input is activated until the time indicated by the time constant has elapsed or a down-flank occurs at input TG4.



With TEN=1 and TRS=0, the timer is activated with an up-flank at input TG4. At that moment, the timer status output (T) changes states (T=1) and the timing starts from "0".

Once the time indicated by the time constant has elapsed, the timing stops and the status output of the timer goes low (T=0).



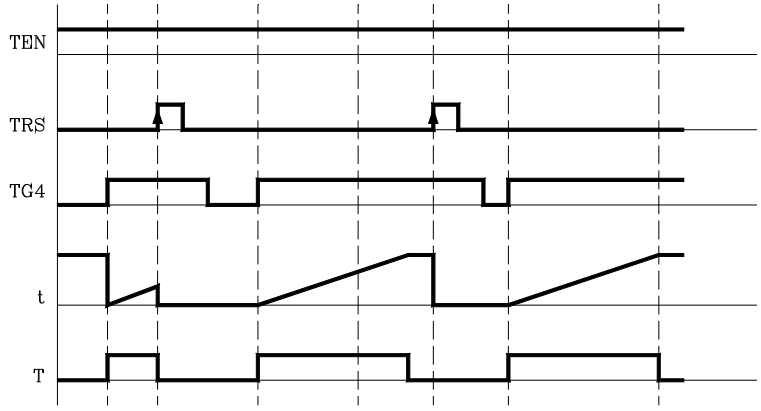
The elapsed time (t) is maintained until a new up-flank takes place at input TG4.

If a down-flank occurs at trigger input TG4 before the time indicated by the time constant has elapsed, the PLC stops the timing, brings the status output low (T=0) and it keeps the current timing value (t).

To trigger the timer again, a new up-flank is required at input TG4.

Operation of the TRS input in this mode.

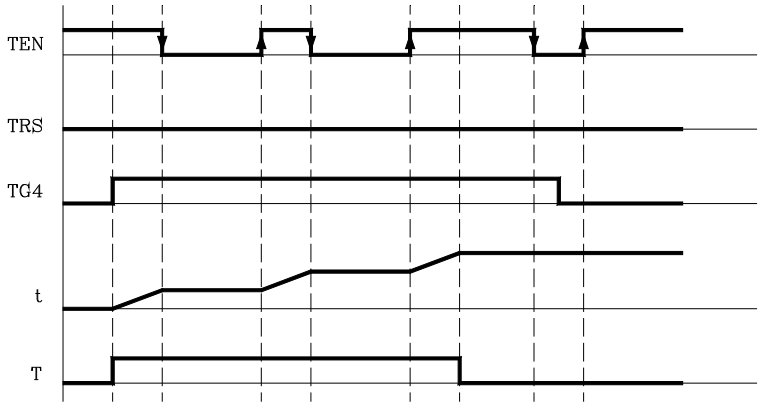
If an up-flank occurs at the TRS input while timing or after it, the PLC resets the timer setting its status output low (T=0) and resetting its timing (t=0).



Since the timer is reset, its trigger input must be activated again to turn it back on.

Operation of the TEN input in this mode.

If once the timer has been activated, TEN = 0, the PLC interrupts the timing and TEN must be set to "1" to resume timing.



3.6 Operation of a counter.

All the counters have a status output "C" and the inputs: CUP, CDW, CEN and CPR. Its internal count can also be checked at any time.

The counter's count is stored in a 32-bit variable. Consequently, its value will be in the ±2147483647 range.

(CUP) Count-up input.

Every time an up-flank occurs at this input, the internal count of the counter increases one unit.

It is referred to by the letters CUP followed by the counter number, for example: CUP 1, CUP 25, CUP 102, etc.

```
I2 = CUP 10
```
Every time an up-flank occurs at I2, the count of the C10 counter increases one unit.

(CDW) Countdown input.

Every time an up-flank occurs at this input, the internal count of the counter decreases one unit.

It is referred to by the letters CDW followed by the counter number, for example CDW 1, CDW 25, CDW 102, etc.

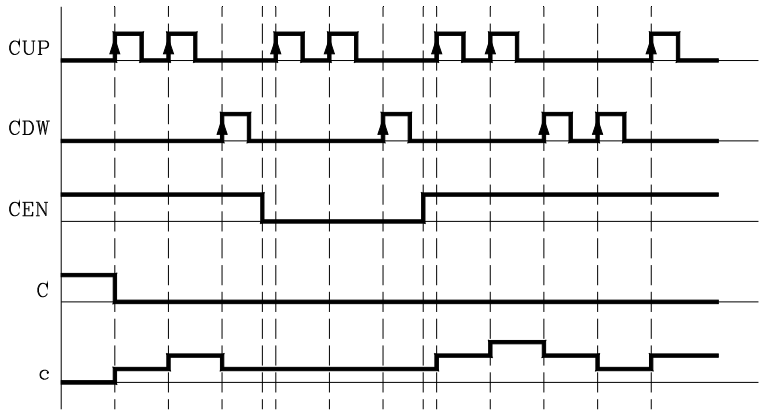
```
I3 = CDW 20
```
Every time an up-flank occurs at I3, the count of the C10 counter decreases one unit.

(CEN) Enable input.

It enables the internal count of the counter.

It is referred to by the letters CEN followed by the counter number, for example: CEN 1, CEN 25, CEN 102, etc.

To change the internal count (CUP and CDW), the CEN input must be high (=1). Setting CEN = 0 stops the counter's count and ignores the CUP and CDW inputs.



(CPR) Preset input.

To preset the counter with the desired value.

It is referred to with the letters CPR followed by the counter number and the preset count value. The counter is preset with the indicated value with an up-flank at the CPR input.

The count value may be indicated by a numeric value or by the internal value of a register "R". Its value must be between 0 and ± 2,147,483.647.

```
CPR 20 100
    Presets counter C20 with a value of 100.
CPR 22 R200
    Presets counter C22 with the value of register R200
```

(C) Status output.

It indicates the logic state of the counter. It is referred to by the letter "C" followed by the counter number, for example: C1, C25, C102, etc.

The logic status of the counter will be C=1 when its count value is "0" and C=0 if otherwise.

(C) Count value.

It indicates the value of the internal count of the counter.

It is referred to by the letter "C" followed by the counter number, for example: C1, C25, C102, etc. Its representation C123 is the same as that of the status output, but different type instructions are used for them.

In binary instruction, it refers to the logic status of the counter.

```
C123 = M100
    Assigns the status (0/1) of C123 to M100
```

In arithmetic and comparison instructions, it refers to the internal count of the counter.

```
I2 = MOV C123 R200
    Transfers the count of C123 to register R200.
CPS C123 GT 1000 = M100
    Compares whether the count of C123 is greater than 1000. If so, it activates mark M100.
```

The PLC has a 32-bit variable to store the count of each counter.

3.

INTRODUCTION TO THE PLC.

Operation of a counter.

FAGOR

CNC 8060
CNC 8065

(REF: 1402)

PLC PROGRAMMING.

4

The PLC program may comprise the following modules. Every module must begin with the directive instruction that defines it (CY1, PRG, PE) and end with the directing instruction END.

- First Cycle module (CY1).
- Main module (PRG).
- Periodic module (PE).

All of them consist of a series of instructions that depending on their function may be either directing or executable instructions.

Directing instructions.

The directing instructions provide the PLC with information on the type of module (PRG, CY1, ...) and on how it must be executed (REA, IMA, ...).

Executable instructions.

With the executable instructions, it is possible to check and/or change the status of the PLC resources. They consist of:

- Logic or Boolean instructions (I28 AND I30).
- Action instructions (=O25).

Logic expressions consist of:

- Consulting instructions (I28, O25).
- Operators (AND).

Logic expressions can be written in two or more different lines without having to include any dividing character. However, to make the program easier to understand, the "\" character may be inserted at the end of each line, but it is not necessary.

The following programming examples are equivalent.

Option 1.
```
I32 AND I36 AND M111 = O25
```

Option 2.
```
I32 AND I36 AND M111
= O25
```

Option 3.
```
I32
AND I36
AND M111
= O25
```

Option 4.
```
I32 \
AND I36 \
AND M111
= O25
```

Comments.

All comments must begin with ";". Lines beginning with a ";" are considered comments and are not executed.

Empty lines are also possible.

Programming example:

```
PRG                    ; Directing instruction
; Example               ; Comment
I100 = M102            ; Executable instruction
I28 AND I30            ; Logic expression
= O25                  ; Action instruction
I32 \                  ; Consulting instruction (1st part of expression)
AND I36                ; Consulting instruction (2nd part of expression)
= M300                 ; Action instruction
END                    ; Directing instruction
```

4.1 Directing instructions.

They provide the PLC with information on the type of program module and on how it must be executed. The available directing instructions are:

PRG, PE t, CY1	Type of module.
END	End of module.
REA, IMA	Real or image values.
L	Label.
SUB	Subroutine definition.
DEF, PDEF	Symbol definition.
NOMONIT	No monitoring.
EXTERN	External subroutine definition.

PRG, PE t, CY1
Type of module.

The PLC program is structured by modules. Every module must begin with its defining instruction (PRG, PE, CY1) and end with the END instruction. See *"3.2 Modular structure of the PLC program."* on page 221.

CY1	First cycle module.
PRG	Main module.
PE t	Periodic module. It is executed every "t" milliseconds.

END
End of module or subroutine.

It must be defined for each module or subroutine.

```
CY1                    ; Beginning of the CY1 module
...
END                    ; End of the CY1 module
PRG                    ; Beginning of the PRG module
...
END                    ; End of the PRG module
```

A carriage return (empty line) is required after the last END.

REA, IMA
Real or image values.

They indicate whether the following consultations are carried out using the real values (REA) or image values (IMA) or the I, O, M resources. The rest of the resources have no image values, only real.

The real value is the one the resource has at that time and the image value is the one it had at the end of the previous cycle scan. Image values (IMA) and real values (REA) may be combined in the same instruction.

```
IMA I3 AND REA M4 = 02
```

By default, all the modules (PRG, CY1, PEt) operate with real resource values. Action instructions (=O32) always update the real values of the PLC resources.

Understanding how real and image values work.

The following example shows how the PLC acts when working with real or image values. For the given PLC program and with the resources initialized to zero, it shows the status of all the resources at the end of each scan or cycle.

		REA				IMA			
		M1	M2	M3	O5	M1	M2	M3	O5
()=M1		0	0	0	0	0	0	0	0
M1 = M2	Scan 1	1	1	1	1	1	0	0	0
M2 = M3	Scan 2	1	1	1	1	1	1	0	0
M3 = O5	Scan 3	1	1	1	1	1	1	1	0
	Scan 4	1	1	1	1	1	1	1	1

With real values (REA), output O5 goes high (=1) at the end of the first cycle scan, whereas it needs 4 cycle scans when using image values (IMA).

In the first cycle, ()=M1 sets the real value of M1=1, but its image value is ·0·. Only at the end of this cycle scan will it be ·1·.

The system is faster when working with real values (REA); whereas with image values (IMA), it is possible to analyze the same resource throughout the whole program with the same value, regardless of its current value.

L
Label.

It is used to identify a program line. It may be defined in two ways:

- With L followed by up to 7 digits (L1 - L9999999).
- With L_ followed by 8 characters (L_GEAR).

If it is defined within a module (CY1, PRG or PE), it identifies a program line and allows making references or jumps.

If it is defined outside the modules, for example at the end of the program after END, it indicates the beginning of a subroutine. It is the same as the SUB directing instruction.

If a program has more than one label with the same name or number, the PLC program will issue the corresponding error message when generating the executable program.

SUB
Subroutine definition.

It indicates the beginning of a subroutine. A subroutine is a portion of the program that may be called upon by any executable instruction.

It is defined with SUB followed by a blank space and up to 24 characters. A subroutine must always end with an END instruction.

```
SUB A22
...
END
```

They must be defined outside the modules (PRG, CY1, PE), for example at the end of the program after the directing instruction END.

A subroutine can also start with the L instruction and end with the END instruction.

EXTERN
External subroutine definition.

The subroutines defined in the "C" language file used by the program must be defined as external at the beginning of the program before the DEF instructions and the modules: CY1, PRE and PEt. With the EXTERN instruction, those subroutines may be defined one by one.

It is defined with EXTERN followed by a blank space and the name of the subroutine with up to 24 characters.

```
EXTERN SUMA
EXTERN TEMPERATURE
```

DEF, PDEF
Symbol definition.

Symbols are always programmed at the beginning of the program, before the modules CY1, PRE and PEt. Since the PLC project may consist of several files and the symbols must be defined before they are used, we recommend to define them in the first file of the PLC project. However, the symbols may be defined in any file as long as they are used in that file or in later files.

The PLC allows to define a number of symbols for easier programming and later understanding of the PLC program. These symbols are programmed using the DEF o PDEF instruction followed by the name of the symbol, a constant or PLC resource. The PLC allows associating a symbol may be associated with any decimal or hexadecimal number or PLC resource such as inputs (I), outputs (O), marks (M), registers (R), register bits, counters (C) and timers (T).

DEF — The DEF instruction may be used to define an unlimited number of symbols to be used only from the PLC.

PDEF — The PDEF instruction may be used to define up to 100 symbols that can be used at the PLC itself, in a part-program or in an external application. Symbols exceeding this limit will be ignored and the CNC will issue the corresponding warning.

The symbols associated with a constant can only be used inside the PLC program; they can be accessed neither from the part-program nor from an external application. The symbols associated with a constant cannot be monitored, used in a trace of the logic analyzer or accessed from external variables.

```
PDEF COOL I12
PDEF CONSTANT $FFFF3
DEF DATA_D1 372893
DEF DATA_D3 -437289
DEF /FAN I23
```

Symbols will consist of a sequence of up to 20 characters with capital letters (A..Z) and digits (0..9). Symbols may also begin with the "/" character; in that case, the next character must be a letter. The name may have a "_" character, but it cannot be its first character. Words reserved for instructions cannot be used in symbols.

Duplicate symbols cannot be defined; but several symbols may be assigned to the same resource. Once a symbol has been associated with a resource or numeric value, it is possible to use the name of the resource, the number or its associated symbol.

Accessing the PDEF symbols from a part-program or from an application.

Accessing the symbols defined with the PDEF instruction from a part-program, MDI or external application is done using variables as follows. Consulting this variable from the part-program interrupts block preparation.

V.PLC.symbol	Access from a part-program or MDI.
PLC.symbol	Access from an external application.

The variables may be read or written depending on the resource assigned to the symbol defined with PDEF.

NOMONIT
No monitoring.

When programming this directing instruction, it does not generate information necessary to monitor the PLC program. In other words, the program is not monitored.

It must always be programmed at the beginning of the program, before the DEF instruction and the modules: CY1, PRG and PEt.

This instruction should only be used when the PLC program execution time is very critical. Define it after debugging the PLC program.

Programming example.

```
;No monitoring
    NOMONIT
;External subroutine
    EXTERN TEMPERATURE
;Symbol definition
    DEF COOL I12
    DEF /FAN I23
;CY1 module
    CY1
    ...
    END
;PRG module
    PRG
    ...
    IMA I3 AND REA M4 = 02
    ...
    L_GEAR
    ...
    END
;PEt module
    PE 100
    ...
    END
;Subroutine
    SUB A22
    ...
    END
```

4.2 Consulting instructions.

They may be used to check the status of PLC resources as well as the marks and registers for CNC-PLC communication. There are the following consulting instructions.

- Simple consulting instructions.
- Flank detection instructions.
- Comparing instructions.

4.2.1 Simple consulting instructions.

They test the status of the resources and they return their logic state.

- Inputs (I1-I1024)
- Outputs (O1-O1024)
- Local inputs (LI1-LI16)
- Local outputs (LO1-LO8)
- Marks (M1-M8192)
- Messages (MSG1-MSG1024)
- Errors (ERR1-ERR1024)
- Clocks (CLK)
- Registers (R1-R1024)
- Register bits (B0-B31 R1-R1024)
- Timer status (T1-T512)
- Counter status (C1-C256)
- CNC-PLC communication marks.

I12

Returns a "1" if the I12 input is active and a "0" if otherwise.

START

Returns a "1" when the CYCLE START key of the front panel is pressed and a "0" if otherwise.

4.2.2 Flank detection instructions.

They check whether the state of a resource has changed since the last time this consultation was made. This consultation may be made on real or image values. The instructions available are:

DFU	Detect an up flank.
DFD	Detect a down flank.

DFU
Detect an up flank.

It detects an up-flank (0-to-1 change) at the indicated resource. It returns a "1" if it happened.

DFD
Detect a down flank.

It detects an down-flank (0-to-1 change) at the indicated resource. It returns a "1" if it happened.

The programming format for DFU and DFD is:

DFU	I1..1024
DFD	O1..1024
	M1..8192
	MSG1..1024
	ERR1..1024
	B0..31 R1..1024
	CLK
	CNC-PLC communication marks

```
DFU I23
DFU B3R120
DFU AUXEND
DFD O32
DFD M45
```

4.2.3 Comparing instructions.

CPS
Compare two operands.

With the CPS instruction, it is possible to make comparisons between two operands, checking whether the first one is greater than (GT), greater than or equal to (GE), equal to (EQ), different from (NE), less than or equal to (LE) or less than (LT) the second one.

It is possible to use as operands, timers (internal count), counters (internal count), registers, registers for CNC-PLC communication and decimal (#) or hexadecimal numbers within ±2147483647 or between 0 and $FFFFFFFF.

If the required condition is met, the consulting instruction returns a logic value "1" and a "0" if otherwise.

The programming format is:

CPS	T1..512	GT	T1..512
	C1..256	GE	C1..256
	R1..1024	EQ	R1..1024
	R CNC-PLC	NE	R CNC-PLC
	#	LE	#
		LT	

```
CPS C12 GT R14 = M100
```

If the internal count of counter "C12" is GREATER than the value of register R14, the PLC will assign the value of "1" to mark M100 and a "0" if otherwise.

```
CPS T2 EQ 100 = TG1 5 2000
```

When the time elapsed at timer "T2" is equal to 100 milliseconds, it will trigger timer "T5" in monostable mode and with time constant of 2 seconds.

4.3 Operators and symbols.

They are used to group and operate with different consulting instructions.

The available operators are NOT, AND, OR, XOR. The operators are associated from left to right and the priorities ordered from the highest to the lowest are: NOT AND XOR OR.

The available symbols are: \, (,).

NOT
Negates the result of the consultation.

```
NOT I2 = O3
```
Output "O3" will be active when input I2 is not.

AND
Logic function "AND".

```
I4 AND I5 = O6
```
Output "O6" will be active when both inputs (I4, I5) are active.

OR
Logic function "OR".

```
I7 OR I8 = O9
```
Output "O9" will be active when either one (or both) inputs are active.

XOR
Logic "Exclusive OR" function.

```
I10 XOR I11 = O12
```
Output "O12" will be active when both inputs I10 and I11 have different logic states.

****
Line feed.

It is used to write a logic expression in more than one line The following may be programmed:

```
DFU MSTROBE AND CPS MFUN* EQ 3 = M1003
```

or also:

```
DFU MSTROBE \
AND CPS MFUN* EQ 3
= M1003
```

( )
Open and close parenthesis.

They help clarify and select the order the logic expression is evaluated.

```
(I2 OR I3) AND (I4 OR (NOT I5 AND I6)) = O7
```

A consulting instruction consisting of only these two operators always has a value of "1".

```
( ) = O2
```
Output O2 will always be high "1".

4.4 Action instructions.

Action instructions allow changing the status of PLC resources and CNC-PLC communication marks depending on the result of the logic expression.

Logic expression = Action instruction

There may be several action instructions associated with a single logic expression. All the action instructions must be preceded by the "=" sign.

All the action instructions admit a prior NOT that inverts the result of the expression for that action.

```
I2 = O3 = NOT M100 = NOT TG1 2 100 = CPR 1 100
```

Output O3 will show the status of input I2.

Mark M100 will show the negated state of input I2.

A down-flank at input I2 will activate the trigger input TG1 of timer T2.

An up-flank at I2 will preset counter C1 with a value of 100.

The action instructions are divided into:

- Assignment binary action instructions.
- Conditional binary action instructions.
- Sequence breaking action instructions.
- Arithmetic action instructions.
- Logic action instructions.
- Specific action instructions.

Action instructions may change the status of all the PLC resources except for the physical inputs being used. When seeing the field "I 1/1024", one must understand that only the status of the unused inputs may be changed.

For example, if physical inputs I1 through I32 are used, only inputs I33 through I1024 may be changed.

4.4.1 Assignment binary instructions.

They assign the value (0/1) resulting from a logic expression to the indicated resource.

= I 1/1024	= O 1/1024	= M 1/8192
= MSG 1/1024	= ERR 1/1024	= TEN 1/512
= TRS 1/512	= TGn 1/512 #/R	= CUP 1/256
= CDW 1/256	= CEN 1/256	= CPR 1/256 #/R
= B 0/31 R 1/499	= CNC-PLC mark	

```
I3 = TG1 4 100
```

Assigns the status of input I3 to the trigger input TG1 of timer T4. Thus, an up-flank at I3 will trigger the TG1 input of timer T4.

```
(I2 OR I3) AND (I4 OR (NOT I5 AND I6)) = M111
```

Assigns to mark M111 the result of evaluating the logic expression: (I2 OR I3) AND (I4 OR (NOT I5 AND I6)).

4.4.2 Conditional binary instructions.

There are 3 instructions: SET, RES and CPL, that are used to change the status of the indicated resource.

The programming format is:

= SET	I 1/1024
= RES	O 1/1024
= CPL	M 1/8192
	MSG 1/1024
	ERR 1/1024
	B 0/31 R 1/1024
	CNC-PLC mark

= SET
If expression = "1", it sets the resource to "1".

If the result of the logic expression is a "1", it sets the indicated resource to "1". If the result is "0", it does not change the resource.

```
CPS T2 EQ 100 = SET B0R100
```

When the timing of timer T2 reaches 100 milliseconds, it sets bit 0 of register R100 to "1".

= RES
If expression = "1", it sets the resource to "0".

If the result of the logic expression is a "1", it sets the indicated resource to "0". If the result is "0", it does not change the resource.

```
I12 OR NOT I22 = RES M55 = NOT RES M65
```

When the result of the logic expression is a "1", the PLC sets "M55 = 0"; but does not change M65. If the result of the logic expression is a "0", the PLC does not change M55; but it sets "M65 = 0".

= CPL
If expression = "1", it complements the resource.

If the result of the logic expression is a "1", it complements the status of the indicated resource. If the result is "0", it will not change the resource.

```
DFU I8 OR DFD M22 = CPL B12R35
```

Every time an up-flank is detected at input I8 or down-flank of mark M22, the PLC complements the state of bit 12 of register R35.

4.4.3 Sequence breaking action instructions.

These actions interrupt the sequence of a program, resuming it somewhere else in the program.

= JMP
Unconditional jump.

If the result of the logic expression is a "1", it jumps to the indicated label. If the result is a "0", it goes on to the next program line.

Its syntax depends on how the label to jump to was defined

= JMP L123	If the label was defined as L123.
= JMP L_ASA2	If the label was defined as L_ASA2.

```
I8 = JMP L12
```

If I8=1 the program continues at L12 and it does not execute the intermediate blocks.

```
NOT M14 AND NOT B7R120 = O8
CPS T2 EQ 2000 = O12
L12
(I12 AND I23) OR M54 = O6
```

= CAL
Call to a subroutine.

If the result of the logic expression is a "1", this action executes the indicated subroutine. If the result of the logic expression is a "0", the PLC will ignore this action and the program will go on without executing that subroutine.

Once the subroutine execution is over, the PLC will continue at the action instruction or executable instruction programmed after CAL.

Its syntax depends on how its associated subroutine was defined.

= CAL OILING	If it was defined as SUB OILING.
= CAL L234	If it was defined as L234.
= CAL L_GEAR	If it was defined as L_GEAR.

```
I2 = CAL L5 = O2
```

With I2=1, subroutine L5 will be executed and once executed, the PLC will set O2 to the value of input I2 (=1). If I2=0, the subroutine is not executed and the PLC sets output O2 to the status of input I2 (=0).

= RET
Return or end of a subroutine.

If the result of the logic expression is a "1", the PLC will treat this action like an END instruction. If the result is a "0", the PLC will ignore it.

If while executing a subroutine, the PLC detects a validated RET, it will conclude the subroutine.

4.4.4 Arithmetic action instructions.

= MOV
Move.

It is used to move data from one PLC resource to another.

The programming format is:

	Origin	Destination	Origin code	Destination code	Bits to transmit
= MOV	I1/1024	I1/1024	0(Bin)	0(Bin)	32
	O1/1024	O1/1024	1(BCD)	1(BCD)	28
	M1/8192	M1/8192			24
	MSG1/1024	MSG1/1024			20
	ERR1/1024	ERR1/1024			16
	T1/512	R1/1024			12
	C1/256	R CNC-PLC			8
	R1/1024				4
	R CNC-PLC				
	#				

The Origin and Destination codes indicate which format (binary or BCD) they have and in which format they will be deposited in the destination resource. 4, 8, 12, 16, 20, 24, 28 or 32 bits may be transmitted.

If the codes and number of bits to be moved are not indicated, 32 binary bits will be moved bit to bit (0032).

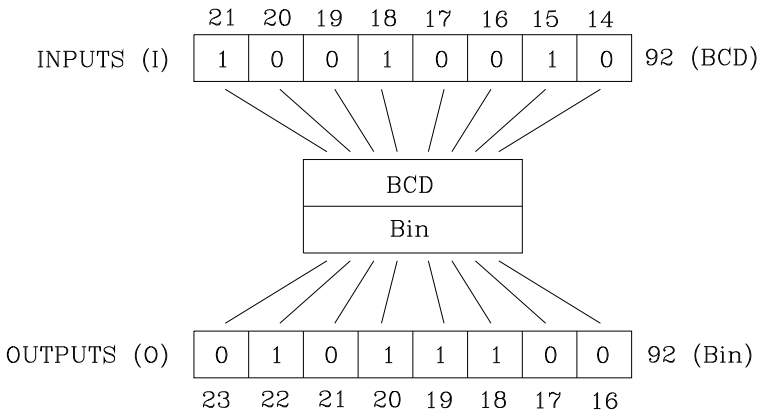
MOV	I12	M100	0032	binary to binary in 32 bits.
MOV	O21	R100	0012	binary to binary in 12 bits.
MOV	C22	O23	0108	binary to BCD in 8 bits.
MOV	T10	M112	1020	BCD to binary in 20 bits.

If the number to be converted from binary to BCD is larger than the maximum BCD, its value will be truncated ignoring the most significant bits.

The maximum BCD value that can be converted is: 9 (with 4 bits), 99 (with 8), 999 (with 12), 9999 (with 16), 99999 (with 20), 999999 (with 24), 9999999 (with 28) and 99999999 (con 32). In these cases, it is recommended to make the move increasing the number of bits by using, if necessary, registers or marks in intermediate steps.

```
I11 = MOV I14 O16 108
```

If input I11 is "=1", the PLC moves the logic states of the 8 inputs (I14 plus the next 7) in BCD code to the 8 outputs (O16 and the next 7) in binary code.



= NGU R 1/1024
Complements the bits of a register.

It changes the state of each one of the 32 bits of a register.

```
I15 = NGU R152
```

If input "I15 is =1", the PLC changes the state of the 32 bits of register R152.

R152 before: 0001 0001 0001 0001 0001 0001 0001 0001

R152 after: 1110 1110 1110 1110 1110 1110 1110 1110

= NGS R 1/1024
Register sign change.

Changes the sign of the register.

```
I16 = NGS R89
```

If input "I16 = 1", the PLC changes the sign of the contents of register R89.

R89 before: 0001 0001 0001 0001 0001 0001 0001 0001

R89 after: 1110 1110 1110 1110 1110 1110 1110 1111

= ADS, = SBS, = MLS, = DVS, = MDS
Arithmetic operations.

for adding (ADS), subtracting (SBS), multiplying (MLS), dividing (DVS) and calculating the module or remainder of a division (MDS).

Its programming format is:

"Operation" "1st operand" "2nd operand" "Result".

- The operands may be: Registers, CNC-PLC communication registers and numbers (#) in the ±2147483647 range or between 0 and $FFFFFFFF.
- The result of the operation may be stored in a register or in a CNC-PLC communication register.

= ADS	R1/1024	R1/1024	R1/1024
= SBS	R CNC-PLC	R CNC-PLC	R CNC-PLC
= MLS	#	#	
= DVS			
= MDS			

Examples with R100=1234 and R101=100.

```
() = ADS    R100    R101    R102    ; R102 = 1234 + 100          = 1334
() = SBS    R100    R101    R103    ; R103 = 1234 - 100          = 1134
() = MLS    R100    R101    R104    ; R104 = 1234 x 100          = 123400
() = DVS    R100    R101    R105    ; R105 = 1234 : 100          = 12
() = MDS    R100    R101    R106    ; R106 = 1234 MOD 100        = 34

() = ADS    1563    R101    R112    ; R112 = 1563 + 100          = 1663
() = SBS    R100    1010    R113    ; R113 = 1234 - 1010         = 224
() = MLS    1563    100     R114    ; R114 = 1563 x 100          = 156300

() = MLS    SANALOG 10000   R115
 = DVS      R115    32767   R115    ; Spindle speed command in mV.
```

4.4.5 Logic action instructions.

= AND
= OR
= XOR
Logic operations

To perform logic operations: AND, OR and XOR between register contents or between a register content and a number. The result is always stored in a register.

Its programming format is:

AND	R1/1024	R1/1024	R1/1024
OR	R CNC-PLC	R CNC-PLC	R CNC-PLC
XOR	#	#	

Examples with R200 = B1001 0010 and R201 = B0100 0101.

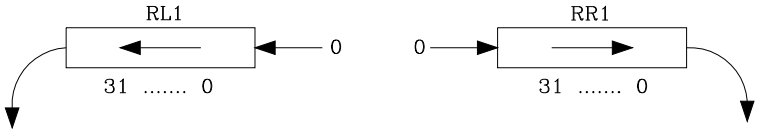
() = AND	R200	R201	R202	; R202 = B0
() = OR	R200	R201	R203	; R203 = B11010111
() = XOR	R200	R201	R204	; R204 = B11010111
() = AND	B1111	R201	R205	; R205 = B00000101
() = OR	R200	B1111	R206	; R206 = B10011111
() = XOR	B1010	B1110	R207	; R207 = B00000100

= RR1
= RR2
= RL1
= RL2
Register rotation.

Register contents may be rotated to the right (RR) or to the left and there are two types of rotations: type 1 (RR1 or RL1) and type 2 (RR2 or RL2).

Rotation type 1 (RL1 or RR1):

It inserts a "0" at the least significant bit (RL1) or at the most significant bit (RR1), pushing the other bits of the register. The value of the last bit disappears.



Rotation type 2 (RL2 or RR2):

Circular rotation of the register in the indicated direction.



Programming format:

The programming format is:

	Origin	Number of repetition	Dtarget
RR1	R1/1024	R1/1024	R1/1024
RR2	R CNC-PLC	R CNC-PLC	R CNC-PLC
RL1		0/31	
RL2			

The origin and destination registers must always be defined, even when they are both the same. The number of repetitions indicates the consecutive number of times the register will be rotated.

RR1 R100 1 R200

It makes one type-1 right-hand rotation of R100 leaving the result in R200.

RL2 R102 4 R101

It makes four type-2 left-hand rotation of R102 leaving the result in R101.

4.4.6 Specific action instructions.

= ERA
Clears a group of resources.

It is used to clear or initialize a group of same-type resources. Indicate the first and last resource to be erased.

The programming format is:

= ERA	I1/1024	1/1024
	O1/1024	1/1024
	M1/8192	1/8192
	MSG1/1024	1/1024
	ERR1/1024	1/1024
	T1/512	1/512
	C1/256	1/256
	R1/1024	1/1024

When erasing a group of I, O, M, MSG, ERR or R, the PLC sets them all to "0".

Erasing a group of timers is like resetting them and erasing a group of counters is like presetting with a "0" value.

This action is especially handy when executed in the first cycle module (CY1) in order to set the desired resources to their initial work conditions (states).

```
I10 = ERA O5 12
```

If input "I10=1", the PLC sets outputs O5 through O12 (both included) to "0".

```
I23 = ERA C15 18
```

If input "I23 =1", the PLC presets counters C15 through C18 (both included) to "0".

=PAR
Parity of a register.

It analyzes the type of parity of a register. If the register has EVEN parity, this instruction sets the selected mark, message or error to "1" and to "0" if it has ODD parity.

The programming format is:

= PAR	R1/1024	M1/8192
	R CNC-PLC	MSG1/1024
		ERR1/1024
		M CNC-PLC

```
I15 = PAR R123 M222
```

If input "I15=1", the PLC analyzes register R123 and sets mark M222 to "1" if EVEN parity or to "0" if ODD parity.

= CNCRD
= CNCWR
Reading (CNCRD) and writing (CNCWR) of CNC variables.

Used to read (CNCRD) and write (CNCWR) the internal CNC variables. Their programming format is:

CNCRD (Variable, Register, Mark)

CNCWR (Register, Variable, Mark)

The CNCRD action loads the contents of the variable into the register and the CNCWR action loads the contents of register into the variable. The mark is set to "1" at the beginning of the operation and it keeps its value until the end of the operation.

When using CNCRD to read the variables of the arithmetic parameters and those of the OEM, it returns the value multiplied by 10000 (float mode reading).

When requesting information on a nonexistent variable (for example the position of an nonexistent axis), it will show the relevant error message. Likewise, when trying to read a value whose range is greater than that of the PLC register, the CNC will return a zero value

and will issue the corresponding error message. Whenever an error occurs when reading a variable, the communication mark will remain at "1".

Syntax of the variables in commands CNCRD and CNCWR.

For these two commands, it is possible to define, in the mnemonic of the variable, the channel number using an integer, a register or a symbol defined with DEF or PDEF.

```
CNCRD ([1].G.FREAL, R10, M1000)
```

```
... = MOV 1 R12
CNCRD ([R12].G.FREAL, R10, M1000)
```

```
DEF CHANNEL 3
CNCRD ([CHANNEL].G.FREAL, R10, M1000)
```

If the mnemonic of the variable contains numerical suffixes, e.g. (V.).G.GUP[i], they too can be defined with an integer, a register or a symbol defined with DEF or PDEF.

```
... = MOV 153 R101
CNCWR (G.GUP[R101], R10, M1000)
```

```
DEF PARAM 153
CNCRD (G.GUP[PARAM], R10, M1000)
```

Consulting synchronous and asynchronous variables.

Synchronous variables are the ones resolved immediately whereas asynchronous variables are the ones requiring several cycle scans to be resolved.

- Example of how to access asynchronous variables:

  `<condition> AND NOT M11 = CNCRD (TM.TOOL, R11, M11)`

  do not repeat this consultation until it ends.

  `DFD M11 AND CPS R11 EQ 3 = ...`

  wait for the consultation to end before comparing the data.

- Examples of how to access synchronous variables:

  `<condition> = CNCRD (G.FREAL, R12, M12)`
  `CPS R12 GT 2000 = ...`

  No need to wait before consulting the data because synchronous variables are resolved immediately.

  `<condition> = CNCWR (R13, PLC.TIMER, M13)`

  It resets the clock enabled by the PLC with the value contained in register R13.

= CNCEX
Execution of a CNC block.

It may be used to execute a CNC block in the indicated channel, including calls to subroutines or to complete programs. It works like when executing a block in MDI. The restrictions for the execution of commands are the same as for the MDI blocks.

Its programming format is:

CNCEX (block, mark, channel)

The mark is set to "1" at the beginning of the operation and it keeps its value until the end of the operation. If the channel is not indicated, the block is executed in the first or main channel.

```
... = CNCEX (G00 X0 Y0, M99, 2)
... = CNCEX (#CALL sub3.nc, M34)
```

Once the block has been executed, the CNC channel activates the FREE mark to let the PLC know that it is ready to accept a new block. The execution of the CNCEX command may be canceled with the mark PLCABORT.

Executing independent movements from the PLC. Commands MOVEABS, MOVEADD and MOVEINF.

The movements of the independent axes may be programmed directly or with the CNCEX command; however, it is not recommended to use both methods in the same PLC program or subroutine.

The treatment for the execution of the commands is different and the order they are executed might not be the desired one. The CNCEX command is executed through a CNC channel whereas the MOVE* commands are executed directly at the interpolator (usually faster execution).

The independent movements may be executed as follows: The two should not be used in the same program or subroutine.

```
() = CNCEX(#MOVE ADD [X100,F100,NULL], M120,1)
() = MOVEADD(X,100000,100000,NULL)
```

4.4.7 Action instruction of the electronic cam.

This function has a specific manual. This manual that you are reading now only offers some information about this function. Refer to the specific documentation to obtain further information regarding the requirements and operation of the electronic cam.

= CAM ON
= TCAM ON
= CAM OFF
Activate the electronic cam with real coordinates (CAM ON), with theoretical coordinates (TCAM ON) or cancel the electronic cam (CAM OFF).

The programming format for each of them is the following.

CAM ON (cam, master/"TIME", slave, master_off, slave_off, range_master, range_slave, type)
TCAM ON (cam, master/"TIME", slave, master_off, slave_off, range_master, range_slave, type)
CAM OFF (slave)

Executing the CAM OFF command involves eliminating the synchronization of the cam. Once this command has been programmed, the cam ends when reaching the end of its profile.

Parameter	Meaning
cam	Cam number.
master	Name of the master axis.
TIME	Time cam. When programming "TIME" instead of an axis name, the cam is interpreted as being a time cam.
slave	Name of the slave axis.
master_off	Offset for the master axis.
slave_off	Offset for the slave axis.
range_master	Master axis activation scale or range.
range_slave	Slave axis activation scale or range.
type	It defines the type of cam; periodic or not periodic. It is programmed using parameters "ONCE" (non-periodic cam) or "CONT" (periodic cam).

Cam mode.

Two types of cams may be activated; cams in time or cams according to the position of the master axis. The activation instruction is the same and it is selected by the call-parameters.

Cam number.

To activate a cam, it must have been previously defined at the cam editor, within the machine parameters.

Master axis activation range.

The cam is activated when the master axis is between the positions "master_off" and "master_off + range_master".

Range for the slave axis.

The cam applies it to the slave axis when the slave axis is between "slave_off" and "slave_off + range_slave".

Cam type.

Depending on the execution mode, the time cams and the position cams may be of two different types; i.e. periodic or non-periodic. It is selected with the type parameter.

Not periodic — It is defined by assigning the "ONCE" value to the type parameter.

This mode maintains the synchronization for the range defined for the master axis. If the master axis moves backwards or if it is a module, the slave axis will keep on executing the cam profile until the cancellation is programmed.

Periodic — It is defined by assigning the "CONT" value to the type parameter.

In this mode, when reaching the end of the range of the master axis, it calculates the offset to execute the cam again shifted in the amount of that range. In other words, identical cams are executed along the path of the master axis.

If the master axis is a rotary module and the cam definition range is that module, the two execution modes are equivalent.

Either mode maintains synchronization until the #CAM OFF command is executed. When reaching that command, the execution of the cam will end the next time the end of the cam profile is reached.

= CAM SELECT
= CAM DESELECT
Select (CAM SELECT) or cancel a file cam (CAM DESELECT).

The data of a cam may be defined in a file that may be loaded from the CNC or from the PLC. When executing a cam from a file, the CNC reads its data dynamically and, consequently, the number of points to define the cam is unlimited.

The following commands only define the location of the cam; to activate it, use the instruction #CAM ON (from the CNC) or the command CAM ON (from the PLC). After selecting a file cam, it stays available until the cam table of the machine parameters is validated or the CNC is turned off.

The programming format for each of them is the following.

```
CAM SELECT (cam, file)
CAM DESELECT (cam)
```

Executing the CAM OFF command involves eliminating the synchronization of the cam. Once this command has been programmed, the cam ends when reaching the end of its profile.

Parameter	Meaning
cam	Cam number.
file	Name and path of the file containing the cam data.

4.4.8 Instructions for independent move: positioning.



This function has a specific manual. This manual that you are reading now only offers some information about this function. Refer to the specific documentation to obtain further information regarding the requirements and operation of the independent axes.

= MOVE ABS
= MOVE ADD
= MOVE INF
Absolute positioning movement (MOV ABS), incremental (MOVE ADD) or no-end (MOVE INF).

The programming format for each of them is the following.

```
MOVE ABS (axis, pos, feed, blend)
MOVE ADD (axis, pos, feed, blend)
MOVE INF (axis, direction, feed, blend)
```

Parameter	Meaning
axis	Axis to position.
pos	Position to reach.
direction	Moving direction. It is programmed with parameters "DIRPOS" (positive direction) or "DIRNEG" (negative direction).
feed	Positioning feedrate
blend	Dynamic blend with the next block It is programmed using parameters "PRESENT", "NULL", "NEXT" or "WAITINPOS.

The programming units are the standard PLC units. The coordinates the feedrates will be given in ten-thousandths if they are in mm or degrees or in hundred-thousandths if they are in inches.

Position to reach.

With MOVE ABS it will be defined in absolute coordinates whereas with MOVE ADD it will be defined in incremental coordinates. For positioning, the zero offset active in the channel is ignored.

The moving direction is determined by the coordinate or the increment programmed. For rotary axes, the moving direction is determined by the type of axis. If it is unidirectional, it positions in the preset direction; otherwise, it positions via the shortest path.

Moving direction.

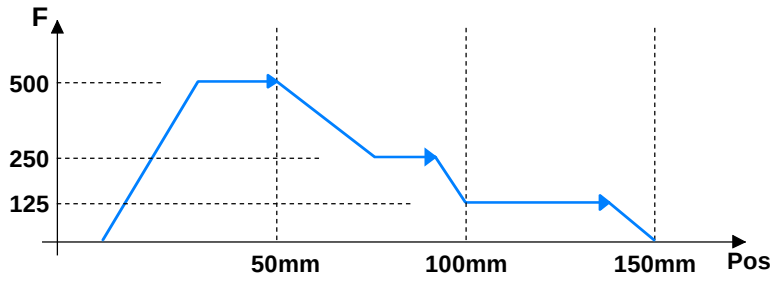
Moving direction. It is used with MOVE INF to execute an endless (infinite) movement until the axis limit is reached or until the movement is interrupted.

Dynamic blend with the next block

It sets the feedrate used to reach the position (dynamic blend with the next block). It is programmed with one of the following parameters.

PRESENT	The axis reaches the indicated position at the feedrate indicated by the block itself.
NEXT	The axis reaches the indicated position at the feedrate indicated in the next block.
NULL	The axis reaches the indicated position at zero feedrate.
WAITINPOS	The axis reaches the indicated position at zero feedrate and it waits to be in position before executing the next block.

```
.. = MOVE ABS (X, 500000, 5000000, PRESENT)
.. = MOVE ABS (X, 1000000, 2500000, NEXT)
.. = MOVE ABS (X, 1500000, 1250000, NULL)
```



4.4.9 Instructions for independent move: synchronization.



This function has a specific manual. This manual that you are reading now only offers some information about this function. Refer to the specific documentation to obtain further information regarding the requirements and operation of the independent axes.

= FOLLOW ON
= TFOLLOW ON
= FOLLOW OFF
Activate the synchronized movement with real coordinates (FOLLOW ON), with theoretical coordinates (TFOLLOW ON) or cancel the synchronized movement (FOLLOW OFF).

The programming format for each of them is the following.

```
FOLLOW ON (master, slave, nratio, dratio, synctype)
TFOLLOW ON (master, slave, nratio, dratio, synctype)
FOLLOW OFF (slave)
```

Parameter	Meaning
master	Name of the master axis.
slave	Name of the slave axis.
nratio	Numerator of the gear ratio. Turns of the slave axis.
dratio	Denominator of the gear ratio. Rotations of the master axis.
synctype	Type of synchronism. It is programmed with parameters "POS" (position synchronism) or "VEL" (velocity synchronism).

The programming units are the standard PLC units. The coordinates the feedrates will be given in ten-thousandths if they are in mm or degrees or in hundred-thousandths if they are in inches.

Example from the PLC program.

```
FOLLOW ON (A1, Z, 3, 1, VEL)
TFOLLOW ON (A1, Z, 3, 1, VEL)
FOLLOW OFF (Z)
```

Velocity synchronization.

Velocity offset.

It may be used to vary the speed of the slave axis independently from that of the master axis. It is defined with the variable V.A.SYNCVELOFF.xn.

Gear ratio.

Ratio (Nslave/Nmaster) between the number of turns of the slave axis (Nslave) and the number of turns of the master axis (Nmaster).

Fine adjustment of the gear ratio.

The gear ratio is determined when programming the instruction and its value remains constant during the whole operation. However, and even if the synchronization is in progress, this ratio may be modified by fine-adjusting it. The fine adjustment of the ratio is defined with the variable GEARADJ.

Position (phase) synchronization.

Position offset.

It may be used to vary the position of the slave axis independently from that of the master axis. It is defined with the variable V.A.SYNCPOSOFF.xn.

Gear ratio.

Ratio (Nslave/Nmaster) between the number of turns of the slave axis (Nslave) and the number of turns of the master axis (Nmaster).

The gear ratio is determined when programming the instruction and its value remains constant during the whole operation. In this synchronization mode, it is not possible to change this value while the system is running, because this behavior is more for an electronic cam than for an electronic gear. To solve this matter, an electronic cam may be programmed instead.

4.4.10 Instructions for coordinate latching using a probe or digital input.

Coordinate latching means that the CNC captures the position value of an axis when a given event occurs. The TOUCHPROBE instruction latches the coordinate of an axis by an event of a probe, of a physical digital input or of a logic input. The events to be used will be digital inputs defined in parameters PRBDI1 and PRBDI2.

- If the selected event is a probe and when activating the TOUCHPROBE command the probe has already reached its logic level (the probe acts either by level or by flank), the latched coordinate will be the one captured when the command was activated, not that of the point where the level reaching event occurred.
- If the selected event is a logic input, it may be activated from the PLC program and, therefore, it may be conditioned indirectly by any other event managed by the PLC.

The coordinate may be latched in any position of the axis travel or a latch window may be defined. If there is a latch window, coordinate latching may be conditioned to whether the axis is inside or outside the window.

The following probe-related machine parameters have no effect: PROBEAXIS, PROBERANGE, PROBEFEED, PRB1MIN, PRB2MIN, PRB3MIN, PRB1MAX, PRB2MAX and PRB3MAX.

= TOUCHPROBE
Coordinate latching.

The programming format is: Optional parameters are indicated between angle brackets.

```
TOUCHPROBE (axis,probe<,wintype><,winminpos,winmaxpos>)
```

Parameter	Meaning
axis	Name of the axis whose coordinate is being latched.
probe	Number of the probe to be used as latching event.
wintype	Type of window for latching. It is programmed with the commands DISABLE (without window), EXCLUSIVE (latch outside the window) and INCLUSIVE (latch inside the window).
winminpos	Lower limit for the latch window.
winmaxpos	Upper limit for the latch window.

The programming units are the standard PLC units. The coordinates the feedrates will be given in ten-thousandths if they are in mm or degrees or in hundred-thousandths if they are in inches.

```
TOUCHPROBE (X, 1, DISABLE)
TOUCHPROBE (X, 1)
TOUCHPROBE (Y, 2, EXCLUSIVE, 1000000, 2300000)
TOUCHPROBE (X3, 1, INCLUSIVE, 500000, 1105000)
```

·axis· Name of the axis whose coordinate is being latched.

Name of the axis whose coordinate is being latched; spindles are not admitted, except if they are active as C axis.

·probe· Number of the probe to be used as latching event.

The names of the probes are set by the order in which the machine parameters have been defined. Probe 1 will be the one assigned to parameter PRBDI1 and probe 2 will be the one assigned to parameter PRBDI2.

Instead of a probe, a physical digital input or a logic input may also be used as an event. In this case, the input bein gused must also be defined in parameter PRBDI1 or PRBDI2.

·wintype· Type of window for latching.

This option is programmed using the following commands.

DISABLE	There is no latch window (default option). A coordinate may be latched in any position of the axis travel.
EXCLUSIVE	The coordinate is latched if the axis is outside the window.
INCLUSIVE	The coordinate is latched if the axis is inside the window.

·winminpos· ·winmaxpos· Size of the latch window.

The size of the window must be defined when the latch window is either exclusive or inclusive. Call parameters ·winminpos· and ·winmaxpos· set the lower and upper limits of the latch window.

Coordinate latching operation.

For the same probe, a coordinate latching process and a probing process programmed with G100 cannot be active at the same time, even if it is on different axes; when executing one of them while the other one is active, the CNC will issue an error message. Coordinate latching is not affected by functions G101 and G102.

CNC	PLC	
#PROBE SELECT 1 G100 X100	·· = TOUCHPROBE (X,1)	Wrong.
#PROBE SELECT 1 G100 X100	·· = TOUCHPROBE (Z,1)	Wrong.
#PROBE SELECT 1 G100 X100	·· = TOUCHPROBE (X,2)	Right.
#PROBE SELECT 2 G100 X100	·· = TOUCHPROBE (X,1)	Right.

The TOUCHPROBE instruction may be activated simultaneously for any of the 2 probes and for any of the axes of the system. An active coordinate latching may also be programmed again.

```
·· = TOUCHPROBE (X,1)
·· = TOUCHPROBE (Y,1)
·· = TOUCHPROBE (X,2)
·· = TOUCHPROBE (Y,2)
```

All active latching processes that are assigned to the same probe are canceled when a latch event occurs on any of them. The event cancels all the latching processes assigned to that probe, even on the axes that are positioned outside the latch window. The independent interpolator will only activate the LATCH1DONE(axis) or LATCH2DONE(axis) mark of the axis where the event took place.

Cancelling the latching processes.

The latching process on the axis is canceled when the PLC activates the IRESET(axis) mark. When this mark cancels a latching process, the independent interpolator cancels the LATCH1ACTIVE(axis) or LATCH2ACTIVE(axis) mark associated with the process.

A reset of the channel cancels the latching processes on all axes of the channel.

Influence on functions M02 and M30.

Functions M02 and M30 will not be considered executed (done) until all the latching processes active on the axes of the channel are finished.

Consultation signals of the PLC associated with coordinate latching.

The independent interpolator has a mark for each probe. Probe 1 will be the one assigned to parameter PRBDI1 and probe 2 will be the one assigned to parameter PRBDI2.

PROBE1ACTIVE
PROBE2ACTIVE

The independent interpolator actives this mark when a latching process is active with the indicated probe and it deactivates when there is no active latching process with the indicated probe.

LATCH1ACTIVE(axis)
LATCH2ACTIVE(axis)

The independent interpolator actives this mark when activating a latching process on the axis with the indicated probe and it deactivates when the latching process ends or it is canceled.

The latching process on the axis is canceled when the PLC activates the IRESET(axis) mark. A reset of the channel cancels the latching processes on all axes of the channel.

Functions M02 and M30 will not be considered executed (done) until all the latching processes active on the axes of the channel are finished.

LATCH1DONE(axis)
LATCH2DONE(axis)

The independent interpolator actives this mark when the latch event on the axis with the indicated probe occurs; it deactivates it when activating a new latching process on the axis with the same probe.

Variables related to coordinate latching.

The coordinate latching processes do not affect the variables associated with probing G100.

(V.)[ch].A.LATCH1.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.

Machine coordinate obtained from latching probe 1 on the ·xn· axis.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.A.LATCH1.Z	Z axis.
V.A.LATCH1.4	Axis with logic number ·4·.
V.[2].A.LATCH1.1	Axis with index ·1· in the channel ·2·.

(V.)[ch].A.LATCH2.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.

Machine coordinate obtained from latching probe 2 on the ·xn· axis.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.A.LATCH2.Z	Z axis.
V.A.LATCH2.4	Axis with logic number ·4·.
V.[2].A.LATCH2.1	Axis with index ·1· in the channel ·2·.

(V.)[ch].A.ACCUDIST.xn

Variable that can be read and written via program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Distance traveled by the axis or spindle since the last coordinate latching. This variable is initialized to ·0· when a latching event takes place. To add a position offset to this variable at the latch point, just add it from the PLC in a later cycle.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.MPA.ACCUDIST.Z	Z axis.
V.MPA.ACCUDIST.4	Axis with logic number ·4·.
V.[2].MPA.ACCUDIST.1	Axis with index ·1· in the channel ·2·.

(V.)[ch].A.PREVACCUDIST.xn

Variable that can be read and written via program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Distance traveled by the axis or spindle between the last two coordinate latchings. This variable updates (refreshes) its value at every latching event; that's why the variable will have a ·0· value until the first one takes place.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.MPA.PREVACCUDIST.Z	Z axis.
V.MPA.PREVACCUDIST.4	Axis with logic number ·4·.
V.[2].MPA.PREVACCUDIST.1	Axis with index ·1· in the channel ·2·.

4.5 Summary programming commands.

Resources available at the PLC.

Inputs (I1..1024)
Outputs (O1..1024)
Marks (M1..8192)
Message marks (MSG1..1024)
Error marks (ERR1..1024)
Clocks (CLK)
CNC-PLC marks
Timers (T1..512)
Counters (C1..256)
Registers (R1..1024)
CNC-PLC registers

The register value may be treated as a decimal or hexadecimal ("S") number.

It is also possible to refer to a register bit using the letter B (0..31) R (1..1024).

Clock	Time	Clock	Time	Clock	Time
CLK1	1 ms	CLK100	100 ms	CLK1000	1 s
CLK2	2 ms	CLK200	200 ms	CLK2000	2 s
CLK4	4 ms	CLK400	400 ms	CLK4000	4 s
CLK8	8 ms	CLK800	800 ms	CLK8000	8 s
CLK16	16 ms	CLK1600	1.6 s	CLK16000	16 s
CLK32	32 ms	CLK3200	3.2 s	CLK32000	32 s
CLK64	64 ms	CLK6400	6.4 s	CLK64000	64 s
CLK128	128 ms	CLK12800	12.8 s	CLK128000	128 s

Directing instructions.

PRG	Main module
PE t	Periodic module. It is executed every "t" milliseconds.
CY1	First Cycle module.
END	End of module.
L	Label.
SUB	Subroutine definition.
DEF:	Symbol definition.
PDEF	External symbol definition.
REA	The consultations will use real values.
IMA	The consultations will use image values.
NOMONIT	No PLC program monitoring.
EXTERN	External subroutine definition.

Consulting instructions.

- Simple consulting instructions.

I1..1024	Inputs.
O1..1024	Outputs.
M1..8192	Marks.
MSG1..1024	Messages.
ERR1..1024	Errors.
T1..512	Timers (status).
C1..256	Counters (status).
B0..31 R1..1024	Register bit.
CLK	Clocks.
M <CNC-PLC>	Marks for CNC-PLC communication.

- Flank detection instructions.

DFU	Up flank detection.
DFD	Down flank detection.

- Comparing instructions.

CPS	For comparisons.

Operators.

NOT	Negates the result of the consultation.
AND	Logic function "AND".
OR	Logic function "OR”.
XOR	Logic "Exclusive OR" function.
\	Line feed.
( )	Consulting instruction whose value is always "1".

Action instructions.

- Assignment binary action instructions.

= I 1/1024	= O 1/1024	= M 1/8192
= MSG 1/1024	= ERR 1/1024	= TEN 1/512
= TRS 1/512	= TGn 1/512 #/R	= CUP 1/256
= CDW 1/256	= CEN 1/256	= CPR 1/256 #/R
= B 0/31 R 1/499	= CNC-PLC mark	

- Conditional binary action instructions.

= SET	If expression = "1", it sets the resource to "1".
= RES	If expression = "1", it sets the resource to "0".
= CPL	If expression = "1", it complements the resource.

- Sequence breaking action instructions.

= JMP L	Unconditional jump.
= CAL	Call to a subroutine.
= RET	Return or end of a subroutine.

- Arithmetic action instructions.

= MOV	Move.
= NGU R1..1024	Complements the bits of a register.
= NGS R1..1024	Register sign change.
= ADS	Add.

= SBS Substract.

= MLS Multiplication.

= DVS Division.

= MDS Module or remainder of a division.

- Logic action instructions.

  = AND Logic operation "AND".

  = OR Logic operation "OR".

  = XOR Logic operation "XOR".

  = RR 1/2 Right-hand register rotation.

  = RL 1/2 Left-hand register rotation.

- Specific action instructions.

  = ERA Erases or resets a group of resources.

  =PAR Parity of a register.

  =CNCRD Reading of internal variables.

  =CNCWR Writing of internal variables.

  = CNCEX Execution of a CNC block.

- Action instruction of the electronic cam.

  = CAM ON Activate the electronic cam (real coordinates).

  = TCAM ON Activate the electronic cam (theoretical coordinates).

  = CAM OFF Cancel the electronic cam.

  = CAM SELECT Selet a file cam.

  = CAM DESELECT Cancel the cam of a file.

- Action instructions for independent axes. Positioning move.

  = MOVE ABS Absolute positioning move.

  = MOVE ADD Incremental positioning move.

  = MOVE INF Infinite (endless) positioning move.

- Action instructions for independent axes. Synchronization movement.

  =FOLLOW ON Activates the synchronization movement (real coordinates).

  = TFOLLOW ON Activates the synchronization movement (theoretical coordinates).

  = FOLLOW OFF Cancels the synchronization movement.

- Action instructions for independent axes. Coordinate latching with the help of a probe or a digital input.

  = TOUCHPROBE Coordinate latching.

CNC-PLC COMMUNICATION.

5

With the data exchange between the CNC and the PLC, it is possible to:

- Control the logic CNC inputs and outputs in peripheral mode using specific PLC marks and registers.
- Transfer auxiliary M, H and S functions from the CNC to the PLC.
- Generate messages and errors at the CNC using PLC marks.
- Read and modify internal CNC variables from the PLC.
- Access all PLC resources from any part-program.
- Monitor PLC resources on the CNC screen.

Abbreviations used in this chapter.

(=0) Low logic level.

(=1) High logic level.

–M– and –H– functions with channels.

The M and H functions are exchanged by channel. When using several channels, the marks and registers of these functions must indicate the channel number they refer to. If no channel number is indicated, the marks and registers refer to the first channel.

–S– functions with multiple spindles.

The exchange of S functions is independent from the channel. When using several spindles, the marks and registers of these functions refer to the spindle number. The spindle number is determined by its logic number.

5.1 Auxiliary –M– functions.

Up to 7 M functions in the same block. The CNC indicates to the PLC which auxiliary M functions are programmed in the execution block using 32-bit registers MFUN1 through MFUN7. Each one of them indicates the number of one of the M functions programmed in the block. If all the registers are not used, the CNC assigns $FFFFFFFF to the unused ones (those with the highest numbers).

This way, if a block contains functions M100, M120 and M135, the CNC will transfer the following information to the PLC.

MFUN1	MFUN2	MFUN3	MFUN4 - MFUN7
100	120	135	$FFFFFFFF

Command MFUN*. Checking if a function has been programmed in the block.

To know whether a particular M function is programmed in the execution block, use one of the following methods:

- Check all the MFUN registers one by one until that particular M function is found or until one of them has the value of $FFFFFFFF.
- Use the MFUN* command to check all the registers at the same time.

Example to detect M30. If it has been programmed, it will return a "1", and a "0" if otherwise.

```
CPS MFUN* EQ 30 = ...
```

Sending the function and synchronizing the execution.

Within the CNC machine parameters, the auxiliary M function table indicates when the function is sent and when the PLC execution is synchronized. In either case, it may be before or after the movement.

The sending and synchronizing types may be the following:

M not synchronized.

M sent out and synchronized before the movement.

M sent out before the move and synchronized after the movement.

M sent out and synchronized after the movement.

M functions with different types of synchronization may be programmed in the same block. Each one of them will be sent out to the PLC at the right moment. The transfer of auxiliary M functions is described later on in this chapter. See *"5.4 Transferring auxiliary functions -M-, -H-, -S-."* on page 282.

The functions may be set as follows:

M11 not synchronized.

M12 is sent and synchronized before the movement.

M13 is sent before and synchronized after the movement.

M14 is sent and synchronized after the movement.

When executing a block like this:

```
X100 F1000 M11 M12 M13 M14
```

The functions are transferred as follows:

1. It sends the M11 M12 and M13 functions out to the PLC.
2. waits for the PLC to execute the M12.
3. The CNC moves the axis to X100.
4. It sends the M14 function to the PLC.
5. It waits until the PLC executes the M13 and M14.

5.1.1 Special considerations with the multi-spindle option and channels.

The CNC may have up to 4 channels and each channel can execute a part-program in parallel with the rest. This means that each channel can execute seven auxiliary functions simultaneously. The auxiliary functions executed from each channel are treated independently; to do that, each channel has its own marks and registers.

Since each channel may have four spindles, it is possible to program in the same block 6 non-spindle-related M functions, the startup of all four spindles M3 / M4 and a speed for each of them involving an automatic gear change. This means that, because some functions are generated automatically, it may exceed the maximum of seven auxiliary functions per block. In this case, the CNC will send the M functions out to the PLC in two stages.

Marks and registers in the channels option.

Each channel has 32-bit registers MFUN1 to MFUN7 to indicate to the PLC which auxiliary M functions are programmed in the execution block.

MFUN1C1 - MFUN7C1	for the first channel.
MFUN1C2 - MFUN7C2	for the second channel.
MFUN1C3 - MFUN7C3	for the third channel.
MFUN1C4 - MFUN7C4	for the fourth channel.

Each one of them indicates the number of one of the M functions programmed in the block. If all the registers are not used, the CNC assigns $FFFFFFFF to the unused ones (those with the highest numbers).

This way, if functions M100 and M135 are programmed in the first channel and functions M88 and M75 in the second channel, the CNC will transfer the following data.

MFUN1C1	MFUN2C1	MFUN3C1 - MFUN7C1
100	135	$FFFFFFFF

MFUN1C2	MFUN2C2	MFUN3C2 - MFUN7C2
88	75	$FFFFFFFF

Commands MFUNC1* - MFUNC4*. Checking if a function has been programmed in the channel.

To know whether a particular M function is programmed in the execution block, use one of the following methods:

- Check all the MFUN registers one by one until that particular M function is found or until one of them has the value of $FFFFFFFF.
- Use one of the following commands to check all the MFUN registers of the channel at the same time.

MFUNC1*	For channel 1
MFUNC2*	For channel 2.
MFUNC3*	For channel 3.
MFUNC4*	For channel 4.

Example for detecting M04 in channel 1. If it has been programmed, it will return a "1", and a "0" if otherwise.

```
CPS MFUNC1* EQ 4 = ...
```

5.2 Auxiliary –H– functions.

Up to 7 M and 7 H functions may be programmed in a block. The treatment of the auxiliary H functions is similar to the M functions without synchronization.

The CNC indicates to the PLC which auxiliary H functions are programmed in the execution block using 32-bit registers HFUN1 through HFUN7. Each one of them indicates the number of one of the H functions programmed in the block. If all the registers are not used, the CNC assigns $FFFFFFFF to the unused ones (those with the highest numbers).

This way, if a block contains functions H12, H20 and H35, the CNC will transfer the following information to the PLC.

HFUN1	HFUN2	HFUN3	HFUN4 - HFUN7
12	20	35	$FFFFFFFF

Command HFUN*. Checking if a function has been programmed in the block.

To know whether a particular H function is programmed in the execution block, use one of the following methods:

- Check all the HFUN registers one by one until that particular H function is found or until one of them has the value of $FFFFFFFF.
- Use the HFUN* format to check all the HFUN registers at the same time.

Example to detect H77: If it has been programmed, it will return a "1", and a "0" if otherwise.

```
CPS HFUN* EQ 77 = …
```

Sending and synchronizing the function.

The H functions are not synchronized and are sent out to the PLC at the beginning of block execution.

The transfer of auxiliary H functions is described later on in this chapter. See *"5.4 Transferring auxiliary functions -M-, -H-, -S-."* on page 282.

When executing a block like this:

```
X100 F1000 H11 H12
```

The functions are transferred as follows:

1. Functions H11 and H12 are sent out to the PLC
2. It does not wait for confirmation and the CNC moves the axis to X100.

5.2.1 Special considerations with the multi-spindle option and channels.

The CNC may have up to 4 channels and each channel can execute a part-program in parallel with the rest. This means that each channel can execute seven auxiliary functions simultaneously. The auxiliary functions executed from each channel are treated independently; to do that, each channel has its own marks and registers.

Marks and registers in the channels option.

Each channel has 32-bit registers HFUN1 to HFUN7 to indicate to the PLC which auxiliary H functions are programmed in the execution block.

HFUN1C1 - HFUN7C1	for the first channel.
HFUN1C2 - HFUN7C2	for the second channel.
HFUN1C3 - HFUN7C3	for the third channel.
HFUN1C4 - HFUN7C4	for the fourth channel.

Each one of them indicates the number of one of the H functions programmed in the block. If all the registers are not used, the CNC assigns $FFFFFFFF to the unused ones (those with the highest numbers).

This way, if functions H10 and H13 are programmed in the first channel and functions H8 and H10 in the second channel, the CNC will transfer the following data.

HFUN1C1	HFUN2C1	HFUN3C1 - HFUN7C1
10	13	$FFFFFFFF

HFUN1C2	HFUN2C2	HFUN3C2 - HFUN7C2
8	10	$FFFFFFFF

Commands HFUNC1* - HFUNC4*. Checking if a function has been programmed in the channel.

To know whether a particular H function is programmed in the execution block, use one of the following methods:

- Check all the HFUN registers one by one until that particular H function is found or until one of them has the value of $FFFFFFFF.
- Use one of the following commands to check all the HFUN registers of the channel at the same time.

HFUNC1*	For channel 1
HFUNC2*	For channel 2.
HFUNC3*	For channel 3.
HFUNC4*	For channel 4.

5.3 Auxiliary –S– function.

The auxiliary S function indicates the spindle turning speed with M03 and M04 or the angular position with M19.

The S function with M03 and M04 is always executed at the beginning of the block and the CNC waits for confirmation before going on executing the program. When working with M19, the CNC treats the spindle like a regular linear axis. It only sends the M19 out to the PLC.

The CNC uses 32-bit register SFUN1 to indicate to the PLC the value of the S function programmed in the block. If not programmed, the value of $FFFFFFFF is sent out. The SFUN command only assumes the value of the programmed S if spindle parameter SPDLTIME is other than zero.

The transfer of the S function is described later on in this chapter. See *"5.4 Transferring auxiliary functions -M-, -H-, -S-."* on page 282.

5.3.1 Special considerations with the multi-spindle option and channels.

The CNC may have up to four spindles. All of them may be controlled independently in the same block; in other words, each spindle may be given a different command.

When using channels, the spindles may be distributed indistinctly between them. In this case, a channel can control a spindle of another channel. The marks and registers refer to the spindle regardless of the channel they belong to.

The spindle number is determined by its logic number that is set by the order they were defined in the machine parameter SPDLNAME.

Marks and registers in the multi-spindle version.

The CNC indicates to the PLC which S functions are programmed in the execution block using 32-bit registers SFUN1 through SFUN4. These registers refer to the spindle number; they are independent from the channel where the spindle is.

Each one of them indicates the value of one of the S functions programmed. If all the registers are not used, the CNC assigns $FFFFFFFF to the unused ones (those with the highest numbers).

This way, if a block contains functions S1000 and S1=550, the CNC will transfer the following information to the PLC:

SFUN1	SFUN2	SFUN3	SFUN4
1000	550	$FFFFFFFF	$FFFFFFFF

Commands SP1FUN* - SP4FUN*. Check if an auxiliary function has been programmed for a spindle.

Considering the possible channels/spindles combinations, these functions are available to make it easier to manage the auxiliary M functions associated with each spindle. Each one indicates if any M3, M4, etc. type M function has been programmed in any channel.

SP1FUN*	For the spindle 1.
SP2FUN*	For the spindle 2.
SP3FUN*	For the spindle 3.
SP4FUN*	For the spindle 4.

Checks if the M5 function has been sent to spindle 1 from a channel.

```
CPS SP1FUN* EQ 5 = ...
```

5.4 Transferring auxiliary functions -M-, -H-, -S-.

The M and H functions are transferred per channel. Transferring S functions does not depend on the channel.

When executing a block that contains M, H, S functions, the following information is transferred to the PLC.

Transferring -M- functions.

The CNC assigns the numbers of the M functions programmed in the block to registers MFUN1 through MFUN7. Some M functions have an associated function (DMxx) that is activated when sending the M to the PLC.

M00	M01	M02	M03	M04
M05	M06	M08	M09	M19
M30	M41	M42	M43	M44

The CNC activates the general logic output MSTROBE to "tell" the PLC that it must execute them. This mark is kept high (=1) for a time period indicated by parameter MINAENDW.

Depending on the type of synchronization, the CNC will either wait or not for the general input AUXEND to be activated indicating the end of the PLC execution. The type of synchronization is defined in the machine parameters

The CNC cancels the general logic output MSTROBE to conclude the execution.

Transferring -H- functions.

The CNC assigns the numbers of the H functions programmed in the block to registers HFUN1 through HFUN7.

The CNC activates the general logic output HSTROBE to "tell" the PLC that it must execute them. This mark is kept high (=1) for a time period indicated by parameter MINAENDW.

After this time period, the CNC considers its execution completed because there is no synchronization.

When sending several blocks in a row just having H functions, the CNC waits twice the time indicated in parameter MINAENDW.

```
N10 H60
N20 H30 H18
N30 H40
```

Transferring -S- functions

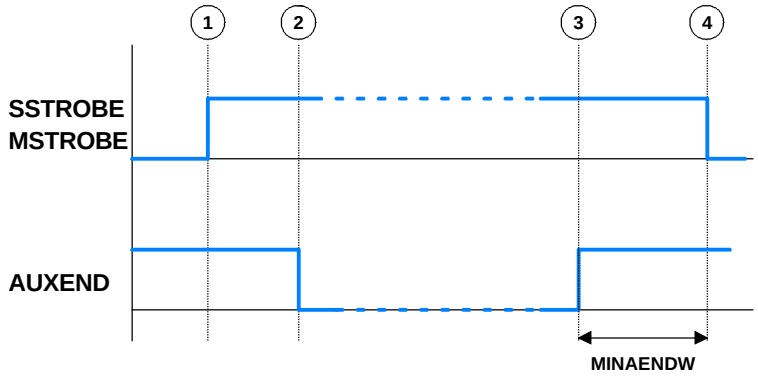
The CNC assigns the values of the S programmed in each spindle to registers SFUN1 through SFUN4.

The CNC activates the general logic output SSTROBE to "tell" the PLC that it must execute it. The CNC waits for the general input AUXEND to be activated indicating the end of the PLC execution.

The CNC cancels the general logic output SSTROBE to conclude the execution.

5.4.1 Synchronized transfer.

This type of transfer takes place with the S function and with the M functions set with synchronization.



When the PLC is requested to execute several M or S functions at the same time, the corresponding SSTROBE or MSTROBE signals are activated; but the CNC waits for a single "AUXEND" signal to end all of them.

Transferring -M- functions

1 The CNC indicates in registers MFUN1 to MFUN7 of the channel the M functions programmed in the block and it activates the MSTROBE mark so the PLC executes them.

2 The PLC must deactivate the AUXEND mark to let the CNC know that the execution has begun.

3 Once the required auxiliary functions have been executed, the PLC must activate the AUXEND mark to let the CNC know that the execution has ended.

The AUXEND mark must be kept high (=1) longer than the time period established by parameter MINAENDW.

4 After this time, the CNC deactivates the MSTROBE mark thus ending the execution of the functions.

Transferring -S- functions

1 The CNC indicates in register SFUN1 to SFUN4 the S value programmed in the block and activates the SSTROBE mark so the PLC executes it.

2 The PLC must deactivate the AUXEND mark to let the CNC know that the execution has begun.

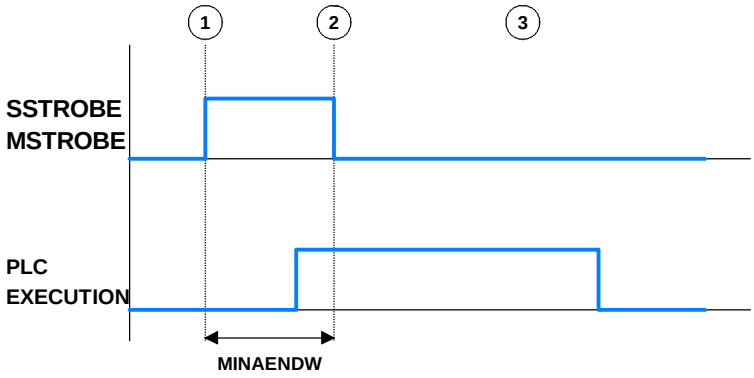
3 After selecting the requested S, the PLC must activate the AUXEND mark to let the CNC know that the execution has ended.

The AUXEND mark must be kept high (=1) longer than the time period established by parameter MINAENDW.

4 After this time, the CNC deactivates the MSTROBE mark thus ending the execution of the function.

5.4.2 Non-synchronized transfer.

This type of transfer takes place with the H function and with the M functions set without synchronization.



Transferring -M- functions

1 The CNC indicates in registers MFUN1 to MFUN7 of the channel the M functions programmed in the block and it activates the MSTROBE mark so the PLC executes them.

2 The CNC keeps the MSTROBE mark active for a time period indicated by parameter MINAENDW.

3 After this time, the CNC goes on executing the program regardless of the time required by the PLC to execute that function.

Transferring -H- functions

1 The CNC indicates in registers HFUN1 to HFUN7 of the channel the H functions programmed in the block and it activates the HSTROBE mark so the PLC executes them.

2 The CNC keeps the HSTROBE mark active for a time period indicated by parameter MINAENDW.

3 After this time, the CNC goes on executing the program regardless of the time required by the PLC to execute that function.

Considerations for transferring these functions.

The value of parameter MINAENDW should be the same or longer than the PLC program execution period parameter PRGFREQ in order to ensure that the PLC detects that signal.

When sending non-synchronized H or M functions corresponding to consecutive blocks of the same program, the CNC waits between blocks for a time period indicated by MINANEDW so the PLC can read all the functions.

5.5 Displaying PLC errors and messages.

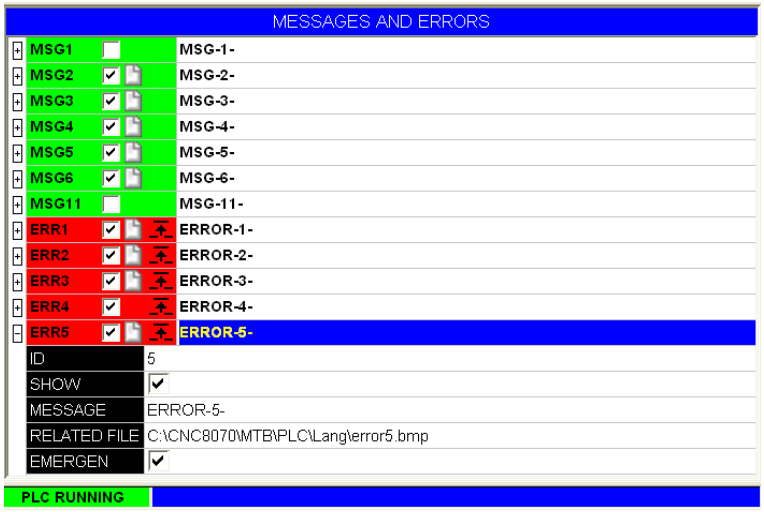
The PLC has 1024 marks for displaying messages and another 1024 marks for displaying errors at the CNC. When the mark is high (=1) the message or the error is active.

MSG1 - MSG1024 for displaying messages.

ERR1 - ERR1024 for displaying errors.

There is a message and error table where each message or error may have associated the following:

- A number (ID field).
- A text ("Message" field).
- An additional information file ("associated file " field) that may be a "bmp, txt, jpg, gif, htm, html or avi" type file.
- Whether when activating an error or message the additional help file must be displayed ("Show" field selected) or only the text of the message or of the error.
- For the errors, whether they open the emergency relay of the central unit or not.



For more information on how to edit this table, refer to the operating manual.

PLC messages.

When activating one of the MSG marks, the status bar of the CNC window for PLC messages shows the message number and its associated text. If the message has a file with additional information associated with it, an access icon will appear to the left of the message.



The message does not have a file with additional information.



The message has a file with additional information.

When there are more than one message activated, it always shows the one with the highest priority (the one with the lowest number). The PLC-messages window shows the "+" sign meaning that there are more messages activated by the PLC. To display the whole list, press [CTRL] + [M].

If the "Show" field of the message is selected, the CNC screen shows the additional information file and if there isn't one, a blue window with the text of the message. If the "Show" field is not selected, to show the additional information file, you must expand the list of messages, select a message and press [ENTER] or click on the message. To close the additional data window, press [ESC].

PLC errors.

When activating one of the ERR marks, the CNC interrupts the execution of the part-program and it displays a window in the middle of the screen showing the error number and its associated text. If the error has the "Emergen" field selected, the error will open the emergency relay of the CNC.

If the error has a file with additional information associated with it, an access icon will appear to the right of the error number. If the error has the "Show" field selected, the CNC shows the additional information file directly on the screen. If the "Show" field is not selected, the additional information file will be displayed when pressing the [HELP] key or when clicking on the icon mentioned earlier. To close the additional data window, press [ESC].

External inputs should be used to activate and deactivate error marks, thus preventing the CNC from receiving those errors at every new PLC cycle scan.

Grouping the additional information text files in a single file.

PLC messages and errors can show an additional information file in text format. The PLC allows grouping several or all these files into a single file as follows.

Defining the additional information file.

It must be a text file (extension txt) and may have any name. The information of each message and error must be structured in the following format:

```
[<id>]
<text>
```

The <id> field, keeping the brackets, will be the code that identifies the help text inside the file, which needs not be the same as the number of the error or message it will be associated with. The <text> field will be the information text that may have up to 500 characters including line feeds.

For example, the OEM.txt file will have the following structure.

```
[10]
Help text.
[27]
Help text.
[33]
Help text.
```

Calling the texts from the PLC message or from the PLC error.

To associate the help message with a PLC message or PLC error, the "associated file" field must be defined like <file>#<id>. The <file> field will be the path and the name of the file. The <id> field will be the code that identifies the help text inside the file.

For example, the "associated file" will be defined as follows.

```
C:\CNC8070\MTB\PLC\LANG\OEM.txt#27
```

LOGIC CNC INPUTS AND OUTPUTS.

6

CNC's physical inputs and outputs are the set of system inputs and outputs that are governed by the PLC and communicate with the outside world through CNC connectors.

The CNC also has a number of logic inputs and outputs to exchange the internal data with the marks and registers of the PLC. This way, the PLC has access to some internal CNC data. Each of these logic inputs and outputs may be referred to using its associated mnemonic. The mnemonics that begin with a "_" sign indicate that the signal is active low (0 V).

CNCREADY	_ALARM
AUXEND	_EMERGEN
MANUAL	_STOP

The CNC's logic outputs or PLC consulting signals are grouped into:

- General consulting signals.
- Axis consulting signals.
- Spindle consulting signals.
- Consultation signals of the independent interpolator.
- Tool manager consulting signals.
- Keystroke consulting signals.

The CNC's logic inputs or signals that may be modified via PLC are grouped into:

- General modifiable signals.
- Axis modifiable signals.
- Spindle modifiable signals.
- Modifiable signals of the independent interpolator.
- Tool manager modifiable signals.
- Keystroke modifiable signals.

Abbreviations used in this chapter.

(=0)	Low logic level.
(=1)	High logic level.
(g.m.p.)	General machine parameter.
(a.m.p.)	Machine parameters for axes and spindles.

6.1 General consulting signals.

CNCREADY

This mark indicates the CNC status. It is set to "0" when the CNC is in an error state (red status window) and to "1" if otherwise.

Include this mark in the PLC maneuver to enable the drives.

```
; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++
; AXIS ENABLE
; ++++++++++++++++++++++++++++++++++++++++++++++++++++++++
CNCREADY \ ; CNC without errors
AND NOT B1R2 \ ; FEEDHOLD due to lack of power at the X axis drive
AND NOT PARKEDX \ ; X axis parked
AND NOT PARKEDX \ ; Unpark X axis
AND NOT PARKEDX \ ; Park X axis
= SERVOXON ; X axis servo drive active
= SPENAX ; X axis sercos enable
= TG3 2 500 ; Delay for DRENAX
;
T2\; Delay to the drop of speed enable
= DRENAX \; Drive enable via Sercos.
```

READY

There is a mark for each channel. The mnemonics for each channel are the following.

READYC1 READYC2 READYC3 READYC4

This mark indicates the status of the CNC channel. It is set to "0" when the CNC is in an error state (red status window) and to "1" if otherwise.

SERCOSRDY

In Sercos configuration, this mark indicates the ring status. The CNC sets this mark high (=1) to indicate that the Sercos ring has been properly initialized.

All the drive status consultations (DRSTAF DRSTAS) and the read/write (CNCRD CNCWR) operations of the drive variables are conditioned by this mark.

START

There is a mark for each channel. The mnemonics for each channel are the following.

STARTC1 (can also be programmed as START)

STARTC2 STARTC3 STARTC4

The CNC channel activates this mark to indicate to the PLC that the [START] key has been pressed at the operator panel. If the rest of the conditions are met (hydraulic, safety, etc.), the PLC must activate the CYSTART mark in order for the program to start running. See *"CYSTART"* on page 315.

```
START AND (rest of conditions) = CYSTART
```

RESETOUT

There is a mark for each channel. The mnemonics for each channel are the following.

RESETOUTC1 (can also be programmed as RESETOUT)

RESETOUTC2 RESETOUTC3 RESETOUTC4

When pressing the [RESET] key or when the PLC activates the "RESETIN" mark, the CNC channel assumes the initial conditions and sets the "RESETOUT" mark high (=1). This mark stays high (=1) for a time period set by (g.m.p.) MINAENDW. See *"RESETIN"* on page 316.

FHOUT

There is a mark for each channel. The mnemonics for each channel are the following.

FHOUTC1 (can also be programmed as FHOUT)

FHOUTC2 FHOUTC3 FHOUTC4

The CNC channel sets this mark high (=1) when the execution of the part-program is interrupted. To interrupt and later resume the execution of a program, proceed as follows:

- If the PLC has set the mark "_STOP" to (=0), to resume the execution, the PLC must set marks "_STOP" and "CYSTART" to (=1).
- If the PLC has set the mark "_FEEDHOL" to (=0), to resume the execution, the PLC must set the "_FEEDHOL" mark to (=1).

_ALARM

There is a mark for each channel. The mnemonics for each channel are the following.

_ALARMC1 (can also be programmed as _ALARM)

_ALARMC2 _ALARMC3 _ALARMC4

The CNC channel activates this mark (at "0") when there is an alarm, an emergency generated by the CNC channel or when the PLC activates the emergency signal _EMERGEN.

The CNC channel cancels this mark (at "1") when the CNC channel message is removed and the cause of the alarm or emergency has been eliminated.

There is no output associated with this mark. The following example shows how to associate output O1.

```
_ALARM AND (rest of conditions) = O1
    If there are no errors, output O1 will be high (=1)
```

MANUAL

There is a mark for each channel. The mnemonics for each channel are the following.

MANUALC1 (can also be programmed as MANUAL)

MANUALC2 MANUALC3 MANUALC4

The CNC sets this mark high (=1) when the Manual Operation (JOG) mode is selected.

AUTOMAT

There is a mark for each channel. The mnemonics for each channel are the following.

AUTOMATC1 (can also be programmed as AUTOMAT)

AUTOMATC2 AUTOMATC3 AUTOMATC4

The CNC channel sets this mark high (=1) when the automatic operation mode is selected.

MDI

There is a mark for each channel. The mnemonics for each channel are the following.

MDIC1 (can also be programmed as MDI)

MDIC2 MDIC3 MDIC4

The CNC sets this mark high (=1) when the MDI (Manual Data Input) mode is selected.

- If the MDI mode is selected while in automatic mode, the AUTOMAT and MDI marks will be activated.
- If the MDI mode is selected in JOG mode, the MANUAL and MDI marks are activated.

SBOUT

There is a mark for each channel. The mnemonics for each channel are the following.

SBOUTC1 (can also be programmed as SBOUT)

SBOUTC2 SBOUTC3 SBOUTC4

The CNC sets this mark high (=1) when the Single Block mode is selected.

- If the –single block– mode is selected while in automatic mode, the AUTOMAT and SBOUT marks are activated.
- If the –single block– mode is selected while in JOG mode, the MANUAL and SBOUT marks.

INCYCLE

There is a mark for each channel. The mnemonics for each channel are the following.

INCYCLEC1 (can also be programmed as INCYCLE)

INCYCLEC2 INCYCLEC3 INCYCLEC4

The CNC sets this mark high (=1) when executing a block or moving an axis.

- In execution. It is set high (=1) at the beginning of the execution and it stays high until the end, until the [STOP] key is pressed or until the "_STOP" mark is set low (=0).
- In MDI or Single Block execution, it is set low (=0) at the end of the block.
- In JOG mode, it stays high (=1) while any of the JOG keys is pressed.

RAPID

There is a mark for each channel. The mnemonics for each channel are the following.

RAPIDC1 (can also be programmed as RAPID)

RAPIDC2 RAPIDC3 RAPIDC4

The CNC channel sets this mark high (=1) when executing a rapid traverse movement (G0).

ZERO

There is a mark for each channel. The mnemonics for each channel are the following.

ZEROC1 (can also be programmed as ZERO)

ZEROC2 ZEROC3 ZEROC4

The CNC sets this mark high (=1) when searching home (G74).

PROBE

There is a mark for each channel. The mnemonics for each channel are the following.

PROBEC1 (can also be programmed as PROBE)

PROBEC2 PROBEC3 PROBEC4

The CNC channel sets this mark high (=1) when executing a probing movement (G100).

THREAD

There is a mark for each channel. The mnemonics for each channel are the following.

THREADC1 (can also be programmed as THREAD)

THREADC2 THREADC3 THREADC4

The CNC sets this mark high (=1) when executing an electronic threading block (G33).

TAPPING

There is a mark for each channel. The mnemonics for each channel are the following.

TAPPINGC1 (can also be programmed as TAPPING)

TAPPINGC2 TAPPINGC3 TAPPINGC4

The CNC sets this mark high (=1) when executing a tapping canned cycle. The mark stays active during the possible dwell programmed at the bottom of the thread.

In milling, cycle G84, the mark stays active throughout the whole cycle (including the blocks with movement to the starting point, etc.). In lathe, cycle G83, the mark stays active only during the actual threading.

RIGID

There is a mark for each channel. The mnemonics for each channel are the following.

RIGIDC1 (can also be programmed as RIGID)

RIGIDC2 RIGIDC3 RIGIDC4

The CNC sets this mark high (=1) when executing an rigid tapping block (G63). Tapping (G84 in milling and G83 in lathe) set as rigid tapping set this mark to (=1) during the whole tapping operation, including the dwell at the bottom of the thread.

CSS

There is a mark for each channel. The mnemonics for each channel are the following.

CSSC1 (can also be programmed as CSS)

CSSC2 CSSC3 CSSC4

The CNC sets this mark high (=1) when Constant Surface Speed is selected (G96).

INTEREND

There is a mark for each channel. The mnemonics for each channel are the following.

INTERENDC1 (can also be programmed as INTEREND)

INTERENDC2 INTERENDC3 INTERENDC4

The CNC sets the INTEREND mark high (=1) when the theoretical movement of the axes (when it no longer outputs the velocity command).

The INTEREND mark may be used to activate external devices before the axes reach their position. The mark ADVINPOS can also be used. See *"ADVINPOS"* on page 294.

INPOSI

There is a mark for each channel. The mnemonics for each channel are the following.

INPOSIC1 (can also be programmed as INPOSI or INPOS)

INPOSIC2 INPOSIC3 INPOSIC4

The CNC channel sets the INPOSI mark to (=1) to indicate that all its active axes and spindles have reached position, except the independent axes programmed from the PLC. The INPOSI mark stays at (=1) while the independent axes are moving.

An axis is in position when it stays within the in-position zone (window) (parameter INPOSW) for a time period indicated by parameter INPOSTIME.

SPN1···SPN7

There is a register for each channel. The mnemonics for each channel are the following. Here is an example of the mnemonics for SPN1; it is the same for the rest of the registers.

SPN1C1 SPN1C2 SPN1C3 SPN1C4

The channel uses these registers to indicate to the PLC which spindle of the channel each auxiliary M function selected for execution is addressed to.

Each channel can have up to 7 M functions in a block. If all the registers are not used, the hexadecimal value $FFFFFFFF is assigned to the unused ones (those with the highest numbers).

This way, if the next block is programmed in the first channel, the CNC will pass on to the PLC the following information.

```
M3.S1 S1=1000 M4.S2 S2=500
```

Clockwise rotation of spindle S1 at 1000 rpm and counterclockwise rotation of spindle S2 at 500 rpm.

MFUN1C1	MFUN2C1	MFUN3C1 - MFUN7C1
3	4	$FFFFFFFF

SPN1C1	SPN2C1	SPN3C1 - SPN7C1
1	2	$FFFFFFFF

If a function is programmed in the block without mentioning the spindle, it will assume the master spindle of the channel.

Commands SP1FUN* - SP4FUN*. Checking if a spindle receives a function from any channel.

In order to know whether a particular spindle has received a particular function or not, it is possible to check all the registers one by one or use the following commands to check all of them at the same time.

SP1FUN* For spindle 1.
SP2FUN* For spindle 2.
SP3FUN* For spindle 3.
SP4FUN* For spindle 4.

```
CPS SP1FUN* EQ 5 = ...
```
Example to check if the first spindle has received an M5 function from any channel. If it has been programmed, it will return a "1", and a "0" if otherwise.

MFUN1···MFUN7
HFUN1···HFUN7

There is a register for each channel. The mnemonics for each channel are the following. Here is an example of the mnemonics for MFUN1 and HFUN1; it is the same for the rest of the registers.

MFUN1C1 (can also be programmed as MFUN1)
MFUN1C2 MFUN1C3 MFUN1C4
HFUN1C1 (can also be programmed as HFUN1)
HFUN1C2 HFUN1C3 HFUN1C4

The channel uses these registers to indicate to the PLC the M or H auxiliary functions selected for execution.

Each channel can have up to 7 M and 7 H functions in a block. If all the registers are not used, the hexadecimal value $FFFFFFFF is assigned to the unused ones (those with the highest numbers).

This way, if functions M100 and M135 are programmed in the first channel and functions M88 and M75 in the second channel, the CNC will transfer the following data.

MFUN1C1	MFUN2C1	MFUN3C1 - MFUN7C1
100	135	$FFFFFFFF

MFUN1C2	MFUN2C2	MFUN3C2 - MFUN7C2
88	75	$FFFFFFFF

If, then, the M88 function is executed in the first channel, then:

MFUN1C1	MFUN2C1	MFUN3C1 - MFUN7C1
88	$FFFFFFFF	$FFFFFFFF

Commands MFUNC1* - MFUNC4* and HFUNC1* - HFUNC4*. Checking if a function has been programmed in the channel.

In order to know whether a particular function is programmed in the block currently being executed, all the registers may be checked one by one or the following commands may be used to check them all at the same time.

MFUNC1*/HFUNC1*	For channel 1. They can also be programmed as MFUN*/HFUN*.
MFUNC2*/HFUNC2*	For channel 2.
MFUNC3*/HFUNC3*	For channel 3.
MFUNC4*/HFUNC4*	For channel 4.

```
CPS MFUNC1* EQ 4 = ...
```

Example for detecting M04 in channel 1. If it has been programmed, it will return a "1", and a "0" if otherwise.

MSTROBE

There is a mark for each channel. The mnemonics for each channel are the following.

MSTROBEC1 (can also be programmed as MSTROBE)

MSTROBEC2 MSTROBEC3 MSTROBEC4

The CNC channel sets this mark high (=1) to indicate to the PLC that it must execute the auxiliary M functions indicated in registers MFUN1 through MFUN7 of the channel.

HSTROBE

There is a mark for each channel. The mnemonics for each channel are the following.

HSTROBEC1 (can also be programmed as HSTROBE)

HSTROBEC2 HSTROBEC3 HSTROBEC4

The CNC channel sets this mark high (=1) to indicate to the PLC that it must execute the auxiliary H functions indicated in registers HFUN1 through HFUN7.

SFUN1···SFUN4

There is a register for each spindle. The mnemonics for each spindle are the following:

SFUN1 SFUN2 SFUN3 SFUN4

These registers indicate the programmed speed for each spindle. These registers refer to the spindle number; they are independent from the channel where the spindle is. The CNC only uses the registers of the spindles whose SPDLTIME parameter has a value other than zero.

Each one of them indicates the value of one of the S functions programmed. If all the registers are not used, the CNC assigns $FFFFFFFF to the unused ones (those with the highest numbers).

This way, if a block contains functions S1000 and S1=550 and the value of SPDLTIME parameter of both spindles is other than zero, the CNC will transfer the following information to the PLC.

SFUN1	SFUN2	SFUN3	SFUN4
1000	550	$FFFFFFFF	$FFFFFFFF

SSTROBE

There is a mark for each spindle. The mnemonics for each spindle are the following:

SSTROBE1 (can also be programmed as SSTROBE)

SSTROBE2 SSTROBE3 SSTROBE4

The CNC channel sets this mark high (=1) to indicate to the PLC that a new spindle speed has been selected. The CNC channel only uses this mark on spindles whose SPDLTIME parameter has a value other than zero.

DMxx

This mark is associated with some M auxiliary functions.

The marks associated with functions M00, M01, M02, M06, M08, M09, M30 have a mark for each channel. Here is an example of the mnemonics for DM00; it is the same for the rest of the marks (DM01, DM02, DM06, DM08, DM09, DM30).

DM00C1 (can also be programmed as DM00)

DM00C2 DM00C3 DM00C4

The marks associated with functions M03, M04, M05, M19, M41, M42, M43, M44 have a mark for each spindle. Here is an example of the mnemonics for DM03; it is the same for the rest of the marks (DM04, DM05, DM19, DM41, DM42, DM43, DM44).

DM03SP1 (can also be programmed as DM03)

DM03SP2 DM03SP3 DM03SP4

The CNC indicates in these marks the status of the spindle auxiliary M functions. The mark is set to (=1) if the function is active and to (=0) if otherwise.

BLKSEARCH

There is a mark for each channel. The mnemonics for each channel are the following.

BLKSEARCHC1 (can also be programmed as BLKSEARCH)

BLKSEARCHC2 BLKSEARCHC3 BLKSEARCHC4

The CNC channel sets this mark high (=1) to indicate that the "Block Search" option is active in the Automatic Operation mode.

ADVINPOS

There is a mark for each channel. The mnemonics for each channel are the following.

ADVINPOSC1 (can also be programmed as ADVINPOS)

ADVINPOSC2 ADVINPOSC3 ADVINPOSC4

The CNC channel sets this signal high for some time before the axes reach position. This time is set by (g.m.p.) ANTIME.

If the total duration of the movement is lower than the value set by (g.m.p.) ANTIME, the mark goes high (=1) immediately.

If (g.m.p.) ANTIME has been set to 0, the mark is always active.

It is used on punch presses that have an eccentric cam as a punching system. This signal may be used to start the movement of the punch before the axes reach the position. This reduces idle time, thus increasing the number of punches per minute.

CAXIS

There is a mark for each channel. The mnemonics for each channel are the following.

CAXISC1 (can also be programmed as CAXIS)

CAXISC2 CAXISC3 CAXISC4

The CNC channel sets this mark to (=1) when the spindle is working as C axis. This mark is kept active while any of the functions #CAX, #FACE or #CYL are kept active.

FREE

There is a mark for each channel. The mnemonics for each channel are the following.

FREEC1 FREEC2 FREEC3 FREEC4

The CNC channel sets this signal high (=1) to indicate to the PLC that it is ready to accept a new block, sent using the CNCEX command.

WAITOUT

There is a mark for each channel. The mnemonics for each channel are the following.

WAITOUTC1 WAITOUTC2 WAITOUTC3 WAITOUTC4

It is applied to channel synchronization. The CNC channel sets this signal high (=1) to indicate to the PLC that it is waiting for a synchronization signal. Synchronization signals may be executed from the part-program using the #WAIT or #MEET instructions.

MMCWDG

This mark indicates the status of the operating system. Its value is (=0) when the CNC operating system is working properly and (=1) when the operating system is locked up.

Include this mark in the PLC maneuver to enable the emergencies when the operating system locks up.

RETRAEND

There is a mark for each channel. The mnemonics for each channel are the following.

RETRAENDC1 RETRAENDC2 RETRAENDC3 RETRAENDC4

The CNC channel sets this mark high (=1) to cancel the retrace function. The CNC channel cancels the retrace function in the following cases.

- The CNC channel has executed all the blocks stored for the retrace function.
- The CNC channel has reached the beginning of the program.
- The CNC channel has reached a block that cannot be executed with the retrace function.

In all these cases, the CNC channel sets its mark RETRAEND high (=1) to indicate to the PLC that all possible blocks have been executed and that it must set its RETRACE mark to "0". See *"RETRACE"* on page 319.

If the PLC automatically removes the RETRACE mark with the RETRAEND mark, the CNC goes on with the normal execution of the program, forward. Otherwise, if the RETRACE mark stays active, the CNC will issue a warning indicating that this mark must be deactivated in order to resume the execution of the program. The same occurs when trying to execute a new program after executing the retrace function.

The retrace function is also canceled after M30, with a reset or when the PLC sets the channel mark RETRACE low (=0)

TANGACTIV

There is a mark for each channel. The mnemonics for each channel are the following.

TANGACTIVC1 TANGACTIVC2 TANGACTIVC3 TANGACTIVC4

The CNC sets this mark high (=1) to let the PLC know that a tangential control is active in the channel. This mark is not initialized when tangential control is frozen (suspended).

PSWSET

The CNC activates this mark when there is an OEM password.

DINDISTC1
DINDISTC2
DINDISTC3
DINDISTC4

The CNC has three marks per channel. The mnemonics for each channel are the following.

DINDISTC1C2	DINDISTC1C3	DINDISTC1C4
DINDISTC2C1	DINDISTC2C3	DINDISTC2C4
DINDISTC3C1	DINDISTC3C2	DINDISTC3C4
DINDISTC4C1	DINDISTC4C2	DINDISTC4C3

These marks are related to dynamic distribution of machining operations between channels (instruction #DINDIST). The option of synchronized identical passes uses these marks; it also uses the marks associated with synchronizing independent axes.

During the roughing operation of the cycle, the CNC channel activates these marks to indicate which channel has the cycle been programmed in and which are the channels involved in the distribution of the passes. The channel of the CNC cancels the marks after finishing the withdrawal from the last roughing pass in the last channel. During the finishing operation, the CNC channel cancels all these marks.

The first channel indicated in the mnemonic refers to the channel that executes the cycle; the second one refers to the channel involved in the distribution of passes. For example:

- If the cycle is executed in channel ·1· and the rest of the passes are distributed with channel ·2·, the CNC activates the DINDISTC1C2 mark.
- If the cycle is executed in channel ·2· and the rest of the passes are distributed with channels ·1· and ·3·, the CNC activates marks DINDISTC2C1 and DINDISTC2C3.

Use these marks so interrupting the program, resuming it and executing it block by block in one of the channels affects all the channels involved in the distribution of passes.

SERPLCAC

This mark is related to the change of the work set or of the parameter set of a Sercos drive (variable (V.)[ch].A.SETGE.xn). The CNC activates the mark to indicate that the requested change is being carried out.

Only one change can be taking place at a time. If other gear or set changes are programmed during this process, even if they are at different drives, the CNC only keeps the one programmed last and ignores the rest of the intermediate changes.

OVERTEMP

This mark indicates the temperature status of the CNC. This mark is deactivated while room temperature is correct. The CNC checks every minute the unit temperature; if in three samples in a row the temperature exceeds 60 ºC (140 ºF), the CNC activates this mark and issues the warning about this situation. The CNC will deactivate this mark when the temperature of the unit goes below the maximum allowed (60 ºC / 140 ºF).

Every time [START] is pressed, the CNC checks that the room temperature does not exceed 65 ºC (149 ºF) and, if it does, inhibits the [START] and issues error E173. The CNC will let complete the execution in progress, but it will not allow to resume it after an interruption, a tool inspection, etc.

Include this mark in the PLC routine to restrict the CNC operation in case of over-temperature.

If the PLC activates this mark, there is an over-temperature risk and the integrity of the unit is at risk. Turn the unit off to avoid possible damage.

MLINKRDY

In Mechatrolink configuration, this mark indicates the bus status. The CNC activates this mark to indicate that the Mechatrolink bus has been properly initialized.

All status consultations of the servos and inverters and the read/write (CNCRD/CNCWR) operations of their variables must be conditioned by this mark. The status of the servos and inverters may be consulted using the following variables.

V.A.MSTATUS.xn V.A.MSUBSTAT.xn V.A.MALARM.xn V.A.MIOMON.xn

6.2 Consulting signals for axes and spindles.

When the spindle is working in closed loop (M19 or G63), it behaves like an axis.

The signal names are generic. Replace the word (axis) with the spindle name or with the name or logic number of the axis.

For example, the name of the ENABLE(axis) mark for a machine with the X, Y, Z, Z2, B axes and spindle S.

ENABLEX, ENABLEZ2, ENABLEB, ENABLES

ENABLE3 for the Z axis.

ENABLE5 for the B axis.

ENABLE(axis)

The CNC sets this mark to (=1) to indicate to the PLC that it is going to move the corresponding axis or spindle in closed loop, so the PLC enables the axis or spindle if necessary. This mark stays active until the axis or spindle gets in position, even if the theoretical movement has not finished; i.e. until the CNC activates the INPOS(axis) mark.

This mark is also enabled in the movements of the independent axis, in spindle movements in closed loop (for example M19 or G74) or when a spindle goes into closed loop using the instruction #SERVO ON.

When the movement of an axis implies the movement of a slave axis, the ENABLE(axis) mark of the slave axis will also be enabled. For example, on gantry axes, slaved with #LINK, synchronized with #FOLLOW, electronic cams or spindles synchronized in closed loop with #SYNC even if the slave axis is in another channel.

DIR(axis)

The CNC sets this mark to (=1) to indicate that the axis is moving in the negative direction and to (=0) when moving in the positive direction. When the axis is stopped, it keeps the last value.

If the PLC stops the movement of the axes (mark _FEEDHOL=0) the ENABLE(axis) signal stays at (=1).

REFPOIN(axis)

This mark is set low (=0) on CNC power-up and is set high (=1) after searching home or after setting the machine coordinate (G174). The mark stays at (=1) until turning the CNC off.

This mark is set to (=0) when a feedback alarm occurs on the analog axes.

On axes and spindles without absolute feedback, the mark is set to (=0) in the following cases.

- When a home search fails.
- When parking the axis or spindle.
- On Sercos axes and spindles, if the ring fails.
- On spindles or rotary axes controlled as a spindle, when switching to open loop.
- On analog axes, when a feedback alarm occurs.

DRSTAF(axis)
DRSTAS(axis)

The CNC uses these marks when communicating with the drive via Sercos® and they indicate the drive's status.

In case of error, both marks stay at (=1) if the axis is moving.

Normal.

1 After actuating the main switch at the electrical cabinet, the drive is supplied with 24 Vdc.

2 The drive runs an internal test. If OK, it activates the SYSTEM OK output. From that instant on, apply power to the Power Supply.

Marks DRSTAF(*)=0 DRSTAS(*)=1

3 When there is power at the bus, the drive is ready to provide torque. To do that, activate the Drive enable and Speed enable inputs.

Marks DRSTAF(*)=1 DRSTAS(*)=0

4 Once the Drive enable and Speed enable inputs are activated, the drive is running properly.

Marks DRSTAF(*)=1 DRSTAS(*)=1

Mark consultation.

When consulting the DRSTAF(*) and DRSTAS(*) marks, the following values may result:

DRSTAF(*)=0 DRSTAS(*)=0

The drive is in error or it does not exist.

DRSTAF(*)=0 DRSTAS(*)=1

The DC bus has no power. The drive cannot be enabled; but it is possible to provide power to the drives' power supply.

DRSTAF(*)=1 DRSTAS(*)=0

The drive's DC bus now has power. The drive may be enabled.

DRSTAF(*)=1 DRSTAS(*)=1

The drive is enabled.

INPOS(axis)

The CNC sets this mark high (=1) when the relevant axis or spindle is in position and there is no request for motion (ENABLE(axis) mark deactivated). An axis is in position when it stays within the in-position zone (window) (parameter INPOSW) for a time period indicated by parameter INPOSTIME.

There is an INPOS(axis) mark for each axis and for the spindle and a general INPOSI mark for the channel that indicates whether all axes and the spindle have reached their position or not, except the independent axes programmed from the PLC. See *"INPOSI"* on page 291.

LUBR(axis)

The CNC sets this mark high (=1) when the relevant axis or spindle needs to be lubricated. See *"LUBRENA(axis) LUBROK(axis)"* on page 325.

(a.m.p.) DISTLUBRI indicates the distance to be moved before being lubricated.

HIRTHON(axis)

This mark is related to Hirth axes. A Hirth axis is a rotary axis that can only be positioned at specific positions, every so many degrees.

The HIRTHON(axis) mark indicates whether it is working as a Hirth axis (=1) or as a regular rotary or linear axis (=0) without having to position at specific positions.

Both linear and rotary axes can be Hirth axes. (a.m.p.) HIRTH indicates whether the axis can work as Hirth axis or not. Functions G170 and G171 indicate whether it is a Hirth axis (G171, by default) or a normal linear or rotary axis (G170).

MATCH(axis)

The MATCH(axis) mark indicates whether the Hirth axis is properly positioned (=1) or not (=0). A Hirth axis is a rotary axis that can only be positioned at specific positions, every so many degrees.

PARK(axis)
UNPARK(axis)

The CNC sets mark PARK(axis) to (=1) to indicate that an axis or spindle is being parked and mark UNPARK(axis) to (=1) to indicate that an axis is being unparked. The mark PARKED(axis) is also used, it may be modified and is described later on. See *"PARKED(axis)"* on page 325.

When parking an axis or spindle, the CNC will not control the axis (it ignores the drive signals, feedback systems, etc.) because, it interprets that the axis is not present in the new machine configuration. When unparking an axis, the CNC will control the axis again because it interprets that the axis is present again in the new machine configuration.

The axes may be parked and unparked from the CNC or from the PLC.

Application example.

There are machines that, depending on the type of machining, may have 2 different axis configurations. For example, a machine that interchanges a regular spindle with an orthogonal one may have the following configurations:

- With a normal spindle, X Y Z axes configuration.
- With an orthogonal spindle, X Y Z A B axes configuration.

When working with a regular spindle, the A and B axes are not present and the CNC issues an error because it takes them into account (drives, feedback systems, etc.). The A and B axes must be parked to avoid this.

Considerations about axis parking.

The CNC will not allow parking an axis in the following cases.

- If the axis belongs to the active kinematics.
- If the axis belongs to an active transformation #AC or #ACS.
- If the axis belongs to an active angular transformation #ANGAX .
- If the axis belongs to a gantry, tandem pair or is a slaved axis.
- If the axis belongs to an active tangential control #TANGCTRL.

Considerations about spindle parking.

The CNC will not allow parking a spindle in the following cases.

- If the spindle is not stopped.
- If the spindle is working as a C axis.
- If G96 or G63 is active and it is the master spindle of the channel.
- If G33 or G95 is active and it is the master spindle of the channel or the spindle is used to synchronize the feedrate.
- If the spindle belongs to a tandem pair or is a synchronized spindle, be it the master or the slave.

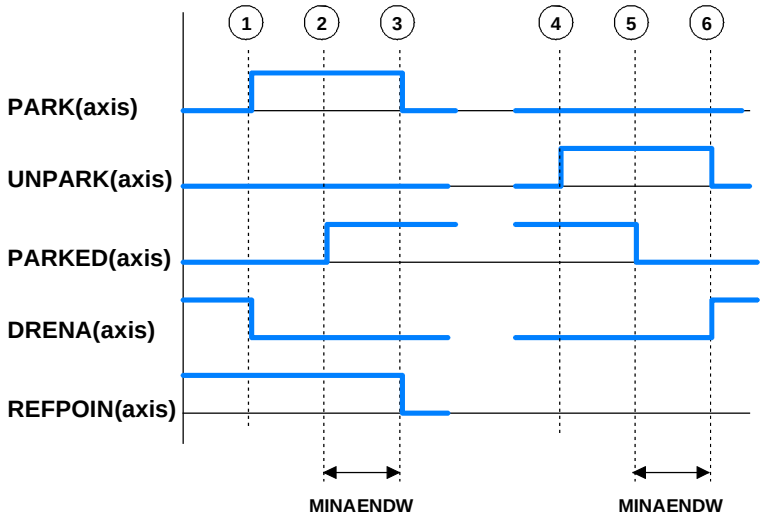
If after parking the spindles, there is only one spindle left in the channel, it will become the new master. If a spindle is unparked and it is the only spindle of the channel, it is also assumed as the new master spindle.

Park/Unpark from the CNC.

This type of maneuver is good for applications that need the axes or spindles to be parked automatically from a part program (for example from a part-program). The parking/unparking maneuver via part-program or MDI is controlled using the programming instructions #PARK and #UNPARK.

The CNC uses the PARK and UNPARK signals to inform the PLC that the parking or unparking process has been engaged.

To park an axis or a spindle, its enable signals must be (=0). Likewise, after unparking the axis, the axis enable signals must be set to (=1).



Maneuver to park an axis or spindle from the CNC.

1 When executing the programming instruction #PARK the CNC checks whether the requested axis can be parked or not. If so, the CNC sets the PARK mark high (=1) to let the PLC know that it must park the corresponding axis.

   For digital axes, the PLC must first remove the enable of the drive of the axis to be parked (DRENA). Also, the CNC sends to the drive the command to park the axis.

2 The PLC, after receiving the PARK signal, parks the requested axis. After verifying that the axis has been parked (presence sensors) the PLC sets mark PARKED to (=1).

3 The CNC will recognize that the axis has been parked when detecting that the PARKED signal has been activated. The PARK and REFPOIN signals are reset completing this process.

Maneuver to unpark an axis or spindle from the CNC.

4 When executing the programming instruction #UNPARK, the CNC sets the UNPARK mark high (=1) to let the PLC know that it must unpark the corresponding axis.

   For digital axes, the CNC sends to the drive the command to unpark the axis.

5 The PLC, after receiving the UNPARK signal, unparks the requested axis. After verifying that the axis has been unparked (presence sensors) the PLC sets mark PARKED to (=0).

6 The CNC will recognize that the process has been completed when detecting that the PARKED signal has been deactivated. The UNPARK and REFPOIN signals are reset.

   For digital axes, the PLC must activate the enable of the drive of the axis (DRENA).

Example for parking and unparking an axis:

Input I15 corresponds to the presence sensor of the "B" axis

PARKB AND NOT I15 = SET PARKEDB

If there is a request to park the "B" axis (PARKB) and the axis is not present (NOT I15), the axis is parked (SET PARKEDB).

UNPARKB AND I15 = RES PARKEDB

If there is a request to unpark the "B" axis (UNPARKB) and the axis is not present (I15), the axis is unparked (RES PARKEDB).

NOT (PARKB OR UNPARKB OR PARKEDB) AND··· = DRENAB = SPENAB = SERVOBON

If the axis is neither parked nor being parked and the enabling conditions are met, the axis gets enabled.

Park/Unpark from the PLC.

This type of maneuver is good for applications that need manual parking of the axes either while the machine is off or on (with or without power).

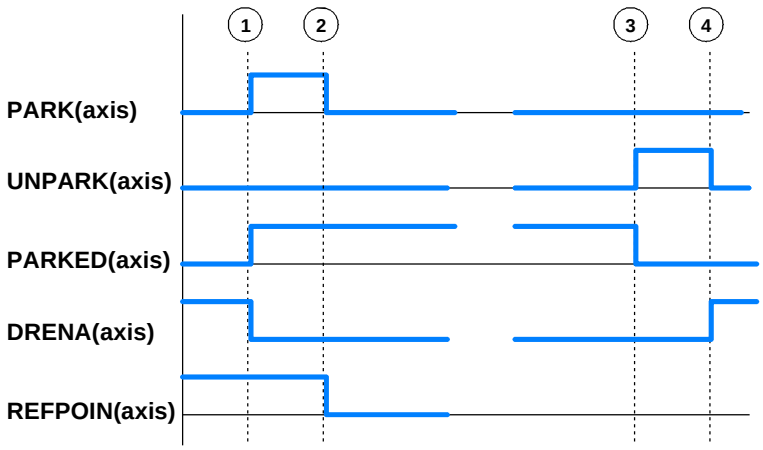
The axis parking/unparking maneuver, handled from the PLC program, is controlled using the PARKED signal. This mark is usually affected by the input for the axis presence sensor. The state of this signal is maintained even if the CNC is turned off.

The CNC uses logic signals PARK and UNPARK to inform the PLC that the parking or unparking processes are in execution.

To park an axis, the axis enable signals must be (=0). Likewise, after unparking the axis, the axis enable signals must be set to (=1).

For safety, after parking and unparking an axis, the REFPOIN signal of the axis is set to (=0).

In versions older than V2.00, at the end of the parking or unparking maneuver, the PLC must generate a reset (RESETIN).



Maneuver to park an axis or spindle from the PLC.

1 From the PLC program, the PARKED mark is set to (=1) to park the relevant axis. The CNC sets the PARK mark to (=1) and begins parking the axis.

For digital axes, the PLC must first remove the enable of the drive of the axis to be parked (DRENA). Also, the CNC sends to the drive the command to park the axis.

2 The CNC considers the operation completed. It resets the PARK signal.

Maneuver to unpark an axis or spindle from the PLC.

3 From the PLC program, the PARKED mark is set to (=0) to unpark the relevant axis. The CNC sets the UNPARK mark to (=1) and begins unparking the axis.

For digital axes, the CNC sends to the drive the command to unpark the axis.

4 The CNC considers the operation completed. It resets the UNPARK signal.

For digital axes, the PLC must activate the enable of the drive of the axis (DRENA).

Example for parking and unparking an axis:

I10 = PARKEDV

Axis present. "V" axis presence sensor

NOT (PARKV OR UNPARKV OR PARKEDV) AND··· = DRENAV = SPENAV = SERVOVON

If the axis is neither parked nor being parked and the enabling conditions are met, the axis gets enabled.

ACTFBACK(axis)

The CNC uses this mark when the system has external+internal feedback. See machine parameter FBACKSRC.

The CNC sets this mark high (=1) to indicate that the external feedback (direct feedback) is being used . The CNC sets this mark high (=0) to indicate that the internal feedback (motor feedback) is being used .

The feedback type may be swapped from the PLC using the mark FBACKSEL(axis). See *"FBACKSEL(axis)"* on page 325.

TANGACT(axis)

The CNC sets this mark high (=1) to indicate that tangential control is active in the axis. The CNC sets this mark high (=0) when freezing (suspending) or canceling the tangential control of the axis.

LOPEN(axis)

The CNC activates this mark to inform the PLC that the axis position loop is open.

MAXDIFF(axis)

This mark is used on gantry axes. The CNC activates this mark when it cannot correct the position difference between the master axis and the slave axis because the difference is greater than the value set in machine parameter MAXDIFF. Position correction must be enabled in machine parameter DIFFCOMP.

6.3 Consulting signals for the spindle.

REVOK

There is a mark for each spindle. The mnemonics for each spindle are the following:

REVOK1 (can also be programmed as REVOK)

REVOK2 REVOK3 REVOK4

It indicates whether the actual (real) spindle rpm match the ones programmed (=1) or not (=0). In other words, whether they are within the percentages set by (a.m.p.) UPSPDLIM and LOSPDLIM.

- When the spindle is stopped, M5, REVOK is (=1).
- With M3 and M4, the CNC sets this mark high (=1) when the actual spindle rpm match the ones programmed.
- When working in closed loop (M19 or G63), the CNC sets this mark low (=0) while the spindle is moving and high (=1) when the spindle is in position.

The REVOK signal may be used to handle the Feedhold signal and avoid machining at lower or higher rpm than desired.

SYNCMASTER

There is a mark for each spindle. The mnemonics for each spindle are the following:

SYNCMASTER1 SYNCMASTER2 SYNCMASTER3 SYNCMASTER4

This mark is activated at the master spindle and indicates that a synchronization has begun with the #SYNC instruction. When activating a synchronization, the CNC activates the ENABLE signal at both spindles and waits for the SERVOON signal (if DWELLis used).

When a spindle synchronization is active, the PLCCNTL, INHIBIT and SPDLEREV signals of both master and slave are ignored. Likewise, while threading, only the feedback and reference signal of the main spindle are taken into account.

SYNCHRON

There is a mark for each spindle. The mnemonics for each spindle are the following:

SYNCHRON1 SYNCHRON2 SYNCHRON3 SYNCHRON4

This mark is activated at the slave spindle and indicates that a synchronization has begun with the #SYNC instruction. When activating a synchronization, the CNC activates the ENABLE signal at both spindles and waits for the SERVOON signal (if DWELLis used).

When a spindle synchronization is active, the PLCCNTL, INHIBIT and SPDLEREV signals of both master and slave are ignored. Likewise, while threading, only the feedback and reference signal of the main spindle are taken into account.

SYNCHRONP

There is a mark for each spindle. The mnemonics for each spindle are the following:

SYNCRONP1 SYNCRONP2 SYNCRONP3 SYNCRONP4

This mark is activated at the slave spindle and indicates that a position synchronization has begun. This mark may be used to distinguish between synchronization in position or in velocity and to know which mark, SYNSPEED or SYNCPOSI, to attend to.

SYNSPEED

There is a mark for each spindle. The mnemonics for each spindle are the following:

SYNSPEED1 SYNSPEED2 SYNSPEED3 SYNSPEED4

This mark is activated at the slave spindle when it is synchronized in speed.

This mark is set to (=0) when exceeding the maximum speed error allowed, whose default value is set by machine parameter DSYNCVELW.

SYNCPOSI

There is a mark for each spindle. The mnemonics for each spindle are the following:

SYNCPOSI1 SYNCPOSI2 SYNCPOSI3 SYNCPOSI4

This mark is activated at the slave spindle when it is synchronized in position.

This mark is set to (=0) when exceeding the maximum position error allowed, whose default value is set by machine parameter DSYNCPOSW.

GEAROK

There is a mark for each spindle. The mnemonics for each channel are the following.

GEAROK1 (can also be programmed as GEAROK)

GEAROK2 GEAROK3 GEAROK4

The spindle activates this mark when the parameter set selected at the CNC and at the PLC are the same. In order for both parameter sets to coincide, funciton M41 must be active at the CNC and the GEAR1 mark at the PLC, M42 with GEAR2 and so on.

If both parameter sets do not coincide, the CNC executes no action. Include this mark in the PLC maneuver to define the actions to carry out when both parameter sets do not coincide, such as stop the spindle or interrupt the execution of the part-program.

6.4 Consultation signals of the independent interpolator.

The signal names are generic. Replace the text (axis) with the name or logic axis number.

IBUSY(axis)

For movements of the independent axis, the independent interpolator of the axis activates this mark when it has an instruction pending to execute.

IFREE(axis)

For movements of the independent axis, the independent interpolator of the axis activates this mark when it is ready to accept a new motion block. The axis interpolator can activate this mark even if there is block in execution, so it can join both blocks at the joining feedrate indicated in the first block.

IFHOUT(axis)

For movements of the independent axis, the independent interpolator of the axis activates this mark when the execution has been interrupted.

IEND(axis)

For movements of the independent axis, the independent interpolator of the axis activates this mark when it finishes generating the theoretical movement.

INSYNC(axis)

For synchronization movements of an independent axis and of an electronic cam, the interpolator of the axis activates this mark when synchronism has been reached. This mark stays active while maintaining synchronism.

PROBE1ACTIVE
PROBE2ACTIVE

It is applied when latching the coordinate of an axis. There is a mark for each probe. Probe 1 will be the one assigned to parameter PRBDI1 and probe 2 will be the one assigned to parameter PRBDI2.

The independent interpolator actives this mark when a latching process is active with the indicated probe and it deactivates when there is no active latching process with the indicated probe.

LATCH1ACTIVE(axis)
LATCH2ACTIVE(axis)

It is applied when latching the coordinate of an axis. There is a mark for each probe. Probe 1 will be the one assigned to parameter PRBDI1 and probe 2 will be the one assigned to parameter PRBDI2.

The independent interpolator actives this mark when activating a latching process on the axis with the indicated probe and it deactivates when the latching process ends or it is canceled.

The latching process on the axis is canceled when the PLC activates the IRESET(axis) mark. A reset of the channel cancels the latching processes on all axes of the channel.

Functions M02 and M30 will not be considered executed (done) until all the latching processes active on the axes of the channel are finished.

LATCH1DONE(axis)
LATCH2DONE(axis)

It is applied when latching the coordinate of an axis. There is a mark for each probe. Probe 1 will be the one assigned to parameter PRBDI1 and probe 2 will be the one assigned to parameter PRBDI2.

The independent interpolator actives this mark when the latch event on the axis with the indicated probe occurs; it deactivates it when activating a new latching process on the axis with the same probe.

6.5 Tool manager consulting signals.

TMOPERATION

There is a register for each channel. The mnemonics for each channel are the following.

TMOPERATIONC1 (same as TMOPERATION)

TMOPERATIONC2 TMOPERATIONC3 TMOPERATIONC4

This register indicates the type of operation to be carried out by the tool manager.

TMOPSTROBE

There is a mark for each channel. The mnemonics for each channel are the following.

TMOPSTROBEC1 (same as TMOPSTROBE)

TMOPSTROBEC2 TMOPSTROBEC3 TMOPSTROBEC4

The CNC sets this mark high (=1) to let the PLC know that it must execute the operation indicated by TMOPERATION.

LEAVEPOS

There is a register for each magazine. The mnemonics for each magazine are the following.

LEAVEPOSMZ1 (can also be programmed as LEAVEPOS)

LEAVEPOSMZ2 LEAVEPOSMZ3 LEAVEPOSMZ4

This register indicates the magazine position to leave the tool. While selecting a turret position (#ROTATEMZ instruction), this register takes the value of ·0· if it is a positive relative positioning and the value of ·1· if it is a negative relative positioning.

TAKEPOS

There is a register for each magazine. The mnemonics for each magazine are the following.

TAKEPOSMZ1 (can also be programmed as TAKEPOS)

TAKEPOSMZ2 TAKEPOSMZ3 TAKEPOSMZ4

This register indicates the magazine position of the tool to be taken. While selecting a turret position (#ROTATEMZ instruction), this register takes the value of ·0· if it is an absolute positioning and the value of ·1· if it is a relative positioning.

NEXTPOS

There is a register for each magazine. The mnemonics for each magazine are the following.

NEXTPOSMZ1 (can also be programmed as NEXTPOS)

NEXTPOSMZ2 NEXTPOSMZ3 NEXTPOSMZ4

This register indicates the magazine position occupied by the next tool. While selecting a turret position (#ROTATEMZ instruction); in an absolute positioning, this register indicates the position to reach and, in a relative positioning, the number of positions to rotate.

TWORNOUT

There is a mark for each channel. The mnemonics for each channel are the following.

TWORNOUTC1 (can also be programmed as TWORNOUT)

TWORNOUTC2 TWORNOUTC3 TWORNOUTC4

The CNC channel sets this mark high (=1) to "tell" the PLC that the tool has been rejected because it is worn out (real life > maximum life span).

TMINEM

There is a register for each magazine. The mnemonics for each magazine are the following.

TMINEMZ1 (can also be programmed as TMINEM)

TMINEMZ2 TMINEMZ3 TMINEMZ4

The CNC sets this mark high (=1) to inform the PLC that an emergency has occurred at the tool manager.

MZID

There is a register for each channel. The mnemonics for each channel are the following.

MZIDC1 MZIDC2 MZIDC3 MZIDC4

This register indicates which magazine contains the tool requested by the channel.

When two magazines are involved in a tool change, the lower portion of this register indicates the destination magazine for the tool and the higher portion the source magazine for the tool.

6.6 Keystroke consulting signals.

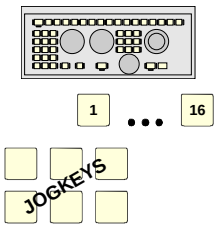
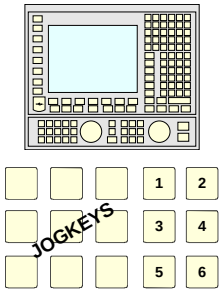
KEYBD1 / KEYBD2 / KEYBD3

These registers are a copy of the map of the keys pressed on the last keyboard used. These registers indicate which key has been pressed (bit=1). If there is only one keyboard, these registers coincide with KEYBD1_1 to KEYBD3_1. When there are several keyboards, the contents of these registers are not always the same as KEYBD1_1 to KEYBD3_1; therefore, they may be used indistinctively.

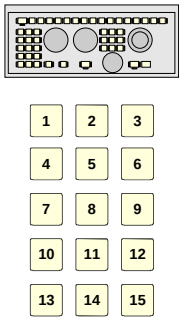
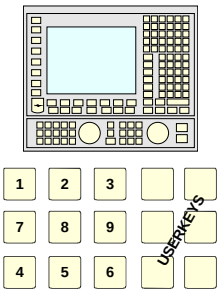
KEYBD1_1 / KEYBD2_1 / KEYBD3_1
..
KEYBD1_8 / KEYBD2_8 / KEYBD3_8

These registers indicate (bit=1) which key has been pressed on each operator panel. Registers KEYBD1_1 to KEYBD3_1 correspond to the first jog panel, KEYBD1_2 to KEYBD3_2 to the second one and so on.

Registers KEYBD1 / KEYBD1_1 to KEYBD1_8. User keys.

Bit	Jog Panel	LCD-10K
		
0	User key 1	User key 1
1	User key 2	User key 2
2	User key 3	User key 3
3	User key 4	User key 4
4	User key 5	User key 5
5	User key 6	User key 6
6	User key 7	- - -
7	User key 8	- - -
8	User key 9	- - -
9	User key 10	- - -
10	User key 11	- - -
11	User key 12	- - -
12	User key 13	- - -
13	User key 14	- - -
14	User key 15	- - -
15	User key 16	- - -

Registers KEYBD1 / KEYBD1_1 to KEYBD1_8. Jog keys.

Bit	Jog Panel	LCD-10K
		
16	Jog key 1	Jog key 1
17	Jog key 2	Jog key 2
18	Jog key 3	Jog key 3
19	Jog key 4	Jog key 4
20	Jog key 5	Jog key 5
21	Jog key 6	Jog key 6
22	Jog key 7	Jog key 7
23	Jog key 8	Jog key 8
24	Jog key 9	Jog key 9
25	Jog key 10	- - -
26	Jog key 11	- - -
27	Jog key 12	- - -
28	Jog key 13	- - -
29	Jog key 14	- - -
30	Jog key 15	- - -
31	- - - -	- - -

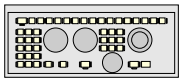
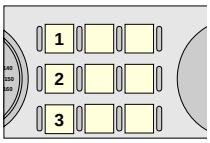
Registers KEYBD2 / KEYBD2_1 to KEYBD2_8.

Bit	Key
0	Spindle override +
1	Spindle clockwise
2	Spindle positioning
3	Spindle stop
4	Spindle override -
5	Spindle counterclockwise
6	START
7	STOP
8	Generic key 1
9	Generic key 2

Bit	Selector
10	Generic key 3
11	ZERO
12	- - -
13	Single block
14	CNC OFF
15	RESET
16 - 20	Feed override
21 - 23	- - -
24 - 27	Mode selector
28 -31	ZERO

When the operator panel has a spindle speed override switch, the keys associated with the speed override (bits 0 and 4) no longer have this functionality and may be configured from the PLC.

Registers KEYBD2 / KEYBD2_1 to KEYBD2_8. Generic keys.

Bit	Jog Panel
	
	
8	Generic key 1
9	Generic key 2
10	Generic key 3

Registers KEYBD2 / KEYBD2_1 to KEYBD2_8. Feedrate override selector

KEYBD2					
20	19	18	17	16	
0	0	0	0	0	0 %
0	0	0	0	1	2 %
0	0	0	1	0	4 %
0	0	0	1	1	10 %
0	0	1	0	0	20 %
0	0	1	0	1	30 %
0	0	1	1	0	40 %
0	0	1	1	1	50 %
0	1	0	0	0	60 %
0	1	0	0	1	70 %
0	1	0	1	0	80 %
0	1	0	1	1	90 %
0	1	1	0	0	100 %
0	1	1	0	1	110 %
0	1	1	1	0	120 %
0	1	1	1	1	130 %
1	0	0	0	0	140 %
1	0	0	0	1	150 %
1	0	0	1	0	160 %
1	0	0	1	1	170 %
1	0	1	0	0	180 %
1	0	1	0	1	190 %
1	0	1	1	0	200 %

Registers KEYBD2 / KEYBD2_1 to KEYBD2_8. Movement selector (handwheel, incremental jog or continuous jog).

KEYBD2				
27	26	25	24	
0	0	0	0	Handwheel x100
0	0	0	1	Handwheel x10
0	0	1	0	Handwheel x1
0	0	1	1	Jog 1
0	1	0	0	Jog 10
0	1	0	1	Jog 100
0	1	1	0	Jog 1000
0	1	1	1	Jog 10000
1	0	0	0	Continuous jog

Registers KEYBD3 / KEYBD3_1 to KEYBD3_8. Spindle speed override selector.

KEYBD3					
4	3	2	1	0	
0	0	0	0	0	0 %
0	0	0	0	1	2 %
0	0	0	1	0	4 %
0	0	0	1	1	10 %
0	0	1	0	0	20 %
0	0	1	0	1	30 %
0	0	1	1	0	40 %
0	0	1	1	1	50 %
0	1	0	0	0	60 %
0	1	0	0	1	70 %
0	1	0	1	0	80 %
0	1	0	1	1	90 %
0	1	1	0	0	100 %
0	1	1	0	1	110 %
0	1	1	1	0	120 %
0	1	1	1	1	130 %
1	0	0	0	0	140 %
1	0	0	0	1	150 %
1	0	0	1	0	160 %
1	0	0	1	1	170 %
1	0	1	0	0	180 %
1	0	1	0	1	190 %
1	0	1	1	0	200 %

6.7 General modifiable signals.

The _EMERGEN, _STOP, _FEEDHOL, _XFERINH and CYSTART signals must be defined in the PLC program.

_EMERGEN

There is a mark for each channel. The mnemonics for each channel are the following.

_EMERGENC1 (can also be programmed as _EMERGEN)

_EMERGENC2 _EMERGENC3 _EMERGENC4

If the PLC sets this mark low (=0), the CNC stops the axes and the spindle and displays the corresponding error message. The CNC interrupts the movement of the axes leaving them without command; the axes do not stop using a set acceleration ramp.

While the _EMERGEN mark is low (=0), the CNC does not allow executing programs and aborts any axis or spindle movement.

I-EMERG AND (rest of conditions) = _EMERGEN

If the emergency-stop button is pressed (I-EMERG=0) or any other emergency situation occurs (=0), the _EMERGEN mark is set low (=0) causing an emergency at the CNC.

_STOP

There is a mark for each channel. The mnemonics for each channel are the following.

_STOPC1 (can also be programmed as _STOP)

_STOPC2 _STOPC3 _STOPC4

If the PLC sets this mark low (=0), the CNC channel interrupts the execution of the part-program; but it keeps the spindle turning. It is similar to pressing the [STOP] button at the keyboard.

The movement of the independent axes is not affected by this mark. Nor is affected by the [STOP] key of the operator panel.

To resume the execution of the program, besides setting this mark high (=1), the CYSTART mark must also be activated. See *"CYSTART"* on page 315.

_XFERINH

There is a mark for each channel. The mnemonics for each channel are the following.

_XFERINHC1 (can also be programmed as _XFERINH)

_XFERINHC2 _XFERINHC3 _XFERINHC4

If the PLC sets this mark low (=0), the CNC channel interrupts the execution of the program at the end of the block being executed and inhibits the execution of the next block. If the axis needs more braking distance than it has with the block being executed, the CNC can continue executing more blocks until the axis comes to a complete stop respecting the dynamics of the machine. When this mark is set back high (=1), the CNC resumes the execution of the program.

Having this mark (=0), the CNC does not allow jog movements for the axes of the channel; pressing a jog key is ignored.

This mark always affects the movements of the independent axis programmed from the CNC; the ones programmed from the PLC depend on parameter XFITOIND. If the PLC deactivates the _XFERINH mark, the CNC channel interrupts the movements of the independent axis when they reach position and inhibits the execution of the next movement.

XFITOIND	Meaning.
Yes	The _XFERINH mark affects the movement of the independent axis programmed from the PLC and from the CNC.
No	The _XFERINH mark does not affect the movements of the independent axis programmed from the PLC; but it does affect the ones programmed from the CNC.

To manage the transfer inhibit in an independent movement, the PLC also has a particular mark per axis (_IXFERINH(axis) mark). See *" _IXFERINH(axis)"* on page 331.

_FEEDHOL

There is a mark for each channel. The mnemonics for each channel are the following.

_FEEDHOLC1 (can also be programmed as _FEEDHOL)

_FEEDHOLC2 _FEEDHOLC3 _FEEDHOLC4

If the PLC sets this mark low (=0), the CNC channel interrupts momentarily the movement of the axes; but keeping the spindle turning. When this mark goes back high (=1), the axes resume their motion. All the stops and starts of the axes are carried out with the corresponding acceleration and deceleration.

The movement of the independent axes is not affected by this mark.

If the _FEEDHOL mark is set low (=0) in a motionless block, the CNC continues executing the program until it detects a motion block.

On hirth axes, if the axis does not stop in a particular position, the CNC does not activate the MATCH(axis) mark.

The "Freal" text of the screens of the automatic and jog modes shows the status of this mark. The text appears in red when the _FEEDHOL mark is active. The status of the mark is not displayed if the screen does not show this text.

CYSTART

There is a mark for each channel. The mnemonics for each channel are the following.

CYSTARTC1 (can also be programmed as CYSTART)

CYSTARTC2 CYSTARTC3 CYSTARTC4

When the operator presses the [START] key, the CNC lets the PLC know by activating the START mark. If the rest of the conditions are met (hydraulic, safety, etc.), the PLC must activate the CYSTART mark in order for the program to start running.

```
START AND (rest of conditions) = CYSTART
```

SBLOCK

There is a mark for each channel. The mnemonics for each channel are the following.

SBLOCKC1 (can also be programmed as SBLOCK)

SBLOCKC2 SBLOCKC3 SBLOCKC4

If the PLC sets this mark high (=1), the CNC switches to SINGLE BLOCK operation mode.

MANRAPID

There is a mark for each channel. The mnemonics for each channel are the following.

MANRAPIDC1 (can also be programmed as MANRAPID)

MANRAPIDC2 MANRAPIDC3 MANRAPIDC4

This mark is treated in a similar way to the Rapid traverse key. If the PLC turns this mark on, the CNC selects the rapid traverse for continuous jog movements. The incremental jog moves are carried out at the active feedrate in jog mode. When the PLC turns this mark off, all jog moves are carried out at the active feedrate in jog mode.

OVRCAN

There is a mark for each channel. The mnemonics for each channel are the following.

OVRCANC1 (can also be programmed as OVRCAN)

OVRCANC2 OVRCANC3 OVRCANC4

If the PLC sets this mark high (=1), the CNC applies 100 % override to the axis feedrate regardless of the value currently selected. While the OVRCAN mark is high (=1), the CNC channel will apply 100% of the feedrate corresponding to each work mode.

LATCHM

With this mark, it is possible to select how the JOG keys will work in JOG mode.

- If this mark is low (=0), the axes will move while the corresponding JOG key kept pressed.
- If it is high (=1), the axes will move from the instant the JOG key is pressed until the software limits are reached or the [STOP] key is pressed or another JOG key is pressed (in this case the new axis will start moving).

RESETIN

There is a mark for each channel. The mnemonics for each channel are the following.

RESETINC1 (can also be programmed as RESETIN)

RESETINC2 RESETINC3 RESETINC4

With an up-flank, (0-to-1 transition), the CNC channel assumes the machining conditions selected by machine parameter and it activates the RESETOUT mark. See *"RESETOUT"* on page 288.

The treatment of this mark is similar to that of the [RESET] key.

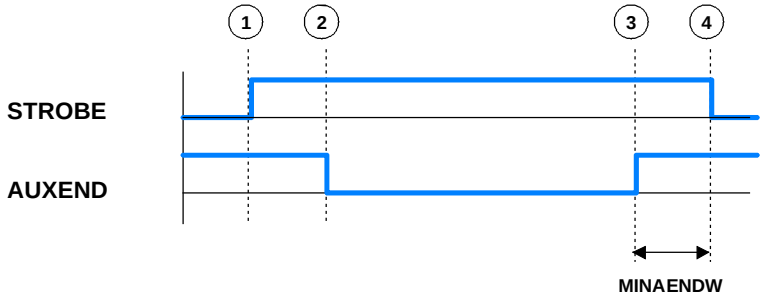
AUXEND

There is a mark for each channel. The mnemonics for each channel are the following.

AUXENDC1 (can also be programmed as AUXEND)

AUXENDC2 AUXENDC3 AUXENDC4

It is used when executing the auxiliary functions M and S. It works as follows:



1 The CNC channel indicates to the PLC, in registers MFUN and SFUN of the channel, the functions that it must execute and it activates the MSTROBE or SSTROBE mark to start the execution.

2 The PLC, when detecting that one of these marks has been activated, it must cancel the AUXEND mark to let the CNC know that the execution begins.

3 Once the required auxiliary functions have been executed, the PLC must activate the AUXEND mark to let the CNC know that the execution has ended. The AUXEND mark must be kept high (=1) longer than the time period established by (g.m.p.) MINAENDW.

4 After this time, the CNC deactivates the corresponding SSTROBE or MSTROBE mark thus ending the execution of the function.

BLKSKIP1

There is a mark for each channel. The mnemonics for each channel are the following.

BLKSKIP1C1 (can also be programmed as BLKSKIP1)

BLKSKIP1C2 BLKSKIP1C3 BLKSKIP1C4

The PLC sets this mark high (=1) to let the CNC channel know that the block skip condition "/" is met. Therefore, blocks with that condition will not be executed.

M01STOP

There is a mark for each channel. The mnemonics for each channel are the following.

M01STOPC1 (can also be programmed as M01STOP)

M01STOPC2 M01STOPC3 M01STOPC4

The PLC sets this mark high (=1) to "tell" the CNC channel not to ignore the conditional stops (M01).

TIMERON

The CNC has a timer to be used at will.

It times in seconds and it is enabled and disabled with this mark. It will be enabled (counting) with TIMERON set high (=1).

To reset and or know its accumulated timing, use the variable: (V.)PLC.TIMER. This variable is accessible from the PLC program, MDI or interface (any supported application).

Example of how to know the machining time.

```
CY1
() = MOV 0 R100
()= CNCWR (R100, PLC.TIMER, M11)
    Resets the timer to "0" on power-up.
END
PRG
AUTOMAT AND INCYCLE = TIMERON
    Timer active while machining.
()= CNCRD (PLC.TIMER, R300, M12)
    Register R300 shows the value of the timer.
END
```

PLCREADY

This mark indicates whether the PLC is running (=1) or stopped (=0).

- It must be high (=1) for the CNC to allow moving the axes and turning the spindle.
- If set low (=0), the PLC program stops and an error message is displayed.

NOWAIT

There is a mark for each channel. The mnemonics for each channel are the following.

NOWAITC1 NOWAITC2 NOWAITC3 NOWAITC4

It is applied to channel synchronization. The PLC sets this mark high (=1) to cancel all the synchronizations with the CNC channel.

For example, with the NOWAITC1 signal set to (=1), the waits programmed in any channel with the #WAIT, instruction and that refer to a mark of channel 1, they finish immediately and the program execution resumes.

DISCROSS1··n

It is applied to the cross compensation tables defined in the machine parameters. There is a mark for each table.

The PLC sets the mark to (=1) to disable the table. The DISCROSS1 mark is for table 1, DISCROSS2 for table 2 and so on.

PLCABORT

There is a mark for each channel. The mnemonics for each channel are the following.

PLCABORTC1 (can also be programmed as PLCABORT)

PLCABORTC2 PLCABORTC3 PLCABORTC4

If the PLC sets this mark low (=0), the CNC channel aborts the CNCEX command launched from the PLC, but without initializing the conditions of the channel and keeping the history of the channnel. This mark is only valid when the channel is executing a CNCEX; not in other program or MDI executions.

Once the execution has been canceled, the CNC deactivates this mark and sets the FREE mark of the channel high (=1). See *"FREE"* on page 295.

The PLCABORT mark is activated without having launched a CNCEX, the mark stays high (=1) until a CNCEX is executed (which will be aborted automatically) or until it is deactivated (RES PLCABORT).

PRGABORT

There is a mark for each channel. The mnemonics for each channel are the following.

PRGABORTC1 (can also be programmed as PRGABORT)

PRGABORTC2 PRGABORTC3 PRGABORTC4

If the PLC sets this mark low (=1), the CNC channel aborts the current execution of the part-program; but without affecting the spindle, the rest of the history is initialized. Then, the CNC resumes the execution of the program from the label indicated in the #ABORT instruction that is active in the part-program.

If no #ABORT instruction is active in the part-program, the PRGABORT mark has no effect.

NEXTMPGAXIS

This mark may be used to select an axis sequentially for jogging it with the handwheel. Only the axes being displayed in the active channel may be selected, regardless of the channel they belong to. The axes of another channel cannot be selected if they are not being displayed.

This mark is only taken into account when the CNC is in jog mode and the selector is in handwheel mode. With an up flank of this mark, transition from (=0) to (=1), the CNC behaves as follows.

- If no axis has been selected, it selects the first one that is displayed.
- If there is an axis selected, it selects the next one; if the last axis is selected, it selects the first one again.

An axis is de-selected when quitting the handwheel mode using the movement selector and after a reset.

This mark is meant for handwheels with push-button. On this type of handwheels, the push-button may be used to select, sequentially, the axis to be jogged. The usual thing in these cases is to connect the push-button of the handwheel to the digital input that will be in charge of managing the NEXTMPGAXIS mark.

PANELOFF
PANELOFF1
..
PANELOFF8

If the PLC activates one of these marks the CNC disables the corresponding jog panel. Each mark may be used to disable the jog panel integrated into the CAN bus; the rest of the elements of the bus are not affected. When the keyboard and the operator panel make up a single element, this mark only disables the jog panel.

There is one mark for each jog panel. The PANELOFF1 mark disables the first one of the bus, the PANELOFF2 mark the second one and so on. Both PANELOFF and PANELOFF1 are valid for the first jog panel.

The "address" switch determines the order of the elements in the CAN bus. The first element will be the one with the lowest number and so on. The element with the lowest number corresponds to PANELOFF1.

Address	Element	PLC mark
0	CNC	
1	Remote (I/O) group	
2	Jog panel	PANELOFF1
3	Remote (I/O) group	
4	Remote (I/O) group	
5	Keyboard + Jog panel	PANELOFF2

SYNC

There is a register for each channel. The mnemonics for each channel are the following.

SYNC1 SYNC2 SYNC3 SYNC4

This register is used when using, from one channel, a particular spindle for synchronization even if the spindle is in another channel. For example, in the case of dual-turret lathe with a single spindle.

- With the G33 function, when threading with a particular spindle.
- With the G95 function, when programming the feedrate as a function of the turning speed of a particular spindle.

To do that, the PLC indicates in channel register SYNC the spindle to be used, only for synchronization. The SYNC register will take values 1 through 4; when assigning a 0 value, it will use the master spindle of the channel.

The CNC will check the contents of this register at the beginning of the block. If the PLC modifies this register during the execution of the block, the change will not effective until the beginning of the next block.

RETRACE

There is a mark for each channel. The mnemonics for each channel are the following.

RETRACEC1 RETRACEC2 RETRACEC3 RETRACEC4

If the PLC sets this mark high (=1) while executing a program in automatic mode, the retrace function is activated in the selected channel. The retrace function stops the execution of the program and starts executing backwards the path traveled so far at the current block plus the last n blocks executed. The maximum number of blocks to execute in retrace mode is set by machine parameter NRETBLK.

The retrace function may be activated during an interpolation, in the middle of a block and also at the end of the block, whether the execution was interrupted by M0 or by the single block mode.

If the PLC sets this mark high (=1), the CNC finishes the RETRACE function. The retrace function also ends after M30, with a reset or when the CNC channel sets mark RETRAEND high (=1) See *"RETRAEND"* on page 295.

The retrace function must be enabled in machine parameter RETRACAC.

CNCOFF

Setting this PLC mark high (=1) initiates the CNC shut-down sequence. Activating this mark is the same as pressing the key combination [ALT]+[F4].

INHIBITMPG1
..
INHIBITMPG12

If the PLC turns one of these marks on, it disables the corresponding handwheel. The PLC has a mark for each handwheel, the INHIBITMPG1 mark disables the first handwheel, the INHIBITMPG2 mark the second one and so on.

If the handwheel is disabled, the CNC ignores the pulses coming from the handwheel and, therefore, it does not move the axis. While the handwheel is disabled, the variable associated with the handwheel does not save the pulses sent by the handwheel.

(V.)G.HANDP[hw] Number of pulses sent by the handwheel since the system was started up.

If it is an individual handwheel, associated with an axis, the ENABLE(axis) mark of the axis will stay active. If the handwheel mode is selected in jog mode for that axis, that axis will appear highlighted even if the PLC has disabled the handwheels that could move it.

EXRAPID

There is a mark for each channel. The mnemonics for each channel are the following.

EXRAPIDC1 (can also be programmed as EXRAPID)

EXRAPIDC2 EXRAPIDC3 EXRAPIDC4

If the PLC activates this mark, the CNC channel enables rapid traverse during the execution of a program for the programmed movements. The behavior of this mark depends on how parameter RAPIDEN has been set.

The treatment of this signal is similar to the rapid key of the operator panel.

KEYBD1CH
..
KEYBD8CH

On power-up, the CNC always assumes the keyboard configuration set in the machine parameters. These registers may be used to change the default behavior of the keyboards with respect to the channels, set by machine parameters. These registers can associate a jog panel with a particular channel, always with the active channel or restore the configuration defined in the machine parameters.

KEYBD1CH .. KEYBD8CH	Meaning.
0	Configuration defined in the machine parameters.
1	Jog panel assigned to channel 1.
2	Jog panel assigned to channel 2.
3	Jog panel assigned to channel 3.
4	Jog panel assigned to channel 4.
FF	Jog panel assigned to the active channel.

VOLCOMP1
..
VOLCOMP4

If the PLC activates one of these marks, the CNC activates the corresponding volumetric compensation (parameter VOLCOMP). All the volumetric compensations may be active at the same time. Volumetric compensation remains active after a reset or end of program (M30).

QWERTYOFF1
..
QWERTYOFF8

If the PLC turns one of these marks on, it disables the corresponding alphanumeric keyboard. Each mark may be used to disable a USB keyboard (10K model) or one integrated into the CAN bus (15" model) ; the rest of the elements of the bus are not affected. The keyboards are not enabled or disabled immediately, especially when activating the USB keyboard, it may take a few seconds.

The way the QWERTYOFF1 works depends on the CNC model.

- On the 10K model, this mark affects the alphanumeric keyboard and the softkeys of the monitor (USB interface).
- On the 15" model, this mark affects the softkeys of the monitor (USB interface) and the first node of the CAN bus. In order for the mark to affect the keyboard associated with the central unit, the number of this keyboard must be the lowest one in the bus.

Marks QWERTYOFF2 to QWERTYOFF8 affect the rest of the CAN keyboards, depending on the logic numbers of the nodes, from lowest to highest.

FLIMITAC

If the PLC activates this mark, the CNC activates the feedrate safety limits (parameter FLIMIT) on all the axes of the system during the execution of the block. If the PLC deactivates this mark, the CNC restores the programmed feedrate.

The feedrate safety limit is applied to the movements in automatic (G0, G1, etc.) and in jog mode (jog, handwheels, etc.). This parameter affects neither the threading operations nor the independent axis movements, which are executed at the programmed feedrate.

FLIMITACCH

There is a mark for each channel. The mnemonics for each channel are the following.

FLIMITACCHC1 (can also be programmed as FLIMITACCH)

FLIMITACCHC2 FLIMITACCHC3 FLIMITACCHC4

If the PLC activates this mark, the CNC activates the feedrate safety limits (parameter FLIMIT) on all the axes of the channel during the execution of the block. If the PLC deactivates this mark, the CNC restores the programmed feedrate.

The feedrate safety limit is applied to the movements in automatic (G0, G1, etc.) and in jog mode (jog, handwheels, etc.). This parameter affects neither the threading operations nor the independent axis movements, which are executed at the programmed feedrate.

SLIMITAC

If the PLC activates this mark, the CNC activates the speed safety limits (parameter SLIMIT) on all the spindles of the system during the execution of the current block. If the PLC deactivates this mark, the CNC restores the programmed speed.

INT1
..
INT4

There is a mark for each channel. The mnemonics for each channel are the following. Here is an example of the mnemonics for INT1; it is the same for the rest of the marks.

INT1C1 (can also be programmed as INT1)

INT1C2 INT1C3 INT1C4

When the PLC activates one of these marks, the channel interrupts the execution of the program and executes the corresponding interruption subroutine associated with parameters INT1SUB to INT4SUB. The CNC executes the subroutine with the current history of the interrupted program (G functions, feedrate, etc.). Once the subroutine is executed, the CNC resumes the execution of the program from the interruption point and maintaining the changes made to the history by the subroutine (G functions, etc.).

If the program is interrupted (STOP) or no program is in execution (channel in READY state), the execution of the subroutine depends on parameter SUBINTSTOP. Also, in order to execute the subroutine when no program is in execution, the channel must be in automatic mode; the subroutine cannot be executed from jog mode.

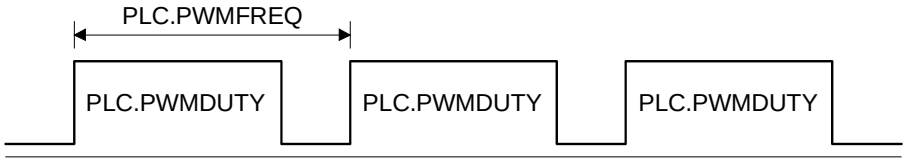
The execution of an interruption subroutine may be interrupted with a STOP, but not by another interruption subroutine. When a subroutine is interrupted, it is not possible to get into tool inspection mode.

The CNC cancels the mark when it starts executing the subroutine or when it rejects the execution of the subroutine.

PWMON

This mark may be used to activate and deactivate the PWM via PLC. The PWM is configured using the following variables.

Variable.	Meaning.
(V.)PLC.PWMFREQ	Frequency of the PWM (between 2 and 5000 Hz; by default, 0).
(V.)PLC.PWMDUTY	Duty cycle of the PWM (between 1 and 1000; by default, 500).



Activating the Laser via PLC has priority over activating it from the CNC.

Laser status from the CNC. Variable (V.)G.PWMON	Laser status from the PLC. PWMON mark	Laser status.
0	0	Laser OFF.
1	0	Laser ON from the CNC.
0	1	Laser ON from the PLC.
1	1	Laser ON from the PLC.

The modifications to the PWM both via program or via PLC are updated (refreshed) without waiting for the PWM cycle in progress to be completed and as continuously as possible with respect to the previous conditions; in other words, it does not wait for the default signal to set to zero or to one in each change.

During tool inspection, the CNC does not deactivate the PWM This mark may be used to deactivate the PWM viaPLC during tool inspection and resume the PWM at the end of the inspection.

6.8 Modifiable signals for axes and spindles.

When the spindle is working in closed loop (M19 or G63), it behaves like an axis.

The signal names are generic. Replace the word (axis) with the spindle name or with the name or logic number of the axis. For example, the name of the LIMITPOS(axis) mark for a machine with the X, Y, Z, Z2, B axes and spindle S.

LIMITPOSX, LIMITPOSZ2, LIMITPOSB, LIMITPOSS

LIMITPOS3 for the Z axis.

LIMITPOS5 for the B axis.

LIMITPOS(axis)
LIMITNEG(axis)

If the PLC sets this mark high (=1), the CNC interprets that the corresponding axis or spindle has exceeded the positive (POS) or negative (NEG) travel limits.

It stops the axes and the spindle and displays the relevant error message.

To take the axis to the work zone, access the JOG mode and move the axis or spindle that overran the travel limit. They can only be moved in the proper direction.

DECEL(axis)

This mark is used during home search. The PLC sets this mark high (=1) to indicate that the home switch is pressed. The CNC decelerates the axis, it switches from the fast homing feedrate indicated by (a.m.p.) REFFEED1, to the slow homing feedrate indicated by the (a.m.p.) REFFEED2.

When searching home on axes that use distance-coded reference marks, the PLC sets this mark to (=1) to indicate that the axis has reached the travel limit. In this case, it is necessary to reverse the moving direction to go on with the home search.

INHIBIT(axis)

If the PLC sets this mark high (=1) the CNC inhibits any movement of the corresponding axis or spindle. This movement will resume when the PLC sets this mark back low (=0). If the axis or spindle is moving with other axes, all the axes will stop.

For independent axes and electronic cam, if the PLC sets this signal to (=1), it interrupts the synchronization movement switching to zero speed. The system waits for the signal to deactivate before resuming the execution and the movement from the interruption point.

For independent axes, this signal also stops the monitoring of the synchronization.

The screens of the automatic and jog modes show the status of this mark.

- For the spindles, the text "Sreal" appears in red when this mark is active for the spindle. The status of the mark is not displayed if the screen does not show this text.
- For the axes, the name of the axis preceding the coordinate appears in red when this mark is active for the axis.

AXISPOS(axis)
AXISNEG(axis)

The CNC uses these marks when operating in JOG mode.

If the PLC sets one of these marks high (=1), the CNC will move the relevant axis in the indicated direction: positive (POS) or negative (NEG). The CNC will move the axis at the corresponding feedrate and selected override (%).

The treatment of these marks is similar to the JOG keys of the operator panel.

SERVO(axis)ON

This mark must be high (=1) in order to be able to move the corresponding axis. If the SERVO(axis)ON mark is set low (=0) while moving the axis or spindle, the CNC stops the axes and the spindle. It also displays the relevant error message.

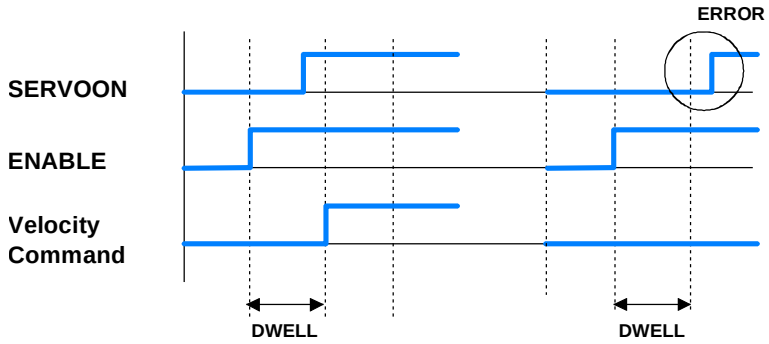
- In order to be able to continuously control the axis, the SERVOON mark must always be high (=1).

  (there are no errors) AND (axis drive OK) = SERVOnON

- To control the axis only when they move, use the ENABLE mark. The CNC sets it high (=1) whenever it has to move the axis. See *"ENABLE(axis)"* on page 298.

  (no errors) AND (drive OK) AND ENABLE = SERVOON

If the axis moves while locked (meaning SERVOON=0), the CNC stores that displacement as axis lag (following error). When controlling it back (SERVOON=1) it restores its position.



After activating the ENABLE mark, the CNC waits for a time period indicated by parameter DWELL only if SERVOnON is low (=0). If after this time period the SERVOnON signal is still deactivated, the CNC shows the error "axis locked up".

To control the axis only while moving, the (a.m.p.) DWELL must be assigned a value greater than 2 PLC cycle scans in order to avoid the error message "axis locked".

When the CNC works in any square-corner mode (G05, G50 or HSC), it can issue the "axis locked up" error in the block-to-block transition if the SERVOnON mark of any of the axes involved has not been activated before. To avoid this problem when the CNC works with dead axes, the PLC must activate the DEAD(axis) mark so the CNC does not apply block transition on these dead axes.

DRO(axis)

This mark, together with the relevant SERVOnON mark allows the axis or spindle as a dro axis. To do that, the DRO mark must be set high (=1) and SERVOnON low (=0).

When working as DRO axis or spindle, the CNC neither closes the position loop nor generates following error because the real and theoretical coordinates are the same.

The programmed movements are considered executed without making any movement or causing error messages. When programming an interpolation with other axes, the other ones will move at the corresponding feedrate; but the dro axis will not move.

When the DRO marks goes back low (=0), the axis is no longer a dro axis and it assumes the current position having a following error of "0".

LIM(axis)OFF

If the PLC sets this mark high (=1), the CNC ignores the software travel limits set for the corresponding axis. With LIM(axis)OFF set to (= 0), those software limits are no longer ignored.

The software travel limits may be set with machine parameters, using functions G198 and G199 or using variables. The CNC always applies the most restrictive travel limits.

SPENA(axis)
DRENA(axis)

The PLC uses these marks to enable the drives when communicating via Sercor or to enable the servo drives and inverters when communicating via Mechatrolink. The SPENA(axis) mark corresponds to the "speed enable" signal and the DRENA(axis) to the "drive enable" signal of the device. The operation of these two signals is the following.

The operation of these two signals is the following.

- On PLC power-up, it must cancel both signals.
- For the device to run normally, both signals must be active. The motor will respond to any changes of velocity command.
- If the PLC cancels the DRENA signal (drive enable), the power circuit of the device shuts down and the motor loses its torque. In this situation, the motor is no longer governed and will turn freely until it stops by inertia.
- If the PLC cancels the SPENA signal (speed enable), the internal velocity command of the drive is switched to "0 rpm". In this situation, the motor keeps its torque while braking and, once stopped, the drive's power circuit shut down and the motor has no torque.

PARKED(axis)

The PLC sets this mark high (=1) to let the CNC know that the corresponding axis or spindle is parked.

For further information, see the consulting signals for axes and spindles: PARK and UNPARK described earlier in this chapter. See *"PARK(axis) UNPARK(axis)"* on page 300.

LUBRENA(axis)
LUBROK(axis)

These marks together with the axis consulting signal LUBR(axis) must be used to lubricate the axes. See *"LUBR(axis)"* on page 299.

The LUBRENA mark indicates whether to use this feature (=1) or not (=0). With LUBRENA set to (=1), the CNC acts as follows:

1. When the axis has traveled the distance set by (a.m.p.) DISTLUBRI, it sets the LUBR mark high (=1) to let the PLC know that the axis must be lubricated.
2. After lubricating the axis, the PLC sets the LUBROK(axis) mark high (=1) to let the CNC know that the axis has been lubricated.
3. The CNC sets the LUBR mark low (=0) and resets its count to "0".

Be careful not to set the LUBROK mark low (=0) so the feature works properly. Otherwise, its count will always stay at "0".

DIFFCOMP(axis)

This mark is used on Gantry axes to correct the position difference between the master and the slave axes. The slave axis will move until reaching the position of the master axis at the feedrate set by parameter REFFEED2.

The correction must be enabled in machine parameter DIFFCOMP and it is applied in the following cases.

- With the up-flank of SERVO*ON if DIFFCOMP is set high.
- With the up-flank of DIFFCOMP if SERVO*ON is set high.

This process can only be interrupted with RESET.

FBACKSEL(axis)

The CNC uses this mark when the system has external+internal feedback. See parameter FBACKSRC.

An up flank of this mark switches to external feedback (direct feedback) and a down flank switches to internal feedback (motor feedback). If the combined feedback is active (parameter FBMIXTIME), it is used with the up flank of this mark.

When the type of feedback is internal+external, the CNC uses the internal feedback (motor feedback) on power-up, when resetting the drive and when initializing the Sercos ring.

The ACTFBACK(axis) mark indicates which is the active feedback. See *"ACTFBACK(axis)"* on page 302.

DEAD(axis)

When the system has a dead axis, the transition between blocks (blending) where the dead axis is involved, it issues the "axis locked up" error message. This is due to the delay between the activation of the ENABLE and SERVOON signals .

To avoid this error, the PLC offers this mark that indicates how the CNC must handle the transitions between blocks when a dead axis is involved.

- If the PLC sets this mark high (=1) whenever the axis is involved in the movement, the CNC does not apply the transition (blend) between blocks; in other words, the movements where the dead axis is involved, they wait a time period set in the DWELL parameter.
- If the PLC sets this mark low (=0), the CNC blends the blocks even when the axis is involved in the movement.

This mark must be set high (=1) when the axis is enabled as dead axis and low (=0) when it is enabled as a live axis.

SWITCH(axis)

When the system has multi-axis groups, this mark may be used to toggle between the various axes or spindles of the group. The PLC activates this mark to let the CNC know which axis or spindle of the group is active. If the PLC turns this mark off, the CNC assumes that the axis or spindle does not belong to the current configuration.

TANDEMOFF(axis)

This mark may be used to temporarily decouple (unslave) the loop of the axes or spindles involved in the tandem so they can be move separately. For example, in a C axis tandem, there is no need to engage each motor which requires generating an oscillation movement on each motor without affecting the other.

This mark refers to the slave axis or spindle of the tandem. If the PLC activates this mark, the slave axis is decoupled from the master and both axes may be moved separately. The slave axis can only be moved from the PLC through the PLCOFFSET. The master axis may be moved as usual from the jog panel, from the MDI/MDA mode, etc. When moving the master axis, the CNC will not generate a velocity command for the slave axis nor will apply any compensation.

Even if the tandem is decoupled, it will remain active in terms of programming, display, etc. The CNC displays the coordinates of the master axis, does not allow programming the slave axis and none of the axes may be parked. The CNC can home the master axis and at the end of the home search, the CNC will also initialize the position value of the slave axis (with the position value of the master).

6.9 Spindle modifiable signals.

GEAR1
GEAR2
GEAR3
GEAR4

There is a mark for each spindle. The mnemonics for each spindle are the following: Here is an example of the mnemonics for GEAR1; it is the same for the rest of the marks.

GEAR1SP1 (can also be programmed as GEAR1)

GEAR1SP2 GEAR1SP3 GEAR1SP4

The PLC uses these marks to let the CNC know which spindle range (gear) is currently selected (=1). When requesting a gearchange, the CNC informs the PLC about it using auxiliary functions: M41, M42, M43 or M44. The gear change concludes when the PLC receives the confirmation signal AUXEND.

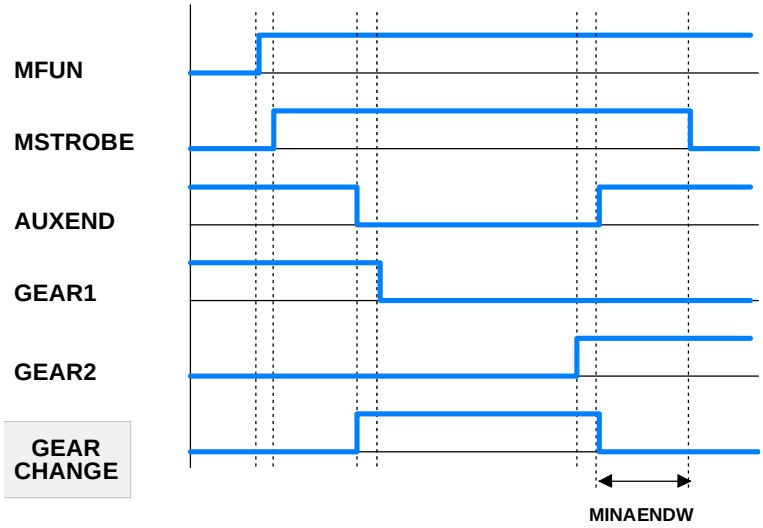
The CNC assumes the parameter set of the new gear when the spindle speed reaches the speed set in parameter SZERO and when the PLC receives the confirmation of one of marks GEAR1 through GEAR4.

On Sercos spindles, when the range change involves a gear change (NPULSES, INPUTREV, OUTPUTREV), functions M41 to M44 also change the gear at the drive.

When the parameter set selected at the CNC and at the PLC are the same, the spindle activates the GEAROK mark. In order for both parameter sets to coincide, funciton M41 must be active at the CNC and the GEAR1 mark at the PLC, M42 with GEAR2 and so on.

Example of a GEAR1 GEAR2 change.

If Gear 2 (M42) is requested while gear 1 is active.



1 The CNC indicates to the PLC the gear requested with MFUN1=42 and sets the MSTROBE mark to (=1).

2 When detecting the request, the PLC sets an internal indicator.

```
DFU MSTROBE AND CPS MFUN* EQ 42 = SET M1002
```

3 The change begins and lets the CNC know by setting AUXEND (=0).

```
NOT M1002 AND <rest of conditions> \
= AUXEND \
= (starts the gear change)
```

During the gear change, the CNC is informed that gear 1 is dropped and gear 2 is now selected. The active gear indicator, GEAR1 through GEAR4 must be set before activating the AUXEND signal.

```
I21 = GEAR1
I22 = GEAR2
```

4 Once the gear change is over, it cancels the indicator (M1002) and it lets the CNC know by setting AUXEND high (=1).

(GEAR change completed) = RES M1002

The AUXEND mark must be kept high (=1) longer than the time period set by g.m.p. MINAENDW so the CNC cancels the MSTROBE mark and concludes the gear change.

PLCCNTL SANALOG

There is one signal for each spindle. The mnemonics for each spindle are the following:

PLCCNTL1 (can also be programmed as PLCCNTL)

PLCCNTL2 PLCCNTL3 PLCCNTL4

SANALOG1 (can also be programmed as SANALOG)

SANALOG2 SANALOG3 SANALOG4

These signals are used with analog spindles, position-Sercos, velocity-Sercos and Mechatrolink.

When the PLC sets the PLCCNTL mark high (=1), the spindle decelerates with a ramp until it stops and it is then controlled by the PLC. The SANALOG register sets the spindle command voltage to be applied. The spindle control via PLC is used, for example, for oscillating the spindle during a gear change.

- Analog spindle.

  10 V of velocity command correspond to SANALOG = 32767. In other words:

  For 4V, program SANALOG = (4x32767)/10 = 13107

  For 4V, program SANALOG = (-4x32767)/10 = -13107

- Velocity-Sercos spindle.

  The command in SANALOG will be given in 0.0001 rpm.

- Sercos spindle in position.

  The command in SANALOG will be given in 0.0001 degrees.

- Mechatrolink spindle.

  The command in SANALOG will be given in hundredths of a Hertz.

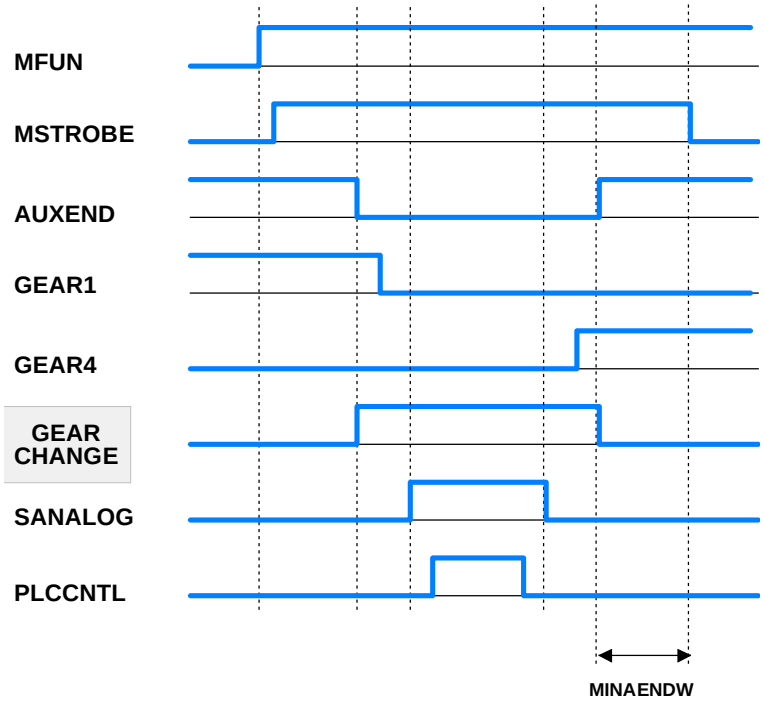
The command indicated in SANALOG is not applied with a ramp; therefore, it is up to the PLC program to apply the command gradually when necessary.

The spindle reference point is not lost when the spindle is controlled via PLC. There is no need to home the spindle again when its control is switched back to the CNC.

The PLC has no priority over a spindle synchronization. When attempting to control a synchronized spindle (either master or slave) using PLCCNTL, a warning will be issued indicating that it is not possible. Also, if the gear change of a synchronized spindle involves a command from the PLC, this change will not be possible.

Example similar to the one used for GEAR1 through GEAR4

The spindle oscillation during a gear change is controlled by the PLC. Gear 4 is requested while gear 1 is active.



The example for the GEAR1 through GEAR4 signals describes how to detect and carry out the gear change. This example shows how to control the spindle oscillation during a gear change.

The PLC sets SANALOG to the value corresponding to the residual velocity command and activates the PLCCNTL mark to indicate that the spindle is controlled by the PLC.

When done, the PLCCNTL mark must be set low (=0) and the SANALOG signal must be set to "0".

SPDLEREV

There is a mark for each spindle. The mnemonics for each spindle are the following:

SPDLEREV1 (can also be programmed as SPDLEREV)

SPDLEREV2 SPDLEREV3 SPDLEREV4

When the PLC sets this mark high (=1), the CNC reverses the spindle turning direction. To do this, it decelerates and accelerates applying the ramps set by machine parameters.

The PLC always applies this mark to the spindles in open loop(M3/M4); On spindles in closed loop (M19), it depends on parameter M19SPDLEREV. If an M2 or M4 function is executed while the SPDLEREV mark is high (=1), the spindle will turn in the opposite direction to the one assigned to the function.

If the PLC activates or cancels the SPDLEREV mark when the spindle is controlled by the PLCA (PLCCNNTL mark active), the CNC does not generate ramps to invert the SANALOG command.

PLCM3
PLCM4
PLCM5

The PLC offers a mark for each spindle. The mnemonics for each spindle are the following: Here is an example of the mnemonics for PLCM3; it is the same for the rest of the spindles.

PLCM3SP1 (can also be programmed as PLCM3)

PLCM3SP2 PLCM3SP3 PLCM3SP4

The PLC turns this mark on to let the CNC know that it must execute the corresponding M function at the indicated spindle. The spindle must belong to a channel, M functions cannot

be sent to spindles that are not assigned to any channel. The PLC may change the spindle speed using the variable "(V.)PLC.S.sn", but without generating a gear change even if the gear change is automatic (parameter AUTOGEAR).

These M functions are treated the same way as those executed from the CNC. When the CNC activates one of these marks, the CNC activates the MSTROBE mark and writes the corresponding M function into the MFUN register. Once the M function has been synchronized at the PLC (AUXEND signal), the CNC starts sending the velocity command to that spindle, updates (refreshes) the M function history and deactivates the mark at the PLC.

The CNC will admit M functions even if the channel is in execution (executing a program, jogging an axis, etc.) as long as the channel status is other than "In error" and "Not ready" and the channel is not doing a reset or validating machine parameters. If during tool inspection, the PLC changes the turning direction of a spindle using these marks, the change will be identified when repositioning and will appear as pending to reposition.

The CNC ignores the PLC requests in the following cases, when the CNC ignores the mark set by the PLC and it erases it so the request does not remain pending.

- When the spindle is working as a C axis.
- When the spindle is threading (rigid tapping, normal tapping or electronic threading).
- When the CNC status is "In error" or "Not ready", is doing a reset or validating parameters.

If the PLC activates these marks while synchronizing another M function of the program being executed or while homing the spindle, the PLC keeps this mark activated until the CNC can attend to it.

SLIMITACSPDL

There is a mark for each spindle. The mnemonics for each spindle are the following:

SLIMITACSPDL1 (can also be programmed as SLIMITACSPDL)

SLIMITACSPDL2 SLIMITACSPDL3 SLIMITACSPDL4

If the PLC activates this mark, the CNC activates the speed safety limits (parameter SLIMIT) on the indicated spindle during the execution of the current block. If the PLC deactivates this mark, the CNC restores the programmed speed.

6.10 Modifiable signals of the independent interpolator.

The signal names are generic. Replace the text (axis) with the name or logic axis number.

_IXFERINH(axis)

If the PLC sets this mark (=0), the movement of the independent axis ends and remains waiting for the PLC to activate this mark again. To manage the transfer inhibit in an independent movement, the PLC also has a general mark per channel (_XFERINH(axis) mark). The CNC evaluates both marks as follows.

- In independent movements programmed from the CNC, it first evaluates the transfer inhibit of the channel (_XFERINH mark) and then the particular one of the axis (_IXFERINH(axis) mark).
- In independent movements programmed from the PLC, the transfer inhibit of the channel is evaluated depending on parameter XFITOIND. The CNC always evaluates the particular transfer inhibit of the axis.

XFITOIND	Meaning.
Yes	The CNC firsts evaluates the transfer inhibit of the channel (_XFERINH mark) and then the particular one of the axis (_IXFERINH(axis) mark).
No	The CNC only evaluates the particular transfer inhibit of the axis (_IXFERINH(axis) mark).

IRESET(axis)

For movements of an independent axis, if the PLC sets this mark to (=1), it interrupts the instruction in execution and eliminates the instructions pending execution.

For electronic-cam movements, it interrupts the cam synchronization movement switching to zero speed.

It sets the initial conditions at the independent interpolator of the axis.

IABORT(axis)

For movements of an independent axis, if the PLC sets this mark to (=1), the positioning block being executed (if any) stops, without eliminating also the rest of the positioning blocks pending execution.

It only affects positioning blocks; neither the pending instructions nor the synchronization movement are eliminated.

6.11 Tool manager modifiable signals.

SETTMEM

There is a mark for each magazine. The mnemonics for each magazine are the following.

SETTMEMZ1 (can also be programmed as SETTMEM)

SETTMEMZ2 SETTMEMZ3 SETTMEMZ4

The PLC sets this mark high (=1) to activate the tool manager emergency.

RESTMEM

There is a mark for each magazine. The mnemonics for each magazine are the following.

RESTMEMZ1 (can also be programmed as RESTMEM)

RESTMEMZ2 RESTMEMZ3 RESTMEMZ4

The PLC sets this mark high (=1) to cancel the tool manager emergency.

CUTTINGON

There is a mark for each channel. The mnemonics for each channel are the following.

CUTTINGONC1 (same as CUTTINGON)

CUTTINGONC2 CUTTINGONC3 CUTTINGONC4

When associating a maximum life span to a tool (monitoring), the CNC checks this mark to find out whether the tool is machining (=1) or not (=0).

```
PRG
()= CNCRD (G.GS0, R300, M12)
    Register R300 shows the status of the G functions.
AUTOMAT AND INCYCLE AND NOT B0R300 = CUTTINGON
    If it is in automatic mode (AUTOMAT), a (INCYCLE) block is being executed and the
    G00 function is not active, then, the tool is considered to be machining.
END
```

TREJECT

There is a mark for each channel. The mnemonics for each channel are the following.

TREJECTC1 (can also be programmed as TREJECT)

TREJECTC2 TREJECTC3 TREJECTC4

If the PLC sets this mark high (=1), the CNC interprets that the tool must be rejected.

MZTOCH1

There is a mark for each magazine. The mnemonics for each magazine are the following.

MZTOCH1MZ1 (can also be programmed as MZTOCH1)

MZTOCH1MZ2 MZTOCH1MZ3 MZTOCH1MZ4

Use it with an asynchronous magazine or synchronous with arm. The PLC must set this mark high (=1) after taking the tool from the magazine to the changer arm 1.

CH1TOSPDL

There is a mark for each magazine. The mnemonics for each magazine are the following.

CH1TOSPDLMZ1 (same as CH1TOSPDL)

CH1TOSPDLMZ2 CH1TOSPDLMZ3 CH1TOSPDLMZ4

Use it with an asynchronous magazine or synchronous with arm. The PLC must set this mark high (=1) after taking the tool from the changer arm 1 to the spindle.

SPDLTOCH1

There is a mark for each magazine. The mnemonics for each magazine are the following.

SPDLTOCH1MZ1 (same as SPDLTOCH1)

SPDLTOCH1MZ2 SPDLTOCH1MZ3 SPDLTOCH1MZ4

Use it with an asynchronous magazine with one changer arm. The PLC must set this mark high (=1) after taking the tool from the spindle to the changer arm 1.

SPDLTOCH2

There is a mark for each magazine. The mnemonics for each magazine are the following.

SPDLTOCH1MZ1 (same as SPDLTOCH1)

SPDLTOCH2MZ2 SPDLTOCH2MZ3 SPDLTOCH2MZ4

Use it with an asynchronous magazine or synchronous with arm. The PLC must set this mark high (=1) after taking the tool from the spindle to the changer arm 2.

CH1TOMZ

There is a mark for each magazine. The mnemonics for each magazine are the following.

CH1TOMZ1 (can also be programmed as CH1TOMZ)

CH1TOMZ2 CH1TOMZ3 CH1TOMZ4

Use it with an asynchronous magazine or synchronous with arm. The PLC must set this mark high (=1) after taking the tool from the tool changer arm 1 to the magazine.

CH2TOMZ

There is a mark for each magazine. The mnemonics for each magazine are the following.

CH2TOMZ1 (can also be programmed as CH2TOMZ)

CH2TOMZ2 CH2TOMZ3 CH2TOMZ4

Use it with an asynchronous magazine or synchronous with arm. The PLC must set this mark high (=1) after taking the tool from the tool changer arm 2 to the magazine.

SPDLTOGR

There is a mark for each channel. The mnemonics for each channel are the following.

SPDLTOGRC1 (can also be programmed as SPDLTOGR)

SPDLTOGRC2 SPDLTOGRC3 SPDLTOGRC4

Use it with a magazine that admits ground tools. The PLC must set this mark high (=1) after taking the tool from the spindle to ground.

GRTOSPDL

There is a mark for each channel. The mnemonics for each channel are the following.

GRTOSPDLC1 (can also be programmed as GRTOSPDL)

GRTOSPDLC2 GRTOSPDLC3 GRTOSPDLC4

Use it with a magazine that admits ground tools. The PLC must set this mark high (=1) after taking the tool from the ground to the spindle.

MZTOSPDL

There is a mark for each magazine. The mnemonics for each magazine are the following.

MZTOSPDLMZ1 (can also be programmed as MZTOSPDL)

MZTOSPDLMZ2 MZTOSPDLMZ3 MZTOSPDLMZ4

Use it with a synchronous magazine (without arm). The PLC must set this mark high (=1) after taking the tool from the magazine to the spindle.

SPDLTOMZ

There is a mark for each magazine. The mnemonics for each magazine are the following.

SPDLTOMZ1 (can also be programmed as SPDLTOMZ)

SPDLTOMZ2 SPDLTOMZ3 SPDLTOMZ4

Use it with a synchronous magazine (without arm). The PLC must set this mark high (=1) after taking the tool from the spindle to the magazine.

MZROT

There is a mark for each magazine. The mnemonics for each magazine are the following.

MZROTMZ1 (can also be programmed as MZROT)

MZROTMZ2 MZROTMZ3 MZROTMZ4

Use it with a turret type magazine. The PLC must set this mark high (=1) when the turret has rotated.

TCHANGEOK

There is a mark for each magazine. The mnemonics for each magazine are the following.

TCHANGEOKMZ1 (same as TCHANGEOK)

TCHANGEOKMZ2 TCHANGEOKMZ3 TCHANGEOKMZ4

The PLC must set this mark high (=1) when the tool change has ended (M06).

MZPOS

There is a register for each magazine. The mnemonics for each magazine are the following.

MZPOSMZ1 (can also be programmed as MZPOS)

MZPOSMZ2 MZPOSMZ3 MZPOSMZ4

The PLC must indicate the current magazine position in this register.

6.12 Keystroke modifiable signals.

KEYLED1 / KEYLED2

These registers control the LED's (lamps) of the keys of all the operator panels at the same time.

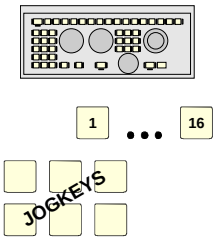
KEYLED1_1 / KEYLED2_1
..
KEYLED1_8 / KEYLED2_8

These registers control the LED's (lamps) of the keys of each operator panel. Registers KEYLED1_1 and KEYLED2_1 correspond to the first jog panel, KEYLED1_2 and KEYLED2_2 to the second one and so on.

The following instruction changes the status of the led of the first user key ()bit 0) every time the key is pressed.

```
DFU B0KEYBD1_2 = CPL B0KEYLED1_2
```

Registers KEYLED1 / KEYLED1_1 to KEYLED1_8. User keys.

Bit	Jog Panel
	
0	User key 1
1	User key 2
2	User key 3
3	User key 4
4	User key 5
5	User key 6
6	User key 7
7	User key 8
8	User key 9
9	User key 10
10	User key 11
11	User key 12
12	User key 13
13	User key 14
14	User key 15
15	User key 16

Registers KEYLED2 / KEYLED2_1 to KEYLED2_8.

Bit	Key	Bit	Key
0	Spindle override +	16	- - -
1	Spindle clockwise	17	- - -
2	Spindle positioning	18	- - -
3	Spindle stop	19	- - -
4	Spindle override -	20	- - -
5	Spindle counterclockwise	21	- - -
6	START	22	- - -
7	STOP	23	- - -
8	- - -	24	- - -
9	- - -	25	- - -
10	- - -	26	- - -
11	ZERO	27	- - -
12	- - -	28	- - -
13	Single block	29	- - -
14	- - -	30	- - -
15	RESET	31	- - -

KEYDIS1 / KEYDIS2 / KEYDIS3 / KEYDIS4

These registers inhibit (bit=1) the keys and the switches on all operator panels at the same time.

KEYDIS1_1 / KEYDIS2_1 / KEYDIS3_1 / KEYDIS4_1
..
KEYDIS1_8 / KEYDIS2_8 / KEYDIS3_8 / KEYDIS4_8

These registers inhibit (bit=1) the keys and the switches on each operator panel. Registers KEYDIS1_1 to KEYDIS4_1 correspond to the first jog panel, KEYDIS1_2 to KEYDIS4_2 to the second one and so on.

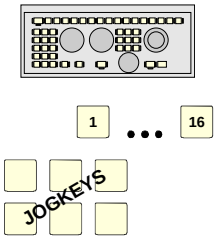
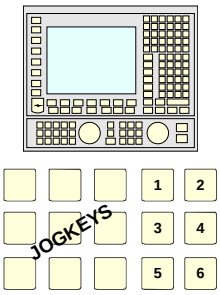
When selecting one of the inhibited positions of the feedrate override, the CNC will take the value of the lowest position allowed. If all of them are inhibited, it will take the value of 0%. For example, being allowed only the 110% and 120% positions, if the 50% position is selected, the CNC will take the value of 0%.

When selecting one of the inhibited positions of the spindle speed override, the CNC will take the value of the lowest position allowed. If all the positions are inhibited, the CNC keeps the active value.

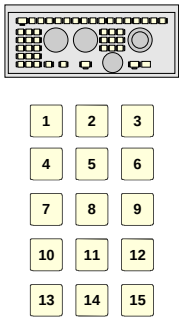
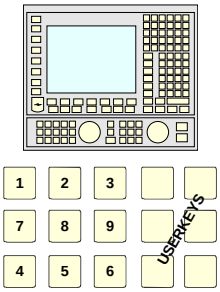
The following instruction inhibits the first jog key (bit 16) of the second keyboard.

```
( ) = B16KEYDIS1_2
```

Registers KEYDIS1 / KEYDIS1_1 to KEYDIS1_8. User keys.

Bit	Jog Panel	LCD-10K
		
0	User key 1	User key 1
1	User key 2	User key 2
2	User key 3	User key 3
3	User key 4	User key 4
4	User key 5	User key 5
5	User key 6	User key 6
6	User key 7	- - -
7	User key 8	- - -
8	User key 9	- - -
9	User key 10	- - -
10	User key 11	- - -
11	User key 12	- - -
12	User key 13	- - -
13	User key 14	- - -
14	User key 15	- - -
15	User key 16	- - -

Registers KEYDIS1 / KEYDIS1_1 to KEYDIS1_8. Jog keys.

Bit	Jog Panel	LCD-10K
		
16	Jog key 1	Jog key 1
17	Jog key 2	Jog key 2
18	Jog key 3	Jog key 3
19	Jog key 4	Jog key 4
20	Jog key 5	Jog key 5
21	Jog key 6	Jog key 6
22	Jog key 7	Jog key 7
23	Jog key 8	Jog key 8
24	Jog key 9	Jog key 9
25	Jog key 10	- - -
26	Jog key 11	- - -

Bit	Jog Panel	LCD-10K
27	Jog key 12	- - -
28	Jog key 13	- - -
29	Jog key 14	- - -
30	Jog key 15	- - -
31	- - - -	- - -

Registers KEYDIS2 / KEYDIS2_1 to KEYDIS2_8.

Bit	Key	Bit	Key
0	Spindle override +	16	- - -
1	Spindle clockwise	17	- - -
2	Spindle positioning	18	- - -
3	Spindle stop	19	- - -
4	Spindle override -	20	- - -
5	Spindle counterclockwise	21	- - -
6	START	22	- - -
7	STOP	23	- - -
8	- - -	24	- - -
9	- - -	25	- - -
10	- - -	26	- - -
11	ZERO	27	- - -
12	- - -	28	- - -
13	Single block	29	- - -
14	- - -	30	- - -
15	RESET	31	- - -

Registers KEYDIS3 / KEYDIS3_1 to KEYDIS3_8. Feedrate override and movement selector (handwheel, incremental jog or continuous jog).

Bit	Selector	Bit	Selector
0	0 %	16	140 %
1	2 %	17	150 %
2	4 %	18	160 %
3	10 %	19	170 %
4	20 %	20	180 %
5	30 %	21	190 %
6	40 %	22	200 %
7	50 %	23	Handwheel x100
8	60 %	24	Handwheel x10
9	70 %	25	Handwheel x1
10	80 %	26	Jog 1
11	90 %	27	Jog 10
12	100 %	28	Jog 100
13	110 %	29	Jog 1000
14	120 %	30	Jog 10000
15	130 %	31	Continuous jog

Registers KEYDIS4 / KEYDIS4_1 to KEYDIS4_8. Spindle speed override selector.

Bit	Selector	Bit	Selector
0	0 %	16	140 %
1	2 %	17	150 %
2	4 %	18	160 %
3	10 %	19	170 %
4	20 %	20	180 %
5	30 %	21	190 %
6	40 %	22	200 %
7	50 %	23	- - -
8	60 %	24	- - -
9	70 %	25	- - -
10	80 %	26	- - -
11	90 %	27	- - -
12	100 %	28	- - -
13	110 %	29	- - -
14	120 %	30	- - -
15	130 %	31	- - -

6.

LOGIC CNC INPUTS AND OUTPUTS.

Keystroke modifiable signals.

FAGOR

CNC 8060
CNC 8065

(REF: 1402)

TOOL AND MAGAZINE MANAGEMENT.

7

To properly configure the tool magazines and the tool change, you must:

- Set the machine parameters.
- Set the tool table and the tool magazine table.
- Write the PLC program.
- Program the subroutine associated with the tool and with the M06 function.

The machine parameters define the number of tool magazines available and their characteristics. Up to four magazines may be used and each may be of a different type. See *"7.1 Types of tool magazine."* on page 343.

When creating the PLC program and the subroutine associated with the tool and with the M06 function, you must bear in mind the number of magazines and channels available. For the communication between the tool manager and the PLC, each channel and each magazine has its own group of marks and registers.

The PLC maneuver will be different depending on the type of tool magazine. Later sections of this chapter show an example for each type of magazine.

About the magazines.

The CNC can have up to four magazines and each may be of a different type. Each magazine has its own configuration parameters.

The number of magazines is independent from the number of spindles and channels available. A magazine is not associated with any particular channel or spindle; i.e. a magazine may be shared by several channels and a channel can request tools from different magazines.



The access to the magazine depends on the mechanical configuration of the machine; in other words, to the physical possibility of the machine to access the magazines.

All the magazines can carry out tool changes simultaneously. However, one magazine can only be involved in a tool change process. If from one channel, one wishes to pick up or leave a tool in a magazine already involved in a tool change, the tool manager will wait for the tool change to be done before attending to the new request.

Two magazines may be involved in a tool change. The magazine receiving the tool and the magazine from where the new tool is picked up may be different.

About the tools.

Each tool is identified by its number, that is unique for the whole system; it cannot be repeated in different magazines nor in ground tools.

The list of available tools is saved in the tool table, the only one for the whole system. This table indicates the position and the magazine where each tool is located, whether it is a ground tool or the tool is active in a channel.

The tools are always stored in the same magazine. When doing a tool change, it is always stored in the magazine from which it was picked up.

Ground tools.

A ground tool is a tool that is not stored in any magazine and is loaded manually when requested. Ground tools are also defined in the tool table, but they are not associated with any magazine position.

Ground tool loading and unloading is global to the system; it is not associated with any particular magazine or channel.

Tool manager.

The CNC has a tool magazine management that knows at all times the location of each tool. When requesting a tool change or tool search, the tool manager "tells" the PLC the operations to be carried out.

- Take a tool from the magazine and insert it in the spindle.
- Leave the tool of the spindle in the magazine and take another one.
- Leave the spindle tool on the ground.
- Etc.

Depending on the type of magazine and the operation requested, several actions may be needed sometimes. For example, in some magazines, to take a tool from the spindle to the main magazine, the tool must be carried from the spindle to the changer arm and then from the changer arm to the magazine.

It is up to the PLC to control those movements. It must inform the manager about the actions carried out so it updates its information.

For the communication between the tool manager and the PLC, each channel and magazine has its own group of marks and registers. See *"7.3 Communication between manager and PLC."* on page 346.

Subroutines associated with the tool change.

There are two subroutines associated with the tool change.

Subroutines associated with the tool.

The subroutine associated with the tool is executed automatically every time a T function (tool selection) is executed.

There is one subroutine in each channel.

Subroutines associated with the M06 function.

The M06 function executes the tool change. The CNC will manage the tool change and update the table for the tool magazine.

It is recommended to set this function in the "M" function table so it executes the subroutine for the tool changer installed on the machine.

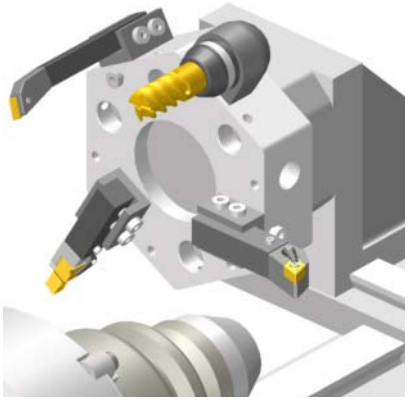
This subroutine is common to the whole system.

Within both subroutines, the "modal" condition of the canned cycles no longer has any influence. This characteristic is restored when the execution of the subroutine ends.

7.1 Types of tool magazine.

Tool magazines may be divided into 4 large groups:

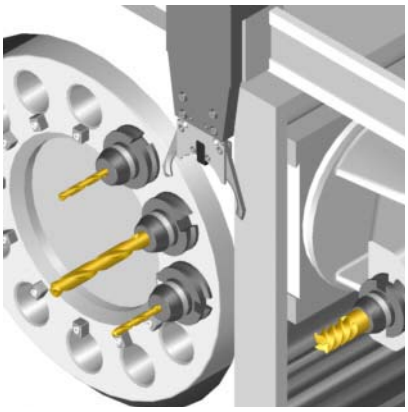
Turret type.



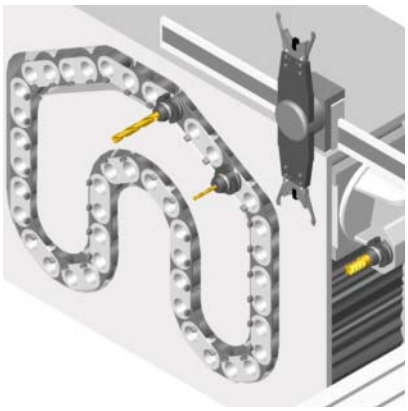
Synchronous magazine without changer arm.



Synchronous magazine with changer arm (1 or 2 holders). The arm cannot move while machining the part.



Asynchronous magazine. Tool changer arm of independent movements that may be moved while machining.



"Random" or "Non-Random" magazine

Depending on how the tools are stored in the magazine during a tool change, the magazine may be either random or non-random. In a random magazine, the tool may occupy any position whereas in a non-random magazine, the tools must always occupy the same position.

In any case and even when all the magazines are random, the tools are always stored in the same magazine from which they were picked up.

"Cyclic" or "Non-Cyclic" magazine

A cyclic magazine requires a tool change command M06 after searching a tool and before searching the next one. In a non-cyclic magazine, it is possible to perform several consecutive tool searches without necessarily having to change the tool.

7.2 Tool table, active tool table and tool magazine table

After setting the machine parameters for the tool magazines, define the tool table and then the magazine table.

Tool table.

The tool table must contain, among other things, the geometry, type of monitoring and tool size. All the tools must be defined, including the ground ones.

The tool manager inserts the special tools always in the same tool pocket regardless of the number of pockets they occupy.

Tool magazine table.

There is one table for each magazine. Each table shows which tool is in each pocket and on each holder of the changer arm (if any).

Although the magazine table may be initialized manually, it is up to the tool manager to dynamically update all its data.

Example:

Having 10 tools and a 10-position magazine. The tools are small, except T2 that is large and T4 that has a special size (0 to the left and 1 to the right).

Load the tools one by one in the magazine using the corresponding softkey.

- T1 in position 1.
- T2 in position 3. Verify that it cannot go in position 2 because the tool is too large.
- T3 in position 5. T2 occupies positions 2-3-4.
- T4 in position 6. T4 occupies positions 6-7.
- T5 in position 8.
- T6 in position 9.
- T7 in position 10.

T8, T9 and T10 do not fit in the magazine; therefore, they will be ground tools.

Active-tools table.

The active-tools table shows which tools are active in the spindles.

7.3 Communication between manager and PLC.

For the communication between the tool manager and the PLC, each channel and magazine has its own group of marks and registers.

The CNC maintains compatibility with the marks and registers of previous versions whose mnemonics do not refer to any particular channel or magazine. From software version V2.00 on, these signals refer to the first channel or magazine accordingly.

The communication between the manager and the PLC takes place in two stages; when executing the T function and when executing the M06 function.

- When executing the T function, the CNC lets the tool manager know about it.

  The tool manager sends a command to the PLC to select the next tool in the magazine (if possible).

  The CNC continues with the execution of the program without waiting for the tool manager to complete the operation.

- When executing the M06 function, its associated subroutine is called upon. The M06 function must also be programmed inside that subroutine so the CNC "tells" the tool manager to start making the tool change.

  The tool manager sends a command to the PLC to make the change.

  The CNC waits for the tool manager to complete the operation before continuing with the execution of the program.

Considerations and recommendations.

The management of the tool change should be included in the subroutine associated with the M06 and leave the control of the external devices up to the PLC.

Use the auxiliary functions to govern the various devices (magazine rotation, magazine movement, tool changer arm, etc.) from the M06 subroutine.

On asynchronous magazines (changer arm with independent movements) when the change implies leaving a tool in the magazine, the TCHANGEOK mark may be activated so the CNC goes on executing the program while the tool is being taken to the magazine.

7.3.1 Manager --> PLC communication.

The manager uses the following registers and marks to inform the PLC about the operations it must carry out. Some signals are per channel whereas others are per magazine. The following table shows the mnemonics for each mark (M) or register (R) in each channel or magazine.

	Channel ·1·	**Channel ·2·**	**Channel ·3·**	**Channel ·4·**
M	TMOPSTROBE TMOPSTROBEC1	TMOPSTROBEC2	TMOPSTROBEC3	TMOPSTROBEC4
R	TMOPERATION TMOPERATIONC1	TMOPERATIONC2	TMOPERATIONC3	TMOPERATIONC4
R	MZIDC1	MZIDC2	MZIDC3	MZIDC4

	Magazine ·1·	**Magazine ·2·**	**Magazine ·3·**	**Magazine ·4·**
R	LEAVEPOS LEAVEPOSMZ1	LEAVEPOSMZ2	LEAVEPOSMZ3	LEAVEPOSMZ4
R	TAKEPOS TAKEPOSMZ1	TAKEPOSMZ2	TAKEPOSMZ3	TAKEPOSMZ4
R	NEXTPOS NEXTPOSMZ1	NEXTPOSMZ2	NEXTPOSMZ3	NEXTPOSMZ4

TMOPSTROBE

The channel tool manager sets this mark high (=1) to let the PLC know that it must execute the operation indicated by the TMOPERATION mark of the channel.

LEAVEPOS

This register indicates the magazine position to leave the tool. While selecting a turret position (#ROTATEMZ instruction), this register takes the value of ·0· if it is a positive relative positioning and the value of ·1· if it is a negative relative positioning.

TAKEPOS

This register indicates the magazine position of the tool to be taken. While selecting a turret position (#ROTATEMZ instruction), this register takes the value of ·0· if it is an absolute positioning and the value of ·1· if it is a relative positioning.

NEXTPOS

This register indicates the magazine position occupied by the next tool. While selecting a turret position (#ROTATEMZ instruction); in an absolute positioning, this register indicates the position to reach and, in a relative positioning, the number of positions to rotate.

MZID

This register indicates which magazine contains the tool requested by the channel.

When two magazines are involved in a tool change, the lower portion of this register indicates the destination magazine for the tool and the higher portion the source magazine for the tool.

TMOPERATION

This register indicates the type of operation to be carried out by the tool manager.

7.3.2 PLC --> Manager communication.

The PLC uses the following marks to inform the manager about the operations that it has carried out. There is one group of marks for each magazine.

The PLC, depending on the type of magazine, must take some actions to execute the operations requested by the manager. After ending each one of them, it must activate certain marks to inform the manager that the action has been completed. The manager sets them back to "0" once they are read.

The following table shows the mnemonics for each mark (M) or register (R) in each channel or magazine.

	Magazine ·1·	Magazine ·2·	Magazine ·3·	Magazine ·4·
M	MZTOCH1 MZTOCH1MZ1	MZTOCH1MZ2	MZTOCH1MZ3	MZTOCH1MZ4
M	CH1TOSPDL CH1TOSPDLMZ1	CH1TOSPDLMZ2	CH1TOSPDLMZ3	CH1TOSPDLMZ4
M	SPDLTOCH1 SPDLTOCH1MZ1	SPDLTOCH1MZ2	SPDLTOCH1MZ3	SPDLTOCH1MZ4
M	SPDLTOCH2 SPDLTOCH2MZ1	SPDLTOCH2MZ2	SPDLTOCH2MZ3	SPDLTOCH2MZ4
M	CH1TOMZ CH1TOMZ1	CH1TOMZ2	CH1TOMZ3	CH1TOMZ4
M	CH2TOMZ CH2TOMZ1	CH2TOMZ2	CH2TOMZ3	CH2TOMZ4
M	SPDLTOGR SPDLTOGRC1	SPDLTOGRC2	SPDLTOGRC3	SPDLTOGRC4
M	GRTOSPDL GRTOSPDLC1	GRTOSPDLC2	GRTOSPDLC3	GRTOSPDLC4
M	MZTOSPDL MZTOSPDLMZ1	MZTOSPDLMZ2	MZTOSPDLMZ3	MZTOSPDLMZ4
M	MZTOSPDL MZTOSPDLMZ1	MZTOSPDLMZ2	MZTOSPDLMZ3	MZTOSPDLMZ4
M	MZROT MZROTMZ1	MZROTMZ2	MZROTMZ3	MZROTMZ4
M	TCHANGEOK TCHANGEOKMZ1	TCHANGEOKMZ2	TCHANGEOKMZ3	TCHANGEOKMZ4
R	MZPOS MZPOSMZ1	MZPOSMZ2	MZPOSMZ3	MZPOSMZ4

MZTOCH1

Use it with an asynchronous magazine or synchronous with arm. The PLC must set this mark high (=1) after taking the tool from the magazine to the changer arm 1.

CH1TOSPDL

Use it with an asynchronous magazine or synchronous with arm. The PLC must set this mark high (=1) after taking the tool from the changer arm 1 to the spindle.

SPDLTOCH1

Use it with an asynchronous magazine with one changer arm. The PLC must set this mark high (=1) after taking the tool from the spindle to the changer arm 1.

SPDLTOCH2

Use it with an asynchronous magazine or synchronous with arm. The PLC must set this mark high (=1) after taking the tool from the spindle to the changer arm 2.

CH1TOMZ

Use it with an asynchronous magazine or synchronous with arm. The PLC must set this mark high (=1) after taking the tool from the tool changer arm 1 to the magazine.

CH2TOMZ

Use it with an asynchronous magazine or synchronous with arm. The PLC must set this mark high (=1) after taking the tool from the tool changer arm 2 to the magazine.

SPDLTOGR

Use it with a magazine that admits ground tools. The PLC must set this mark high (=1) after taking the tool from the spindle to ground.

GRTOSPDL

Use it with a magazine that admits ground tools. The PLC must set this mark high (=1) after taking the tool from the ground to the spindle.

MZTOSPDL

Use it with a synchronous magazine (without arm). The PLC must set this mark high (=1) after taking the tool from the magazine to the spindle.

SPDLTOMZ

Use it with a synchronous magazine (without arm). The PLC must set this mark high (=1) after taking the tool from the spindle to the magazine.

MZROT

Use it with a turret-type magazine and with a synchronous magazine.

The PLC must set this mark high (=1) when the turret has rotated. In the synchronous magazine, it is used to optimize the change by orienting the magazine while machining. The PLC must set this mark to (=1) to indicate that the operation has been completed, whether it has been oriented or not.

TCHANGEOK

On asynchronous magazines (with changer arm of independent movements) the following mark may be activated so the CNC goes on executing the program while the tool is being taken to the magazine.

The PLC must set this mark high (=1) to "tell" the manager to go on executing the program while the tool is being taken to the magazine.

MZPOS

On random magazines, magazine orientations may be optimized if the manager knows the position selected at all times. The PLC must indicate in this register the current position of the magazine; when not using this register, the PLC must set it to ·0·.

On turret magazines, it is possible to select a position (#ROTATEMZ instruction). The PLC must indicate in this register the current position of the magazine; if the register has a value of ·0·, the PLC will issue the corresponding error.

7.3.3 Manager Emergency.

The manager may be set in an Emergency state when a malfunction occurs (the PLC has executed the wrong action, incomplete tool change, etc.) or if the PLC activates the emergency.

The PLC marks related to manager emergency are the following: There is one group of marks for each magazine.

The following table shows the mnemonics for each mark (M) in each channel or magazine.

	Magazine ·1·	Magazine ·2·	Magazine ·3·	Magazine ·4·
M	SETTMEM SETTMEMZ1	SETTMEMZ2	SETTMEMZ3	SETTMEMZ4
M	RESTMEM RESTMEMZ1	RESTMEMZ2	RESTMEMZ3	RESTMEMZ4
M	TMINEM TMINEMZ1	TMINEMZ2	TMINEMZ3	TMINEMZ4

SETTMEM

The PLC sets this mark high (=1) to activate the tool manager emergency.

RESTMEM

The PLC sets this mark high (=1) to cancel the tool manager emergency.

TMINEM

The CNC sets this mark high (=1) to inform the PLC that an emergency has occurred at the tool manager.

To generate an emergency at the manager from the PLC:

1 Activate the emergency.

DFU (error condition) = SET SETTMEM

2 Make sure that the emergency has occurred before canceling it.

TMINEM AND DFU (removal condition) = SET RESTMEM

The manager sets the SETMEM and RESTMEM signals low (=0).

7.3.4 Tool monitoring.

The PLC marks related to tool monitoring are the following. There is one group of marks for each channel.

The following table shows the mnemonics for each mark (M) in each channel or magazine.

	Channel ·1·	Channel ·2·	Channel ·3·	Channel ·4·
M	CUTTINGON CUTTINGONC1	CUTTINGONC2	CUTTINGONC3	CUTTINGONC4
M	TREJECT TREJECTC1	TREJECTC2	TREJECTC3	TREJECTC4
M	TWORNOUT TWORNOUTC1	TWORNOUTC2	TWORNOUTC3	TWORNOUTC4

The PLC marks related to tool monitoring are:

CUTTINGON

When a tool is assigned a maximum life span (monitoring), the CNC checks this mark in order to know whether the tool is machining (=1) or not (=0).

It is usually considered to be machining when the following conditions are met:

- The spindle is turning (M3 or M4) or when the tool is threadcutting and the 0% of feedrate is not selected.
- The automatic operating mode is selected, there is a block in execution and the G00 function is not active.
- The execution is not interrupted.

TREJECT

If the PLC sets this mark high (=1), the CNC interprets that the tool must be rejected.

TWORNOUT

The CNC sets this mark high (=1) to "tell" the PLC that the tool has been rejected because it is worn out (real life > maximum life span).

7.4 Variables related to tool magazine management.

The variables associated with the magazine that are involved in the tool magazine management are the following. There is one group of variables for each channel. Replace the [n] character with the channel number, maintaining the brackets.

V.[n].TM.MZMODE

Operating mode of the tool magazine manager.

With a ·0· value	Normal mode (by default and after reset).
With a ·1· value	Magazine loading mode.
With a ·2· value	Magazine unloading mode.

V.[n].TM.MZSTATUS

Tool manager status.

With a ·0· value	Normal.
With a ·1· value	An error has occurred.
With a ·2· value	An error has occurred; but it will wait for the current maneuver to be completed.
With a ·4· value	Emergency.

V.[n].TM.MZRUN

Tool manager running.

With a ·0· value	There is no sequence in execution.
With a ·1· value	There is a sequence in execution.

V.[n].TM.MZWAIT

Tool manager executing a maneuver. It indicates whether to wait or not for the maneuver to end.

With a ·0· value	No need to wait.
With a ·1· value	It has to wait.

There is no need to program it in the subroutine associated with M06. The subroutine itself waits for the manager's maneuvers to finish. This way, block preparation is not interrupted.

7.5 Tool loading and unloading from the magazines.

Tool loading and unloading from the magazine.

Each magazine table has softkeys for initializing, loading and unloading tools in the magazine either manually or automatically. Refer to the operating manual.

The tools may also be loaded and unloaded in the magazine by program or in MDI mode.

Tool loading and unloading from the spindle.

The tools must always be loaded and unloaded in the spindle and on the changer arm in manual mode from the magazine table. Refer to the operating manual.

Loading the magazine by program or in MDI mode.

The tools are loaded in the magazine by taking them one by one from ground to the magazine going through the spindle.

Set the variable: V.TM.MZMODE=1 to "tell" the tool manager that the magazine loading mode has be chosen.

If a T1 M6 is executed next, the tool manager interprets that T1 must be taken from ground to the magazine going through the spindle and it will let the PLC know by setting TMOPERATION=9.

The subroutine associated with the M06 function and the PLC program must contain the maneuver needed to make the tool change.

After loading the tool, the tool manager updates the magazine table.

Unloading the magazine by program or MDI.

The tools are unloaded from the magazine taking them one by one from the magazine to the ground going through the spindle.

Set variable: V.TM.MZMODE=2 to "tell" the tool manager that the magazine unloading mode has been chosen.

If a T1 M6 is executed next, the tool manager interprets that T1 must be taken from magazine to the ground going through the spindle and it will let the PLC know by setting TMOPERATION=10.

The subroutine associated with the M06 function and the PLC program must contain the maneuver needed to make the tool change.

After unloading the tool, the tool manager updates the magazine table.

Placing a tool in the spindle by program or in MDI mode

Set variable: V.TM.MZMODE=0 to "tell" the tool manager that the normal mode has been chosen.

If T1 M6 is executed next, the tool manager checks whether there is already a tool in the spindle (to remove it first) and whether the requested tool is in the magazine or it has to be picked up from the ground. In any case, it lets the PLC know which operation it must carry out by setting TMOPERATION to the proper value.

The subroutine associated with the M06 function and the PLC program must contain the maneuver needed to make the tool change.

After placing the tool, the tool manager updates the magazine table.

7.6 Magazine-less system.

When there is no tool magazine, changing the tool only requires programming the T function without the need for M6. In this circumstances, a programmed M6 function is not interpreted as a tool change, but as just another M function without any special meaning. This does not prevent it from having an associated subroutine like any other M function.

It will be synchronized with the tool manager after the tool change (MZWAIT) after managing the T block in execution.

Execution of the T function.

1. When the CNC executes the T function, it lets the tool manager know about it.
2. The tool manager sends a command to the PLC to make the change.
3. The CNC waits for the tool manager to complete the operation before continuing with the execution of the program.

Behavior of function M06.

When there is no tool magazine, the M6 function behaves as follows.

- The M6 function will not activate PLC marks (e.g. DM06) nor will it execute anything related to the tool change (like the tool change subroutine, etc.).
- The M6 function will not be displayed in the M function history.
- The subroutine associated with the M6 function (if it has one) will execute modal canned cycles or modal subroutines with motion.
- The subroutine associated with M6 will have no special treatment when called upon from hidden files, canned cycles, etc.

7.6.1 Valid operations and marks activated by the PLC with each one of them.

Valid operations of the tool manager for this magazine.

The possible operations in this type of magazine are:

TMOPERATION	Meaning.
0	Do nothing.
3	Insert a ground tool in the spindle.
4	Leave the spindle tool on the ground.
8	Leave the spindle tool on the ground and take a tool from ground.

Values of registers TAKEPOS and LEAVEPOS used by the manager.

The TAKEPOS and LEAVEPOS signals do not assume any value. Here is a summary table of the marks that must be activated by the PLC at the end of each operation.

TM => PLC			PLC => TM
TMOPERATION	TAKEPOS	LEAVEPOS	
3	0	0	GRTOSPDL
4	0	0	SPDLTOGR
8	0	0	SPDLTOGR + GRTOSPDL

Example of operation.

The following example shows, assuming that there is no tool in the spindle, it shows the functions executed by the CNC, the values sent by the tool manager to the PLC in each operation and the marks to be activated by the PLC in each case.

It is a non-random magazine, each tool occupies the position of its own number and T7 and T8 are ground tools.

CNC	TMOPERATION	TAKEPOS	LEAVEPOS	
T7	3	0	0	GRTOSPDL
T8	8	0	0	SPDLTOGR + GRTOSPDL
T0	4	0	0	SPDLTOGR

7.6.2 Detailed description of the operations of the magazine.

Here is a sample of a detailed description of the valid operations for this magazine. In a system without magazines, the TAKEPOS, LEAVEPOS, NEXTPOS and MZID signals are not involved.

TMOPERATION = 3

Insert a ground tool in the spindle.

The sequence of this operation is the following.

1 When the operation is completed, activate the mark: GRTOSPDL.

TMOPERATION = 4

Leave the spindle tool on the ground.

The sequence of this operation is the following.

1 When the operation is completed, activate the mark: SPDLTOGR.

TMOPERATION = 8

Leave the spindle tool on the ground and take a tool from ground.

The sequence of this operation is the following.

1 Leave the tool of the spindle on the ground and activate the mark SPDLTOGR.

2 Leave the ground tool in the spindle and activate the mark GRTOSPDL.

7.6.3 Basic PLC programming.

When executing the T function.

When executing the T function, the tool manager sends to the PLC, in the TMOPERATION register, the code for the operation to be carried out.

DFU TMOPSTROBE AND (CPS TMOPERATION EQ 3) = SET GRTOSPDL

A tool change has been requested and TMOPERATION=3. The PLC has taken the tool from ground to the spindle.

DFU TMOPSTROBE AND (CPS TMOPERATION EQ 4) = SET SPDLTOGR

A tool change has been requested and TMOPERATION=4. The PLC has taken the tool from the spindle to ground.

DFU TMOPSTROBE AND (CPS TMOPERATION EQ 8) = SET SPDLTOGR

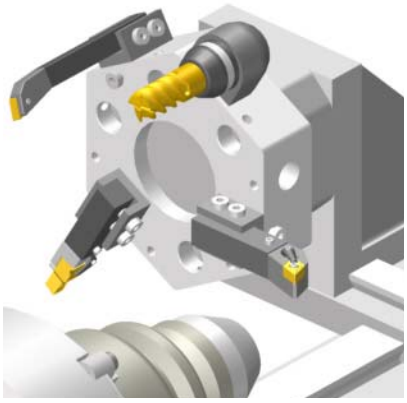
A tool change has been requested and TMOPERATION=4. The PLC has taken the tool from the spindle to ground.

DFD SPDLTOGR AND (CPS TMOPERATION EQ 8) = SET GRTOSPDL

The manager has canceled the SPDLTOGR mark and TMOPERATION=8. The PLC has taken the tool from ground to the spindle.

7.7 Turret type magazine.

It is a typical magazine for lathes. To make the change, the magazine rotates until it places the new tool in the work position. The tool cannot be changed while the part is being machined.



Usually, in this type of magazine, the communication between the manager and the PLC is configured so it is carried out in a single stage, with the execution of the T function.

When selecting a tool, the subroutine associated with the T executes the M06 function (tool change), machining stops and the tool manager sends to the PLC the code for the operation to be carried out.

1. When the CNC executes the T function, it lets the tool manager know about it and calls the associated subroutine. The M06 function must be programmed inside that subroutine so the CNC "tells" the tool manager to start making the tool change.
2. The tool manager sends a command to the PLC to select the next tool in the magazine.
3. The subroutine associated with the T function executes the M06 function. The M06 subroutine can also have an associated subroutine. If a subroutine associated with the M06 function is defined, the M06 function must also be programmed inside that subroutine so the CNC "tells" the tool manager to start making the tool change.
4. The tool manager sends a command to the PLC to make the change.
5. The CNC waits for the tool manager to complete the operation before continuing with the execution of the program.

The management of the tool change should be included in the subroutine associated with the M06 and leave the control of the external devices up to the PLC. Use the auxiliary functions to govern the various devices (magazine rotation, magazine movement, tool changer arm, etc.) from the M06 subroutine.

Ground tools in a turret type magazine.

This magazine offers the possibility of working with ground tools. If there is a tool in the work position and another one is requested that is not in the turret, the CNC considers it to be a ground tool.

The maneuver for loading/unloading a tool in the spindle (in the work position) either directly or going through the spindle is the same. In the latter case, the variable V.TM.MZMODE must have the proper value to load or unload.

7.7.1 Valid operations and marks activated by the PLC with each one of them.

Valid operations of the tool manager for this magazine.

The possible operations in this type of magazine are:

TMOPERATION	Meaning.
0	Do nothing.
1	Assume the current tool as active tool.
3	Insert a ground tool in the spindle.
4	Leave the spindle tool on the ground.
9	Take a ground tool to the magazine going through the spindle.
10	Take a tool from the magazine and leave on the ground going through the spindle.
11	Orient the turret.
15	Select a turret position.

Values of registers TAKEPOS and LEAVEPOS used by the manager.

The TAKEPOS and LEAVEPOS signals can assume the following values.

Value	Meaning
0	Do nothing.
#	Magazine position.
-4	Ground tool.

If TMOPERATION = 15, it is a special case wehre the TAKEPOS, LEAVEPOS and NEXTPOS signals can assume the following values.

Signal.	Meaning.
TAKEPOS	This register takes the value of ·0· if it is an absolute positioning and the value of ·1· if it is a relative positioning.
LEAVEPOS	This register takes the value of ·0· if it is a positive relative positioning and the value of ·1· if it is a negative relative positioning.
NEXTPOS	In an absolute positioning, this register indicates the position to reach and, in a relative positioning, the number of positions to rotate.

Here is a summary table of the TAKEPOS and LEAVEPOS values used by the manager in each operation as well as the marks that must be activated by the PLC at the end of each operation.

TM => PLC			PLC => TM
TMOPERATION	TAKEPOS	LEAVEPOS	
1	#	0	MZTOSPDL
3	0	0	GRTOSPDL
4	0	0	SPDLTOGR
9	-4	#	GRTOSPDL
10	#	-4	SPDLTOGR
11	0	0	MZROT
15	- - -	- - -	MZROT

Example of operation.

The following example shows, assuming that there is no tool in the spindle, it shows the functions executed by the CNC, the values sent by the tool manager to the PLC in each operation and the marks to be activated by the PLC in each case.

It is a non-random magazine, each tool occupies the position of its own number and T7 is a ground tool.

CNC	TMOPERATION	TAKEPOS	LEAVEPOS	
T1 M6	11 1	0 1	0 0	MZROT MZTOSPDL
T2 M6	11 1	0 2	0 0	MZROT MZTOSPDL
T7 M6	11 3	0 0	0 0	 GRTOSPDL

7.7.2 Detailed description of the operations of the magazine.

Here is a sample of a detailed description of the valid operations for this magazine. For each operation, it indicates which of the TAKEPOS, LEAVEPOS, NEXTPOS and MZID signals are involved and their meanings. It also shows the sequence to follow to complete the operation.

The TAKEPOS, LEAVEPOS and NEXTPOS signals can assume the following values.

Value	Meaning
0	Do nothing.
#	Magazine position.
-4	Ground tool.

TMOPERATION = 1

Assume the current tool as active tool.

TAKEPOS Magazine position to pick up the tool.

The sequence of this operation is the following.

1 After ending this operation, activate the MZTOSPDL mark.

TMOPERATION = 3

Insert a ground tool in the spindle.

The sequence of this operation is the following.

1 When the operation is completed, activate the mark: GRTOSPDL.

TMOPERATION = 4

Leave the spindle tool on the ground.

The sequence of this operation is the following.

1 When the operation is completed, activate the mark: SPDLTOGR.

TMOPERATION = 9

Take a ground tool to the magazine going through the spindle.

TAKEPOS Ground tool.

LEAVEPOS Magazine position to leave the tool.

The sequence of this operation is the following.

1 Leave the ground tool in the spindle and activate the mark GRTOSPDL.

TMOPERATION = 10

Take a tool from the magazine and leave on the ground going through the spindle.

TAKEPOS Magazine position to pick up the tool.

LEAVEPOS Ground tool.

The sequence of this operation is the following.

1 Leave the tool of the spindle on the ground and activate the mark SPDLTOGR.

TMOPERATION = 11

Orient the magazine.

The sequence of this operation is the following.

1 When the operation is completed, activate the mark MZROT.

TMOPERATION = 15

Select a magazine position.

TAKEPOS	This register takes the value of ·0· if it is an absolute positioning and the value of ·1· if it is a relative positioning.
LEAVEPOS	This register takes the value of ·0· if it is a positive relative positioning and the value of ·1· if it is a negative relative positioning.
NEXTPOS	In an absolute positioning, this register indicates the position to reach and, in a relative positioning, the number of positions to rotate.

The sequence of this operation is the following.

1 When the operation is completed, activate the mark MZROT.

7.7.3 Communication between the PLC and the M06 subroutine.

The communication between the PLC and the M06 subroutine takes place using a series of generic marks and registers. The program of the M06 subroutine offered as an example uses the following marks and registers.

Communication between the PLC and the M06 subroutine.

Registers used by the PLC to transfer information to the M06 subroutine:

R101 Type of operation requested by the tool manager (value of TMOPERATION).

Communication between the PLC and the M06 subroutine

Marks activated by the M06 subroutine so the PLC activates the relevant mark of the tool manager.

M1107 SPDLTOGR

The tool has been taken from the spindle to ground.

M1108 GRTOSPDL

The tool has been taken from ground to the spindle.

M1109 MZTOSPDL

Assume the current tool as active tool.

M1110 SPDLTOGR

Assume the current tool as active tool.

M1111 MZROT

Mark that indicates that the magazine has rotated.

M functions at the PLC

M functions that imply movements controlled by the PLC:

M109 Select in the magazine the position indicated by TAKEPOS and insert the tool in the spindle.

M110 Select in the magazine the position indicated by LEAVEPOS and leave the spindle tool.

Set all the M functions with "before-before" synchronization to the program continues after the M function is completed.

7.7.4 Program of the M06 subroutine.

%L SUB_MZ_ROT

V.PLC.M[1111]=1

MZROT mark to the tool manager.

#RET

%L SUB_SPD_TO_GR

Remove the tool from the spindle (take it to ground).

It shows the message for the operator to extract the tool and waits for the operation to be completed before removing the message.

#MSG ["Extract tool T%D and press START", V.TM.TOOL]

M0

#MSG [""]

V.PLC.M[1107]=1

SPDLTOGR mark to the tool manager.

#RET

%L SUB_GR_TO_SPD

Insert the ground tool in the spindle.

It shows the message for the operator to insert the tool and waits for the operation to be completed before removing the message.

#MSG ["Insert tool T%D and press START", V.TM.NXTOOL]

M0

#MSG [""]

V.PLC.M[1108]=1

GRTOSPDL mark to the tool manager.

#RET

%L SUB_MZ_TO_SPD

Take the tool from the magazine and insert it in the spindle.

M109

Auxiliary function to execute an action.

V.PLC.M[1109]=1

MZTOSPDL mark to the tool manager.

#RET

%L SUB_SPD_TO_MZ

Leave the spindle tool in the magazine.

M110

Auxiliary function to execute an action.

V.PLC.M[1110]=1

SPDLTOGR mark to the tool manager.

#RET

%L SUB_SPD_GMCHG

Move the spindle to the manual tool change point.

G1 Z_ F_

Move the spindle.

#RET

%L SUB_SPD_AUTCHG

Move the spindle to the automatic tool change point

G1 Z_ F_

Move the spindle.

#RET

%SUB_M6.nc

M6

Order the tool manager to start the tool change.

$SWITCH V.PLC.R[101]

It analyzes the type of operation.

$CASE 1

Assume the current tool as active tool.

LL SUB_MZ_TO_SPD

$BREAK

$CASE 3

Insert the ground tool in the spindle.

LL SUB_SPD_GMCHG

Move the spindle to the manual tool change point.

LL SUB_GR_TO_SPD

Insert the ground tool in the spindle.

$BREAK

$CASE 4

Leave the spindle tool on the ground.

LL SUB_SPD_GMCHG

Move the spindle to the manual tool change point.

LL SUB_SPD_TO_GR

Remove the tool from the spindle.

$BREAK

$CASE 9

Take a ground tool to the magazine going through the spindle.

LL SUB_SPD_AUTCHG

Move the spindle to the automatic tool change point

LL SUB_SPD_TO_MZ

Leave the spindle tool in the magazine.

$BREAK

$CASE 10

Take a tool from the magazine and leave on the ground going through the spindle.

LL SUB_SPD_AUTCHG

Move the spindle to the automatic tool change point

LL SUB_MZ_TO_SPD

Take the tool from the magazine to the spindle.

$BREAK

$CASE 11

Orient the turret.

LL SUB_SPD_AUTCHG

Move the spindle to the automatic tool change point

LL SUB_MZ_ROT

$BREAK

$ENDSWITCH

End of analysis of type of operation.

$WHILE V.TM.MZWAIT == 1

$ENDWHILE

Wait for the tool manager.

#RET

End of M06 subroutine.

7.7.5 Basic PLC programming.

When executing the -T- function

When executing a T function, the tool manager sends the code TMOPERATION=11 to the PLC. In general, it is an optimization of the change that permits orienting the magazine while machining.

```
DFU TMOPSTROBE AND CPS TMOPERATION EQ 11 = SET MZROT
```

Activate the MZROT mark to "tell" the tool manager that the operation has finished.

When executing the M06 function

When executing the M06 function, the tool manager sends to the PLC, in the TMOPERATION register, the code for the operation to be carried out.

```
DFU TMOPSTROBE = MOV TMOPERATION R101
```

This instruction transfers the TMOPERATION value to register R101 so it is managed by the M06 subroutine.

Every time the M06 subroutine ends an action, it lets the PLC know so it activates the relevant mark of the tool manager.

```
DFU M1107 = SET SPDLTOGR
DFD SPDLTOGR = RES M1107
```

The tool has been taken from the spindle to ground.

```
DFU M1108 = SET GRTOSPDL
DFD GRTOSPDL = RES M1108
```

The tool has been taken from ground to the spindle.

```
DFU M1109 = SET MZTOSPDL
DFD MZTOSPDL = RES M1109
```

The tool has been taken from the magazine to the spindle.

```
DFU M1110 = SET SPDLTOGR
DFD SPDLTOGR = RES M1110
```

The tool has been taken from the spindle to the magazine.

```
DFU M1111 = SET MZROT
DFD MZROT = RES M1111
```

The magazine has rotated already.

The M06 subroutine uses the following M functions to "tell" the PLC which movements it must carry out.

M109	Select in the magazine the position indicated by TAKEPOS and insert the tool in the spindle.
M110	Select in the magazine the position indicated by LEAVEPOS and leave the spindle tool.

Programming it depends on the type of machine. The auxiliary function will conclude after executing the requested movement.

Certain operations require using the information transferred by the tool manager in the following registers:

LEAVEPOS	This register indicates the magazine position to leave the tool.
TAKEPOS	This register indicates the magazine position of the tool to be taken.

Manager emergency signal.

Treatment of the tool manager emergency signal.

```
DFU B11KEYBD1 AND NOT TMINEM = SET SETTMEM
DFU TMINEM = RES SETTMEM
```

Pressing the USER12 key activates the emergency.

```
TMINEM = B11KEYLED1
```

The lamp of the USER12 key turns on when there is an emergency.

```
TMINEM AND DFU B12KEYBD1 = SET RESTMEM
```

Pressing the USER13 key removes the emergency.

7.8 Synchronous magazine without changer arm.

In a synchronous tool changer without arm, the magazine must move up to the spindle to change the tool. The tool cannot be changed while the part is being machined.



The communication between the manager and the PLC takes place in two stages; the first one when executing the T function and the second one when executing the M06 function.

Execution of the T function.

1. When the CNC executes the T function, it lets the tool manager know about it.
2. The tool manager sends a command to the PLC to select the next tool in the magazine (if possible).
3. The CNC continues with the execution of the program without waiting for the tool manager to complete the operation.

Execution of the M06 function.

1. When the CNC executes the M06 function, it calls the associated subroutine. The M06 function must also be programmed inside that subroutine so the CNC "tells" the tool manager to start making the tool change.
2. The tool manager sends a command to the PLC to make the change.
3. The CNC waits for the tool manager to complete the operation before continuing with the execution of the program.

The management of the tool change should be included in the subroutine associated with the M06 and leave the control of the external devices up to the PLC. Use the auxiliary functions to govern the various devices (magazine rotation, magazine movement, etc.) from the M06 subroutine.

7.8.1 Valid operations and marks activated by the PLC with each one of them.

Valid operations of the tool manager for this magazine.

The possible operations in this type of magazine are:

TMOPERATION	Meaning.
0	Do nothing.
1	Take a tool from the magazine and insert it in the spindle.
2	Leave the spindle tool in the magazine.
3	Insert a ground tool in the spindle.
4	Leave the spindle tool on the ground.
5	Leave the spindle tool in the magazine and take a tool from the same magazine.
6	Leave the spindle tool in the magazine and take a tool from ground.
7	Leave the spindle tool on the ground and take a tool from the magazine.
8	Leave the spindle tool on the ground and take a tool from ground.
9	Take a ground tool to the magazine going through the spindle.
10	Take a tool from the magazine and leave on the ground going through the spindle.
11	Orient the magazine.
12	Leave the spindle tool in the magazine and take a tool from the same magazine (as TMOPERATION=5). It is an optimized operation, only valid when the magazine is random and the tool is special.
13	Orienting two magazines.
14	Leave the spindle tool in the magazine and take a tool from another magazine.

Values of registers TAKEPOS and LEAVEPOS used by the manager.

The TAKEPOS and LEAVEPOS signals can assume the following values.

Value	Meaning
0	Do nothing.
#	Magazine position.
-4	Ground tool.

Here is a summary table of the TAKEPOS and LEAVEPOS values used by the manager in each operation as well as the marks that must be activated by the PLC at the end of each operation.

TM => PLC			PLC => TM
TMOPERATION	TAKEPOS	LEAVEPOS	
1	#	0	MZTOSPDL
2	0	#	SPDLTOMZ
3	0	0	GRTOSPDL
4	0	0	SPDLTOGR
5	#	#	SPDLTOMZ + MZTOSPDL
6	-4	#	SPDLTOMZ + GRTOSPDL
7	#	-4	SPDLTOGR + MZTOSPDL
8	0	0	SPDLTOGR + GRTOSPDL
9	-4	#	GRTOSPDL + SPDLTOMZ

TM => PLC			PLC => TM
TMOPERATION	TAKEPOS	LEAVEPOS	
10	#	-4	MZTOSPDL + SPDLTOGR
11	0	0	MZROT
12	#	#	SPDLTOMZ + MZTOSPDL
13	#	0	MZROT + MZROT
14	#	#	SPDLTOMZ + MZTOSPDL

Example of operation.

The following example shows, assuming that there is no tool in the spindle, it shows the functions executed by the CNC, the values sent by the tool manager to the PLC in each operation and the marks to be activated by the PLC in each case.

It is a non-random magazine, each tool occupies the position of its own number and T7, T8 and T9 are ground tools.

CNC	TMOPERATION	TAKEPOS	LEAVEPOS	
T1 M6	11 1	0 1	0 0	MZROT MZTOSPDL
T2 M6	11 5	0 2	0 1	MZROT SPDLTOMZ + MZTOSPDL
T7 M6	11 6	0 -4	0 2	MZROT SPDLTOMZ + GRTOSPDL
T8 M6	11 8	0 0	0 0	MZROT SPDLTOGR + GRTOSPDL
T3 T4 M6	11 11 7	0 0 4	0 0 -4	MZROT MZROT SDPLTOGR + MZTOSPDL
T0 M6	11 2	0 0	0 4	MZROT SPDLTOMZ
T9 M6	11 3	0 0	0 0	MZROT GRTOSPDL
T0 M6	11 4	0 0	0 0	MZROT SPDLTOGR

7.8.2 Detailed description of the operations of the magazine.

Here is a sample of a detailed description of the valid operations for this magazine. For each operation, it indicates which of the TAKEPOS, LEAVEPOS, NEXTPOS and MZID signals are involved and their meanings. It also shows the sequence to follow to complete the operation.

The TAKEPOS, LEAVEPOS and NEXTPOS signals can assume the following values.

Value	Meaning
0	Do nothing.
#	Magazine position.
-4	Ground tool.

TMOPERATION = 1

Take a tool from the magazine and insert it in the spindle.

TAKEPOS=# Magazine position to pick up the tool.

The sequence of this operation is the following.

1 After ending this operation, activate the MZTOSPDL mark.

TMOPERATION = 2

Leave the spindle tool in the magazine.

LEAVEPOS=# Magazine position to leave the tool.

The sequence of this operation is the following.

1 After ending this operation, activate the SPDLTOMZ mark.

TMOPERATION = 3

Insert a ground tool in the spindle.

The sequence of this operation is the following.

1 When the operation is completed, activate the mark: GRTOSPDL.

TMOPERATION = 4

Leave the spindle tool on the ground.

The sequence of this operation is the following.

1 When the operation is completed, activate the mark: SPDLTOGR.

TMOPERATION = 5

Leave the spindle tool in the magazine and take a tool from the same magazine.

TAKEPOS=# Magazine position to pick up the tool.

LEAVEPOS=# Magazine position to leave the tool.

The sequence of this operation is the following.

1 Leave the tool of the spindle in the magazine and activate the mark SPDLTOMZ.

2 Leave the tool of the magazine in the spindle and activate the mark MZTOSPDL.

TMOPERATION = 6

Leave the spindle tool in the magazine and take a tool from ground.

TAKEPOS=-4 Ground tool.

LEAVEPOS=# Magazine position to leave the tool.

The sequence of this operation is the following.

1 Leave the tool of the spindle in the magazine and activate the mark SPDLTOMZ.

2 Leave the ground tool in the spindle and activate the mark GRTOSPDL.

TMOPERATION = 7

Leave the spindle tool on the ground and take a tool from the magazine.

TAKEPOS=# Magazine position to pick up the tool.

LEAVEPOS=-4 Ground tool.

The sequence of this operation is the following.

1 Leave the tool of the spindle on the ground and activate the mark SPDLTOGR.

2 Leave the tool of the magazine in the spindle and activate the mark MZTOSPDL.

TMOPERATION = 8

Leave the spindle tool on the ground and take a tool from ground.

The sequence of this operation is the following.

1 Leave the tool of the spindle on the ground and activate the mark SPDLTOGR.

2 Leave the ground tool in the spindle and activate the mark GRTOSPDL.

TMOPERATION = 9

Take a ground tool to the magazine going through the spindle.

TAKEPOS=-4 Ground tool.

LEAVEPOS=# Magazine position to leave the tool.

The sequence of this operation is the following.

1 Leave the ground tool in the spindle and activate the mark GRTOSPDL.

2 Leave the tool of the spindle in the magazine and activate the mark SPDLTOMZ.

TMOPERATION = 10

Take a tool from the magazine and leave on the ground going through the spindle.

TAKEPOS=# Magazine position to pick up the tool.

LEAVEPOS=-4 Ground tool.

The sequence of this operation is the following.

1 Leave the tool of the magazine in the spindle and activate the mark MZTOSPDL.

2 Leave the tool of the spindle on the ground and activate the mark SPDLTOGR.

TMOPERATION = 11

Orient the magazine.

This operation is an optimization of the tool change that permits orienting the magazine while machining. Activate the MZROT mark to indicate that the operation has been completed, whether it has been oriented or not.

TMOPERATION = 12

Leave the spindle tool in the magazine and take a tool from the same magazine (as TMOPERATION=5). It is an optimized operation, only valid when the magazine is random and the tool is special.

TAKEPOS=# Magazine position to pick up the tool.

LEAVEPOS=# Magazine position to leave the tool.

TMOPERATION = 13

Orienting two magazines.

This operation is an optimization of the tool change that permits orienting two magazines while machining. Activate the MZROT mark in both magazines to indicate that the operation has been completed, whether the magazines have been oriented or not.

NEXTPOS=#	Magazine position to leave the tool.
TAKEPOS=#	Magazine position to pick up the tool.
MZID	The lower portion of this register indicates the destination magazine for the tool and the higher portion the source magazine for the tool.

TMOPERATION = 14

Leave the spindle tool in the magazine and take a tool from another magazine.

TAKEPOS=#	Magazine position to pick up the tool.
LEAVEPOS=#	Magazine position to leave the tool.
MZID	The lower portion of this register indicates the destination magazine for the tool and the higher portion the source magazine for the tool.

The sequence of this operation is the following.

1 Leave the tool of the spindle in the magazine and activate the mark SPDLTOMZ.

2 Leave the tool of the magazine in the spindle and activate the mark MZTOSPDL.

7.8.3 Communication between the PLC and the M06 subroutine.

The communication between the PLC and the M06 subroutine takes place using a series of generic marks and registers. The program of the M06 subroutine offered as an example uses the following marks and registers.

Communication between the PLC and the M06 subroutine.

Registers used by the PLC to transfer information to the M06 subroutine:

R101 Type of operation requested by the tool manager (value of TMOPERATION).

Communication between the PLC and the M06 subroutine

Marks activated by the M06 subroutine so the PLC activates the relevant mark of the tool manager.

M1107 SPDLTOGR

The tool has been taken from the spindle to ground.

M1108 GRTOSPDL

The tool has been taken from ground to the spindle.

M1109 MZTOSPDL

The tool has been taken from the magazine to the spindle.

M1110 SPDLTOMZ

The tool has been taken from the spindle to the magazine.

M functions at the PLC

M functions that imply movements controlled by the PLC:

M109 Select in the magazine the position indicated by TAKEPOS and insert the tool in the spindle.

M110 Select in the magazine the position indicated by LEAVEPOS and leave the spindle tool.

Set all the M functions with "before-before" synchronization to the program continues after the M function is completed.

7.8.4 Program of the M06 subroutine.

%L SUB_SPD_TO_GR

Remove the tool from the spindle (take it to ground).

It shows the message for the operator to extract the tool and waits for the operation to be completed before removing the message.

#MSG ["Extract tool T%D and press START", V.TM.TOOL]

M0

#MSG [""]

V.PLC.M[1107]=1

SPDLTOGR mark to the tool manager.

#RET

%L SUB_GR_TO_SPD

Insert the ground tool in the spindle.

It shows the message for the operator to insert the tool and waits for the operation to be completed before removing the message.

#MSG ["Insert tool T%D and press START", V.TM.NXTOOL]

M0

#MSG [""]

V.PLC.M[1108]=1

GRTOSPDL mark to the tool manager.

#RET

%L SUB_MZ_TO_SPD

Take the tool from the magazine and insert it in the spindle.

M109

Auxiliary function to execute an action.

V.PLC.M[1109]=1

MZTOSPDL mark to the tool manager.

#RET

%L SUB_SPD_TO_MZ

Leave the spindle tool in the magazine.

M110

Auxiliary function to execute an action.

V.PLC.M[1110]=1

SPDLTOMZ mark to the tool manager.

#RET

%L SUB_SPD_GMCHG

Move the spindle to the manual tool change point.

G1 Z_ F_

Move the spindle.

#RET

%L SUB_SPD_AUTCHG

Move the spindle to the automatic tool change point

G1 Z_ F_

Move the spindle.

#RET

%SUB_M6.nc

M6

Order the tool manager to start the tool change.

$SWITCH V.PLC.R[101]

It analyzes the type of operation.

$CASE 1

Take a tool from the magazine and insert it in the spindle.

LL SUB_SPD_AUTCHG

Move the spindle to the automatic tool change point

LL SUB_MZ_TO_SPD

Take the tool from the magazine to the spindle.

$BREAK

$CASE 2

Leave the spindle tool in the magazine.

LL SUB_SPD_AUTCHG

Move the spindle to the automatic tool change point

LL SUB_SPD_TO_MZ

Leave the spindle tool in the magazine.

$BREAK

$CASE 3

Insert the ground tool in the spindle.

LL SUB_SPD_GMCHG

Move the spindle to the manual tool change point.

LL SUB_GR_TO_SPD

Insert the ground tool in the spindle.

$BREAK

$CASE 4

Leave the spindle tool on the ground.

LL SUB_SPD_GMCHG

Move the spindle to the manual tool change point.

LL SUB_SPD_TO_GR

Remove the tool from the spindle.

$BREAK

$CASE 5

Leave the spindle tool in the magazine and take another one from the magazine.

LL SUB_SPD_AUTCHG

Move the spindle to the automatic tool change point

LL SUB_SPD_TO_MZ

Leave the spindle tool in the magazine.

LL SUB_MZ_TO_SPD

Take the tool from the magazine to the spindle.

$BREAK

$CASE 6

Leave the spindle tool in the magazine and take another one from ground.

LL SUB_SPD_AUTCHG

Move the spindle to the automatic tool change point

LL SUB_SPD_TO_MZ

Leave the spindle tool in the magazine.

LL SUB_SPD_GMCHG

Move the spindle to the manual tool change point.

LL SUB_GR_TO_SPD

Insert the ground tool in the spindle.

$BREAK

$CASE 7

Leave the spindle tool on the ground and take another one from the magazine.

LL SUB_SPD_GMCHG

Move the spindle to the manual tool change point.

LL SUB_SPD_TO_GR

Remove the tool from the spindle.

LL SUB_SPD_AUTCHG

Move the spindle to the automatic tool change point

LL SUB_MZ_TO_SPD

Take the tool from the magazine to the spindle.

$BREAK

$CASE 8

Leave the spindle tool on the ground and take another one from ground.

LL SUB_SPD_GMCHG

Move the spindle to the manual tool change point.

LL SUB_SPD_TO_GR

Remove the tool from the spindle.

LL SUB_GR_TO_SPD

Insert the ground tool in the spindle.

$BREAK

$CASE 9

Take a ground tool to the magazine going through the spindle.

LL SUB_SPD_GMCHG

Move the spindle to the manual tool change point.

LL SUB_GR_TO_SPD

Insert the ground tool in the spindle.

LL SUB_SPD_AUTCHG

Move the spindle to the automatic tool change point

LL SUB_SPD_TO_MZ

Leave the spindle tool in the magazine.

$BREAK

$CASE 10

Take a tool from the magazine and leave on the ground going through the spindle.

LL SUB_SPD_AUTCHG

Move the spindle to the automatic tool change point

LL SUB_MZ_TO_SPD

Take the tool from the magazine to the spindle.

LL SUB_SPD_GMCHG

Move the spindle to the manual tool change point.

LL SUB_SPD_TO_GR

Remove the tool from the spindle.

$BREAK

$ENDSWITCH

End of analysis of type of operation.

$WHILE V.TM.MZWAIT == 1

$ENDWHILE

Wait for the tool manager.

#RET

End of M06 subroutine.

7.8.5 Basic PLC programming.

When executing the -T- function

When executing a T function, the tool manager sends the code TMOPERATION=11 to the PLC. In general, it is an optimization of the change that permits orienting the magazine while machining.

In this case, the magazine is not oriented and the MZROT mark is activated to indicate that the operation has been completed.

```
DFU TMOPSTROBE AND CPS TMOPERATION EQ 11 = SET MZROT
```

Activate the MZROT mark to "tell" the tool manager that the operation has finished.

When executing the M06 function

When executing the M06 function, the tool manager sends to the PLC, in the TMOPERATION register, the code for the operation to be carried out.

```
DFU TMOPSTROBE = MOV TMOPERATION R101
```

This instruction transfers the TMOPERATION value to register R101 so it is managed by the M06 subroutine.

Every time the M06 subroutine ends an action, it lets the PLC know so it activates the relevant mark of the tool manager.

```
DFU M1107 = SET SPDLTOGR
DFD SPDLTOGR = RES M1107
```

The tool has been taken from the spindle to ground.

```
DFU M1108 = SET GRTOSPDL
DFD GRTOSPDL = RES M1108
```

The tool has been taken from ground to the spindle.

```
DFU M1109 = SET MZTOSPDL
DFD MZTOSPDL = RES M1109
```

The tool has been taken from the magazine to the spindle.

```
DFU M1110 = SET SPDLTOMZ
DFD SPDLTOMZ = RES M1110
```

The tool has been taken from the spindle to the magazine.

```
DFU M1111 = SET MZROT
DFD MZROT = RES M1111
```

The magazine has rotated already.

The M06 subroutine uses the following M functions to "tell" the PLC which movements it must carry out.

M109	Select in the magazine the position indicated by TAKEPOS and insert the tool in the spindle.
M110	Select in the magazine the position indicated by LEAVEPOS and leave the spindle tool.

Programming it depends on the type of machine. The auxiliary function will conclude after executing the requested movement.

Certain operations require using the information transferred by the tool manager in the following registers:

LEAVEPOS	This register indicates the magazine position to leave the tool.
TAKEPOS	This register indicates the magazine position of the tool to be taken.

Manager emergency signal.

Treatment of the tool manager emergency signal.

```
DFU B11KEYBD1 AND NOT TMINEM = SET SETTMEM
DFU TMINEM = RES SETTMEM
```

Pressing the USER12 key activates the emergency.

```
TMINEM = B11KEYLED1
```

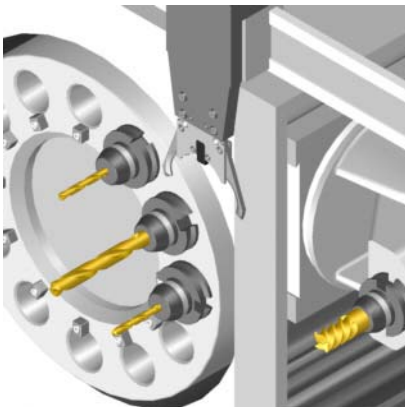
The lamp of the USER12 key turns on when there is an emergency.

TMINEM AND DFU B12KEYBD1 = SET RESTMEM

Pressing the USER13 key removes the emergency.

7.9 Synchronous magazine with changer arm and 1 claw

Synchronous magazines with tool changer arm (1 or 2 claws) have the magazine close to the spindle and the tool is changen using a tool changer arm. The tool cannot be changed while the part is being machined.



The communication between the manager and the PLC takes place in two stages; the first one when executing the T function and the second one when executing the M06 function.

Execution of the T function.

1. When the CNC executes the T function, it lets the tool manager know about it.
2. The tool manager sends a command to the PLC to select the next tool in the magazine (if possible).
3. The CNC continues with the execution of the program without waiting for the tool manager to complete the operation.

Execution of the M06 function.

1. When the CNC executes the M06 function, it calls the associated subroutine. The M06 function must also be programmed inside that subroutine so the CNC "tells" the tool manager to start making the tool change.
2. The tool manager sends a command to the PLC to make the change.
3. The CNC waits for the tool manager to complete the operation before continuing with the execution of the program.

The management of the tool change should be included in the subroutine associated with the M06 and leave the control of the external devices up to the PLC. Use the auxiliary functions to govern the various devices (magazine rotation, magazine movement, tool changer arm, etc.) from the M06 subroutine.

7.9.1 Valid operations and marks activated by the PLC with each one of them.

Valid operations of the tool manager for this magazine.

The possible operations in this type of magazine are:

TMOPERATION	Meaning.
0	Do nothing.
1	Take a tool from the magazine and insert it in the spindle.
2	Leave the spindle tool in the magazine.
3	Insert a ground tool in the spindle.
4	Leave the spindle tool on the ground.
5	Leave the spindle tool in the magazine and take a tool from the same magazine.
6	Leave the spindle tool in the magazine and take a tool from ground.
7	Leave the spindle tool on the ground and take a tool from the magazine.
8	Leave the spindle tool on the ground and take a tool from ground.
9	Take a ground tool to the magazine going through the spindle.
10	Take a tool from the magazine and leave on the ground going through the spindle.
11	Orient the magazine.
12	Leave the spindle tool in the magazine and take a tool from the same magazine (as TMOPERATION=5). It is an optimized operation, only valid when the magazine is random and the tool is special.
13	Orienting two magazines.
14	Leave the spindle tool in the magazine and take a tool from another magazine.

Values of registers TAKEPOS and LEAVEPOS used by the manager.

The TAKEPOS and LEAVEPOS signals can assume the following values.

Value	Meaning
0	Do nothing.
#	Magazine position.
-4	Ground tool.

Here is a summary table of the TAKEPOS and LEAVEPOS values used by the manager in each operation as well as the marks that must be activated by the PLC at the end of each operation. In the case of the marks shown between parenthesis, the manager can execute the operations in any order, but it must always execute both.

TM => PLC			PLC => TM
TMOPERATION	TAKEPOS	LEAVEPOS	
1	#	0	MZTOCH1 + CH1TOSPDL
2	0	#	SPDLTOCH1 + CH1TOMZ
3	0	0	GRTOSPDL
4	0	0	SPDLTOGR
5	#	#	SPDLTOCH1 + CH1TOMZ + MZTOCH1 + CH1TOSPDL
6	-4	#	SPDLTOCH1 + (CH1TOMZ & GRTOSPDL)
7	#	-4	(SPDLTOGR & MZTOCH1) + CH1TOSPDL

TM => PLC			PLC => TM
TMOPERATION	TAKEPOS	LEAVEPOS	
8	0	0	SPDLTOGR + GRTOSPDL
9	-4	#	GRTOSPDL + SPDLTOCH1 + CH1TOMZ
10	#	-4	MZTOCH1 + CH1TOSPDL + SPDLTOGR
11	0	0	MZROT
12	#	#	SPDLTOCH1 + CH1TOMZ + MZTOCH1 + CH1TOSPDL
13	#	0	MZROT + MZROT
14	#	#	SPDLTOCH1 + CH1TOMZ + MZTOCH1 + CH1TOSPDL

Example of operation.

The following example shows, assuming that there is no tool in the spindle, it shows the functions executed by the CNC, the values sent by the tool manager to the PLC in each operation and the marks to be activated by the PLC in each case.

It is a non-random magazine, each tool occupies the position of its own number and T7, T8 and T9 are ground tools.

CNC	TMOPERATION	TAKEPOS	LEAVEPOS	
T1 M6	11 1	0 1	0 0	MZROT MZTOCH1 + CH1TOSPDL
T2 M6	11 5	0 2	0 1	MZROT SPDLTOCH1 + CH1TOMZ + MZTOCH1 + CH1TOSPDL
T7 M6	11 6	0 -4	0 2	MZROT SPDLTOCH1 + CH1TOMZ + GRTOSPDL
T8 M6	11 8	0 0	0 0	MZROT SPDLTOGR + GRTOSPDL
T3 T4 M6	11 11 7	0 0 4	0 0 -4	MZROT MZROT SDPLTOGR + MZTOCH1 + CH1TOSPDL
T0 M6	11 2	0 0	0 4	MZROT SPDLTOCH1 + CH1TOMZ
T9 M6	11 3	0 0	0 0	MZROT GRTOSPDL
T0 M6	11 4	0 0	0 0	MZROT SPDLTOGR

7.9.2 Detailed description of the operations of the magazine.

Here is a sample of a detailed description of the valid operations for this magazine. For each operation, it indicates which of the TAKEPOS, LEAVEPOS, NEXTPOS and MZID signals are involved and their meanings. It also shows the sequence to follow to complete the operation.

The TAKEPOS, LEAVEPOS and NEXTPOS signals can assume the following values.

Value	Meaning
0	Do nothing.
#	Magazine position.
-4	Ground tool.

TMOPERATION = 1

Take a tool from the magazine and insert it in the spindle.

TAKEPOS=# Magazine position to pick up the tool.

The sequence of this operation is the following.

1 Take the tool from the magazine with holder 1 and activate the mark MZTOCH1.
2 Insert the tool of holder 1 in the spindle and activate CH1TOSPDL.

TMOPERATION = 2

Leave the spindle tool in the magazine.

LEAVEPOS=# Magazine position to leave the tool.

The sequence of this operation is the following.

1 Take the tool from the spindle with holder 1 and activate the mark SPDLTOCH1.
2 Leave the tool of holder 1 in the magazine and activate the mark CH1TOMZ.

TMOPERATION = 3

Insert a ground tool in the spindle.

The sequence of this operation is the following.

1 When the operation is completed, activate the mark: GRTOSPDL.

TMOPERATION = 4

Leave the spindle tool on the ground.

The sequence of this operation is the following.

1 When the operation is completed, activate the mark: SPDLTOGR.

TMOPERATION = 5

Leave the spindle tool in the magazine and take a tool from the same magazine.

TAKEPOS=# Magazine position to pick up the tool.
LEAVEPOS=# Magazine position to leave the tool.

The sequence of this operation is the following.

1 Take the tool from the spindle with holder 1 and activate the mark SPDLTOCH1.
2 Leave the tool of holder 1 in the magazine and activate the mark CH1TOMZ.
3 Take the tool from the magazine with holder 1 and activate the mark MZTOCH1.
4 Insert the tool of holder 1 in the spindle and activate CH1TOSPDL.

TMOPERATION = 6

Leave the spindle tool in the magazine and take a tool from ground.

TAKEPOS=-4 Ground tool.
LEAVEPOS=# Magazine position to leave the tool.

The sequence of this operation is the following. The tool manager admits the sequence 1-3-2.

1 Take the tool from the spindle with holder 1 and activate the mark SPDLTOCH1.
2 Leave the tool of holder 1 in the magazine and activate the mark CH1TOMZ.
3 Leave the ground tool in the spindle and activate the mark GRTOSPDL.

TMOPERATION = 7

Leave the spindle tool on the ground and take a tool from the magazine.

TAKEPOS=# Magazine position to pick up the tool.
LEAVEPOS=-4 Ground tool.

The sequence of this operation is the following. The tool manager admits the sequence 2-1-3.

1 Leave the tool of the spindle on the ground and activate the mark SPDLTOGR.
2 Take the tool from the magazine with holder 1 and activate the mark MZTOCH1.
3 Insert the tool of holder 1 in the spindle and activate CH1TOSPDL.

TMOPERATION = 8

Leave the spindle tool on the ground and take a tool from ground.

The sequence of this operation is the following.

1 Leave the tool of the spindle on the ground and activate the mark SPDLTOGR.
2 Leave the ground tool in the spindle and activate the mark GRTOSPDL.

TMOPERATION = 9

Take a ground tool to the magazine going through the spindle.

TAKEPOS=-4 Ground tool.
LEAVEPOS=# Magazine position to leave the tool.

The sequence of this operation is the following.

1 Leave the ground tool in the spindle and activate the mark GRTOSPDL.
2 Take the tool from the spindle with holder 1 and activate the mark SPDLTOCH1.
3 Leave the tool of holder 1 in the magazine and activate the mark CH1TOMZ.

TMOPERATION = 10

Take a tool from the magazine and leave on the ground going through the spindle.

TAKEPOS=# Magazine position to pick up the tool.
LEAVEPOS=-4 Ground tool.

The sequence of this operation is the following.

1 Take the tool from the magazine with holder 1 and activate the mark MZTOCH1.
2 Insert the tool of holder 1 in the spindle and activate CH1TOSPDL.
3 Leave the tool of the spindle on the ground and activate the mark SPDLTOGR.

TMOPERATION = 11

Orient the magazine.

This operation is an optimization of the tool change that permits orienting the magazine while machining. Activate the MZROT mark to indicate that the operation has been completed, whether it has been oriented or not.

TMOPERATION = 12

Leave the spindle tool in the magazine and take a tool from the same magazine (as TMOPERATION=5). It is an optimized operation, only valid when the magazine is random and the tool is special.

TAKEPOS=# Magazine position to pick up the tool.
LEAVEPOS=# Magazine position to leave the tool.

TMOPERATION = 13

Orienting two magazines.

This operation is an optimization of the tool change that permits orienting two magazines while machining. Activate the MZROT mark in both magazines to indicate that the operation has been completed, whether the magazines have been oriented or not.

NEXTPOS=#	Magazine position to leave the tool.
TAKEPOS=#	Magazine position to pick up the tool.
MZID	The lower portion of this register indicates the destination magazine for the tool and the higher portion the source magazine for the tool.

TMOPERATION = 14

Leave the spindle tool in the magazine and take a tool from another magazine.

TAKEPOS=#	Magazine position to pick up the tool.
LEAVEPOS=#	Magazine position to leave the tool.
MZID	The lower portion of this register indicates the destination magazine for the tool and the higher portion the source magazine for the tool.

The sequence of this operation is the following.

1. Take the tool from the spindle with holder 1 and activate the mark SPDLTOCH1.
2. Leave the tool of holder 1 in the magazine and activate the mark CH1TOMZ.
3. Take the tool from the other magazine with holder ·1· and activate the mark MZTOCH1.
4. Insert the tool of holder 1 in the spindle and activate CH1TOSPDL.

7.9.3 Communication between the PLC and the M06 subroutine.

The communication between the PLC and the M06 subroutine takes place using a series of generic marks and registers. The program of the M06 subroutine offered as an example uses the following marks and registers.

Communication between the PLC and the M06 subroutine.

Registers used by the PLC to transfer information to the M06 subroutine:

R101 Type of operation requested by the tool manager (value of TMOPERATION)

Communication between the PLC and the M06 subroutine

Marks activated by the M06 subroutine so the PLC activates the relevant mark of the tool manager.

M1101 MZTOCH1

The tool has been taken from the magazine to holder 1.

M1102 CH1TOSPDL

The tool has been taken from holder 1 to the spindle.

M1103 SPDLTOCH1

The tool has been taken from the spindle to holder 1.

M1105 CH1TOMZ

The tool has been taken from holder 1 to the magazine.

M1107 SPDLTOGR

The tool has been taken from the spindle to ground.

M1108 GRTOSPDL

The tool has been taken from ground to the spindle.

M functions at the PLC

M functions that imply movements controlled by the PLC:

M101 Select in the magazine the position indicated by TAKEPOS and take the tool with holder 1.

M102 Take the tool of holder 1 to the spindle.

M103 Take the spindle tool with holder 1.

M105 Select in the magazine the position indicated by LEAVEPOS and leave the tool of holder 1.

Set all the M functions with "before-before" synchronization to the program continues after the M function is completed.

7.9.4 Program of the M06 subroutine.

%L SUB_MZ_TO_CH1

Take the tool from the magazine with holder 1.

M101

Auxiliary function to execute an action.

V.PLC.M[1101]=1

MZTOCH1 mark to the tool manager.

#RET

%L SUB_CH1_TO_SPD

Take the tool of holder 1 to the spindle.

M102

Auxiliary function to execute an action.

V.PLC.M[1102]=1

CH1TOSPDL mark to the tool manager.

#RET

%L SUB_SPD_TO_CH1

Take the spindle tool with holder 1.

M103

Auxiliary function to execute an action.

V.PLC.M[1103]=1

SPDLTOCH1 mark to the tool manager.

#RET

%L SUB_CH1_TO_MZ

Take the tool of holder 1 to the magazine.

M105

Auxiliary function to execute an action.

V.PLC.M[1105]=1

CH1TOMZ mark to the tool manager.

#RET

%L SUB_SPD_TO_GR

Remove the tool from the spindle (take it to ground).

It shows the message for the operator to extract the tool and waits for the operation to be completed before removing the message.

#MSG ["Extract tool T%D and press START", V.TM.TOOL]

M0

#MSG [""]

V.PLC.M[1107]=1

SPDLTOGR mark to the tool manager.

#RET

%L SUB_GR_TO_SPD

Insert the ground tool in the spindle.

It shows the message for the operator to insert the tool and waits for the operation to be completed before removing the message.

#MSG ["Insert tool T%D and press START", V.TM.NXTOOL]

M0

#MSG [""]

V.PLC.M[1108]=1

GRTOSPDL mark to the tool manager.

#RET

%L SUB_SPD_GMCHG

Move the spindle to the manual tool change point.

G1 Z_ F_

Move the spindle.

#RET

%L SUB_SPD_AUTCHG

Move the spindle to the automatic tool change point

G1 Z_ F_

Move the spindle.

#RET

%SUB_M6.nc

M6

Order the tool manager to start the tool change.

$SWITCH V.PLC.R[101]

It analyzes the type of operation.

$CASE 1

Take a tool from the magazine and insert it in the spindle.

LL SUB_SPD_AUTCHG

Move the spindle to the automatic tool change point

LL SUB_MZ_TO_CH1

Take the tool from the magazine with holder 1.

LL SUB_CH1_TO_SPD

Take the tool of holder 1 to the spindle.

$BREAK

$CASE 2

Leave the spindle tool in the magazine.

LL SUB_SPD_AUTCHG

Move the spindle to the automatic tool change point

LL SUB_SPD_TO_CH1

Take the spindle tool with holder 1.

LL SUB_CH1_TO_MZ

Leave the tool of holder 1 in the magazine.

$BREAK

$CASE 3

Insert the ground tool in the spindle.

LL SUB_SPD_GMCHG

Move the spindle to the manual tool change point.

LL SUB_GR_TO_SPD

Insert the ground tool in the spindle.

$BREAK

$CASE 4

Leave the spindle tool on the ground.

LL SUB_SPD_GMCHG

Move the spindle to the manual tool change point.

LL SUB_SPD_TO_GR

Remove the tool from the spindle.

$BREAK

$CASE 5

Leave the spindle tool in the magazine and take another one from the magazine.

LL SUB_SPD_AUTCHG

Move the spindle to the automatic tool change point

LL SUB_SPD_TO_CH1

Take the spindle tool with holder 1.

LL SUB_CH1_TO_MZ

Leave the tool of holder 1 in the magazine.

LL SUB_MZ_TO_CH1

Take the tool from the magazine with holder 1.

LL SUB_CH1_TO_SPD

Take the tool of holder 1 to the spindle.

$BREAK

$CASE 6

Leave the spindle tool in the magazine and take another one from ground.

LL SUB_SPD_AUTCHG

Move the spindle to the automatic tool change point

LL SUB_SPD_TO_CH1

Take the spindle tool with holder 1.

LL SUB_CH1_TO_MZ

Leave the tool of holder 1 in the magazine.

LL SUB_SPD_GMCHG

Move the spindle to the manual tool change point.

LL SUB_GR_TO_SPD

Insert the ground tool in the spindle.

$BREAK

$CASE 7

Leave the spindle tool on the ground and take another one from the magazine.

LL SUB_SPD_GMCHG

Move the spindle to the manual tool change point.

LL SUB_SPD_TO_GR

Remove the tool from the spindle.

LL SUB_SPD_AUTCHG

Move the spindle to the automatic tool change point

LL SUB_MZ_TO_CH1

Take the tool from the magazine with holder 1.

LL SUB_CH1_TO_SPD

Take the tool of holder 1 to the spindle.

$BREAK

$CASE 8

Leave the spindle tool on the ground and take another one from ground.

LL SUB_SPD_GMCHG

Move the spindle to the manual tool change point.

LL SUB_SPD_TO_GR

Remove the tool from the spindle.

LL SUB_GR_TO_SPD

Insert the ground tool in the spindle.

$BREAK

$CASE 9

Take a ground tool to the magazine going through the spindle.

LL SUB_SPD_GMCHG

Move the spindle to the manual tool change point.

LL SUB_GR_TO_SPD

Insert the ground tool in the spindle.

LL SUB_SPD_AUTCHG

Move the spindle to the automatic tool change point

LL SUB_SPD_TO_CH1

Take the spindle tool with holder 1.

LL SUB_CH1_TO_MZ

Leave the tool of holder 1 in the magazine.

$BREAK

$CASE 10

Take a tool from the magazine and leave on the ground going through the spindle.

LL SUB_SPD_AUTCHG

Move the spindle to the automatic tool change point

LL SUB_MZ_TO_CH1

Take the tool from the magazine with holder 1.

LL SUB_CH1_TO_SPD

Take the tool of holder 1 to the spindle.

LL SUB_SPD_GMCHG

Move the spindle to the manual tool change point.

LL SUB_SPD_TO_GR

Remove the tool from the spindle.

$BREAK

$ENDSWITCH

End of analysis of type of operation.

$WHILE V.TM.MZWAIT == 1

$ENDWHILE

Wait for the tool manager.

#RET

End of M06 subroutine.

7.9.5 Basic PLC programming.

When executing the -T- function

When executing a T function, the tool manager sends the code TMOPERATION=11 to the PLC. In general, it is an optimization of the change that permits orienting the magazine while machining.

In this case, the magazine is not oriented and the MZROT mark is activated to indicate that the operation has been completed.

```
DFU TMOPSTROBE AND CPS TMOPERATION EQ 11 = SET MZROT
```

Activate the MZROT mark to "tell" the tool manager that the operation has finished.

When executing the M06 function

When executing the M06 function, the tool manager sends to the PLC, in the TMOPERATION register, the code for the operation to be carried out.

```
DFU TMOPSTROBE = MOV TMOPERATION R101
```

This instruction transfers the TMOPERATION value to register R101 so it is managed by the M06 subroutine.

Every time the M06 subroutine ends an action, it lets the PLC know so it activates the relevant mark of the tool manager.

```
DFU M1101 = SET MZTOCH1
DFD MZTOCH1 = RES M1101
```

The tool has been taken from the magazine to holder 1.

```
DFU M1102 = SET CH1TOSPDL
DFD CH1TOSPDL = RES M1102
```

The tool has been taken from holder 1 to the spindle.

```
DFU M1103 = SET SPDLTOCH1
DFD SPDLTOCH1 = RES M1103
```

The tool has been taken from the spindle to holder 1.

```
DFU M1105 = SET CH1TOMZ
DFD CH1TOMZ = RES M1105
```

The tool has been taken from holder 1 to the magazine.

```
DFU M1107 = SET SPDLTOGR
DFD SPDLTOGR = RES M1107
```

The tool has been taken from the spindle to ground.

```
DFU M1108 = SET GRTOSPDL
DFD GRTOSPDL = RES M1108
```

The tool has been taken from ground to the spindle.

The M06 subroutine uses the following M functions to "tell" the PLC which movements it must carry out.

M101	Select in the magazine the position indicated by TAKEPOS and take the tool with holder 1.
M102	Take the tool of holder 1 to the spindle.
M103	Take the spindle tool with holder 1.
M105	Select in the magazine the position indicated by LEAVEPOS and leave the tool of holder 1.

Programming it depends on the type of machine. The auxiliary function will conclude after executing the requested movement.

Certain operations require using the information transferred by the tool manager in the following registers:

LEAVEPOS	This register indicates the magazine position to leave the tool.
TAKEPOS	This register indicates the magazine position of the tool to be taken.

Manager emergency signal.

Treatment of the tool manager emergency signal.

```
DFU B11KEYBD1 AND NOT TMINEM = SET SETTMEM
DFU TMINEM = RES SETTMEM
```

Pressing the USER12 key activates the emergency.

```
TMINEM = B11KEYLED1
```

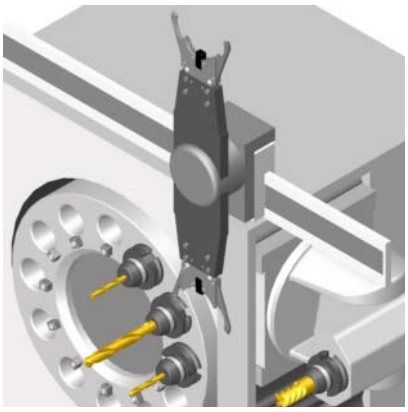
The lamp of the USER12 key turns on when there is an emergency.

```
TMINEM AND DFU B12KEYBD1 = SET RESTMEM
```

Pressing the USER13 key removes the emergency.

7.10 Synchronous magazine with changer arm and 2 claws.

Synchronous magazines with tool changer arm (1 or 2 claws) have the magazine close to the spindle and the tool is changen using a tool changer arm. The tool cannot be changed while the part is being machined.



The communication between the manager and the PLC takes place in two stages; the first one when executing the T function and the second one when executing the M06 function.

Execution of the T function.

1. When the CNC executes the T function, it lets the tool manager know about it.
2. The tool manager sends a command to the PLC to select the next tool in the magazine (if possible).
3. The CNC continues with the execution of the program without waiting for the tool manager to complete the operation.

Execution of the M06 function.

1. When the CNC executes the M06 function, it calls the associated subroutine. The M06 function must also be programmed inside that subroutine so the CNC "tells" the tool manager to start making the tool change.
2. The tool manager sends a command to the PLC to make the change.
3. The CNC waits for the tool manager to complete the operation before continuing with the execution of the program.

The management of the tool change should be included in the subroutine associated with the M06 and leave the control of the external devices up to the PLC. Use the auxiliary functions to govern the various devices (magazine rotation, magazine movement, tool changer arm, etc.) from the M06 subroutine.

7.10.1 Valid operations and marks activated by the PLC with each one of them.

Valid operations of the tool manager for this magazine.

The possible operations in this type of magazine are:

TMOPERATION	Meaning.
0	Do nothing.
1	Take a tool from the magazine and insert it in the spindle.
2	Leave the spindle tool in the magazine.
3	Insert a ground tool in the spindle.
4	Leave the spindle tool on the ground.
5	Leave the spindle tool in the magazine and take a tool from the same magazine.
6	Leave the spindle tool in the magazine and take a tool from ground.
7	Leave the spindle tool on the ground and take a tool from the magazine.
8	Leave the spindle tool on the ground and take a tool from ground.
9	Take a ground tool to the magazine going through the spindle.
10	Take a tool from the magazine and leave on the ground going through the spindle.
11	Orient the magazine.
12	Leave the spindle tool in the magazine and take a tool from the same magazine (as TMOPERATION=5). It is an optimized operation only valid in the following types of synchronous magazine. • Non-random having a tool changer arm with two claws. • Random when having special tools.
13	Orienting two magazines.
14	Leave the spindle tool in the magazine and take a tool from another magazine.

Values of registers TAKEPOS and LEAVEPOS used by the manager.

The TAKEPOS and LEAVEPOS signals can assume the following values.

Value	Meaning
0	Do nothing.
#	Magazine position.
-4	Ground tool.

Here is a summary table of the TAKEPOS and LEAVEPOS values used by the manager in each operation as well as the marks that must be activated by the PLC at the end of each operation. In the case of the marks shown between parenthesis, the manager can execute the operations in any order, but it must always execute both.

TM => PLC			PLC => TM
TMOPERATION	TAKEPOS	LEAVEPOS	
1	#	0	MZTOCH1 + CH1TOSPDL
2	0	#	SPDLTOCH2 + CH2TOMZ
3	0	0	GRTOSPDL
4	0	0	SPDLTOGR
5	#	#	(SPDLTOCH2 & MZTOCH1) + (CH1TOSPDL & CH2TOMZ)
6	-4	#	SPDLTOCH2 + (CH2TOMZ & GRTOSPDL)

TM => PLC			PLC => TM
TMOPERATION	TAKEPOS	LEAVEPOS	
7	#	-4	(SPDLTOGR & MZTOCH1) + CH1TOSPDL
8	0	0	SPDLTOGR + GRTOSPDL
9	-4	#	GRTOSPDL + SPDLTOCH2 + CH2TOMZ
10	#	-4	MZTOCH1 + CH1TOSPDL + SPDLTOGR
11	0	0	MZROT
12	#	#	SPDLTOCH2 + CH2TOMZ + MZTOCH1 + CH1TOSPDL
13	#	0	MZROT + MZROT
14	#	#	SPDLTOCH2 + CH2TOMZ + MZTOCH1 + CH1TOSPDL

Example of operation.

The following example shows, assuming that there is no tool in the spindle, it shows the functions executed by the CNC, the values sent by the tool manager to the PLC in each operation and the marks to be activated by the PLC in each case.

It is a non-random magazine, each tool occupies the position of its own number and T7, T8 and T9 are ground tools.

CNC	TMOPERATION	TAKEPOS	LEAVEPOS	
T1 M6	11 1	0 1	0 0	MZROT MZTOCH1 + CH1TOSPDL
T2 M6	11 5	0 2	0 1	MZROT MZTOCH1 + SPDLTOCH2 + CH1TOSPDL + CH2TOMZ
T7 M6	11 6	0 -4	0 2	MZROT SPDLTOCH2 + CH2TOMZ + GRTOSPDL
T8 M6	11 8	0 0	0 0	MZROT SPDLTOGR + GRTOSPDL
T3 T4 M6	11 11 7	0 0 4	0 0 -4	MZROT MZROT SDPLTOGR + MZTOCH1 + CH1TOSPDL
T0 M6	11 2	0 0	0 4	MZROT SPDLTOCH2 + CH2TOMZ
T9 M6	11 3	0 0	0 0	MZROT GRTOSPDL
T0 M6	11 4	0 0	0 0	MZROT SPDLTOGR

7.10.2 Detailed description of the operations of the magazine.

Here is a sample of a detailed description of the valid operations for this magazine. For each operation, it indicates which of the TAKEPOS, LEAVEPOS, NEXTPOS and MZID signals are involved and their meanings. It also shows the sequence to follow to complete the operation.

The TAKEPOS, LEAVEPOS and NEXTPOS signals can assume the following values.

Value	Meaning
0	Do nothing.
#	Magazine position.
-4	Ground tool.

TMOPERATION = 1

Take a tool from the magazine and insert it in the spindle.

TAKEPOS=# Magazine position to pick up the tool.

The sequence of this operation is the following.

1 Take the tool from the magazine with holder 1 and activate the mark MZTOCH1.
2 Insert the tool of holder 1 in the spindle and activate CH1TOSPDL.

TMOPERATION = 2

Leave the spindle tool in the magazine.

LEAVEPOS=# Magazine position to leave the tool.

The sequence of this operation is the following.

1 Take the tool from the spindle with holder ·2· and activate the mark SPDLTOCH2.
2 Leave the tool of holder ·2· in the magazine and activate the mark CH2TOMZ.

TMOPERATION = 3

Insert a ground tool in the spindle.

The sequence of this operation is the following.

1 When the operation is completed, activate the mark: GRTOSPDL.

TMOPERATION = 4

Leave the spindle tool on the ground.

The sequence of this operation is the following.

1 When the operation is completed, activate the mark: SPDLTOGR.

TMOPERATION = 5

Leave the spindle tool in the magazine and take a tool from the same magazine.

TAKEPOS=# Magazine position to pick up the tool.

LEAVEPOS=# Magazine position to leave the tool.

The sequence of this operation is the following. The tool manager also admits the sequences 1-2-4-3, 2-1-3-4, 2-1-4-3.

1 Take the tool from the spindle with holder ·2· and activate the mark SPDLTOCH2.
2 Take the tool from the magazine with holder 1 and activate the mark MZTOCH1.
3 Leave the tool of holder ·2· in the magazine and activate the mark CH2TOMZ.
4 Insert the tool of holder 1 in the spindle and activate CH1TOSPDL.

TMOPERATION = 6

Leave the spindle tool in the magazine and take a tool from ground.

TAKEPOS=-4 Ground tool.

LEAVEPOS=# Magazine position to leave the tool.

The sequence of this operation is the following. The tool manager admits the sequence 1-3-2.

1 Take the tool from the spindle with holder ·2· and activate the mark SPDLTOCH2.
2 Leave the tool of holder ·2· in the magazine and activate the mark CH2TOMZ.
3 Leave the ground tool in the spindle and activate the mark GRTOSPDL.

TMOPERATION = 7

Leave the spindle tool on the ground and take a tool from the magazine.

TAKEPOS=# Magazine position to pick up the tool.
LEAVEPOS=-4 Ground tool.

The sequence of this operation is the following. The tool manager admits the sequence 2-1-3.

1 Leave the tool of the spindle on the ground and activate the mark SPDLTOGR.
2 Take the tool from the magazine with holder 1 and activate the mark MZTOCH1.
3 Insert the tool of holder 1 in the spindle and activate CH1TOSPDL.

TMOPERATION = 8

Leave the spindle tool on the ground and take a tool from ground.

The sequence of this operation is the following.

1 Leave the tool of the spindle on the ground and activate the mark SPDLTOGR.
2 Leave the ground tool in the spindle and activate the mark GRTOSPDL.

TMOPERATION = 9

Take a ground tool to the magazine going through the spindle.

TAKEPOS=-4 Ground tool.
LEAVEPOS=# Magazine position to leave the tool.

The sequence of this operation is the following.

1 Leave the ground tool in the spindle and activate the mark GRTOSPDL.
2 Take the tool from the spindle with holder ·2· and activate the mark SPDLTOCH2.
3 Leave the tool of holder ·2· in the magazine and activate the mark CH2TOMZ.

TMOPERATION = 10

Take a tool from the magazine and leave on the ground going through the spindle.

TAKEPOS=# Magazine position to pick up the tool.
LEAVEPOS=-4 Ground tool.

The sequence of this operation is the following.

1 Take the tool from the magazine with holder 1 and activate the mark MZTOCH1.
2 Insert the tool of holder 1 in the spindle and activate CH1TOSPDL.
3 Leave the tool of the spindle on the ground and activate the mark SPDLTOGR.

TMOPERATION = 11

Orient the magazine.

This operation is an optimization of the tool change that permits orienting the magazine while machining. Activate the MZROT mark to indicate that the operation has been completed, whether it has been oriented or not.

TMOPERATION = 12

Leave the spindle tool in the magazine and take a tool from the same magazine (as TMOPERATION=5). It is an optimized operation only valid in the following types of synchronous magazine.

- Non-random having a tool changer arm with two claws.

- Random when having special tools.

TAKEPOS=#	Magazine position to pick up the tool.
LEAVEPOS=#	Magazine position to leave the tool.

TMOPERATION = 13

Orienting two magazines.

This operation is an optimization of the tool change that permits orienting two magazines while machining. Activate the MZROT mark in both magazines to indicate that the operation has been completed, whether the magazines have been oriented or not.

NEXTPOS=#	Magazine position to leave the tool.
TAKEPOS=#	Magazine position to pick up the tool.
MZID	The lower portion of this register indicates the destination magazine for the tool and the higher portion the source magazine for the tool.

TMOPERATION = 14

Leave the spindle tool in the magazine and take a tool from another magazine.

TAKEPOS=#	Magazine position to pick up the tool.
LEAVEPOS=#	Magazine position to leave the tool.
MZID	The lower portion of this register indicates the destination magazine for the tool and the higher portion the source magazine for the tool.

The sequence of this operation is the following.

1. Take the tool from the spindle with holder ·2· and activate the mark SPDLTOCH2.
2. Leave the tool of holder ·2· in the magazine and activate the mark CH2TOMZ.
3. Take the tool from the other magazine with holder ·1· and activate the mark MZTOCH1.
4. Insert the tool of holder 1 in the spindle and activate CH1TOSPDL.

7.10.3 Communication between the PLC and the M06 subroutine.

The communication between the PLC and the M06 subroutine takes place using a series of generic marks and registers. The program of the M06 subroutine offered as an example uses the following marks and registers.

Communication between the PLC and the M06 subroutine.

Registers used by the PLC to transfer information to the M06 subroutine:

R101 Type of operation requested by the tool manager (value of TMOPERATION)

Communication between the PLC and the M06 subroutine

Marks activated by the M06 subroutine so the PLC activates the relevant mark of the tool manager.

M1101 MZTOCH1

The tool has been taken from the magazine to holder 1.

M1102 CH1TOSPDL

The tool has been taken from holder 1 to the spindle.

M1104 SPDLTOCH2

The tool has been taken from the spindle to holder 2.

M1106 CH2TOMZ

The tool has been taken from holder 2 to the magazine.

M1107 SPDLTOGR

The tool has been taken from the spindle to ground.

M1108 GRTOSPDL

The tool has been taken from ground to the spindle.

M functions at the PLC

M functions that imply movements controlled by the PLC:

M101 Select in the magazine the position indicated by TAKEPOS and take the tool with holder 1.

M102 Take the tool of holder 1 to the spindle.

M104 Take the spindle tool with holder 2.

M106 Select in the magazine the position indicated by LEAVEPOS and leave the tool of holder 2.

Set all the M functions with "before-before" synchronization to the program continues after the M function is completed.

7.10.4 Program of the M06 subroutine.

%L SUB_MZ_TO_CH1

Take the tool from the magazine with holder 1.

M101

Auxiliary function to execute an action.

V.PLC.M[1101]=1

MZTOCH1 mark to the tool manager.

#RET

%L SUB_CH1_TO_SPD

Take the tool of holder 1 to the spindle.

M102

Auxiliary function to execute an action.

V.PLC.M[1102]=1

CH1TOSPDL mark to the tool manager.

#RET

%L SUB_SPD_TO_CH2

Take the spindle tool with holder 2.

M104

Auxiliary function to execute an action.

V.PLC.M[1104]=1

SPDLTOCH2 mark to the tool manager.

#RET

%L SUB_CH2_TO_MZ

Take the tool of holder 2 to the magazine.

M106

Auxiliary function to execute an action.

V.PLC.M[1106]=1

CH2TOMZ mark to the tool manager.

#RET

%L SUB_SPD_TO_GR

Remove the tool from the spindle (take it to ground).

It shows the message for the operator to extract the tool and waits for the operation to be completed before removing the message.

#MSG ["Extract tool T%D and press START", V.TM.TOOL]

M0

#MSG [""]

V.PLC.M[1107]=1

SPDLTOGR mark to the tool manager.

#RET

%L SUB_GR_TO_SPD

Insert the ground tool in the spindle.

It shows the message for the operator to insert the tool and waits for the operation to be completed before removing the message.

#MSG ["Insert tool T%D and press START", V.TM.NXTOOL]

M0

#MSG [""]

V.PLC.M[1108]=1

GRTOSPDL mark to the tool manager.

#RET

%L SUB_SPD_GMCHG

Move the spindle to the manual tool change point.

G1 Z_ F_

Move the spindle.

#RET

%L SUB_SPD_AUTCHG

Move the spindle to the automatic tool change point

G1 Z_ F_

Move the spindle.

#RET

%SUB_M6.nc

M6

Order the tool manager to start the tool change.

$SWITCH V.PLC.R[101]

It analyzes the type of operation.

$CASE 1

Take a tool from the magazine and insert it in the spindle.

LL SUB_SPD_AUTCHG

Move the spindle to the automatic tool change point

LL SUB_MZ_TO_CH1

Take the tool from the magazine with holder 1.

LL SUB_CH1_TO_SPD

Take the tool of holder 1 to the spindle.

$BREAK

$CASE 2

Leave the spindle tool in the magazine.

LL SUB_SPD_AUTCHG

Move the spindle to the automatic tool change point

LL SUB_SPD_TO_CH2

Take the spindle tool with holder 2.

LL SUB_CH2_TO_MZ

Leave the tool of holder 2 in the magazine.

$BREAK

$CASE 3

Insert the ground tool in the spindle.

LL SUB_SPD_GMCHG

Move the spindle to the manual tool change point.

LL SUB_GR_TO_SPD

Insert the ground tool in the spindle.

$BREAK

$CASE 4

Leave the spindle tool on the ground.

LL SUB_SPD_GMCHG

Move the spindle to the manual tool change point.

LL SUB_SPD_TO_GR

Remove the tool from the spindle.

$BREAK

$CASE 5

Leave the spindle tool in the magazine and take another one from the magazine.

LL SUB_SPD_AUTCHG

Move the spindle to the automatic tool change point

LL SUB_SPD_TO_CH2

Take the spindle tool with holder 2.

LL SUB_MZ_TO_CH1

Take the tool from the magazine with holder 1.

LL SUB_CH2_TO_MZ

Leave the tool of holder 2 in the magazine.

LL SUB_CH1_TO_SPD

Take the tool of holder 1 to the spindle.

$BREAK

$CASE 6

Leave the spindle tool in the magazine and take another one from ground.

LL SUB_SPD_AUTCHG

Move the spindle to the automatic tool change point

LL SUB_SPD_TO_CH2

Take the spindle tool with holder 2.

LL SUB_CH2_TO_MZ

Leave the tool of holder 2 in the magazine.

LL SUB_SPD_GMCHG

Move the spindle to the manual tool change point.

LL SUB_GR_TO_SPD

Insert the ground tool in the spindle.

$BREAK

$CASE 7

Leave the spindle tool on the ground and take another one from the magazine.

LL SUB_SPD_GMCHG

Move the spindle to the manual tool change point.

LL SUB_SPD_TO_GR

Remove the tool from the spindle.

LL SUB_SPD_AUTCHG

Move the spindle to the automatic tool change point

LL SUB_MZ_TO_CH1

Take the tool from the magazine with holder 1.

LL SUB_CH1_TO_SPD

Take the tool of holder 1 to the spindle.

$BREAK

$CASE 8

Leave the spindle tool on the ground and take another one from ground.

LL SUB_SPD_GMCHG

Move the spindle to the manual tool change point.

LL SUB_SPD_TO_GR

Remove the tool from the spindle.

LL SUB_GR_TO_SPD

Insert the ground tool in the spindle.

$BREAK

$CASE 9

Take a ground tool to the magazine going through the spindle.

LL SUB_SPD_GMCHG

Move the spindle to the manual tool change point.

LL SUB_GR_TO_SPD

Insert the ground tool in the spindle.

LL SUB_SPD_AUTCHG

Move the spindle to the automatic tool change point

LL SUB_SPD_TO_CH2

Take the spindle tool with holder 2.

LL SUB_CH2_TO_MZ

Leave the tool of holder 2 in the magazine.

$BREAK

$CASE 10

Take a tool from the magazine and leave on the ground going through the spindle.

LL SUB_SPD_AUTCHG

Move the spindle to the automatic tool change point

LL SUB_MZ_TO_CH1

Take the tool from the magazine with holder 1.

LL SUB_CH1_TO_SPD

Take the tool of holder 1 to the spindle.

LL SUB_SPD_GMCHG

Move the spindle to the manual tool change point.

LL SUB_SPD_TO_GR

Remove the tool from the spindle.

$BREAK

$ENDSWITCH

End of analysis of type of operation.

$WHILE V.TM.MZWAIT == 1

$ENDWHILE

Wait for the tool manager.

#RET

End of M06 subroutine.

7.10.5 Basic PLC programming.

When executing the -T- function

When executing a T function, the tool manager sends the code TMOPERATION=11 to the PLC. In general, it is an optimization of the change that permits orienting the magazine while machining.

In this case, the magazine is not oriented and the MZROT mark is activated to indicate that the operation has been completed.

```
DFU TMOPSTROBE AND CPS TMOPERATION EQ 11 = SET MZROT
```

Activate the MZROT mark to "tell" the tool manager that the operation has finished.

When executing the M06 function

When executing the M06 function, the tool manager sends to the PLC, in the TMOPERATION register, the code for the operation to be carried out.

```
DFU TMOPSTROBE = MOV TMOPERATION R101
```

This instruction transfers the TMOPERATION value to register R101 so it is managed by the M06 subroutine.

Every time the M06 subroutine ends an action, it lets the PLC know so it activates the relevant mark of the tool manager.

```
DFU M1101 = SET MZTOCH1
DFD MZTOCH1 = RES M1101
```

The tool has been taken from the magazine to holder 1.

```
DFU M1102 = SET CH1TOSPDL
DFD CH1TOSPDL = RES M1102
```

The tool has been taken from holder 1 to the spindle.

```
DFU M1104 = SET SPDLTOCH2
DFD SPDLTOCH2 = RES M1104
```

The tool has been taken from the spindle to holder 2.

```
DFU M1106 = SET CH2TOMZ
DFD CH2TOMZ = RES M1106
```

The tool has been taken from holder 2 to the magazine.

```
DFU M1107 = SET SPDLTOGR
DFD SPDLTOGR = RES M1107
```

The tool has been taken from the spindle to ground.

```
DFU M1108 = SET GRTOSPDL
DFD GRTOSPDL = RES M1108
```

The tool has been taken from ground to the spindle.

The M06 subroutine uses the following M functions to "tell" the PLC which movements it must carry out.

M101	Select in the magazine the position indicated by TAKEPOS and take the tool with holder 1.
M102	Take the tool of holder 1 to the spindle.
M104	Take the spindle tool with holder 2.
M106	Select in the magazine the position indicated by LEAVEPOS and leave the tool of holder 2.

Programming it depends on the type of machine. The auxiliary function will conclude after executing the requested movement.

Certain operations require using the information transferred by the tool manager in the following registers:

LEAVEPOS	This register indicates the magazine position to leave the tool.
TAKEPOS	This register indicates the magazine position of the tool to be taken.

Manager emergency signal.

Treatment of the tool manager emergency signal.

```
DFU B11KEYBD1 AND NOT TMINEM = SET SETTMEM
DFU TMINEM = RES SETTMEM
```

Pressing the USER12 key activates the emergency.

```
TMINEM = B11KEYLED1
```

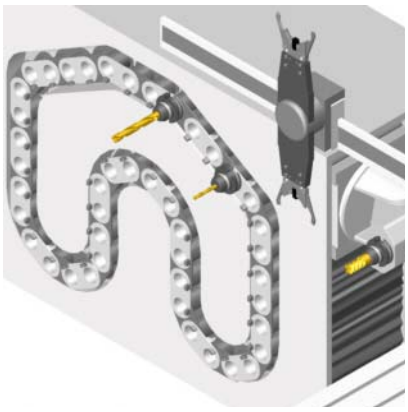
The lamp of the USER12 key turns on when there is an emergency.

```
TMINEM AND DFU B12KEYBD1 = SET RESTMEM
```

Pressing the USER13 key removes the emergency.

7.11 Asynchronous magazine with changer arm.

Asynchronous magazines have the magazine far fromto the spindle and the tool is changen using a tool changer arm. Most of the movements may be carried out while machining the part, thus minimizing machining time.



The communication between the manager and the PLC takes place in two stages; the first one when executing the T function and the second one when executing the M06 function.

Execution of the T function.

1. When the CNC executes the T function, it lets the tool manager know about it.
2. The tool manager sends a command to the PLC to select the next tool in the magazine.
3. The CNC continues with the execution of the program without waiting for the tool manager to complete the operation.

Execution of the M06 function.

1. When the CNC executes the M06 function, it calls the associated subroutine. The M06 function must also be programmed inside that subroutine so the CNC "tells" the tool manager to start making the tool change.
2. The tool manager sends a command to the PLC to make the change.
3. The CNC waits for the tool manager to complete the operation before continuing with the execution of the program.

The management of the tool change should be included in the subroutine associated with the M06 and leave the control of the external devices up to the PLC. Use the auxiliary functions to govern the various devices (magazine rotation, magazine movement, tool changer arm, etc.) from the M06 subroutine.

If the tool change implies leaving a tool in the magazine, once executed the change and with the tool in the arm, it is possible to activate the TCHANGEOK mark for the CNC to continue executing the program while the tool is being left in the magazine.

7.11.1 Valid operations and marks activated by the PLC with each one of them.

Valid operations of the tool manager for this magazine.

The possible operations in this type of magazine are:

TMOPERATION	Meaning.
0	Do nothing.
1	Take a tool from the arm and insert it in the spindle.
2	Leave the spindle tool in the magazine.
3	Insert a ground tool in the spindle.
4	Leave the spindle tool on the ground.
5	Leave the spindle tool in the magazine and take a tool from the arm. The tools of the spindle and of the arm are from the same magazine.
6	Leave the spindle tool in the magazine and take a tool from ground.
7	Leave the spindle tool on the ground and take a tool from the arm.
8	Leave the spindle tool on the ground and take a tool from ground.
9	Take a ground tool to the magazine going through the spindle.
10	Take a tool from the magazine and leave on the ground going through the spindle.
11	Take a tool from the magazine and insert it in the arm.
14	Leave the spindle tool in the magazine and take a tool from the arm. The tools of the spindle and of the arm are from the different magazines.

Values of registers TAKEPOS and LEAVEPOS used by the manager.

The TAKEPOS and LEAVEPOS signals can assume the following values.

Value	Meaning
0	Do nothing.
#	Magazine position.
-1	Claw ·1· of the tool changer arm.
-4	Ground tool.

Here is a summary table of the TAKEPOS and LEAVEPOS values used by the manager in each operation as well as the marks that must be activated by the PLC at the end of each operation.

The TCHANGEOK mark is optional and must only be used when possible. Once the tool change is done and the tool is in the arm, activate the TCHANGEOK mark to resume the execution of the program while leaving the tool in the magazine.

TM => PLC TMOPERATION	TAKEPOS	LEAVEPOS	PLC => TM
1	-1	0	CH1TOSPDL
2	0	#	SPDLTOCH2 + TCHANGEOK + CH2TOMZ
3	0	0	GRTOSPDL
4	0	0	SPDLTOGR
5	-1	#	(a) SPDLTOCH2 + CH2TOMZ + CH1TOSPDL (b) SPDLTOCH2 + CH1TOSPDL + TCHANGEOK + CH2TOMZ

TM => PLC TMOPERATION	TAKEPOS	LEAVEPOS	PLC => TM
6	-4	#	(a) SPDLTOCH2 + CH2TOMZ + GRTOSPDL (b) SPDLTOCH2 + GRTOSPDL + TCHANGEOK + CH2TOMZ
7	-1	-4	SPDLTOGR + CH1TOSPDL
8	0	0	SPDLTOGR + GRTOSPDL
9	-4	#	GRTOSPDL + SPDLTOCH2 + TCHANGEOK + CH2TOMZ
10	#	-4	MZTOCH1 + CH1TOSPDL + SPDLTOGR
11	# #	0 #	MZTOCH1 CH1TOMZ + MZTOCH1
14	-1	#	SPDLTOCH2 + CH2TOMZ + CH1TOSPDL

Example of operation.

The following example shows, assuming that there is no tool in the spindle, it shows the functions executed by the CNC, the values sent by the tool manager to the PLC in each operation and the marks to be activated by the PLC in each case.

It is a non-random magazine, each tool occupies the position of its own number and T7, T8 and T9 are ground tools.

CNC	TMOPERATION	TAKEPOS	LEAVEPOS	
T1 M6	11 1	1 -1	0 0	MZTOCH1 CH1TOSPDL
T2 M6	11 5	2 -1	0 1	MZTOCH1 SPDLTOCH2 + CH1TOSPDL + CH2TOMZ
T7 M6	11 6	0 -4	0 2	MZROT SPDLTOCH2 + CH2TOMZ + GRTOSPDL
T8 M6	11 8	0 0	0 0	MZROT SPDLTOGR + GRTOSPDL
T3 T4 M6	11 11 7	0 0 4	0 0 -4	MZTOCH1 CH1TOMZ + MZTOCH1 SDPLTOGR + CH1TOSPDL
T0 M6	11 2	0 0	0 4	MZROT SPDLTOCH2 + CH2TOMZ
T9 M6	11 3	0 0	0 0	MZROT GRTOSPDL
T0 M6	11 4	0 0	0 0	MZROT SPDLTOGR

7.11.2 Detailed description of the operations of the magazine.

Here is a sample of a detailed description of the valid operations for this magazine. For each operation, it indicates which of the TAKEPOS, LEAVEPOS, NEXTPOS and MZID signals are involved and their meanings. It also shows the sequence to follow to complete the operation.

The TAKEPOS, LEAVEPOS and NEXTPOS signals can assume the following values.

Value	Meaning
0	Do nothing.
#	Magazine position.
-1	Claw ·1· of the tool changer arm.
-4	Ground tool.

TMOPERATION = 1

Take a tool from the arm and insert it in the spindle.

TAKEPOS=-1 Tool in the changer arm.

The sequence of this operation is the following. Previously, while machining (when executing the T) the manager sends the code: TMOPERATION=11 to take the tool from the magazine with claw 1.

1 Insert the tool of holder 1 in the spindle and activate CH1TOSPDL.

TMOPERATION = 2

Leave the spindle tool in the magazine.

LEAVEPOS=# Magazine position to leave the tool.

The sequence of this operation is the following.

1 Take the tool from the spindle with holder ·2· and activate the mark SPDLTOCH2.
2 Start sending the changer arm to the magazine to leave the tool in holder 2.
3 When the arm leaves the collision zone, activate TCHANGEOK, if so desired, to continue executing the program.
4 Leave the tool of holder ·2· in the magazine and activate the mark CH2TOMZ.

TMOPERATION = 3

Insert a ground tool in the spindle.

The sequence of this operation is the following.

1 When the operation is completed, activate the mark: GRTOSPDL.

TMOPERATION = 4

Leave the spindle tool on the ground.

The sequence of this operation is the following.

1 When the operation is completed, activate the mark: SPDLTOGR.

TMOPERATION = 5

Leave the spindle tool in the magazine and take a tool from the arm. The tools of the spindle and of the arm are from the same magazine.

TAKEPOS=-1 Tool in the changer arm.

LEAVEPOS=# Magazine position to leave the tool.

In this operation, the tool manager admits 2 sequences. Previously, while machining (when executing the T) the manager sends the code: TMOPERATION=11 to take the tool from the magazine with claw 1.

First sequence:

1 Take the tool from the spindle with holder ·2· and activate the mark SPDLTOCH2.
2 Leave the tool of holder ·2· in the magazine and activate the mark CH2TOMZ.
3 Insert the tool of holder 1 in the spindle and activate CH1TOSPDL.

Second sequence:

1 Take the tool from the spindle with holder ·2· and activate the mark SPDLTOCH2.
2 Insert the tool of holder 1 in the spindle and activate CH1TOSPDL.
3 Start sending the changer arm to the magazine to leave the tool in holder 2.
4 When the arm leaves the collision zone, activate TCHANGEOK, if so desired, to continue executing the program.
5 Leave the tool of holder ·2· in the magazine and activate the mark CH2TOMZ.

TMOPERATION = 6

Leave the spindle tool in the magazine and take a tool from ground.

TAKEPOS=-4 Ground tool.
LEAVEPOS=# Magazine position to leave the tool.

In this operation, the tool manager admits 2 sequences.

First sequence:

1 Take the tool from the spindle with holder ·2· and activate the mark SPDLTOCH2.
2 Leave the tool of holder ·2· in the magazine and activate the mark CH2TOMZ.
3 Leave the ground tool in the spindle and activate the mark GRTOSPDL.

Second sequence:

1 Take the tool from the spindle with holder ·2· and activate the mark SPDLTOCH2.
2 Leave the ground tool in the spindle and activate the mark GRTOSPDL.
3 Start sending the changer arm to the magazine to leave the tool in holder 2.
4 When the arm leaves the collision zone, activate TCHANGEOK, if so desired, to continue executing the program.
5 Leave the tool of holder ·2· in the magazine and activate the mark CH2TOMZ.

TMOPERATION = 7

Leave the spindle tool on the ground and take a tool from the arm.

TAKEPOS=-1 Tool in the changer arm.
LEAVEPOS=-4 Ground tool.

The sequence of this operation is the following. Previously, while machining (when executing the T) the manager sends the code: TMOPERATION=11 to take the tool from the magazine with claw 1.

1 Leave the tool of the spindle on the ground and activate the mark SPDLTOGR.
2 Insert the tool of holder 1 in the spindle and activate CH1TOSPDL.

TMOPERATION = 8

Leave the spindle tool on the ground and take a tool from ground.

The sequence of this operation is the following.

1 Leave the tool of the spindle on the ground and activate the mark SPDLTOGR.
2 Leave the ground tool in the spindle and activate the mark GRTOSPDL.

TMOPERATION = 9

Take a ground tool to the magazine going through the spindle.

TAKEPOS=-4 Ground tool.
LEAVEPOS=# Magazine position to leave the tool.

The sequence of this operation is the following.

1 Leave the ground tool in the spindle and activate the mark GRTOSPDL.
2 Take the tool from the spindle with holder ·2· and activate the mark SPDLTOCH2.
3 Start sending the changer arm to the magazine to leave the tool in holder 2.
4 When the arm leaves the collision zone, activate TCHANGEOK, if so desired, to continue executing the program.
5 Leave the tool of holder ·2· in the magazine and activate the mark CH2TOMZ.

TMOPERATION = 10

Take a tool from the magazine and leave on the ground going through the spindle.

TAKEPOS=#	Magazine position to pick up the tool.
LEAVEPOS=-4	Ground tool.

The sequence of this operation is the following.

1 Insert the tool of holder 1 in the spindle and activate CH1TOSPDL.
2 Leave the tool of the spindle on the ground and activate the mark SPDLTOGR.

TMOPERATION = 11

Take a tool from the magazine and insert it in the arm.

This operation is an optimization of the tool change that permits placing a tool in claw ·1· of the tool changer arm while machining. This operation may be used in the following cases.

First case. When executing a T function while machining.

TAKEPOS=#	Position occupied by the tool.

1 Take the tool from the magazine with holder 1 and activate the mark MZTOCH1.

Second case. When requesting a new tool and there is another one in holder 1 of the arm.

TAKEPOS=#	Position occupied by the tool.
LEAVEPOS=#	Position to leave the tool.

1 Leave the tool of holder 1 in the magazine and activate the mark CH1TOMZ.
2 Take the tool from the magazine with holder 1 and activate the mark MZTOCH1.

TMOPERATION = 14

Leave the spindle tool in the magazine and take a tool from the arm. The tools of the spindle and of the arm are from the different magazines.

TAKEPOS=-1	Tool in the changer arm.
LEAVEPOS=#	Magazine position to leave the tool.
MZID	The lower portion of this register indicates the destination magazine for the tool and the higher portion the source magazine for the tool.

The sequence of this operation is the following. In this operation, the tool manager admits 2 sequences. Previously, while machining (when executing the T) the manager sends the code: TMOPERATION=11 to take the tool from the magazine with claw 1.

1 Take the tool from the spindle with holder ·2· and activate the mark SPDLTOCH2.
2 Leave the tool of holder ·2· in the magazine and activate the mark CH2TOMZ.
3 Insert the tool of holder 1 in the spindle and activate CH1TOSPDL.

7.11.3 Communication between the PLC and the M06 subroutine.

The communication between the PLC and the M06 subroutine takes place using a series of generic marks and registers. The program of the M06 subroutine offered as an example uses the following marks and registers.

Communication between the PLC and the M06 subroutine.

Registers used by the PLC to transfer information to the M06 subroutine:

R101 Type of operation requested by the tool manager (value of TMOPERATION)

Communication between the PLC and the M06 subroutine

Marks activated by the M06 subroutine so the PLC activates the relevant mark of the tool manager.

M1100 TCHANGEOK

Continue executing the program.

M1101 MZTOCH1

The tool has been taken from the magazine to holder 1.

M1102 CH1TOSPDL

The tool has been taken from holder 1 to the spindle.

M1104 SPDLTOCH2

The tool has been taken from the spindle to holder 2.

M1107 SPDLTOGR

The tool has been taken from the spindle to ground.

M1108 GRTOSPDL

The tool has been taken from ground to the spindle.

The PLC sets the CH2TOMZ mark when the tool has been left.

M functions at the PLC

M functions that imply movements controlled by the PLC:

M101 Select in the magazine the position indicated by TAKEPOS and take the tool with holder 1.

M102 Take the tool of holder 1 to the spindle.

M104 Take the spindle tool with holder 2.

M106 Start sending the changer arm to the magazine to leave the tool in holder 2.

M121 Take the changer arm to the change point.

M122 Take the changer arm to the magazine.

M123 Retract the changer arm.

Set all the M functions with "before-before" synchronization to the program continues after the M function is completed.

The PLC must consider the M106 completed when the arm exits the collision zone and machining is possible.

7.11.4 Program of the M06 subroutine.

%L SUB_MZ_TO_CH1

Take the tool from the magazine with holder 1.

M101

Auxiliary function to execute an action.

V.PLC.M[1101]=1

MZTOCH1 mark to the tool manager.

#RET

%L SUB_CH1_TO_SPD

Take the tool of holder 1 to the spindle.

M102

Auxiliary function to execute an action.

V.PLC.M[1102]=1

CH1TOSPDL mark to the tool manager.

#RET

%L SUB_SPD_TO_CH2

Take the spindle tool with holder 2.

M104

Auxiliary function to execute an action.

V.PLC.M[1104]=1

SPDLTOCH2 mark to the tool manager.

#RET

%L SUB_CH2_TO_MZ

Start sending the changer arm to the magazine to leave the tool in holder 2.

M106

Auxiliary function to execute an action.

The PLC must consider the M106 completed when the arm exits the collision zone and machining is possible.

The PLC sets the CH2TOMZ mark when the tool has been left.

#RET

%L SUB_SPD_TO_GR

Remove the tool from the spindle (take it to ground).

It shows the message for the operator to extract the tool and waits for the operation to be completed before removing the message.

#MSG ["Extract tool T%D and press START", V.TM.TOOL]

M0

#MSG [""]

V.PLC.M[1107]=1

SPDLTOGR mark to the tool manager.

#RET

%L SUB_GR_TO_SPD

Insert the ground tool in the spindle.

It shows the message for the operator to insert the tool and waits for the operation to be completed before removing the message.

#MSG ["Insert tool T%D and press START", V.TM.NXTOOL]

M0

#MSG [""]

V.PLC.M[1108]=1

GRTOSPDL mark to the tool manager.

#RET

%L SUB_SPD_GMCHG

Move the spindle to the manual tool change point.

G1 Z_ F_

Move the spindle.

#RET

%L SUB_SPD_AUTCHG

Move the spindle to the automatic tool change point

G1 Z_ F_

Move the spindle.

#RET

%L SUB_ARM_TO_CHG

Take the changer arm to the change point.

M121

Auxiliary function to execute an action.

#RET

%L SUB_ARM_TO_MZ

Take the changer arm to the magazine.

M122

Auxiliary function to execute an action.

#RET

%L SUB_ARM_BACK

Retract the changer arm.

M123

Auxiliary function to execute an action.

#RET

%SUB_M6.nc

M6

Order the tool manager to start the tool change.

$SWITCH V.PLC.R[101]

It analyzes the type of operation.

$CASE 1

Take a tool from the magazine and insert it in the spindle.

Previously, While machining (when executing the T), the tool is taken from the magazine to holder 1.

LL SUB_SPD_AUTCHG

Move the spindle to the automatic tool change point

LL SUB_ARM_TO_CHG

Take the changer arm to the change point.

LL SUB_CH1_TO_SPD

Take the tool of holder 1 to the spindle.

LL SUB_ARM_BACK

Retract the changer arm.

$BREAK

$CASE 2

Leave the spindle tool in the magazine.

LL SUB_SPD_AUTCHG

Move the spindle to the automatic tool change point

LL SUB_ARM_TO_CHG

Take the changer arm to the change point.

LL SUB_SPD_TO_CH2

Take the spindle tool with holder 2.

LL SUB_ARM_BACK

Retract the changer arm.

LL SUB_CH2_TO_MZ

Start sending the changer arm to the magazine to leave the tool in holder 2.

V.PLC.M[1100]=1

Order the PLC to activate the TCHANGEOK mark "telling" the tool manager that it can continue executing.

$BREAK

$CASE 3

Insert the ground tool in the spindle.

LL SUB_SPD_GMCHG

Move the spindle to the manual tool change point.

LL SUB_GR_TO_SPD

Insert the ground tool in the spindle.

$BREAK

$CASE 4

Leave the spindle tool on the ground.

LL SUB_SPD_GMCHG

Move the spindle to the manual tool change point.

LL SUB_SPD_TO_GR

Remove the tool from the spindle.

$BREAK

$CASE 5

Leave the spindle tool in the magazine and take another one from the magazine.

Previously, While machining (when executing the T), the tool is taken from the magazine to holder 1.

LL SUB_SPD_AUTCHG

Move the spindle to the automatic tool change point

LL SUB_ARM_TO_CHG

Take the changer arm to the change point.

LL SUB_SPD_TO_CH2

Take the spindle tool with holder 2.

LL SUB_CH1_TO_SPD

Take the tool of holder 1 to the spindle.

LL SUB_ARM_BACK

Retract the changer arm.

LL SUB_CH2_TO_MZ

Start sending the changer arm to the magazine to leave the tool.

V.PLC.M[1100]=1

Order the PLC to activate the TCHANGEOK mark "telling" the tool manager that it can continue executing.

$BREAK

$CASE 6

Leave the spindle tool in the magazine and take another one from ground.

LL SUB_SPD_AUTCHG

Move the spindle to the automatic tool change point

LL SUB_ARM_TO_CHG

Take the changer arm to the change point.

LL SUB_SPD_TO_CH2

Take the spindle tool with holder 2.

LL SUB_ARM_BACK

Retract the changer arm.

LL SUB_SPD_GMCHG

Move the spindle to the manual tool change point.

LL SUB_GR_TO_SPD

Insert the ground tool in the spindle.

LL SUB_CH2_TO_MZ

Start sending the changer arm to the magazine to leave the tool.

V.PLC.M[1100]=1

Order the PLC to activate the TCHANGEOK mark "telling" the tool manager that it can continue executing.

$BREAK

$CASE 7

Leave the spindle tool on the ground and take another one from the magazine.

Previously, While machining (when executing the T), the tool is taken from the magazine to holder 1.

LL SUB_SPD_GMCHG

Move the spindle to the manual tool change point.

LL SUB_SPD_TO_GR

Remove the tool from the spindle.

LL SUB_SPD_AUTCHG

Move the spindle to the automatic tool change point

LL SUB_ARM_TO_CHG

Take the changer arm to the change point.

LL SUB_CH1_TO_SPD

Take the tool of holder 1 to the spindle.

LL SUB_ARM_BACK

Retract the changer arm.

$BREAK

$CASE 8

Leave the spindle tool on the ground and take another one from ground.

LL SUB_SPD_GMCHG

Move the spindle to the manual tool change point.

LL SUB_SPD_TO_GR

Remove the tool from the spindle.

LL SUB_GR_TO_SPD

Insert the ground tool in the spindle.

$BREAK

$CASE 9

Take a ground tool to the magazine going through the spindle.

LL SUB_SPD_GMCHG

Move the spindle to the manual tool change point.

LL SUB_GR_TO_SPD

Insert the ground tool in the spindle.

LL SUB_SPD_AUTCHG

Move the spindle to the automatic tool change point

LL SUB_ARM_TO_CHG

Take the changer arm to the change point.

LL SUB_SPD_TO_CH2

Take the spindle tool with holder 2.

LL SUB_ARM_BACK

Retract the changer arm.

LL SUB_CH2_TO_MZ

Start sending the changer arm to the magazine to leave the tool.

V.PLC.M[1100]=1

Order the PLC to activate the TCHANGEOK mark "telling" the tool manager that it can continue executing.

$BREAK

$CASE 10

Take a tool from the magazine and leave on the ground going through the spindle.

LL SUB_MZ_TO_CH1

Take the tool from the magazine with holder 1.

LL SUB_SPD_AUTCHG

Move the spindle to the automatic tool change point

LL SUB_ARM_TO_CHG

Take the changer arm to the change point.

LL SUB_CH1_TO_SPD

Take the tool of holder 1 to the spindle.

LL SUB_ARM_BACK

Retract the changer arm.

LL SUB_SPD_GMCHG

Move the spindle to the manual tool change point.

LL SUB_SPD_TO_GR

Remove the tool from the spindle.

$BREAK

$ENDSWITCH

End of analysis of type of operation.

$WHILE V.TM.MZWAIT == 1

$ENDWHILE

Wait for the tool manager.

#RET

End of M06 subroutine.

7.11.5 Basic PLC programming.

When executing the -T- function

When executing the T function, the tool manager sends to the PLC the code: TMOPERATION=11 to take the next tool in the arm and approach it to the spindle while machining.

```
DFU TMOPSTROBE AND CPS TMOPERATION EQ 11 = ···
```

The following operations must be carried out:

Send the changer arm to the magazine.

If LEAVEPOS indicates a magazine position, leave the tool of holder 1 in that position and activate the CH1TOMZ mark.

Take the tool of the TAKEPOS position of the magazine with holder 1. Activate the MZTOCH1 mark to "tell" the tool manager that the tool has been taken.

When executing the M06 function

When executing the M06 function, the tool manager sends to the PLC, in the TMOPERATION register, the code for the operation to be carried out.

```
DFU TMOPSTROBE = MOV TMOPERATION R101
```

This instruction transfers the TMOPERATION value to register R101 so it is managed by the M06 subroutine.

Every time the M06 subroutine ends an action, it lets the PLC know so it activates the relevant mark of the tool manager.

```
DFU M1100 = SET TCHANGEOK
DFD TCHANGEOK = RES M1100
```

Continue executing the program.

```
DFU M1101 = SET MZTOCH1
DFD MZTOCH1 = RES M1101
```

The tool has been taken from the magazine to holder 1.

```
DFU M1102 = SET CH1TOSPDL
DFD CH1TOSPDL = RES M1102
```

The tool has been taken from holder 1 to the spindle.

```
DFU M1104 = SET SPDLTOCH2
DFD SPDLTOCH2 = RES M1104
```

The tool has been taken from the spindle to holder 2.

```
DFU M1107 = SET SPDLTOGR
DFD SPDLTOGR = RES M1107
```

The tool has been taken from the spindle to ground.

```
DFU M1108 = SET GRTOSPDL
DFD GRTOSPDL = RES M1108
The tool has been taken from ground to the spindle.
```

The M06 subroutine uses the following M functions to "tell" the PLC which movements it must carry out.

M101	Select in the magazine the position indicated by TAKEPOS and take the tool with holder 1.
M102	Take the tool of holder 1 to the spindle.
M104	Take the spindle tool with holder 2.
M106	Start sending the changer arm to the magazine to leave the tool in holder 2.
M121	Take the changer arm to the change point.
M122	Take the changer arm to the magazine.
M123	Retract the changer arm.

Programming it depends on the type of machine. The auxiliary functions will be completed after executing the requested movement.

Treatment of the M106 function:

- Completed when the changer arm exits the collision zone and machining is possible.
- Activate the CH2TOMZ mark when the tool has been left to let the tool manager know that the tool has been taken from holder 2 to the magazine".

Certain operations require using the information transferred by the tool manager in the following registers:

LEAVEPOS	This register indicates the magazine position to leave the tool.
TAKEPOS	This register indicates the magazine position of the tool to be taken.

Manager emergency signal.

Treatment of the tool manager emergency signal.

```
DFU B11KEYBD1 AND NOT TMINEM = SET SETTMEM
DFU TMINEM = RES SETTMEM
```

Pressing the USER12 key activates the emergency.

```
TMINEM = B11KEYLED1
```

The lamp of the USER12 key turns on when there is an emergency.

```
TMINEM AND DFU B12KEYBD1 = SET RESTMEM
```

Pressing the USER13 key removes the emergency.

7.

TOOL AND MAGAZINE MANAGEMENT.

Asynchronous magazine with changer arm.

FAGOR

CNC 8060
CNC 8065

(REF: 1402)

KEY CODES.

8

The key codes may be used to know which is the last key accepted by the CNC and also to simulate the keyboard from the PLC.

The codes are associated with the physical key (the location of the key on the keyboard), not with the printed character. For example, the code $27 on a Spanish keyboard corresponds to the character "Ñ" and on an English keyboard corresponds to the character ";".

The print on the keyboards supplied by Fagor only coincides with the codes if the Spanish language has been selected on the keyboard. See *"8.2 Assigning codes for a Spanish keyboard."* on page 430.

Codes to press and release the keys.

Each key is assigned two codes; one for pressing it and another one for releasing it. The following drawings show both codes in each key; the top one is for pressing the key and the bottom one for releasing it.

$10 $90

The code $10 indicates that the key has been pressed and the code $90 indicates that the key has been released.

Both codes must be written when sending each key from the PLC. When sending the two codes, it is recommended to use a delay between them (for safety).

Consulting the last key accepted by the CNC.

This consultation is made using the variable (V.)G.KEY. This variable may be consulted from the CNC, from the PLC and from an external interface. The consultation from the PLC is made with the command CNCRD.

```
Reading from the PLC. The key code is saved in register R101.
    ···=CNCRD(G.KEY,R101,M100)
```

Keyboard simulation from the PLC.

This consultation is made using the variable (V.)G.KEY. To read the last key that has been accepted by the CNC or simulate the keyboard from the PLC by writing in it the code of the desired key.

This variable is read and written using the commands CNCRD and CNCWR.

```
CNCRD(variable, register, mark)
CNCWR(register, variable, mark)
```

The CNCRD command saves the value of the variable in the register. The CNCWR writes the value of the register in the variable. In both cases, the mark stays active during the whole operation.

```
Reading from the PLC. The key code is saved in register R101.
    ···=CNCRD(G.KEY,R101,M100)
Writing from the PLC. The key code is read from register R102.
    ···=CNCWR(R102,G.KEY,M100)
```

Keystroke combination.

To simulate from a keystroke combination from the PLC, for example [CTRL]+[F1], proceed as follows.

1 [CTRL] key. Send the code for pressing the key.
2 [F1] key. Send the code for pressing the key.
3 [CTRL] key. Send the code for releasing the key.
4 [F1] key. Send the code for releasing the key.

When sending the key pressing code and the key releasing codes, it is recommended to use a delay between them (for safety).

CNC's own keys.

The [START], [STOP] and [RESET] keys may be actuated from the PLC using the CYSTART, _STOP and RESETIN keys respectively. The rest of the CNC's own keys, such as the ones for accessing work modes, may be simulated through their hotkeys.

Hotkey.	Work mode.	
[CTRL] + [A]	Task window.	
[CTRL] + [SHIFT] + [F1]	Main menu.	MAIN MENU
[CTRL] + [F6]	Automatic mode.	AUTO
[CTRL] + [F7]	Jog mode.	MANUAL
[CTRL] + [F8]	MDI mode.	MDI
[CTRL] + [F9]	EDISIMU mode.	EDIT
[CTRL] + [F10]	User tables.	TABLES
[CTRL] + [F11]	Tool and magazine table.	TOOLS
[CTRL] + [F12]	Utilities mode.	UTILITIES
[SHIFT] + [F1]	User mode.	

Hotkey.	Using the screens.	
[CTRL] + [F1]	Previous horizontal menu.	BACK
[CTRL] + [F2]	Window change.	FOCUS
[CTRL] + [F3]	Screen change.	NEXT

Hotkey.	Operations at the CNC.	
[ALT] + [B]	Two-color key.	
[CTRL] + [F4]	HELP.	HELP
[CTRL] + [F5]	RECALL.	RECALL
[CTRL] + [B]	"Single block" mode.	SINGLE
	Home search.	ZERO
[CTRL] + [J]	Show/Hide virtual operator panel.	

Hotkey.	Operations at the CNC.	
[CTRL] + [M]	Show/Hide PLC messages.	
[CTRL] + [O]	Show/Hide CNC messages.	
[CTRL] + [W]	Minimize/Restore the CNC.	
[ALT] + [F4]	Turn the CNC off.	
[ALT] + [S]	Channel synchronization window.	

8.1 Example for simulating the keyboard from the PLC.

Pressing the first customizable key (B0KEYBD1) carries out the following operations:

1. The manual mode is accessed at the CNC.
2. Then, the MDI mode is accessed.
3. It homes the X axis.
4. After the home search, it exits the MDI mode.

For each key to be sent from the PLC, the codes for "press key" and for the "release key" must be written. When sending both codes, the example uses a 200 ms delay between them (for safety).

```
START OR DFU M313 = CYSTART

() = MOV $1D R200 = MOV $9D R201          ;CTRL
() = MOV $41 R202 = MOV $C1 R203          ;F7
() = MOV $42 R204 = MOV $C2 R205          ;F8
() = MOV $22 R206 = MOV $A2 R207          ;G
() = MOV $08 R208 = MOV $88 R209          ;7
() = MOV $05 R210 = MOV $85 R211          ;4
() = MOV $2D R212 = MOV $AD R213          ;X
() = MOV $02 R214 = MOV $82 R215          ;1
() = MOV $01 R216 = MOV $81 R217          ;ESC

; The CNC goes into jog mode. [CTRL] + [F7]
DFU B0KEYBD1 = CNCWR(R200,G.KEY,M200) = CNCWR(R202,G.KEY,M201)=TG1 200 200
T200 = M300
DFD M300 = CNCWR(R201,G.KEY,M202)= CNCWR(R203,G.KEY,M203)=TG1 201 200
T201 = M301

; The CNC goes into MDI mode. [CTRL] + [F8]
DFD M301 = CNCWR(R200,G.KEY,M200) = CNCWR(R204,G.KEY,M204) = TG1 202 200
T202 = M302
DFD M302 = CNCWR(R201,G.KEY,M202)= CNCWR(R205,G.KEY,M205) = TG1 203 200
T203 =M303

; Home search. G74
DFD M303 = CNCWR(R206,G.KEY,M206) = TG1 204 200      ;G
T204 = M304
DFD M304 = CNCWR(R207,G.KEY,M207) = TG1 205 200
T205 = M305
DFD M305 = CNCWR(R208,G.KEY,M208) = TG1 206 200      ;7
T206 = M306
DFD M306 = CNCWR(R209,G.KEY,M209) = TG1 207 200
T207 = M307
DFD M307 = CNCWR(R210,G.KEY,M210) = TG1 208 200      ;4
T208 = M308
DFD M308 = CNCWR(R211,G.KEY,M211) = TG1 209 200
T209 = M309
DFD M309 = CNCWR(R212,G.KEY,M212) = TG1 210 200      ;X
T210 = M310
DFD M310 = CNCWR(R213,G.KEY,M213) = TG1 211 200
T211 = M311
DFD M311 = CNCWR(R214,G.KEY,M214) = TG1 212 200      ;1
T212 = M312
DFD M312 = CNCWR(R215,G.KEY,M215) = TG1 213 200
T213 = M313
DFD M313 = SET M500                                  ;Execute Cycle Start (CYSTART=1)
DFD ZERO = SET M501

;Quit MDI mode
()= CNCRD(G.STATUS,R220,M220)                        ;CNC status ("1"=READY)
```

M500 AND M501 AND (CPS R220 EQ 1) = CNCWR(R216,G.KEY,M216) = TG1 214 200

T214 = M314

DFD M314 = CNCWR(R217,G.KEY,M217)= RES M500=RES M501

8.2 Assigning codes for a Spanish keyboard.

The following tables show the character associated with each code on a keyboard configured for the Spanish language.

Alphanumeric keyboard (alphabetical keys).

Press key.	Release key.	Character.	[SHIFT]	[ALT]+[CTRL]
$1E	$9E	a	A	
$30	$B0	b	B	
$2E	$AE	c	C	
$20	$A0	d	D	
$12	$92	e	E	
$21	$A1	f	F	
$22	$A2	g	G	
$23	$A3	h	H	
$17	$97	i	I	
$24	$A4	j	J	
$25	$A5	k	K	
$26	$A6	l	L	
$32	$B2	m	M	
$31	$B1	n	N	
$27	$A7	ñ	Ñ	
$18	$98	o	O	
$19	$99	p	P	
$10	$90	q	Q	
$13	$93	r	R	
$1F	$9F	s	S	
$14	$94	t	T	
$16	$96	u	U	
$2F	$AF	v	V	
$11	$91	w	W	
$2D	$AD	x	X	
$15	$95	y	Y	
$2C	$AC	z	Z	
$2B	$AB	ç	Ç	}

Alphanumeric keyboard (numerical keys).

Press key.	Release key.	Character.	[SHIFT]	[ALT]+[CTRL]
$0B	$8B	0	=	|
$02	$82	1	!	@
$03	$83	2		"#
$04	$84	3	·	
$05	$85	4	$	
$06	$86	5	%	?
$07	$87	6	&	¬
$08	$88	7	/	
$09	$89	8	(	
$0A	$8A	9	)	

Alphanumeric keyboard (other characters).

Press key	Release key	Character	[SHIFT]	[ALT]+[CTRL]
$29	$A9	º	ª	\
$0C	$8C	'	?	
$0D	$8D	¡	¿	
$1A	$9A	`	^	[
$1B	$9B	+	*	]
$28	$A8	´	¨	{
$33	$B3	,	;	
$34	$B4	.	:	
$35	$B5	-	_	
$56	$D6	<	>	

Alphanumeric keyboard (other keys).

Press key.	Release key.	
$01	$81	ESC
$0F	$8F	Tab
$3A	$BA	Shift lock
$2A	$AA	Left SHIFT
$1D	$9D	Left CTRL
$5B	$DB	Left Windows
$38	$B8	ALT
$39	$B9	Space bar
$E0$38	$E0$B8	ALT GR
$E0$5C	$E0$DC	Right Windows
$E0$5D	$E0$DD	Windows menu
$E0$1D	$E0$9D	Right CTRL
$36	$B6	Right SHIFT
$1C	$9C	ENTER
$0E	$8E	Backspace - Delete

Numeric keypad.

Press key.	Release key.	
$45	$C5	Num Lock
$E0$35	$E0$B5	Divide key
$37	$B7	Multiply key.
$4A	$CA	Subtract key.
$4E	$CE	Add key.
$E0$1C	$E0$9C	ENTER
$4F	$CF	Number 1.
$50	$D0	Number 2.
$51	$D1	Number 3.
$4B	$CB	Number 4.
$4C	$CC	Number 5.
$4D	$CD	Number 6.
$47	$C7	Number 7.
$48	$C8	Number 8.

Press key.	Release key.	
$0E	$8E	Number 9.
$52	$D2	Number 0.
$53	$D3	Decimal point key.

Function keys.

Press key.	Release key	
$3B	$BB	F1
$3C	$BC	F2
$3D	$BD	F3
$3E	$BE	F4
$3F	$BF	F5
$40	$C0	F6
$41	$C1	F7
$42	$C2	F8
$43	$C3	F9
$44	$C4	F10
$57	$D7	F11
$58	$D8	F12

Other keys

Press key.	Release key.	
$E0$37	$E0$B7	Print screen.
$46	$C6	Lock/Unlock
$E0$45	$E0$C5	Pause.
$E0$46	$E0$C6	Interrupt
$E0$52	$E0$D2	Insert.
$E0$47	$E0$C7	Begin.
$E0$49	$E0$C9	Page up
$E0$53	$E0$D3	Delete
$E0$4F	$E0$CF	End
$E0$51	$E0$D1	Page down
$E0$48	$E0$C8	Up arrow.
$E0$50	$E0$D0	Down arrow.
$E0$4D	$E0$CD	Right arrow.
$E0$4B	$E0$CB	Left arrow.

9.1 Configure the name and number of axes and spindles.

Valid names for axes and spindles.

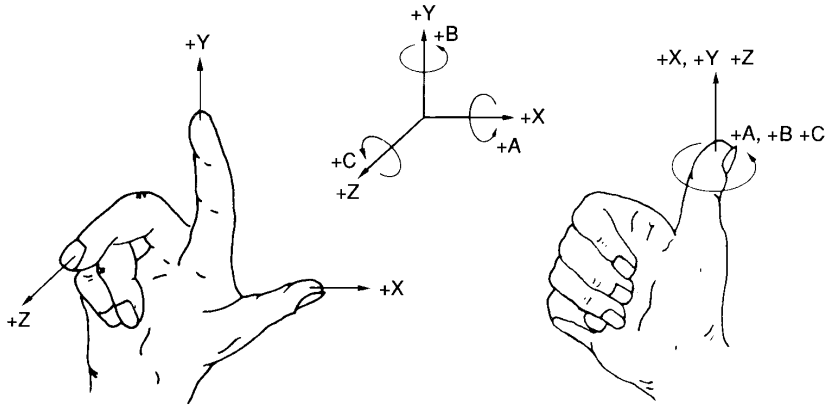
The axis name is defined by 1 or 2 characters. The first character must be one of the letters X - Y - Z - U - V - W - A - B - C. The second character is optional and will be a numerical suffix between 1 and 9. This way, the name of the axes may be within the range X, X1…X9,...C, C1…C9. For example X, X1, Y3, Z9, W, W7, C...

The spindle name is defined by 1 or 2 characters. The first character must be the letter -S- . The second character is optional and will be a numerical suffix between 1 and 9. This way, the name of the spindles may be within the range S, S1 ... S9.

Name of the axes according to their type. DIN66217 standard.

Any of the names mentioned earlier may be assigned to any type of axis (rotary, auxiliary, etc.). However, if possible, we recommend to apply the DIN 66217 standard when naming the axes of the machine. The DIN 66217 standard names the various types of axes as follows.

Name.	Type of axis according to the DIN 66217 standard.
Xn Yn Zn	Main axes. Two axes make up the work plane and the third axis corresponds to the axis perpendicular to the plane.
Un Vn Wn	Auxiliary axes, parallel to X-Y-Z respectively.
An Bn Cn	Rotary axes, on X-Y-Z respectively.



9.1.1 Configure the number of axes and spindles of the system.

The axes and spindles of the system are configured using machine parameters NAXIS, AXISNAME, NSPDL and SPDLNAME. The order the axes and spindles are defined in parameters AXISNAME and SPDLNAME determines their logic number.

Parameter.	Meaning.
NAXIS	Number of axes of the system.
AXISNAME	List of axes of the system.
NSPDL	Number of spindles of the system
SPDLNAME	List of spindles of the system.

Logic number of the axes and spindles.

As with the name, the logic number permits identifying the axis or spindle in PLC variables, marks, etc.

The logic number of the axes and spindles is determined by the order in which they have been defined in machine parameters AXISNAME and SPDLNAME. The first axis of the AXISNAME table will be logic axis ·1· and so on. The logic numbering of the spindles continues from the last logic axis; hence, in a 3-axis system, the first spindle of the SPDLNAME table will be logic spindle ·4· and so on.

AXISNAME	SPDLNAME	Logic order.
AXISNAME 1		Logic number ·1·.
AXISNAME 2		Logic number ·2·.
AXISNAME 3		Logic number ·3·.
	SPDLNAME 1	Logic number ·4·.
	SPDLNAME 2	Logic number ·5·.

Index of the spindles in the system.

As with the name, the index may be used to identify the spindle in the variables.

The index of a spindle in the system is determined by the order in which it has been defined in machine parameter SPDLNAME. The index of the first spindle of the SPDLNAME table will be ·1· and so on.

AXISNAME SPDLNAME	Logic order.	Index in the system.
AXISNAME 1	Logic number ·1·.	
AXISNAME 2	Logic number ·2·.	
AXISNAME 3	Logic number ·3·.	
SPDLNAME 1	Logic number ·4·.	Spindle with index ·1·.
SPDLNAME 2	Logic number ·5·.	Spindle with index ·2·.

9.1.2 Configure the number of axes and spindles of the channels.

In a single-channel or multi-channel system, the axes and spindles defined in the system must be distributed among the different channels. The axes and spindles of a channel are configured using parameters CHNAXIS, CHAXISNAME, CHNSPDL and CHSPDLNAME. The order the axes and spindles are defined in parameters CHAXISNAME and CHSPDLNAME determines their index in the channel.

Parameter.	Meaning.
CHNAXIS	Number of axes of the channel.
CHAXISNAME	List of axes of the channel.
CHNSPDL	Number of spindles of the channel.
CHSPDLNAME	List of spindles of the channel.

A channel may have initially associated with it one, several or no of the axes defined in the system. In any case, the number of axes assigned to the channel cannot be higher than the number of axes of the system, defined by parameter NAXIS. The sum of the axes assigned to the channels cannot exceed the number of axes of the system either. This is also valid for the spindles.

Index of an axis or spindle in the channel.

As with the name, the index in the channel permits identifying the axis or spindle in PLC variables, marks, etc.

The logic number of the axes and spindles is determined by the order in which they have been defined in machine parameters CHAXISNAME and CHSPDLNAME. The index of the first axis of the CHAXISNAME table will be ·1· and so on. The index of the first spindle of the CHSPDLNAME table will be ·1· and so on.

CHAXISNAME	CHSPDLNAME	Index in the channel.
CHAXISNAME 1		Axis with index ·1·.
CHAXISNAME 2		Axis with index ·2·.
CHAXISNAME 3		Axis with index ·3·.
	CHSPDLNAME 1	Spindle with index ·1·.
	CHSPDLNAME 2	Spindle with index ·2·.

9.1.3 Configuration examples.

Here are various machine configurations. Each one indicates the logic number and index of each axis and spindle in the channel. Each example also shows the work plane associated with functions G17, G18 and G19.

The lathe examples indicate the value of machine parameter GEOCONFIG because the behavior of the functions related to the work planes changes depending on this parameter.

Example list.	Page.
Milling machine with 1 channel, 3 axes and 1 spindle.	437
Milling machine with 1 channel, 5 axes (2 free) and 1 spindle.	438
Milling machine with 3 channels, 9 axes and 2 spindles. Channel 1: 3 axes and 1 spindle. Channel 2: 4 axes and 1 spindle. Channel 3: 2 axes and no spindle.	439
Lathe with 1 channel, 2 axes and 1 spindle. Configuration of "plane" type axes.	440
Lathe with 1 channel, 3 axes and 1 spindle. Configuration of "plane" type axes.	441
Lathe with 1 channel, 3 axes and 1 spindle. Configuration of trihedron type axes.	442
Lathe with 1 channel, 3 axes (1 free) and 1 spindle. Configuration of "plane" type axes.	443
Lathe with 2 channels, 4 axes and 2 spindles. Channel 1: 2 axes and 1 spindle. Configuration of "plane" type axes. Channel 2: 2 axes and 1 spindle. Configuration of "plane" type axes.	444
Lathe with 3 channels, 6 axes and 2 spindles. Channel 1: 2 axes and 1 spindle. Configuration of "plane" type axes. Channel 2: 2 axes and 1 spindle. Configuration of "plane" type axes. Channel 3: 2 axes and no spindle.	445

MILLING MACHINE WITH 1 CHANNEL, 3 AXES AND 1 SPINDLE.

The axes and spindles are distributed as follows.

Channel ·1·. 3 axes (X Y Z) 1 spindle (S).

General configuration of the channels, axes and spindles.

System.	Channel ·1·.
NCHANNEL = 1	
NAXIS = 3	CHNAXIS = 3
NSPDL = 1	CHNSPDL = 1

Configuration of the axes and spindles of the system.

AXISNAME	Value.	Logic order.	Index in the system.
AXISNAME 1	X	Logic number ·1·.	
AXISNAME 2	Y	Logic number ·2·.	
AXISNAME 3	Z	Logic number ·3·.	
SPDLNAME 1	S	Logic number ·4·.	Spindle with index ·1·.

Configuration of the axes and spindles of channel ·1·.

CHAXISNAME	Value.	Index in the channel.	Logic order.
CHAXISNAME 1	X	Axis with index ·1·.	Logic number ·1·.
CHAXISNAME 2	Y	Axis with index ·2·.	Logic number ·2·.
CHAXISNAME 3	Z	Axis with index ·3·.	Logic number ·3·.

CHSPDLNAME	Value.	Index in the channel.	Logic order.
CHSPDLNAME 1	S	Spindle with index ·1·.	Logic number ·4·.

Plane selection in channel ·1·.

Function.	Selected plane.
G17	X-Y plane. Perpendicular axis Z.
G18	Z-X plane. Perpendicular axis Y.
G19	Y-Z plane. Perpendicular axis X.
G20	Any plane and/or longitudinal axis.

MILLING MACHINE WITH 1 CHANNEL, 5 AXES (2 FREE) AND 1 SPINDLE.

Let us suppose a single-channel machine with three axes and one spindle and two axes that have not been assigned initially. The axes and spindles are distributed as follows.

Channel ·1·. 3 axes (X Y Z) 1 spindle (S).

Not assigned. 2 axes (A B)

General configuration of the channels, axes and spindles.

System	Channel ·1·
NCHANNEL = 1	
NAXIS = 5	CHNAXIS = 3
NSPDL = 1	CHNSPDL = 1

Configuration of the axes and spindles of the system.

AXISNAME	Value.	Logic order.	Index in the system.
AXISNAME 1	X	Logic number ·1·.	
AXISNAME 2	Y	Logic number ·2·.	
AXISNAME 3	Z	Logic number ·3·.	
AXISNAME 4	A	Logic number ·4·.	
AXISNAME 5	B	Logic number ·5·.	
SPDLNAME 1	S1	Logic number ·6·.	Spindle with index ·1·.

Configuration of the axes and spindles of channel ·1·.

CHAXISNAME	Value.	Index in the channel.	Logic order.
CHAXISNAME 1	X	Axis with index ·1·.	Logic number ·1·.
CHAXISNAME 2	Y	Axis with index ·2·.	Logic number ·2·.
CHAXISNAME 3	Z	Axis with index ·3·.	Logic number ·3·.

CHSPDLNAME	Value.	Index in the channel.	Logic order.
CHSPDLNAME 1	S	Spindle with index ·1·.	Logic number ·10·.

Plane selection in channel ·1·.

Function.	Selected plane.
G17	X-Y plane. Perpendicular axis Z.
G18	Z-X plane. Perpendicular axis Y.
G19	Y-Z plane. Perpendicular axis X.
G20	Any plane and/or longitudinal axis.

MILLING MACHINE WITH 3 CHANNELS, 9 AXES AND 2 SPINDLES.

Let us suppose tree-channel machine where the first two channels have axes and spindles for machining and the third channel is a loading and unloading system controlled by two axes. The axes and spindles are distributed as follows.

Channel ·1·.	3 axes (X Y Z)	1 spindle (S1).
Channel ·2·.	4 axes (X1 Y1 Z1 W)	1 spindle (S2).
Channel ·3·.	2 axes (U U1)	0 spindles.

General configuration of the channels, axes and spindles.

System	Channel ·1·	Channel ·2·	Channel ·3·
NCHANNEL = 3			
NAXIS = 9	CHNAXIS = 3	CHNAXIS = 4	CHNAXIS = 2
NSPDL = 2	CHNSPDL = 1	CHNSPDL = 1	CHNSPDL = 0

Configuration of the axes and spindles of the system.

AXISNAME	Value.	Logic order.	Index in the system.
AXISNAME 1	X	Logic number ·1·.	
AXISNAME 2	Y	Logic number ·2·.	
AXISNAME 3	Z	Logic number ·3·.	
AXISNAME 4	X1	Logic number ·4·.	
AXISNAME 5	Y1	Logic number ·5·.	
AXISNAME 6	Z1	Logic number ·6·.	
AXISNAME 7	W	Logic number ·7·.	
AXISNAME 8	U	Logic number ·8·.	
AXISNAME 9	U1	Logic number ·9·.	
SPDLNAME 1	S1	Logic number ·10·.	Spindle with index ·1·.
SPDLNAME 2	S2	Logic number ·11·.	Spindle with index ·2·.

Configuration of the axes and spindles of channel ·1·.

CHAXISNAME	Value.	Index in the channel.	Logic order.
CHAXISNAME 1	X	Axis with index ·1·.	Logic number ·1·.
CHAXISNAME 2	Y	Axis with index ·2·.	Logic number ·2·.
CHAXISNAME 3	Z	Axis with index ·3·.	Logic number ·3·.

CHSPDLNAME	Value.	Index in the channel.	Logic order.
CHSPDLNAME 1	S1	Spindle with index ·1·.	Logic number ·10·.

Configuration of the axes and spindles of channel ·2·.

CHAXISNAME	Value.	Index in the channel.	Logic order.
CHAXISNAME 1	X1	Axis with index ·1·.	Logic number ·4·.
CHAXISNAME 2	Y1	Axis with index ·2·.	Logic number ·5·.
CHAXISNAME 3	Z1	Axis with index ·3·.	Logic number ·6·.
CHAXISNAME 3	W	Axis with index ·4·.	Logic number ·7·.

CHSPDLNAME	Value.	Index in the channel.	Logic order.
CHSPDLNAME 1	S2	Spindle with index ·1·.	Logic number ·11·.

Configuration of the axes and spindles of channel ·3·.

CHAXISNAME	Value.	Index in the channel.	Logic order.
CHAXISNAME 1	U1	Axis with index ·1·.	Logic number ·9·.
CHAXISNAME 2	U	Axis with index ·2·.	Logic number ·8·.

Plane selection in channels ·1· and ·2·.

Function.	Channel ·1·. Selected plane.	Channel ·2·. Selected plane.
G17	X-Y plane. Perpendicular axis Z.	X1-Y1 plane. Perpendicular axis Z1.
G18	Z-X plane. Perpendicular axis Y.	Z1-X1 plane. Perpendicular axis Y1.
G19	Y-Z plane. Perpendicular axis X.	Y1-Z1 plane. Perpendicular axis X1.
G20	Any plane and/or longitudinal axis.	Any plane and/or longitudinal axis.

LATHE WITH 1 CHANNEL, 2 AXES AND 1 SPINDLE. CONFIGURATION OF "PLANE" TYPE AXES.

The axes and spindles are distributed as follows.

Channel ·1·. 2 axes (X Z) 1 spindle (S).

General configuration of the channels, axes and spindles.

System	Channel ·1·
NCHANNEL = 1	GEOCONFIG = Plane
NAXIS = 2	CHNAXIS = 2
NSPDL = 1	CHNSPDL = 1
	IPLANE = G18

Configuration of the axes and spindles of the system.

AXISNAME	Value	Logic order.	Index in the system.
AXISNAME 1	X	Logic number ·1·.	
AXISNAME 2	Z	Logic number ·2·.	
SPDLNAME 1	S	Logic number ·3·.	Spindle with index ·1·.

Configuration of the axes and spindles of channel ·1·.

CHAXISNAME	Value	Index in the channel.	Logic order.
CHAXISNAME 1	X	Axis with index ·1·.	Logic number ·1·.
CHAXISNAME 2	Z	Axis with index ·2·.	Logic number ·2·.

CHSPDLNAME	Value	Index in the channel.	Logic order.
CHSPDLNAME 1	S	Spindle with index ·1·.	Logic number ·3·.

Plane selection in channel ·1·.

With this configuration, the work plane is always G18 and will be formed by the first two axes defined in the channel. If the X (firs axis of the channel) and Z (second axis of the channel) have been defined, the work plane will be the ZX (Z as abscissa and X as ordinate). Functions G17 and G19 are not valid. Since there are only two axes, function G20 makes no sense.

LATHE WITH 1 CHANNEL, 3 AXES AND 1 SPINDLE. CONFIGURATION OF "PLANE" TYPE AXES.

The axes and spindles are distributed as follows.

Channel ·1·. 3 axes (X Z Y) 1 spindle (S).

General configuration of the channels, axes and spindles.

System	Channel ·1·
NCHANNEL = 1	GEOCONFIG = Plane
NAXIS = 3	CHNAXIS = 3
NSPDL = 1	CHNSPDL = 1
	IPLANE = G18

Configuration of the axes and spindles of the system.

AXISNAME	Value	Logic order.	Index in the system.
AXISNAME 1	X	Logic number ·1·.	
AXISNAME 2	Z	Logic number ·2·.	
AXISNAME 3	Y	Logic number ·3·.	
SPDLNAME 1	S	Logic number ·4·.	Spindle with index ·1·.

Configuration of the axes and spindles of channel ·1·.

CHAXISNAME	Value	Index in the channel.	Logic order.
CHAXISNAME 1	X	Axis with index ·1·.	Logic number ·1·.
CHAXISNAME 2	Z	Axis with index ·2·.	Logic number ·2·.
CHAXISNAME 3	Y	Axis with index ·3·.	Logic number ·3·.

CHSPDLNAME	Value	Index in the channel.	Logic order.
CHSPDLNAME 1	S	Spindle with index ·1·.	Logic number ·4·.

Plane selection in channel ·1·.

Function	Selected plane.
G18	ZX plane Z axis for longitudinal compensation.

With a plane-type-axis configuration, the G18 plane is always active; in this case the ZX plane. Functions G17 and G19 are not valid. Function G20 permits selecting the rest of the axes (Y axis) as the axis for longitudinal compensation.

LATHE WITH 1 CHANNEL, 3 AXES AND 1 SPINDLE. CONFIGURATION OF TRIHEDRON TYPE AXES.

The axes and spindles are distributed as follows.

Channel ·1·. 3 axes (X Y Z) 1 spindle (S).

General configuration of the channels, axes and spindles.

System	Channel ·1·
NCHANNEL = 1	GEOCONFIG = Trihedron
NAXIS = 3	CHNAXIS = 3
NSPDL = 1	CHNSPDL = 1

Configuration of the axes and spindles of the system.

AXISNAME	Value	Logic order.	Index in the system.
AXISNAME 1	X	Logic number ·1·.	
AXISNAME 2	Y	Logic number ·2·.	
AXISNAME 3	Z	Logic number ·3·.	
SPDLNAME 1	S	Logic number ·4·.	Spindle with index ·1·.

Configuration of the axes and spindles of channel ·1·.

CHAXISNAME	Value	Index in the channel.	Logic order.
CHAXISNAME 1	X	Axis with index ·1·.	Logic number ·1·.
CHAXISNAME 2	Y	Axis with index ·2·.	Logic number ·2·.
CHAXISNAME 3	Z	Axis with index ·3·.	Logic number ·3·.

CHSPDLNAME	Value	Index in the channel.	Logic order.
CHSPDLNAME 1	S	Spindle with index ·1·.	Logic number ·4·.

Plane selection in channel ·1·.

Function.	Selected plane.
G17	X-Y plane. Perpendicular axis Z.
G18	Z-X plane. Perpendicular axis Y.
G19	Y-Z plane. Perpendicular axis X.
G20	Any plane and/or longitudinal axis.

With a trihedron-type-axis configuration, the planes behave in the same way as on a milling machine except that the usual work plane will be G18 (if it has been configured like that in parameter IPLANE).

LATHE WITH 1 CHANNEL, 3 AXES (1 FREE) AND 1 SPINDLE. CONFIGURATION OF "PLANE" TYPE AXES.

The axes and spindles are distributed as follows.

Channel ·1·. 3 axes (X Z U) 1 spindle (S).

General configuration of the channels, axes and spindles.

System	Channel ·1·
NCHANNEL = 1	GEOCONFIG = Plane
NAXIS = 3	CHNAXIS = 2
NSPDL = 1	CHNSPDL = 1
	IPLANE = G18

Configuration of the axes and spindles of the system.

AXISNAME	Value	Logic order.	Index in the system.
AXISNAME 1	X	Logic number ·1·.	
AXISNAME 2	Z	Logic number ·2·.	
AXISNAME 3	U	Logic number ·3·.	
SPDLNAME 1	S	Logic number ·4·.	Spindle with index ·1·.

Configuration of the axes and spindles of channel ·1·.

CHAXISNAME	Value	Index in the channel.	Logic order.
CHAXISNAME 1	X	Axis with index ·1·.	Logic number ·1·.
CHAXISNAME 2	Z	Axis with index ·2·.	Logic number ·2·.

CHSPDLNAME	Value	Index in the channel.	Logic order.
CHSPDLNAME 1	S	Spindle with index ·1·.	Logic number ·4·.

Plane selection in channel ·1·.

Function	Selected plane.
G18	ZX plane Z axis for longitudinal compensation.

With a plane-type-axis configuration, the G18 plane is always active; in this case the ZX plane. Functions G17 and G19 are not valid.

LATHE WITH 2 CHANNELS, 4 AXES AND 2 SPINDLES. CONFIGURATION OF "PLANE" TYPE AXES.

The axes and spindles are distributed as follows.

Channel ·1·. 2 axes (X Z) 1 spindle (S).

Channel ·2·. 2 axes (X1 Z1) 1 spindle (S1).

General configuration of the channels, axes and spindles.

System	Channel ·1·	Channel ·2·
NCHANNEL = 2	GEOCONFIG = Plane	GEOCONFIG = Plane
NAXIS = 4	CHNAXIS = 2	CHNAXIS = 2
NSPDL = 2	CHNSPDL = 1	CHNSPDL = 1
	IPLANE = G18	IPLANE = G18

Configuration of the axes and spindles of the system.

AXISNAME	Value	Logic order.	Index in the system.
AXISNAME 1	X	Logic number ·1·.	
AXISNAME 2	Z	Logic number ·2·.	
AXISNAME 3	X1	Logic number ·3·.	
AXISNAME 4	Z1	Logic number ·4·.	
SPDLNAME 1	S	Logic number ·5·.	Spindle with index ·1·.
SPDLNAME 2	S1	Logic number ·6·.	Spindle with index ·2·.

Configuration of the axes and spindles of channel ·1·.

CHAXISNAME	Value	Index in the channel.	Logic order.
CHAXISNAME 1	X	Axis with index ·1·.	Logic number ·1·.
CHAXISNAME 2	Z	Axis with index ·2·.	Logic number ·2·.

CHSPDLNAME	Value	Index in the channel.	Logic order.
CHSPDLNAME 1	S	Spindle with index ·1·.	Logic number ·5·.

Configuration of the axes and spindles of channel ·2·.

CHAXISNAME	Value	Index in the channel.	Logic order.
CHAXISNAME 1	X1	Axis with index ·1·.	Logic number ·3·.
CHAXISNAME 2	Z1	Axis with index ·2·.	Logic number ·4·.

CHSPDLNAME	Value	Index in the channel.	Logic order.
CHSPDLNAME 1	S1	Spindle with index ·1·.	Logic number ·6·.

Plane selection in channels ·1· and ·2·.

With this configuration, the work plane is always G18 and will be formed by the first two axes defined in the channel. If the X (firs axis of the channel) and Z (second axis of the channel) have been defined, the work plane will be the ZX (Z as abscissa and X as ordinate). Functions G17 and G19 are not valid. Since there are only two axes, function G20 makes no sense.

LATHE WITH 3 CHANNELS, 6 AXES AND 2 SPINDLES. CONFIGURATION OF "PLANE" TYPE AXES.

The axes and spindles are distributed as follows.

Channel ·1·.	2 axes (X Z)	1 spindle (S).
Channel ·2·.	2 axes (X1 Z1)	1 spindle (S1).
Channel ·3·.	2 axes (U V)	

General configuration of the channels, axes and spindles.

System	Channel ·1·	Channel ·2·	Channel ·3·
NCHANNEL = 3	GEOCONFIG = Plane	GEOCONFIG = Plane	
NAXIS = 6	CHNAXIS = 2	CHNAXIS = 2	CHNAXIS = 2
NSPDL = 2	CHNSPDL = 1	CHNSPDL = 1	CHNSPDL = 0
	IPLANE = G18	IPLANE = G18	

Configuration of the axes and spindles of the system.

AXISNAME	Value	Logic order.	Index in the system.
AXISNAME 1	X	Logic number ·1·.	
AXISNAME 2	Z	Logic number ·2·.	
AXISNAME 3	X1	Logic number ·3·.	
AXISNAME 4	Z1	Logic number ·4·.	
AXISNAME 5	U	Logic number ·5·.	
AXISNAME 6	V	Logic number ·6·.	
SPDLNAME 1	S	Logic number ·7·.	Spindle with index ·1·.
SPDLNAME 2	S1	Logic number ·8·.	Spindle with index ·2·.

Configuration of the axes and spindles of channel ·1·.

CHAXISNAME	Value	Index in the channel.	Logic order.
CHAXISNAME 1	X	Axis with index ·1·.	Logic number ·1·.
CHAXISNAME 2	Z	Axis with index ·2·.	Logic number ·2·.

CHSPDLNAME	Value	Index in the channel.	Logic order.
CHSPDLNAME 1	S	Spindle with index ·1·.	Logic number ·7·.

Configuration of the axes and spindles of channel ·2·.

CHAXISNAME	Value	Index in the channel.	Logic order.
CHAXISNAME 1	X1	Axis with index ·1·.	Logic number ·3·.
CHAXISNAME 2	Z1	Axis with index ·2·.	Logic number ·4·.

CHSPDLNAME	Value	Index in the channel.	Logic order.
CHSPDLNAME 1	S1	Spindle with index ·1·.	Logic number ·8·.

Configuration of the axes and spindles of channel ·3·.

CHAXISNAME	Value	Index in the channel.	Logic order.
CHAXISNAME 1	U	Axis with index ·1·.	Logic number ·5·.
CHAXISNAME 2	V	Axis with index ·2·.	Logic number ·6·.

Plane selection in channels ·1· and ·2·.

With this configuration, the work plane is always G18 and will be formed by the first two axes defined in the channel. If the X (firs axis of the channel) and Z (second axis of the channel) have been defined, the work plane will be the ZX (Z as abscissa and X as ordinate). Functions G17 and G19 are not valid. Since there are only two axes, function G20 makes no sense.

9.2 Configure an axis as rotary axis.

The CNC admits different ways to configure a rotary axis depending on how it is going to move. Hence, the CNC can have rotary axes with travel limits, for example between 0º and 180º (linearlike rotary axis); axes that always move in the same direction (unidirectional rotary axis); axes that choose the shortest path (positioning-only rotary axis).

All rotary axes must be programmed in degrees; therefore, they will not be affected by the mm-inch conversion. The number of revolutions the axis will turn when programming a distance greater than the module depends on the type of axis. The limits to display the position values (coordinates) also depend on the type of axis.

Linearlike rotary axis.

The axis behaves like a linear axis, but it is programmed in degrees. The CNC displays the position values between the travel limits.

Machine parameter configuration.

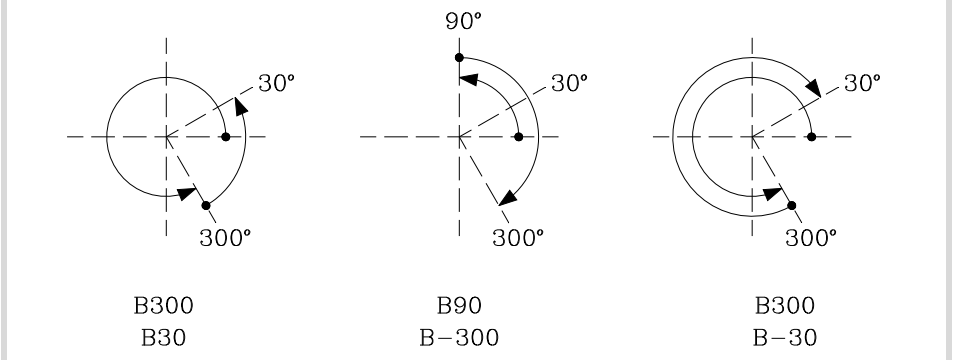
Machine parameters LIMIT+ and LIMIT- set the travel limits for the axis; there are no module limits.

Parameter.	Value.
AXISTYPE	Rotary.
AXISMODE	Linearlike.
UNIDIR	Not being used.
SHORTESTWAY	Not being used.
LIMIT+ LIMIT-	Travel limits.
MODUPLIM MODLOWLIM	Not being used.

Normal rotary axis.

This type of rotary axis can turn in both directions. The CNC displays the position values between the limits of the module.

G90 movements.	G91 movements.
The sign of the position value indicates the moving direction; the absolute position value indicates the target position.	Normal incremental movement. The sign of the position value indicates the moving direction; the absolute position value indicates the position increment.
Even if the programmed distance is greater than the module, the axis never turns more than one revolution.	If the programmed distance is greater than the module, the axis turns more than one revolution.



Machine parameter configuration.

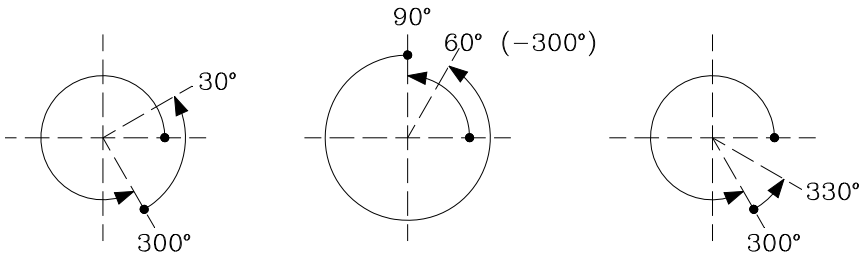
Machine parameters MODUPLIM and MODLOWLIM set the module limits for the axis; there are no travel limits.

Parameter.	Value.
AXISTYPE	Rotary.
AXISMODE	Module.
UNIDIR	No.
SHORTESTWAY	No.
LIMIT+ LIMIT-	Not being used. There are no travel limits.
MODUPLIM MODLOWLIM	Limits of the module. Limits to display the position values.

Unidirectional rotary axis.

This type of rotary axis only moves in one direction, the one that has been preset for it. The CNC displays the position values between the limits of the module.

G90 movements.	G91 movements.
The axis moves in the preset direction up to the programmed position.	The axis only admits movements in the preset direction. The sign of the position value indicates the moving direction; the absolute position value indicates the position increment.
Even if the programmed distance is greater than the module, the axis never turns more than one revolution.	If the programmed distance is greater than the module, the axis turns more than one revolution.



Machine parameter configuration.

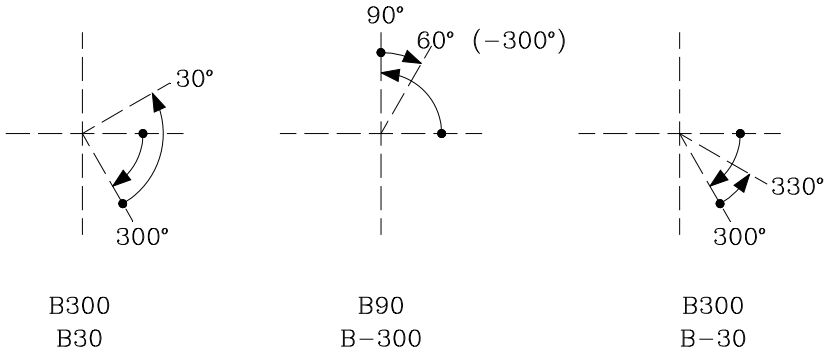
Machine parameters MODUPLIM and MODLOWLIM set the module limits for the axis; there are no travel limits.

Parameter.	Value.
AXISTYPE	Rotary.
AXISMODE	Module.
UNIDIR	Positive / Negative.
SHORTESTWAY	No.
LIMIT+ LIMIT-	Not being used. There are no travel limits.
MODUPLIM MODLOWLIM	Limits of the module. Limits to display the position values.

Positioning-only rotary axis.

This type of rotary axis can move in both directions; but in absolute movements, it only moves via the shortest path. The CNC displays the position values between the limits of the module.

G90 movements.	G91 movements.
The axis moves via the shortest path up to the programmed position.	Normal incremental movement. The sign of the position value indicates the moving direction; the absolute position value indicates the position increment.
Even if the programmed distance is greater than the module, the axis never turns more than one revolution.	If the programmed distance is greater than the module, the axis turns more than one revolution.



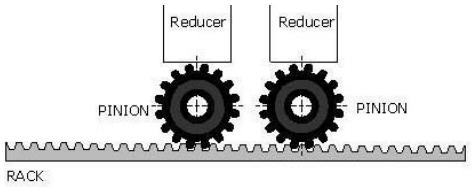
Machine parameter configuration.

Machine parameters MODUPLIM and MODLOWLIM set the module limits for the axis; there are no travel limits.

Parameter.	Value.
AXISTYPE	Rotary.
AXISMODE	Module.
UNIDIR	No.
SHORTESTWAY	Yes.
LIMIT+ LIMIT-	Not being used. There are no travel limits.
MODUPLIM MODLOWLIM	Limits of the module. Limits to display the position values.

9.3 Configure two axes as a tandem axis.

A tandem axis consists of two motors mechanically coupled to each other forming a single transmission system (axis or spindle). Only one axis (master tandem axis) can be displayed and programmed at the CNC. The other axis (slave tandem axis) is not displayed and is not programmable; it is controlled by the CNC.

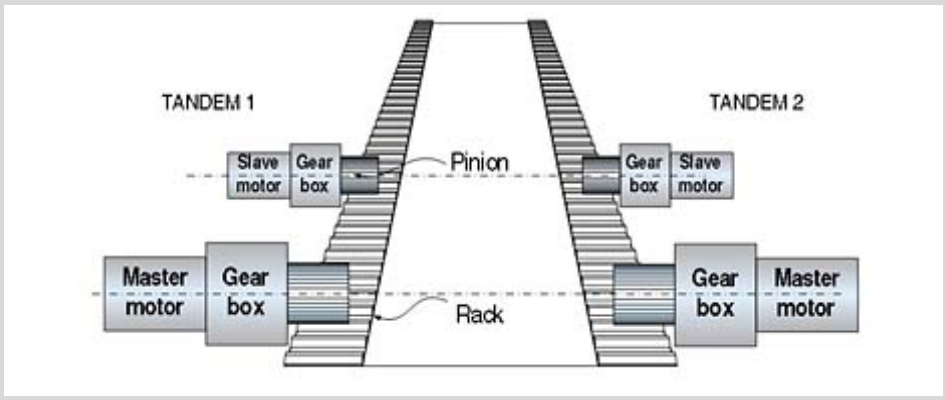


Example of a tandem axis made up of two pinions and a rack.

The following aspects may be pointed out in this configuration:

- A tandem axis helps provide the necessary torque to move an axis when a single motor is not capable of supplying enough torque to do it.
- Applying a pre-load torque between the main motor and the slave motor reduces the backlash on the rack and pinion.
- The rigidity of the rack-and-pinion system is greater than that of long leadscrews.

One of the many applications of controlling a tandem axis has to do with gantry machines. The following example shows a gantry axis with two tandem axes.



Requirements and limitations of tandem axes.

Each tandem pair (master and slave) must meet the following requirements:

- Each master tandem axis (spindle) admits one single slave tandem axis (spindle).
- A preload may be applied between the two motors.
- Each motor may have a different rated torque.
- The turning direction of each motor may be different from the other's.
- The torque distribution between both motors may be different from 1:1 ratio. For example, on motors whose rated torque is different.
- Both axes and drives must be of the same type (same AXISTYPE and DRIVETYPE parameters for both axes).
- The channel changing permission (parameter AXISEXCH) must be the same in both axes.
- Both axes and drives must have the same software limits (same LIMIT+ and LIMIT- parameters for both axes).
- When the axes are rotary, both axes must be of the same type (same AXISMODE and SHORTESTWAY parameters for both axes).
- The fast and slow home search speeds (parameters REFFEED1 and REFFEED2) must be the same for both axes.

9.3.1 Tandem axis configuration. Machine parameters.

The tandem pairs of the system are configured by machine parameters. Each tandem pair has the following general machine parameters to configure it.

Parameter.	Meaning.
TANDEM	Table of tandem axes or spindles.
MASTERAXIS	Master or main axis/spindle.
SLAVEAXIS	Slave axis/spindle.
TORQDIST	Torque distribution.
PRELOAD	Preload between both motors.
PRELFITI	Filter time to apply the preload.
TPROGAIN	Proportional gain (Kp) for the tandem.
TINTIME	Integral gain (Ki) for the tandem.
TCOMPLIM	Compensation limit.

Axes or spindles of the tandem system.

A tandem system can consist of a pair of linear axes, rotary axes or spindles. The pair of axes or spindles making up the tandem system is set by machine parameters MASTERAXIS and SLAVEAXIS inside the TANDEM table. The axes must be previously defined in parameter AXISNAME and the spindles in parameter SPDLNAME.

The master motor of the tandem, besides generating torque, it is in charge of positioning. The slave motor only provides torque.

The CNC closes the position loop only with the position of the master axis of the tandem. The velocity command of the master axis/spindle of the tandem is also sent to the slave axis/spindle of the tandem closing the velocity loop. The tandem control changes the velocity command of the master axis/spindle and that of the slave axis/spindle according to the torque distribution and the selected preload.

Type of servo system and feedback.

Tandem axes or spindles must be velocity-Sercos; if not, both axes will move, but it will not be a tandem system. All the axes that will be interpolated with the tandem axis must also operate in velocity-Sercos mode.

In a tandem system, the master axis or spindle can carry either external or internal feedback. The slave axis or spindle normally uses internal feedback. The tandem does not admit combined feedback (external+internal).

Considerations for tandem spindles.

- The PLC must enable and disable the SPENA(axis) and DRENA(axis) signals of the master and slave spindles at the same time.
- Both spindles must be in the same gear (range).
- To control the tandem spindle from the PLC, for example in a gear change, it is only necessary to activate the marks PLCCNTL, SPDLEREV and SANALOG of the master spindle.

Torque load distribution in the tandem.

The torque distribution sets the torque percentage supplied by each motor to obtain the total necessary torque in the tandem. The torque distribution in the tandem is defined with parameter TORQDIST that indicates the percentage to be provided by the master motor. The difference between the value of this parameter and 100% is the percentage applied to the slave motor.

The torque distribution between both motors may be different from 1:1 ratio; for example, on motors whose rated torque is different. If the motors are identical and they're both supposed to output the same torque, this parameter should be set to 50%.

Apply a preload value in the tandem.

The preload is the amount of torque applied before, in both directions, to both motors of the tandem to set a traction between them in order to eliminate the backlash when the tandem is in rest position. In order for the two motors to supply opposite torques, the preload value must be greater than the maximum torque needed at all times, including accelerations.



Applying the preload necessarily implies mechanically joining the master and slave axes that make up the tandem; otherwise, the motors will move even without control command.

The preload is defined with parameter PRELOAD that indicates the percentage of the rated torque of the master motor that will be applied as preload.

Time to apply the preload.

Parameter PRELFITI may be set in order to apply the preload gradually avoiding torque steps at the tandem compensator. This parameter enables a frst order filter that sets the time it takes the CNC to apply the preload gradually.

Proportional gain (Kp) for the tandem.

The proportional controller generates an output proportional to the torque error between the two motors. The proportional gain may be set with parameter TPROGAIN.

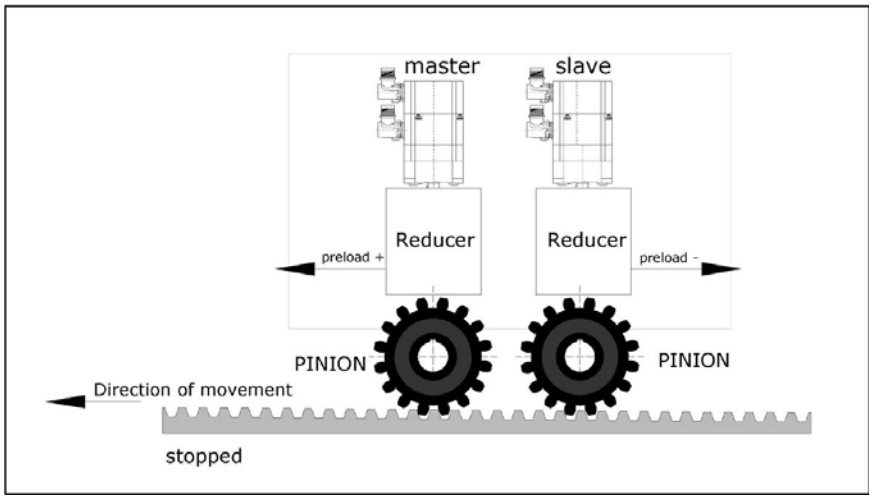
Integral gain (Ki) for the tandem.

The integral controller generates an output proportional to the integral of the torque error between the two motors. The integral gain may be set with parameter TINTTIME.

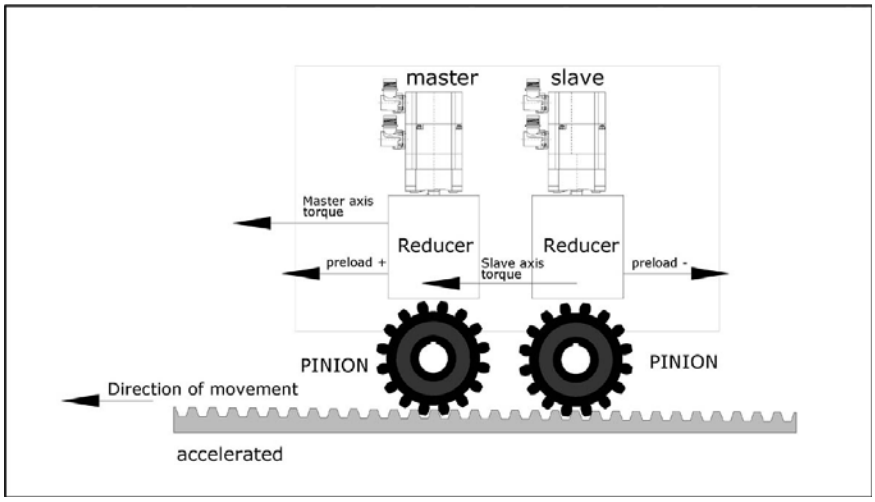
9.3.2 Effect of the preload.

The following diagrams show the effect of preload in different situations.

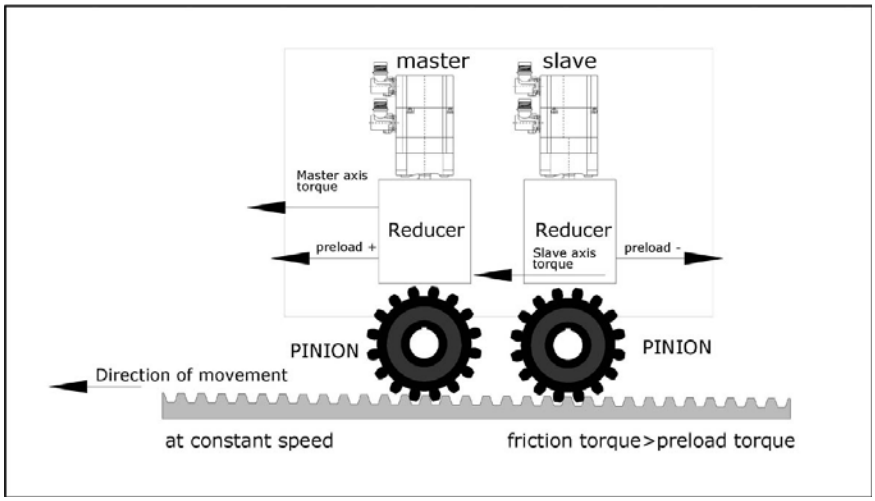
Preload at rest.



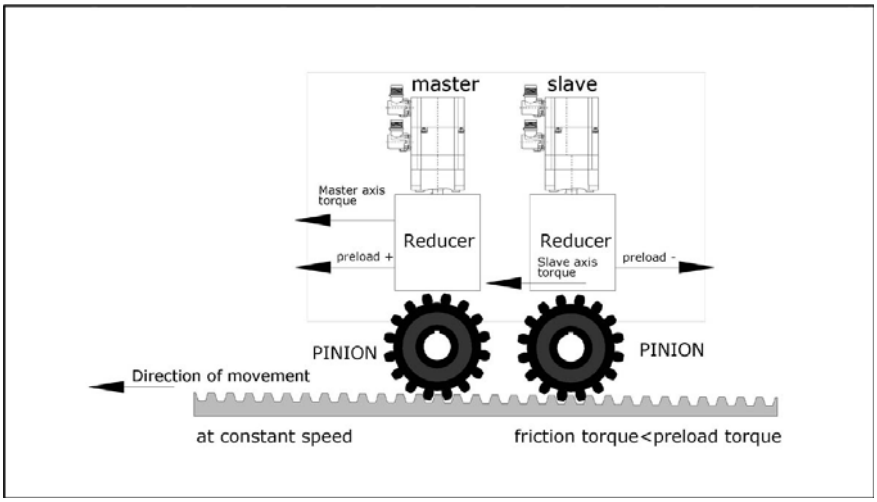
Preload with acceleration.



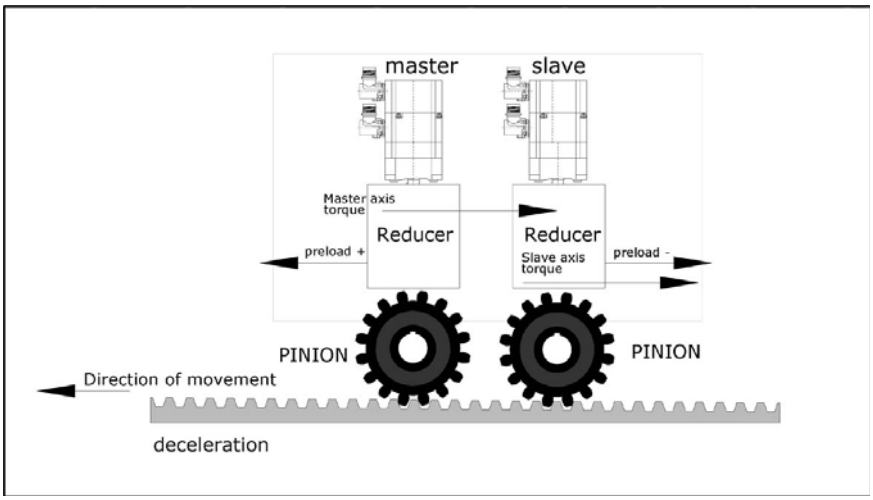
Preload at constant speed. Friction torque > Preload.



Preload at constant speed. Friction torque < Preload.

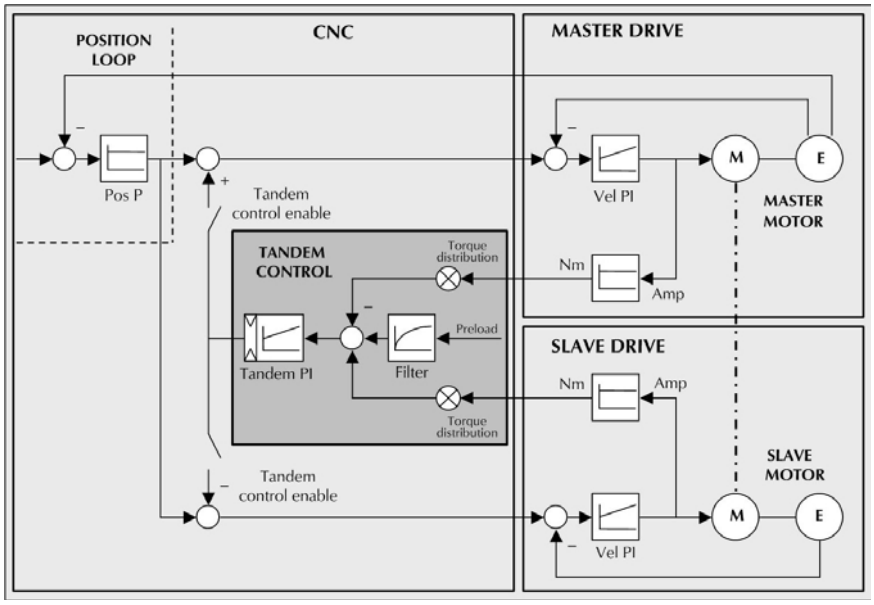


Preload with deceleration.



9.3.3 Tandem axis configuration. Block diagram.

The block diagram of the tandem control system shows the master tandem axis with its slave tandem axis. The block diagram for a gantry machine consists of two identical diagrams to the one shown in the figure.

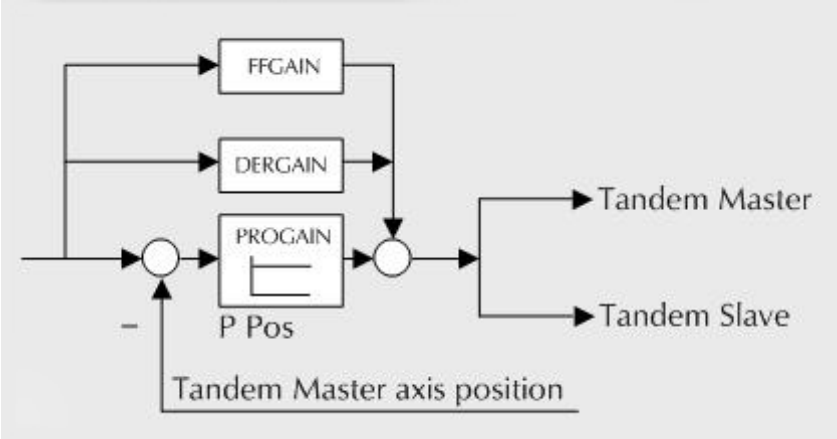


The block diagram has an area for the drive and another one for the CNC that comprises the position loop and the control of the tandem.

Position and velocity loop.

The position loop is closed only with the position of the master axis of the tandem. The velocity command of the master tandem axis is also sent to the slave tandem axis closing the velocity loop.

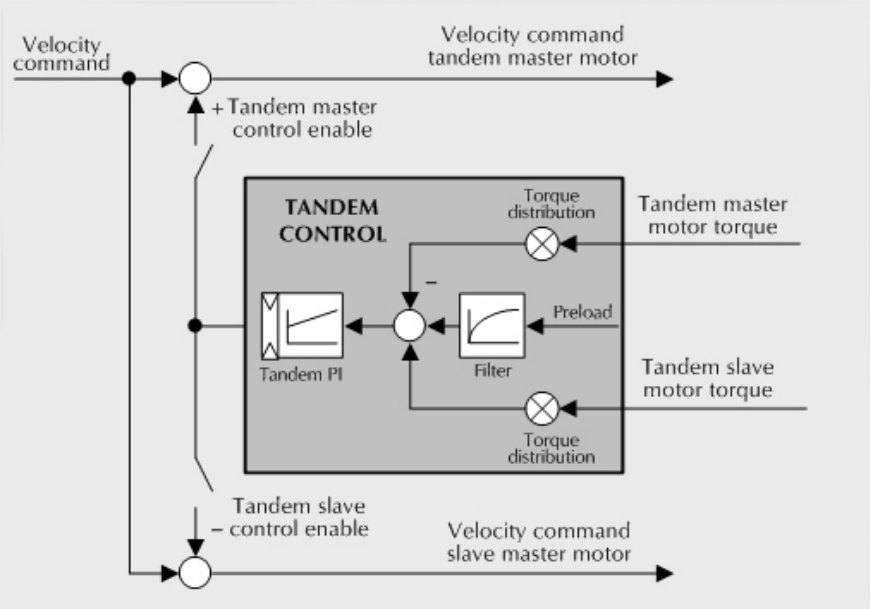
The tandem control changes the velocity command of the master axis and that of the slave axis according to the torque distribution and the selected preload.



The feed-forward and AC-forward values of the master axis are applied to the slave axis; consequently, they must have the same gear ratios.

Tandem axis control.

The block diagram showing the application of the tandem axis control is the following. The meaning of the nomenclature being used is:



Torque of the master motor of the tandem.

It is the percentage of the rated torque shown by the sercos variable TV2 of the drive that governs the master axis of the tandem. It is read through the sercos fast channel in each loop.

Torque of the slave motor of the tandem.

It is the percentage of the rated torque shown by the sercos variable TV2 of the drive that governs the slave axis of the tandem. It is read through the sercos fast channel in each loop.

Torque distribution.

Normalizing gain of the torque generated by the motors in order to distribute the torque in a ratio other than 1:1.

Preload.

Previous torque applied to both tandem axes in opposite direction. This sets a traction between them in order to eliminate the rack-and-pinion backlash when it is in rest position. It is determined as the torque difference supplied by each axis.



Applying the preload necessarily implies mechanically joining the master and slave axes that make up the tandem axis. Otherwise, the motors will move even without the control velocity command.

Preload filter.

First-order filter to prevent torque steps from coming in when configuring the preload.

PI of the tandem.

PI for making each motor provide its corresponding torque. It increases its velocity command if the torque being supplied is too low and it decreases it if the torque being supplied is too high.

When defining a tandem axis, at each loop, the CNC reads via Sercos the torque supplied by the master and the slave axes. This may result in a smaller number of read and write variables available in the Sercos fast channel.

9.3.4 Tandem related variables.

Variables related to general machine parameters.

These variables are read-only synchronous and are evaluated during block preparation. They have generic names. Replace the letter "nb" with numbers, keeping the brackets.

Variable	PRG	PLC	INT
(V.)MPG.TMASTERAXIS[nb] Tandem [nb]. Logic number of the master axis	R	R	R
(V.)MPG.TSLAVEAXIS[nb] Tandem [nb]. Logic number of the slave axis	R	R	R
(V.)MPG.TORQDIST[nb] Tandem [nb]. Torque distribution	R	R	R
(V.)MPG.PRELOAD[nb] Tandem [nb]. Preload	R	R	R
(V.)MPG.PRELFITI[nb] Tandem [nb]. Time to apply the preload	R	R	R
(V.)MPG.TPROGAIN[nb] Tandem [nb]. Proportional gain	R	R	R
(V.)MPG.TINTIME[nb] Tandem [nb]. Integral gain	R	R	R
(V.)MPG.TCOMPLIM[nb] Tandem [nb]. Compensation Limit	R	R	R

Tandem related variables.

These variables are read/write (R/W) synchronous and are evaluated during execution. They correspond to linear and rotary axes and to the spindle. The mnemonics of the variables have generic names.

- Replace "Xn" with the name or logic number of the axis.
- Replace the "ch" character with the channel number, maintaining the brackets.

	PRG	PLC	INT
(V.)[ch].A.TPIIN.xn Input of the PI of the master axis of the tandem (in rpm)	R	R	R
(V.)[ch].A.TPIOUT.xn Output of the PI of the master axis of the tandem (in rpm)	R	R	R
(V.)[ch].A.TFILTOUT.xn Output of the pre-load filter	R	R	R
(V.)[ch].A.PRELOAD.xn Preload	R/W	R/W	R/W
(V.)[ch].A.FTEO.xn Velocity command for Sercos	R	R	R
(V.)[ch].A.TORQUE.xn Current torque in Sercos	R	R	R

(V.)A.TPIOUT.[Xn] (V.)A.TPIIN.[Xn]

The axis must be a valid tandem master, otherwise, it returns a value of zero.

(V.)A.TORQUE

The PLC reading of this variable comes in tenths (x10).

9.3.5 Tandem adjustment procedure.

This procedure must bear in mind the type of machine. In general, the resonance frequency of a tandem machine is low; therefore, the CNC must generate position commands without frequency components higher than the resonance frequency.

The process should be initiated with low jerk values (lower than 10 m/sg3) and low Kv. They can always be increased in a later readjustment.

Steps for the adjustment.

1 Move both axes independently.

The first step is to ensure the perfect operation of both the master and the slave axes separately. Also verify that both axes move in the same direction with similar dynamics.

2 Move one of the axes at a slow and constant speed.

- Do not make jerky movements because the second motor is being dragged by the first one. In this situation, any acceleration or deceleration forces it to go from one side of the backlash to the other thus making it jerky.
- Verify that the turning direction of both motors are coherent once the movement has been carried out.

Observe that reversing the turning direction of a motor reverses the direction of the torque and, consequently, it will be necessary to change the direction of its values monitored with drive parameters SP43 and TP85.

- Verify that the gear ratio in both motors is the same (same feed for same turning speed).
- Carry out a basic adjustment of the velocity loop so the machine can move. It will be readjusted later with both motors together.
- Do not set the friction parameters (there is already enough torque to move the machine).

3 Repeat the procedure with the second axis.

When adjusting the loops, use the same parameters if the motors are identical and the torque distribution is 50%. If the motors are different, the axes must be adjusted so their dynamic response is the same or very similar. When using AC-forward ("ACFGAIN" = YES), remember that each motor has half the inertia for a 50% torque distribution.

4 Enable the tandem with both motors.

First, disable the PI of he tandem, apply power and verify that the system is at rest.

Then, enter a low proportional value and eliminate the integral value of the PI of the tandem. Without preload, verify that the machine moves and that each motor supplies its corresponding torque according to the parameter "TORQDIST" (e.g. half the torque for 50% distribution).

To define the pre-load, monitor the torque of each motor (Sercos variable TV2). While stopped, increase the preload gradually until the motors supply torque in opposite directions.

Having enabled the tandem, move slowly in both directions and verify that it works properly. Make sure that it is not jerky and that each motor supplies its corresponding torque according to parameters "TORQDIST" and "PRELOAD".

Finally, readjust the velocity loop in both motors with the method used normally.

While changing the parameters of the velocity loop, the best thing would be to change them on both drives at the same time. But, since this is not possible, it is recommended to make small changes in the values or make them while the motor is stopped.

9.4 Analog axes.

9.4.1 Configure the number of the analog output and of the feedback input.

Both the feedback input and the velocity command output are configured by machine parameters. Each axis has the following machine parameters to configure it.

Parameter.	Meaning.
ANAOUTID	Number of the analog output associated with the axis.
COUNTERTYPE	Type of feedback input for the axis.
COUNTERID	Number of the feedback input for the axis.

Configure the analog output of the axis.

The velocity command for an analog axis may be taken from an analog output of the remote modules or of a Sercos drive. It is selected with parameter ANAOUTID. In either case, this parameter indicates the analog output used for the command.

Managing the analog command from CANfagor remote modules.

Parameter.	Value.
ANAOUTID	1 - 16

In this case, this parameter indicates the number of the analog output used for the command. The analog output modules are numbered following the logic order of the remote groups (rotary switch of the power supply). If there are several analog-output modules in each group, the order is from top to bottom and from left to right. The analog outputs of the first module will be 1 through 4, those of the second module 5 through 8 and so on.

Managing the analog command from Sercos drives.

Parameter.	Value.
ANAOUTID	101 -132 201 - 232

In this case, this parameter indicates the number of the analog output and the drive to be used. The parameter is set with a three-digit number; the first digit indicates the number of the analog output to be used (1 or 2) and the other two digits indicate the logic address of the drive (1 through 32). The "address" rotary selector switch sets the logic address.

For example, if the parameter is with a value of ·107·, it indicates that the CNC uses analog input ·1· of the drive whose logic address is ·7·

Configure the feedback input for the axis.

The feedback inputs for an analog axis may be located in remote modules, in a Sercos drive or locally in the central unit itself. It is selected with parameter COUNTERTYPE.

Once it has been determined where the feedback input is, Parameter COUNTERID must be set indicating which feedback input will be used (either local or remote) or the drive number.

When using the feedback input of a Sercos drive, home search is not possible; therefore, the axis must have absolute feedback.

When using the feedback input of a Sercos drive, set drive parameter PP5=-0.0001.

Feedback input in CANfagor remote modules.

Parameter.	Value.
COUNTERTYPE	Remote.
COUNTERID	Number of the feedback input.

A counter module is used as feedback input in the remote modules. Parameter COUNTERID indicates the number of the feedback input to be used.

The counter modules are numbered following the logic order of the remote groups (rotary switch of the power supply). If there are several Counter modules in each group, the order is from top to bottom and from left to right. The counters of the first module will be 1 through 4, those of the second module 5 through 8 and so on.

Feedback input in Sercos drives.

Parameter.	Value.
COUNTERTYPE	Drive.
COUNTERID	Number of the feedback input.

In this case, the second feedback input of the drive will always be used. Parameter COUNTERID indicates the logic address of the drive (from 1 to 32). The "address" rotary selector switch sets the logic address.

Local feedback input.

Parameter.	Value.
COUNTERTYPE	Local.
COUNTERID	Number of the feedback input.

The central unit has two feedback inputs. Parameter COUNTERID indicates the feedback input being used.

9.4.2 Configure 2 axes with the same feedback input and analog output.

There are machines having 2 axes that are never in the same configuration at the same time, because one of them is always parked; for example when changing spindles with an internal motor used to move its articulation.

In these cases, both motors are controlled by the same servo drive and there is a single feedback input and a single analog output for the CNC. This way, although the motors are identical, the axes may be totally different, compensations, dynamics, etc.

Configuring the axes.

To use the same feedback input, parameters COUNTERTYPE and COUNTERID of both axes must be the same. To use the same analog output, parameter ANAOUTID of both axes must be identical.

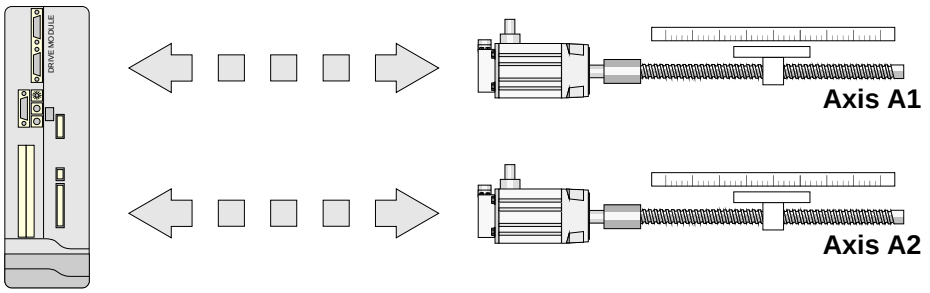
Besides, the motors must necessarily be identical because the following machine parameters must have the same values for both axes.

Parameter	Meaning
ABSFEEDBACK	Type of feedback.
SINMAGNI	Multiplying factor for the sinusoidal signal.
I0TYPE	Reference mark (I0) type
REFPULSE	Flank of the reference signal (I0).
ABSOFF	Offset of the feedback device with respect to I0.
EXTMULT	External multiplying factor.
I0CODDI1	Pitch between 2 fixed I0 signals
I0CODDI2	Pitch between 2 variable I0 signals

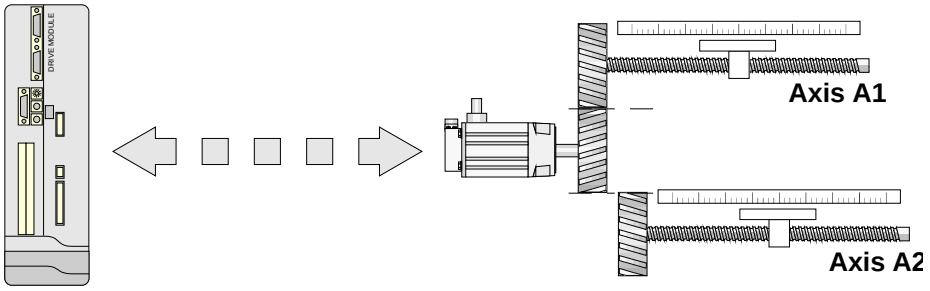
The PLC maneuver must include the detection of the active axis and of the parked axis.

9.5 Multi-axis management.

A multi-axis group consists of several Sercos axes or spindles controlled by a single drive, although not simultaneously. The axes and spindles may have different dynamics and may be controlled either by the same motor or by different motors.



Multi-axis group composed by two axes, each one controlled by a motor.



Multi-axis group composed by two axes controlled by a single motor.

The internal (motor) feedback of both axes will be located at the drive whereas the external (direct) feedback of each axis may be located at the remote modules, at a Sercos drive or locally in the central unit itself. When using a single drive, it is possible to use the free feedback input of that drive or of another one. Each axis can have its own feedback system or both axes can share the same feedback system.

Requirements and limitations of a multi-axis group.

The axes associated with the same drive must meet the following requirements.

- Any Sercos axis of the system may be part of a multi-axis group, except those that belong to gantry or tandem axes.
- Since each axis and spindle defined at the CNC can have several parameter sets, the sum of sets of all the axes and/or spindles included in a group must not exceed the 8 sets available at the drive. For example, a group may consist of 8 axes with 1 parameter set each, a spindle with 4 sets plus 2 axes with 2 sets each, etc.
- Since the axes are controlled by the same drive, all the axes of a group must have the same Sercos address (parameter DRIVEID).
- All the axes of a group must work in the same Sercos operating mode, position or speed, (parameter OPMODEP). If the external feedback (direct feedback) of the axis is connected to a drive, the axis may be operated in either position-Sercos mode or velocity-Sercos mode (velocity-Sercos mode is recommended). If the external feedback (direct feedback) of the axis is connected to local or remote feedback input, the axis can only work in velocity-Sercos mode.
- If the external feedback is local or remote, the CNC homes the axis as if it were analog; the drive does not know that the axis has been homed and does not update its position value, it only receives the velocity command. If the external feedback is connected to another drive, it will not be possible to home the axis.
- The axis being governed by the drive is selected via PLC (mark SWITCH(axis)). The PLC must either park the axes not governed by the drive or set them as DRO axes.
- The CNC keeps monitoring (following error, tendency test, etc.) all the axes sharing the drive even if the axis is not selected.

Behavior of the CNC and the drive on power-up and when switching axes.

Behavior of the CNC and the drive on power-up.

On power-up, the drive controls the master axis of the group, the first axis defined in the group (parameter MULAXISNAME_1). The CNC assumes default set of the axis (parameter DEFAULTSET). The PLC program must define the set and the gear assumed by the drive.

Behavior of the CNC and the drive when switching axes.

When the PLC switches an axis and/or spindle at the drive, the PLC program must define the set and the gear assumed by the drive. The CNC assumes default set (parameter DEFAULTSET).

Parameter matching between the CNC and the Sercos drive.

In a multi-axis group, the CNC matches the drive with the general parameters and the parameters of the first set of the master axis. During parameter matching, the CNC only uses the parameters of the first set of the drive, the rest of the sets of the drive must be set via DDSSetup or at the drive itself. The CNC uses the following criterion for parameter matching.

- The CNC sends the general parameters of the master axis to the general parameters of the drive.
- The CNC sends the parameters of the default set of the master axis (set defined in parameter DEFAULTSET) to the general parameters or of the first set of the drive accordingly.
- The CNC sends the gear parameters of the master axis to the gear parameters of the drive.

The gears that correspond to the slave axes must be set at the drive itself.

9.5.1 Configuration of a multi-axis group. Machine parameters.

Configuring a multi-axis group.

Multi-axis groups are configured via machine parameters. Each multi-axis group has the following general machine parameters to configure it.

Parameter.	Meaning.
MULTIAXIS	Table of multi-axis groups.
MULNGROUP	Number of multi-axis groups in the system.
GROUP n	Multi-axis groups.
MULNAXIS	Number of axes and/or spindles making up the multi-axis group.
MULAXISNAME n	Name of the axes and/or spindles of the group.

Defining the multi-axis groups of the system.

Parameter MULTIAXIS shows the multi-axis group definition table for defining the number of groups of the system and the composition of each group.

Parameter MULNGROUP sets the number of multi-axis groups of the system. The table shows a GROUP_n parameter for each multi-axis group.

Defining the axes or spindles that make up the multi-axis group.

Parameter MULNAXIS sets the number of axes and/or spindles making up the multi-axis group. Since each axis or spindle of the CNC can have several parameter sets, the number of axes of the group is limited by the 8 parameter sets available at the drive. The parameter sets of the drive are distributed among all the axes and/or spindles that make up the multi-axis group, so a group may be formed by 8 axes with 1 parameter set for each, 1 spindle with 4 sets plus 2 axes with 2 sets each, etc.

The table shows a MUMLAXISNAME_n parameter for each axis or spindle that belong to the multi-axis group. Any Sercos axis of the system may be part of a group, except those that belong to gantry or tandem axes. The first axis of the group will be the master axis, the one the drive will use by default; the order of the rest of the axes is irrelevant. On system power-up, the CNC will use the parameters of the master axis to run the parameter matching with the drive.

Setting the drive to be used.

Parameter.	Meaning.
DRIVESET	Set of the drive to be used.

At the axes that belong to a multi-axis group, parameter DRIVESET indicates the set to be activated at the drive after a set or gear change at the CNC (G112 and M41 to M44). The CNC sends the changes of Kv, feed-forward (parameter FFGAIN) and AC-forward (parameter ACFGAIN) to the set indicated in this parameter.

Being parameter matching active, the CNC sends the parameters of set ·1· of the master axis to the set ·0· and to the gear ·1· (first) of the drive. For the rest of the sets of the master axis, the CNC only sends the gear data to the drive; set ·2· of the CNC to the gear 2 of the drive and so on. At the master axis of multi-axis group, in order for parameter matching to work properly, this parameter must coincide with the set number where it is; i.e. DRIVESET=1 in set ·1·, DRIVESET=2 in ·2· and so on. At the slave axis of a multi-axis group, this parameter indicates the gear to be used at the drive. If the axis belongs to a multi-axis group, the parameter does not admit a ·0· (zero) value.

Configuring the external feedback (direct feedback).

Parameter.	Meaning.
COUNTERTYPE	Type of feedback input for the axis; drive, local or remote.
COUNTERID	Number of the feedback input for the axis.

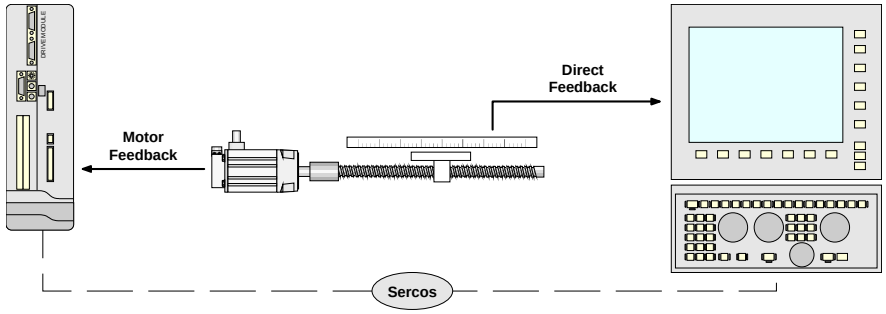
Type of feedback input for the axis.

The internal feedback (motor) of both axes is located at the drive. Parameter COUNTERTYPE indicates the location of the external (direct) feedback of the axis; at the remotte modules (4 inputs per module, differential TTL, 1 Vpp sinusoidal), at the free input of a Sercos drive or at the local feedback inputs of the central unit (2 TTL and differential TTL inputs).

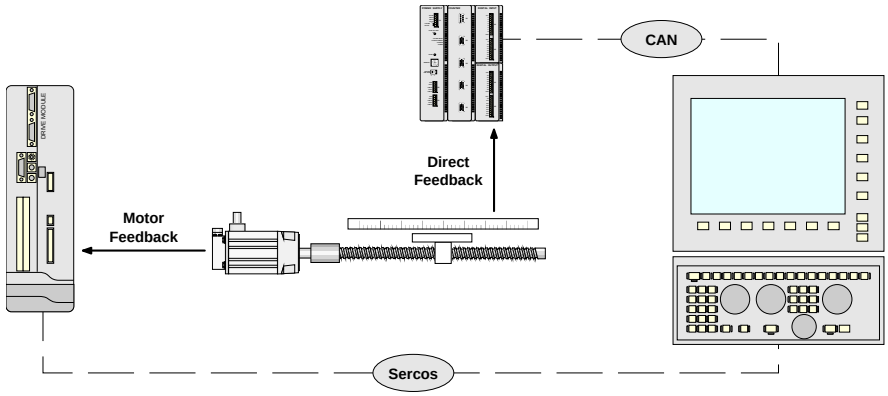
When using a Sercos drive, the axis can use the second free feedback input of that drive or of another one. When the axis uses the second feedback input of another drive, the CNC does not allow homing the axis; therefore, the axis must have absolute feedback.

When the internal (motor) feedback and the external (direct) feedback of an axis are at different drives, set parameter PP5=-0,0001 at the drive that receives the external feedback.

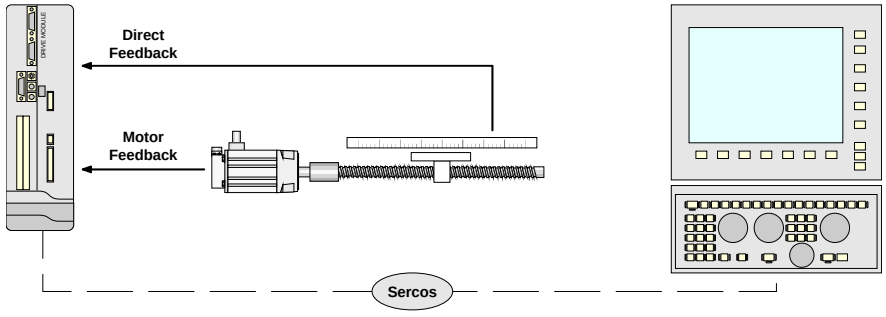
COUNTERTYPE = Drive



COUNTERTYPE = Remote



COUNTERTYPE = Local



Number of the feedback input for the axis.

Depending on the type of feedback, this parameter indicates which feedback input will be used (either local or remote) or the drive number. When two axes will not be in the configuration at the same time, the CNC allows using the same feedback input for both axes.

Type of feedback.	Number of the feedback input for the axis.
Feedback input in CANfagor remote modules.	A counter module is used as feedback input in the remote modules. This parameter indicates the number of the feedback input being used. The counter modules are numbered following the logic order of the remote groups (rotary switch of the power supply). If there are several Counter modules in each group, the order is from top to bottom and from left to right. The counters of the first module will be 1 through 4, those of the second module 5 through 8 and so on.
Feedback input in Sercos drives.	In this case, the second feedback input of the drive will always be used. The parameter indicates the logic address of the drive (from 1 to 32). The "address" rotary selector switch sets the logic address.
Local feedback input.	The central unit has two feedback inputs. This parameter indicates the feedback input being used.

9.5.2 Configuration of a multi-axis group. The PLC program generates an error.

Multi-axis groups are managed from the PLC using one of its marks.

PLC signal.	Meaning.
SWITCH(axis)	The PLC activates this mark to let the CNC know which axis or spindle that it must govern.

The PLC program must define the set and the gear assumed by the drive on power-up, toggle between the various axes or spindle of the group and define the set and gear assumed by the drive for each change of axis or spindle. The axis or spindle changing process must consider the following aspects (assuming that the X axis is active and the X1 axis is parked)

1. Park the X axis.
2. Verify that the X axis is parked.
3. Switch to the X1 axis (SWITCHX1=1).
4. Otherwise, write the parameters at the drive. Verify that the parameters have been written at the drive and, if necessary, validate the parameters of the drive
5. Change the set and/or gear at the drive.
6. Verify that the set and/or gear has been changed (mark SERPLCAC) and that the set and/gear of the drive is correct (parameters GV21 and GV25 of the drive).
7. Enable the X1 axis.

9.5.3 Changing the set and the gear at the CNC and at the drive.

A set is a bunch of parameters that may be activated and gear is a bunch of parameters specifically related to gear ratios. The parameters that define the gear are included in the set, one gear per set. A set change activates all its parameters, including those of the gear, whereas a gear change only activates the specific gear-related parameters.

Axes and spindles included in a multi-axis group.

Function.	CNC	Drive.
G112 (axes only)	The CNC activates the programmed set.	The drive activates the gear defined in parameter DRIVESET of the program set.
M41-M44 (spindles only)	The CNC activates the programmed set.	The drive activates the gear defined in parameter DRIVESET of the program set.
(V.)A.SETGE.xn	Not being used.	The drive activates the programmed set and/or gear.

If the axis or spindle does not belong to a multi-axis group, functions G112 and M41 to M44 act in the same way at the CNC; but at the drive they activate the programmed gear, not the one defined in parameter DRIVESET.

Spindle and C axis.

Function.	CNC	Drive.
#CAX ON	The CNC activates the set defined in parameter CAXSET.	The drive activates the gear defined in parameter CAXSET.
#CAX OFF	The CNC restores the set that was active before enabling the spindle as C axis.	The drive restores the gear that was active before enabling the spindle as C axis.

9.5.4 Configuration examples.

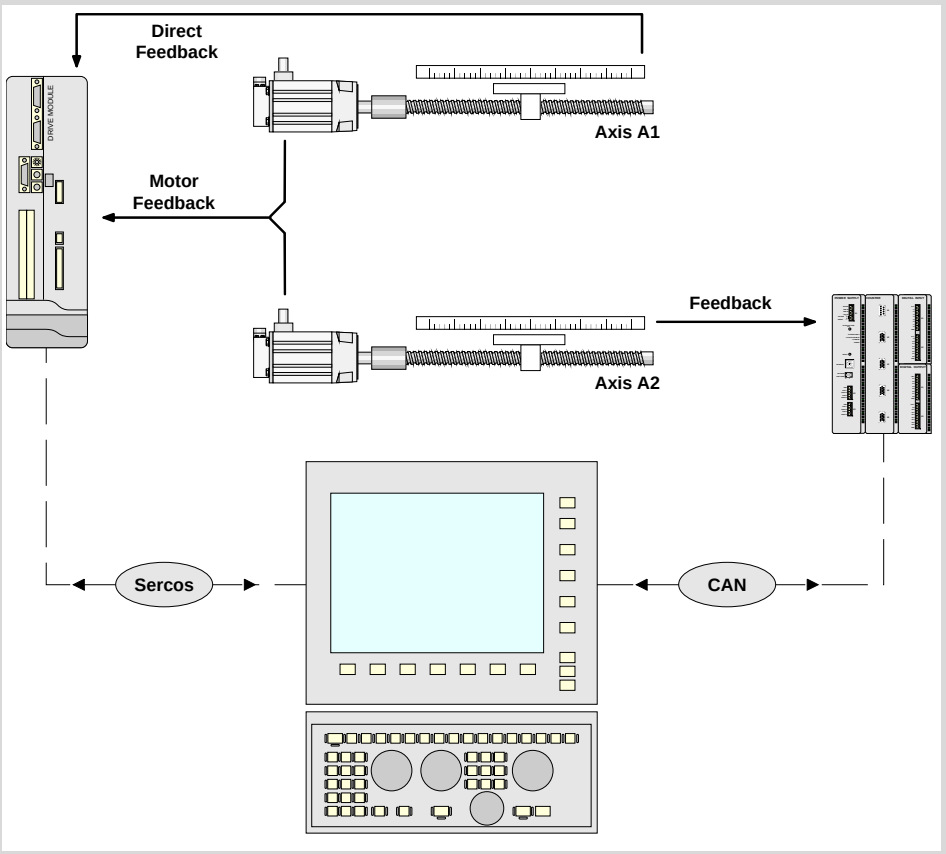
Example 1. 2-axis system, each with its own motor.

The system has the following characteristics.

- A single drive controls both axes, each with its own motor.
- Each axis uses a different type of feedback.

A1 axis The external (direct) feedback is connected to the drive itself.

A2 axis The external (direct) feedback is connected to the remote modules.



Configuration of the multi-axis group.

MULTIAXIS	GROUP_1
MULNGROUP = 1	MULNAXIS = 2 MULAXISNAME_1 = A1 MULAXISNAME_2 = A2

Configuring the axes.

	DRIVEID	OPMODEP	FBACKSRC	COUNTERTYPE	COUNTERID
A1 axis	1	Speed	External	Drive	1
A2 axis	1	Speed	External	Remote	1

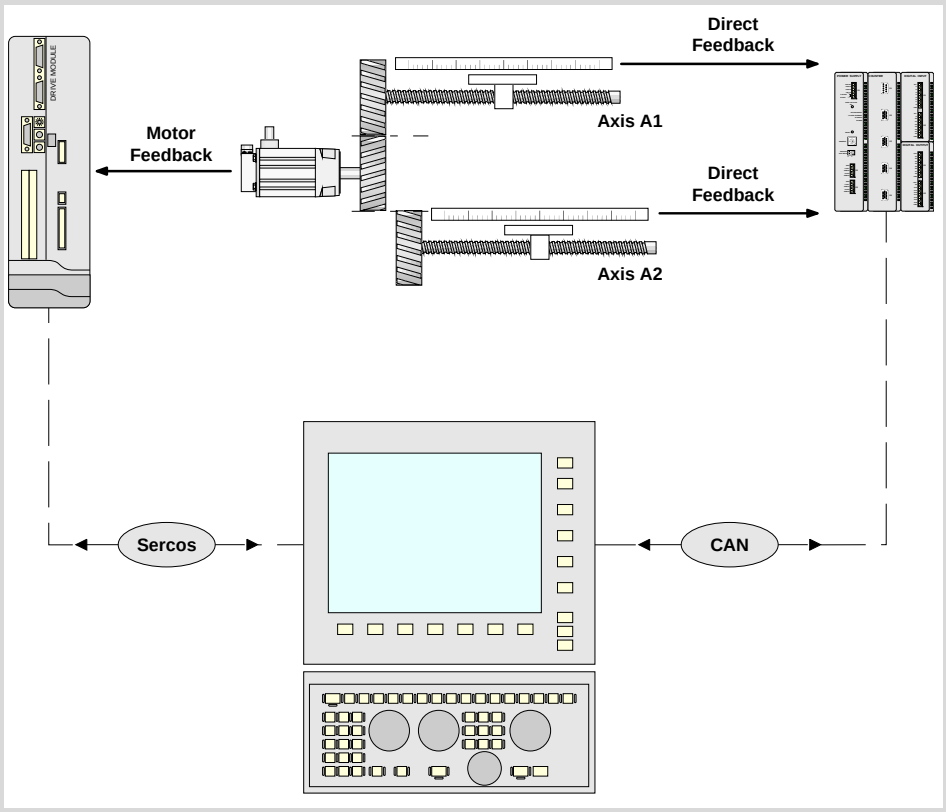
Example 2. System with two axes and a single motor.

The system has the following characteristics.

- A single drive controls both axes with the same motor.
- Each axis uses a different type of feedback.

A1 axis The external (direct) feedback is connected to the remote modules.

A2 axis The external (direct) feedback is connected to the remote modules.



Configuration of the multi-axis group.

MULTIAXIS	GROUP_1
MULNGROUP = 1	MULNAXIS = 2 MULAXISNAME_1 = A1 MULAXISNAME_2 = A2

Configuring the axes.

	DRIVEID	OPMODEP	FBACKSRC	COUNTERTYPE	COUNTERID
A1 axis	1	Speed	External	Remote	1
A2 axis	1	Speed	External	Remote	2

9.6 Home search.

What is home search?.

Home search is the operation used to synchronize the system. This operation must be carried out when the CNC loses the position of the origin point (e.g. by turning the machine off). Home search may be done in three ways.

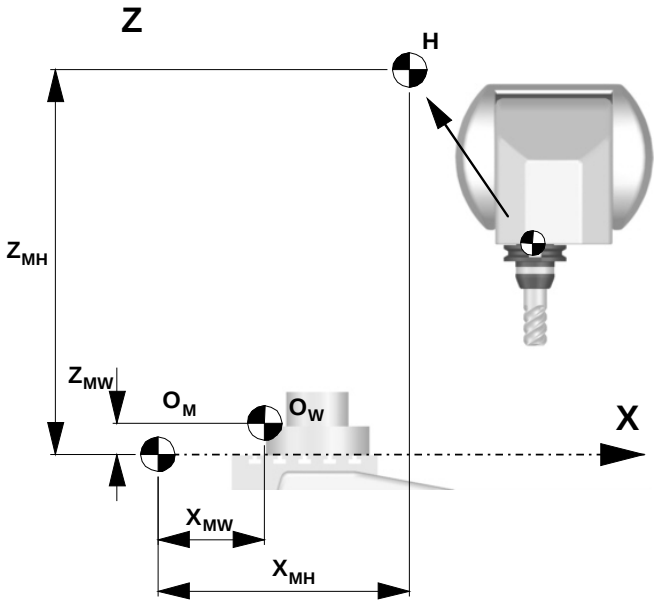
- Manual home search, in JOG mode. The axes are homed one by one. The CNC does not keep the part zero and the coordinates are displayed referred to machine reference zero.
- Automatic home search, in JOG mode. This kind is only available when the home search subroutine associated with function G74 (parameter REFPSUB) has been defined. All the axes are homed at the same time. The zero offsets are not canceled. The position values are displayed in the active reference system.
- Home search by program or MDI using function G74. The zero offsets are not canceled. The position values are displayed in the active reference system.

When searching home, the axes move to a known point of the machine and the CNC assumes the coordinate values assigned to that point by the machine manufacturer, referred to machine zero. If the system uses distance-coded reference marks or absolute feedback, the axes will only move the distance necessary to verify their position.

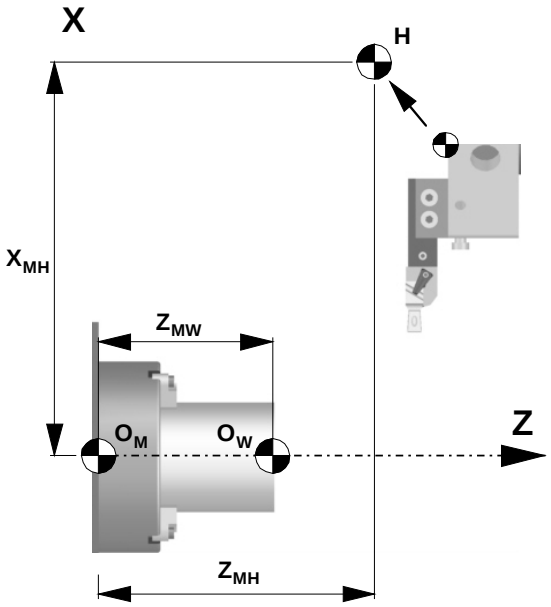
Machine reference system and machine reference point.

In order to perform the home search, the machine manufacturer has set particular points of the machine; the machine zero and the machine reference point.

- The machine reference zero is the origin point of the machine reference system, set by the machine manufacturer.
- The machine reference point is a point set by the manufacturer and referred to the machine reference zero. This point may be located anywhere on the machine. The position of the reference point, for each axis, is set at parameter REFVALUE.



OM	Machine reference zero (home).
OW	Part zero
H	Machine reference point.
XMW, YMW, ZMW,	Coordinates of the part zero.
XMH, YMH, ZMH,	Coordinates of the reference point.


X
H
X_{MH}
Z_{MW}
O_M
O_W
Z
Z_{MH}

9.6.1 Home search (axes and spindles).

Home search with a feedback system that does not have distance-coded reference marks.

Parameter.	Meaning.
I0TYPE	Reference mark (I0) type
DECINPUT	The axis has a home switch.
REFDIREC	Home searching direction.
REFFEED1	Fast home searching feedrate.
REFFEED2	Slow home searching feedrate.

When the feedback system does not have distance-coded reference marks, the axis must always be homed at a specific point of the machine which is referred to as Machine Reference Point. The installer must place a home switch for each axis at the machine reference point.

When starting the search, the axis moves in the direction set by parameter REFDIREC and at the feedrate indicated by parameter REFFEED1 until reaching the home switch. When the axis presses the home switch, it starts moving in the opposite direction at the feedrate set by parameter REFFEED2. It keeps on moving after releasing the home switch until the CNC detects a marker pulse (I0) from the feedback device.

Home search with a feedback system that has distance-coded reference marks.

Parameter.	Value.
I0TYPE	Increasing distance-coded Decreasing distance-coded
REFDIREC	Home searching direction.
REFFEED2	Slow home searching feedrate.

When the feedback system has distance-coded reference marks (coded I0) may be referenced (homed) at any point of the machine; there is no need for home switches. However, it will be necessary to set the machine reference point if the axis uses leadscrew error compensation because the amount of error at the machine reference point must be ·0·.

The axis moves the minimum distance possible, less than 200 mm, in the direction set by parameter REFDIREC and at the feedrate indicated in "REFFEED2" until the CNC receives a reference marker pulse from the feedback system.. If before receiving the reference signal, the axis presses the home switch, it starts moving in the opposite direction and goes on with the process getting away from the switch until the CNC receives the reference marker pulse of the feedback system.

Redefine the machine reference point after having dismounted or replaced the feedback system.

Parameter.	Value.
REFVALUE	Home position.
REFSHIFT	Offset of the reference point (home)

Sometimes, to readjust the machine, it is necessary to take down the feedback device, thus when putting back up, the new home point might no coincide with the previous one.

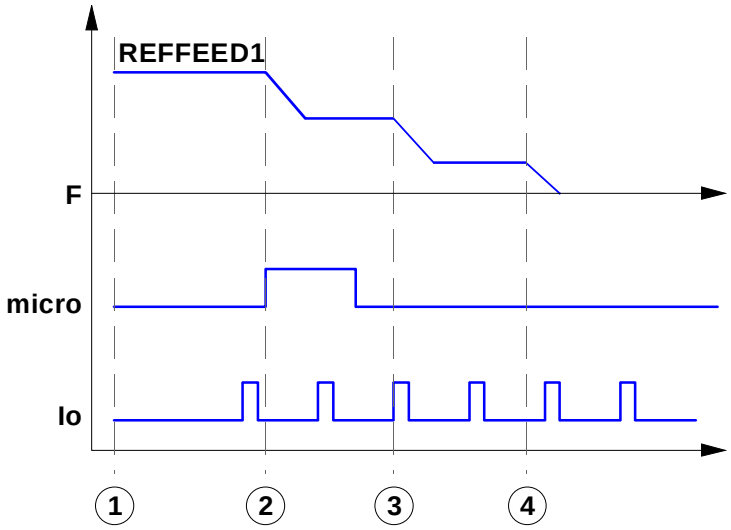
Since the home point must still be the same, the difference between the new point and the old point must be assigned to parameter REFSHIFT . This way, when the axis finds the I0, it moves the distance indicated in REFSHIFT and at that point updates its coordinate to the value of REFVALUE..

When the axes are position-Sercos; during parameter matching, the CNC sends the value of parameter REFSHIFT to the drive so it takes it into account; this way the CNC coordinate and that of the drive will be the same.

Peculiarities of home search for Mechatrolink axes.

Parameter.	Value.
REFFEED1	Fast home searching feedrate.

The servo carries out the home search and the home switch must be wired to the servo. The homing procedure follows this sequence. Refer to the documentation of the device for further detail.



1 When homing starts, the axis moves in the homing direction (set by the parameters of the servo) at the feedrate set by parameter REFFEED1.

2 When the axis presses the home switch, it switches to approach feedrate 1 (set by the parameters of the servo).

3 When the axis releases the home switch and receives the home signal, it switches to approach feedrate 2 (set by the parameters of the servo).

4 Once the axis has traveled the final positioning distance (set by the parameters of the servo), the servo concludes the home search.

Peculiarities of home search for spindles, tandem system and analog axes.

Peculiarities of home search for spindles.

Parameter.	Value.
REFINI	Home search in the first movement.
NPULSES	Number of encoder pulses.
NPULSES2	Number of pulses of the encoder (external feedback).

Parameter REFINI determines whether the CNC homes the spindle in its first movement or not. The CNC takes this parameter into account only when parameters NPULSES y NPULSES2 have been set with a value other than 0.

If the spindle has been homed, the spindle orientation in M19 does not force a new home search.

The CNC will home the spindle again in the following situations.

- When programming a new home search with function G74, via program or MDI/MDA mode.
- When the spindle exceeds the feedback pulses reading limit.
- After a failure at the Sercos ring.
- When replacing the encoder.

Peculiarities of home search for a tandem system.

In a tandem system, only the master axis is homed; the home search is transparent for the slave that just moves along with the master.

Peculiarities of home search for analog axes homed in parallel.

When programming a home search on several analog axes in parallel; i.e. with the same index (for example "G74 X1 Y1 Z1"), the execution sequence is the following.

1. All the axes move at the same time, each one at its feedrate REFFEED1 until each one detects its home switch. The axes wait on the switch until all the axes involved in the home search detect the home switch.
2. Once all the axes have reached their switches, the home search begins sequentially on each axis at feedrate REFFEED2 starting with the last axis reaching its home switch and following their order according to the logic number of the axes.
3. If any of the axes of the programmed group does not have a home switch, the axis waits for the rest of the axes to reach their switches and then starts looking for its reference mark at REFFEED2 feedrate in its corresponding order (sequence).
4. If none of the axes of the programmed group has a home switch, the home search starts at REFFEED2 with the axis having the lowest logic number and when that axis has finished, it goes on sequentially with the rest.

9.6.2 Home search (gantry axes).

Gantry axes may be homed in the same three ways described for the rest of the axes and spindles. The following requirements must be met when setting the parameters for the Gantry axes.

- The type of reference mark (parameter I0TYPE) must be the same for both axes.
- When not using distance-coded reference marks (I0), either both axes or just the master axis may have a home switch (parameter DECINPUT).

Analog axes and velocity-Sercos axes.

Home search with a feedback system that does not have distance-coded reference marks. The master and slave axes have a home switch.

Parameter.	Meaning.
I0TYPE	Reference mark (I0) type Normal (non-distance-coded) on both axes. To be defined for the master and slave axes.
DECINPUT	The axis has a home switch. To be defined for the master and slave axes.
REFDIREC	Home searching direction. To be defined for the master axis.
REFFEED1	Fast home searching feedrate. To be defined for the master axis.
REFFEED2	Slow home searching feedrate. To be defined for the master axis.
REFVALUE	Home position. To be defined for the master and slave axes.
REFSHIFT	Offset of the reference point (home) To be defined for the master and slave axes.

The CNC starts moving both axes in the direction indicated by parameter REFDIREC of the master axis. This movement is carried out at the feedrate indicated by parameter REFFEED1 of the master axis until any of the axis presses its home switch. After pressing the home switch, both axes move at REFFEED2 feedrate of the master axis until the axis that pressed the home switch detects its reference mark. After detecting the reference mark, the CNC resets the position value of this axis to the value set by parameter REFVALUE and starts homing the second axis.

To home the second axis, both axes move at the feedrate indicated by parameter REFFEED1 of the master axis until the second axis presses its home switch. After pressing the home switch, both axes move at REFFEED2 feedrate of the master axis until the axis that pressed the home switch detects its reference mark. After detecting the reference mark, the CNC resets the position value of this axis to the value set by parameter REFVALUE.

If the first axis pressing its home switch was the master and its parameter REFSHIFT is other than zero, the slave axis does not start homing until the master axis ends the movement corresponding to parameter REFSHIFT. The REFSHIFT parameter of the slave axis is applied when it resets the position value after having detected the reference mark without making the movement.

Home search with a feedback system that does not have distance-coded reference marks. Only the master axis has a home switch.

The CNC starts moving both axes in the direction indicated by parameter REFDIREC of the master axis. This movement is carried out at the feedrate indicated by parameter REFFEED1 of the master axis until this axis presses its home switch. After pressing its home switch, both axes move at REFFEED2 feedrate of the master axis until the slave axis detects its reference mark. After detecting the reference mark, the CNC resets the position value of the slave axis to the value set by parameter REFVALUE and considering parameter REFSHIFT. To update the position value, the axis does not move.

Then, the master axis searches the reference mark. Once the reference mark has been detected, the CNC resets its position value to the one set by parameter REFVALUE and

considering parameter REFSHIFT. In this case, the axis will move to apply the REFSHIFT value.

Home search with a feedback system that has distance-coded reference marks or where none of the axes has a home switch.

The CNC starts moving both axes in the direction indicated by parameter REFDIREC of the master axis. This movement is carried out at the feedrate indicated by parameter REFFEED2 of the master axis until the slave axis detects its reference mark. After detecting the reference mark, the CNC resets the position value of this axis to the value set by parameter REFVALUE and considering parameter REFSHIFT if there is one.

Then, the master axis searches the reference mark. After detecting the reference mark, the CNC resets the position value of the master axis to the value set by parameter REFVALUE and considering parameter REFSHIFT if there is one.

Position-Sercos axes

To homing these axes, both must either have distance-coded reference marks or a home switch. If both axes have a home switch but do not have distance-coded reference marks, the home switches must be placed in such way that the master axis is not pressing its home switch while homing the slave axis (first to one to search its reference mark).

The CNC sends to the drive the command to home the slave axis. In the process, the CNC reads the increment in the real (actual) slave position value and sends it to the master axis as an increment of the theoretical position value, hence making the master axis follow the slave axis only by the lag between the theoretical and real position values.

When the slave axis has been homed, the CNC sends to the drive the command to home the master axis. In this case, the slave axis follows the master axis.

In order to prevent the axis following the one being homed from jerking due to the change in position value, the CNC makes sure that the increment of the real position value corresponds at the most with the homing feedrate (parameter REFFEED1) and that it does not exceed the maximum acceleration allowed for that axis.

When both axes have been homed, the CNC takes into account the difference between the position values to correct it, if so permitted by parameter DIFFCOMP.

Coordinate (position) difference compensation after a home search.

Parameter.	Meaning.
DIFFCOMP	Compensate for the coordinate (position) difference between the two axes after G74.

The CNC corrects the position difference between the master and the slave axes after they are homed. The position value correction (compensation) may be applied to any type of axis; analog, position-Sercos and velocity-Sercos.

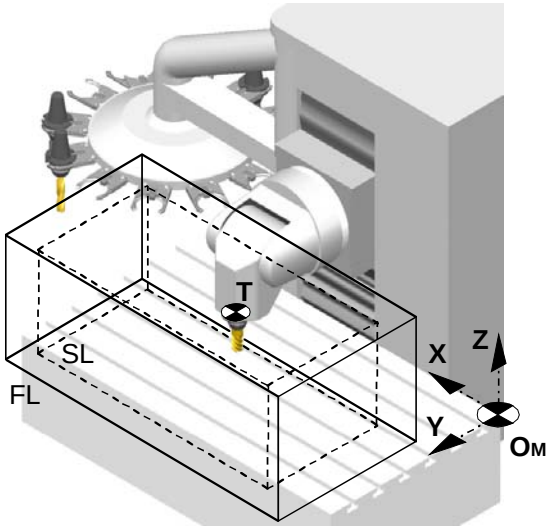
Position value compensation is enabled by parameter DIFFCOMP and it is applied with marks DIFFCOMP(axis) and SERVO*ON(axis) as follows.

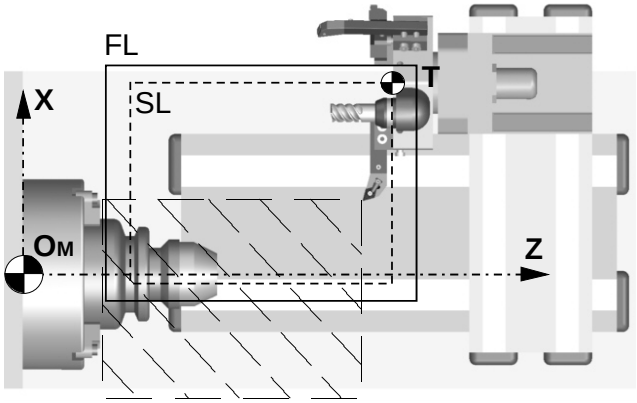
- With the up-flank of SERVO*ON if DIFFCOMP is active.
- With the up-flank of DIFFCOMP if SERVO*ON is active.

To compensate the position value, the slave axis will move until reaching the position of the master axis at the feedrate set by parameter REFFEED2. This process can only be interrupted with RESET. The REFPOIN(axis) mark of both axes is activated when the position value compensation is completed.

9.7 Software limits of the axes.

The software limits set the travel limits for the axes to prevent the carriages from reaching the mechanical hard stops. The carriages reach the hard stops when the reference point of the tool holder is placed at the physical limits. The CNC lets set the software limits on linear axes and on linearlike rotary axes.



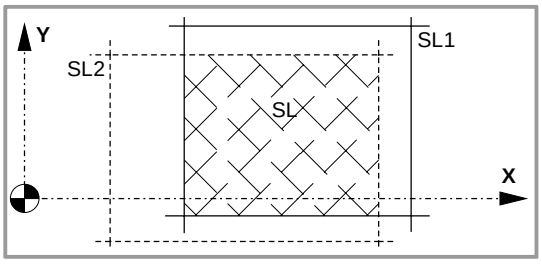


OM	Machine zero (home).
T	Reference point of the tool holder.
FL	Physical limits.
SL	Software limits applied by the CNC.
	Programmable positions of the axes (they depend on the active tool).

The programmable positions of the axes depend on the dimensions of each tool. When programming a position where the reference point of the tool holder is beyond the software limits, the CNC interrupts the execution and shows the pertinent error message.

Software limits applied by the CNC.

The CNC can have two groups of software limits active, where each group may consist of an upper limit and a lower limit for each axis; i.e. each axis may have two upper limits and two lower limits. Of the four possible software limits for each axis, the CNC will apply the most restrictive upper and lower limit even if they do not belong to the same group.



CNC behavior when an axis reaches the software limits.

When an axis reaches the software limits, the CNC interrupts the execution and shows the pertinent error message. To take the axis to the work zone, access the JOG mode and move the axis or spindle that overran the travel limit. The axis or spindle can only move in the direction that places it within the limits.

The CNC offers the following variable to indicate that an axis has reached the software limits.

Variable.	Meaning.
(V.)[ch].G.SOFTLIMIT	Software limits reached.

The PLC activates one of these marks to indicate to the CNC that the corresponding axis or spindle has reached the positive or negative travel limit.

PLC mark.	Meaning.
LIMITPOS(axis)	The axis has reached the upper software travel limit.
LIMITNEG(axis)	The axis has reached the lower software travel limit.

9.7.1 How to set the software travel limits.

Considerations for setting the software travel limits.

The software travel limits may be positive or negative, but the lower limits must be smaller than the upper ones; otherwise, the axis might not move in any direction.

If when changing the limits, an axis is positioned beyond them, that axis can only be moved in the direction that places it within those limits.

If for an axis, both upper and lower limits of a group are set to ·0·, the CNC cancels those limits on that axis and applies the ones of the other group.

The software limits are always applied in radius, regardless of the setting of parameter DIAMPROG, of the active function G151/G152 and of the method chosen to set them.

Default software travel limits (first software travel limits).

The default software travel limits, those assumed by the CNC on power-up, are set by machine parameters; if both parameters are set to ·0·, these limits are canceled.

Parameter.	Meaning.
LIMIT+	Upper software travel limits.
LIMIT-	Lower software travel limits.

These software travel limits may be changed by program or MDI using the following functions and variables. When modifying the limits with these functions or variables, the CNC assumes those values as the new limits in this group.

Function.	Variable.	Meaning.
G198	(V.)[ch].A.NEGLIMIT.xn	Lower software travel limits.
G199	(V.)[ch].A.POSLIMIT.xn	Upper software travel limits.

Setting both upper and lower limits of an axis to ·0· cancels the limits, regardless of their values in the machine parameters. In this case, the CNC will apply the second software travel limits to the axis.

The new limits will be maintained after executing an M02 or M30, and after an emergency or a reset. On power-up or after validating the axis machine parameters the CNC assumes the software limits set by the machine parameters.

Modifying the software travel limits with functions G198/G199.

When programming G198 or G199, the CNC interprets that the coordinates programmed next set the new software limits.

```
G198 X-1000 Y-1000
    (New lower limits X=-1000 Y=-1000)
G199 X1000 Y1000
    (New upper limits X=1000 Y=1000)
```

Depending on the active work mode G90 or G91, the position of the new limits will be defined in absolute coordinates (G90) in the machine reference system or in incremental coordinates (G91) referred to the current active limits.

```
G90 G198 X-800
    (New lower limit X=-800)
G91 G198 X-700
    (New incremental lower limit X=-1500)
```

Second software travel limits.

These software travel limits are set with the following variables that can be written via part-program, MDI, PLC or interface. These variables are initialized on power-up assuming the maximum value possible. When modifying the limits with these variables, the CNC assumes those values as the new limits in this group.

Variable.	Meaning.
(V.)[ch].A.RTNEGLIMIT.xn	Lower software travel limits (second limits).
(V.)[ch].A.RTPOSLIMIT.xn	Upper software travel limits (second limits).

Setting both upper and lower limits of an axis to ·0· cancels the limits and the CNC applies the first software travel limits to the axis.

9.7.2 Set the tolerance for an axis located at the software travel limits.

The tolerance for the software travel limits is set with machine parameter SWLIMITTOL. The CNC applies this tolerance to the active software limits.

Parameter.	Meaning.
SWLIMITTOL	Software limit tolerance.

This tolerance indicates the maximum variation or oscillation allowed to a real coordinate of an axis referred to the software limits, before issuing an error indicating travel limit overrun. The programmed theoretical movement of the axis is only possible up to the exact limit, but the real axis coordinate is allowed this margin before the error is issued. When they are DRO axes, the error is also issued when the real coordinate exceeds the limit over the tolerance.

When no theoretical movement has been programmed, the limit overrun error will only be issued when exceeding the tolerance in a sampling period (cycle time); for example, when hitting the axis causing it to overrun the limits abruptly. In any other case, if no theoretical movement has been programmed for the axis, the error will not be issued even if it overruns the limits.

9.8 Configure a handwheel as "feed handwheel".

Usually, when machining a part for the first time, the feedrate is controlled by the switch on the operator panel. The "feed handwheel" allows using one of the handwheels of the machine to control that feedrate depending on how fast the handwheel is turned.

The "feed handwheel" must be managed by the PLC. Usually, this feature is turned on and off using an external push button or key configured for that purpose.

Know the number of pulses sent by the handwheel.

The following variable reads the number of handwheel pulses.

Variable.	Meaning.
(V.)G.HANDP[hw]	Number of pulses sent by the handwheel since the system was started up.

PLC maneuver.

The PLC maneuver must consider the following elements.

- Inhibit via PLC all the positions of the feedrate override switch.
- Detect how much the handwheel has turned (reading - counting - the pulses received).
- Set via PLC the feedrate override depending on the handwheel pulses.

Example of a PLC maneuver. The machine has a push button to enable and disable the "feed handwheel" (input I71) and the feedrate control is handled by the second handwheel.

```
; Resources used in the PLC program.
; I71 _____ Push button to enable and disable the "feed handwheel"
; R100 _____ Total number of handwheel pulses.
; R101 _____ Number of handwheel pulses until previous reading.
; R102 _____ Number of handwheel pulses since previous reading.
; R103 _____ Calculated percentage of feedrate.
; M1000 _____ "Feed handwheel" on.
; M1001 _____ Auxiliary mark.

CY1
() = ERA R101
    Initialize the register that contains the reading (counting) of handwheel pulses.
END

PRG
DFU I71 = CPL M1000
    Every time the button associated with the "feed handwheel" is pressed, complement
    mark M1000.
NOT M1000
= AND KEYDIS3 $FF800000 KEYDIS3
= JMP L101
    If the "feed handwheel" is not off, inhibit all the positions of the feedrate override
    switch and resume the execution of the program.
M1000 = MSG1
    If the "feed handwheel" is on, show a message.

DFU CLK100
= CNCRD (G.HANDP[2], R100, M1001)
= SBS R101 R100 R102
= MOV R100 R101
= MLS R102 3 R103
= OR KEYDIS3 $7FFFFF KEYDIS3
    If the "feed handwheel" is on and an up flank occurs at the clock CLK100, the PLC
    saves in register R100 the handwheel pulses; calculates in register R102 the pulses
    received since the previous reading (count); updates register R101 for the next
    reading; calculates in register R103 the right feedrate override percentage; inhibits
    all the positions of the feedrate override switch (KEYDIS3).
```

```
CPS R103 LT 0 = SBS 0 R103 R103
CPS R103 GT 120 = MOV 120 R103
    Adjust the value of register R103; ignore the handwheel turning direction (sign) and
    limit the value to 120%.
DFU CLK100
= CNCWR (R103,PLC.FRO, M1001)
    Use the up flank at the clock CLK100 to set the calculated feedrate override value.
L101
END
```

9.9 Configuration of the HSC mode (High Speed Cutting).

Part-programs generated by CAD/CAM systems consist of a large number of blocks of all sizes, from a few millimeters to a few tenths of a micron.

In this type of parts, the CNC must be capable to analyze a large number of points in advance so it can generate a continuous path that goes through (or near) the points of the program while keeping (the best way possible) the programmed feedrate and the restrictions of maximum acceleration, jerk, etc of each axis and of the path.

The command to execute this type of programs, typical of high speed machining, is carried out with the instruction #HSC. This function offers several ways to work; optimizing the chordal error (CONTERROR mode) or the machining feedrate (FAST mode).

9.9.1 Configuration of the HSC mode.

The HSC mode is configured with the following axis and general machine parameters.

Parameter.	Meaning.
FEEDAVRG	Calculating feedrate average.
SMOOTHFREQ	Smoothing frequency in the interpolation.
HSCFILTFREQ	Filter frequency (CONTERROR mode).
FASTFACTOR	Default feedrate percentage (FAST mode).
FTIMELIM	Time difference permitted in feedrate interpolation (FAST mode).
MINCORFEED	Minimum feedrate at the corners.
FSMOOTHFREQ	Smoothing frequency in the interpolation (FAST mode).
FASTFILTFREQ	Filter frequency (FAST mode).
CORNER	Maximum angle of the corner to machine it in square corner mode.

Parameter.	Meaning.
CORNERACC	Maximum acceleration permitted at the corners.
CURVACC	Maximum contouring acceleration permitted.
CORNERJERK	Maximum Jerk permitted at the corners.
CURVJERK	Maximum contouring Jerk permitted.
FASTACC	Maximum acceleration permitted (FAST mode).

General machine parameters.

FEEDAVRG
Calculating feedrate average.

This parameter enables the adjustment of the feedrate according to the block reading speed and their size. This adjustment eliminates the need to slow down due to a poor supply of small blocks; although as a result, the feedrate reached is lower, the overall machining time will improve.

The block reading speed depends on machine parameter PREPFREQ.

SMOOTHFREQ
Smoothing frequency in the interpolation..

This parameter sets the smoothing frequency in path interpolation. This parameter avoids accelerating and decelerating throughout a path, beyond a particular frequency when generating an averaged speed.

CORNER
Maximum angle of the corner to machine it in square corner mode.

This parameter indicates the maximum angle between two paths, under which the machining operation is carried out in square corner mode.

HSCFILTFREQ
Filter frequency (CONTERROR mode).

This parameter activates an automatic FIR filter for all the axes of the channel during the execution in CONTERROR mode. This filter may be used to smooth the response of the axes generating a smoother tool path and, if necessary, reduce the feedrate so the path error is adjusted to the programmed path.

This parameter inserts a variable (non constant) phase shift depending on the frequencies. This phase shift can cause a change on the path if it is not executed at the same feedrate

as, for example, when changing the feedrate override percentage or when moving back and forth on the same path.

FASTFACTOR
Default feedrate percentage (FAST mode).

This parameter indicates the feedrate at the corners and indicates the feedrate percentage to be reached, of the maximum that the CNC can reach in FAST mode. This parameter indicates the default value and it may be changed from the part program.

FTIMELIM
Time difference permitted in feedrate interpolation (FAST mode).

The feedrate interpolation in HSC FAST mode improves the machining quality. In rather large blocks where the feedrate adaptation represents a longer additional time than the one indicated in FTIMELIM, the CNC will adapt the feedrate using the maximum dynamics to avoid unnecessary waste of time. When programmed with a 0 value, the CNC does not carry out the velocity interpolation.

MINCORFEED
Minimum feedrate at the corners.

This value should not be modified because it is possible to exceed the dynamics of the axes.

FSMOOTHFREQ
Smoothing frequency in the interpolation (FAST mode).

This parameter sets the smoothing frequency in path interpolation for the HSC FAST mode. This parameter avoids accelerating and decelerating throughout a path, beyond a particular frequency when generating an averaged speed. It is only applied when the HSC FAST mode is active.

FASTFILTFREQ
Filter frequency (FAST mode).

This parameter activates an automatic filter for all the axes of the channel while executing in HSC FAST mode that permits smoothing the response of the axes by generating a smoother path. This filter has the drawback of rounding the corners slightly.

This parameter inserts a constant phase shift regardless of the frequencies.

Axis machine parameters.

CORNERACC
Maximum acceleration permitted at the corners.

This parameter sets the maximum acceleration allowed for the axis during block transition. If this parameter is set with a ·0· value, the maximum acceleration of the axis is respected.

CURVACC
Maximum contouring acceleration permitted.

If this parameter is set with a ·0· value, the maximum acceleration of the axis is respected.

CORNERJERK
Maximum Jerk permitted at the corners.

If this parameter is set with a ·0· value, the maximum jerk of the axis is respected.

CURVJERK
Maximum contouring Jerk permitted.

If this parameter is set with a ·0· value, the maximum jerk of the axis is respected.

FASTACC
Maximum acceleration permitted (FAST mode).

If this parameter is set with a ·0· value, the maximum acceleration of the axis is respected.

9.9.2 Influence of the type of acceleration and of the filters in HSC mode.

HSC optimizing the chordal error (CONTERROR mode).

It is the recommended way (mode) to work. In this mode, the CNC modifies the geometry through intelligent algorithms for eliminating unnecessary points and generating splines or polynomial transitions between blocks always respecting the error limits set for this mode. This way, the contour is traveled at a variable feedrate according to the curvature and the programmed parameters (acceleration and feedrate) but without going beyond the set error limits.

This mode uses this type of acceleration is defined with parameter SLOPETYPE. The following situations can occur when the type of acceleration has jerk.

- Jerk control permits pretty smooth velocity and acceleration profiles.

  In this case, if there is no need to eliminate resonance frequencies, there is no need to activate filters to make the path smoother. Since there are no filters, the part will be more accurate because the path gores through the programmed points. The exception is on sections where the CNC disconnects the spline because it exceeds the error limit and, in that case, it executes a rounding polynomial that passes at a distance from the programmed point lower than the error limit.

- Since the HSC mode tries to respect the jerk throughout the path, some irregularities may result on the velocity profile, especially at the corners, and, consequently, affect the quality of the surface finish.

  In this case, the amount of jerk can be increased at the corners (parameter CORNERJERK) or apply the smoothing frequency in the interpolation of the path (parameter SMOOTHFREQ). Activating the smoothing frequency generates smoother velocity and acceleration profiles, but increasing the execution time.

The effects of the filters set by the FILTER parameter are added to the effects of the filters active in HSC; parameter HSCFILTFREQ for the CONTERROR mode.

HSC optimizing the feedrate (FASTmode).

It is the recommended way (mode) to work in the following circumstances.

- When feedrate is more important than accuracy.
- When the CONTERROR mode does not provide the desired results. When the error with which the part has been generated is not significantly lower than the error demanded to the CONTERROR, the execution is slower and the machined surfaces show ridges.
- When the dynamics of the machine does not respond in wide range of frequencies; i.e. the machine shows resonance or a limited bandwidth.

The FAST mode provides a smoother surface finish at a more constant feedrate. From the program, it is possible to indicate the percentage of feedrate to apply, over the maximum that the CNC can reach depending on the setting of parameters LACC1 and LACC2. Programming it is optional; if not defined, it assumes the percentage set in parameter FASTFACTOR. This parameter affect directly the feedrate when going through the corners and it must be borne in mind when reaching a compromise between the feedrate at the corner and its finish quality.

This mode uses linear acceleration, regardless of the setting of parameter SLOPETYPE. For best results, it is essential to set parameters LACC1 and LACC2 properly. Since there is no jerk control, a filter is required for all the axes, inserted automatically, to make the machine movement smoother. The frequency of this filter for all the axis is set in parameter FASTFILTFREQ.

Using axis filters results in smoother velocity and acceleration profiles, but lower precision at the corners depending on geometry and type of filter. The higher the linear acceleration or the lower the range of frequencies of the machine, the lower the frequency of the filter needed to generate a greater loss of accuracy to narrow the bandwidth of the system.

The effects of the filters set by the FILTER parameter are added to the effects of the filters active in HSC; parameter FASTFILTFREQ for the FAST mode.

9.9.3 Procedure for analysis and adjustment of the HSC.

To adjust the HSC mode, the CNC axes must have parameters G00FEED, ACCEL, LACC1, LACC2 and ACCJERK already tested and adjusted to the maximum value allowed by the dynamics of the machine. The following procedure is meant for analyzing the behavior of the CNC while machining and evaluate whether the result may be improved or not by adjusting a parameter. The steps to follow are.

1 Evaluate the time the CNC has for its calculations and if the blocks are supplied properly.

2 Evaluate the feedrate on the path in case there are irregularities from one pass to the next.

3 Analyze the problems occurring at specific points.

The CNC has a set of variables that, when analyzed while machining, help find aspect of the HSC that may be improved in terms of execution time and machining quality. The oscilloscope will serve as a tool to analyze the evolution HSC while machining.

Evaluate the time the CNC has for its calculations and if the blocks are supplied properly.

An improper supply of blocks forces to lower the feedrate unnecessarily. The CNC offers the following variables to check whether the HSC has a proper supply of blocks or not.

Variable.	Meaning.
V.[ch].G.PERFRATE	Percentage of blocks managed by the CNC in block preparation, with respect to the best possible.
V.[ch].G.DROPRATE	Percentage of reduction of the maximum feedrate possible.
V.G.NCTIMERATE	Percentage of loop time (cycle time) used by the real time portion of the CNC.
V.G.LOOPTIMERATE	Percentage of loop time (cycle time) used by the position loop.
V.[ch].G.CHTIMERATE	Percentage of loop time (cycle time) used by the channel.
V.[ch].G.PREPTIMERATE	Percentage of loop time (cycle time) used by the channel for block preparation.
V.[ch].G.IPOTIMERATE	Percentage of loop time (cycle time) used by the interpolator of the channel.

In order to ensure a proper supply of blocks to the HSC, the following variables must return a value near 100%.

V.[ch].G.PERFRATE A value lower than 100 means that the CNC can process more blocks.

V.[ch].G.DROPRATE A value lower than 100 means that the CNC could increase the feedrate if it had more blocks.

In either case, parameter PREPFREQ may be increased to increase the supply of blocks as long as the system has enough time; in other words, the percentage of cycle time used by the CNC is not too close to the total cycle time (about 50%). This information may be checked with the following variable.

V.[ch].G.NCTIMERATE With a value lower than 50, parameter PREPFREQ may be increased.

To properly analyze this aspect, parameter FEEDAVRG must be set to ·No· to prevent the CNC from limiting the feedrate depending on the supply of blocks.

Evaluate the feedrate on the path in case there are irregularities from one pass to the next.

For similar machining passes, the velocity graph should look the same. However, in some passes, the feedrate could decrease even more and the machining could be less symmetrical

than what appears on the graph. The CNC has the following variables to analyze the cause of this feedrate reduction.

Variable.	Meaning.
V.[ch].G.PARLIMF	Cause that limits the feedrate at the block being executed.
V.[ch].G.AXLIMF	Logic number of the axis that limits the feedrate at the block being executed.
V.[ch].G.PARLIMC	Cause that limits the feedrate at the corner in the current block.
V.[ch].G.AXLIMC	Logic number of the axis that limits the feedrate at the corner, in the block being executed.
V.[ch].G.LIMERROR	Value of the error that cancels the splines (CONTERROR mode).
V.[ch].G.PATHFEED	Theoretical feedrate on the tool path.
V.[ch].G.FREAL	Real feedrate on the tool path.

To analyze the cause of feedrate reduction, use a trace of the oscilloscope that shows an undesired feedrate reduction. Move the cursor to the value where the theoretical feedrate (V.[ch].G.PATHFEED) decreases and check whether the cause that limits the feedrate may be adjusted or the generated error is greater than the one programmed.

Causes that limit the feedrate on the tool path.

Check the V.[ch].G.AXLIMF variable to know which axis is limiting the feedrate and the V.[ch].G.PARLIMF variable to know the reason why this axis is limiting the feedrate. This last variable returns one of the following values.

Value.	Meaning.
1	Maximum feedrate of the axis.
2	Acceleration due to curvature (parameter CURVACC).
3	Jerk due to curvature (parameter CURVJERK).
6	Error committed by the spline.
7	Insufficient memory due to very small blocks and/or buffer full.
10	Maximum axis feedrate in transformations.
11	Maximum axis acceleration in transformations.

The error incurred by the spline may be checked with the V.G.LIMERROR variable. Depending on the cause, some HSC adjusting parameter may be tweaked.

- Parameter CURVACC may be changed if the feedrate is limited by the acceleration due to curvature).
- Parameter CURVJERK may be changed if the feedrate is limited by the jerk due to curvature).

Causes that limit the feedrate at the corners.

Check the V.[ch].G.AXLIMC variable to know which axis is limiting the feedrate and the V.[ch].G.PARLIMC variable to know the reason why this axis is limiting the feedrate. This last variable returns one of the following values.

Value.	Meaning.
1	Maximum feedrate of the axis.
4	The acceleration at the corner (parameter CORNERACC).
5	The jerk at the corner (parameter CORNERJERK).
8	The chordal error at the corner (CONTERROR).
9	Geometry at the corner (FAST mode).

Check whether the generated error is greater than the one programmed or not.

Check the vV.[ch].G.LIMERROR variable to know the value of the error that cancels the splines. This last variable returns one of the following values.

Value.	Meaning.
-1	The generated profile does not exceed the programmed error. The programmed error does not limit the maximum feedrate of the axis.
Value	Value of the error that cancels the splines.

If the value returned by this variable is greater than the one programmed, it would be a good idea to execute the HSC with more error or generate the program with less error in the CAM.

Analyze the problems occurring at specific points.

If a problem comes up at a specific point, use a trace of the oscilloscope to analyze the following variables and locate the velocity profile that causes the problem.

Variable.	Meaning.
V.[ch].G.BLKN	Last block executed (number).
V.[ch].G.LINEN	Number of program blocks executed or prepared.
V.[ch].G.PATHFEED	Theoretical feedrate on the tool path.

Then, use the oscilloscope to analyze the following variables, sequentially because it only admits four, to find the cause of the feedrate problem.

Variable.	Meaning.
V.[ch].G.PARLIMF	Cause that limits the feedrate at the block being executed.
V.[ch].G.AXLIMF	Logic number of the axis that limits the feedrate at the block being executed.
V.[ch].G.PARLIMC	Cause that limits the feedrate at the corner in the current block.
V.[ch].G.AXLIMC	Logic number of the axis that limits the feedrate at the corner, in the block being executed.
V.[ch].G.LIMERROR	Value of the error that cancels the splines (CONTERROR mode).
V.[ch].G.PATHFEED	Theoretical feedrate on the tool path.
V.[ch].G.FREAL	Real feedrate on the tool path.

Analyze the various variables of the axes in the problem area to determine if there is an error on the theoretical path, in the control loop or in the feedback. The following variables may be used for this latter analysis.

Variable.	Meaning.
V.[ch].A.IPOPOS.xn	Theoretical position value (coordinate) at the output of the interpolator, before the transformation; i.e. in part coordinates.
V.[ch].A.ADDMANOF.xn	Distance moved with G201.
V.[ch].A.INDPOS.xn	Theoretical coordinate of the independent interpolator.
V.[ch].A.FILTERIN.xn	Theoretical coordinate of the interpolator before the filter.
V.[ch].A.FILTEROUT.xn	Theoretical coordinate of the interpolator after the filter.
V.[ch].A.LOOPTPOS.xn	Theoretical coordinate at the input of the position loop.
V.[ch].A.LOOPPOS.xn	Real coordinate at the input of the position loop.
V.[ch].A.TFEED.xn	Instantaneous theoretical speed value at the input of the position loop.
V.[ch].A.FEED.xn	Instantaneous real speed value at the input of the position loop.
V.[ch].A.POSCMD.xn	Position command for Sercos.
V.[ch].A.FTEO.xn	Velocity command for Sercos (in rpm).
V.[ch].A.POSNC.xn	Position feedback.

9.9.4 Summary of the useful variables to analyze the HSC.

ANALYSIS OF THE LOOP TIME (CYCLE TIME) AT THE CNC.

V.G.NCTIMERATE

Percentage of loop time (cycle time) used by the real time portion of the CNC.

This variable serves to evaluate the load of the system and the time that the periodic interruption that controls the movement of the axes leaves the operating time to manage other applications that may be executed in parallel with the CNC. Applications that need time of the operating system are, for example, displaying the screens, user interface applications, refreshing variables on the screen, file management (subroutines or programs that are opened and closed while machining), etc.

If there isn't enough free time for the applications, it is possible to decrease parameter PREPFREQ, increase parameter LOOPTIME, group the subroutines in a the same file or reduce the number of external applications

V.G.LOOPTIMERATE

Percentage of loop time (cycle time) used by the position loop.

This variable serves as reference to know whether the consumption of interruption time is due to the number of axes or to the path preparation process itself.

If the position loop uses most of the interruption time, it means that the system is overloaded by the axes and, therefore, you will have to evaluate the possibility to increase the LOOPTIME parameter.

ANALYSIS OF THE LOOP TIME (CYCLE TIME) IN THE CHANNEL.

V.[ch].G.CHTIMERATE

Percentage of loop time (cycle time) used by the channel.

This variable helps determine whether the particular execution of a channel is taking up too much time.

V.[ch].G.PREPTIMERATE

Percentage of loop time (cycle time) used by the channel for block preparation.

This variable serves to evaluate the load in path preparation and to know whether it is possible to increase the PREPFREQ parameter or not.

V.[ch].G.IPOTIMERATE

Percentage of loop time (cycle time) used by the interpolator of the channel.

This variable serves to evaluate the overload in the path generating algorithm and the smoothing algorithm.

VARIABLES ASSOCIATED WITH FEEDRATE LIMITATION.

V.[ch].G.PERFRATE

Percentage of blocks managed by the CNC in block preparation, with respect to the best possible to reach the maximum feedrate in each section.

Remarks.

This variable can return a value close to 100; if the percentage of blocks is lower than 100%, the HSC might be losing room to increase the feedrate, respecting the available room to brake. To know if this is the case, it will be necessary to analyze the V.G.DROPRATE variable because the reason the feedrate is not increasing might be geometrical and not due to the number of blocks available.

V.[ch].G.DROPRATE

Percentage of reduction of the maximum feedrate possible.

Feedrate reduction may be due to an improper supply of blocks or because a feedrate lower than possible has been programmed.

Remarks.

This variable can return a value close to 100; if the value is lower than 100%, the CNC may increase the feedrate if it has more blocks. To know whether the CNC can supply more blocks or not, it will be necessary to analyze the V.G.PERFRATE variable.

If both variables have a value lower than 100, parameter PREPFREQ may be increased to increase the supply of blocks as long as the system has enough time; in other words, the percentage of cycle time used by the CNC is not too close to the total cycle time (about 50%). This information may be checked in the V.G.NCTIMERATE variable.

V.[ch].G.LIMERROR

Value of the error that cancels the splines (CONTERROR mode).

This variable may be used to evaluate whether the program is generated with more error than we demand from the HSC mode or not.

Special returned values.

This variable returns one of the following values.

Value.	Meaning.
-1	The generated profile does not exceed the programmed error. The programmed error does not limit the maximum feedrate of the axis.
###	Value of the error that cancels the splines (CONTERROR mode).

V.[ch].G.AXLIMF

Logic number of the axis that limits the feedrate at the block being executed.

Together with the V.G.PARLIMF variable, it may be used to evaluate the behavior of the machining operation in a particular section where the feedrate decreases too much or is irregular.

V.[ch].G.PARLIMF

Cause that limits the feedrate at the block being executed.

Special returned values.

This variable returns one of the following values.

Value.	Meaning.
1	Maximum feedrate of the axis.
2	Acceleration due to curvature (parameter CURVACC).
3	Jerk due to curvature (parameter CURVJERK).
6	Error committed by the spline.

Value.	Meaning.
7	Insufficient memory due to very small blocks and/or buffer full.
10	Maximum axis feedrate in transformations.
11	Maximum axis acceleration in transformations.

V.[ch].G.AXLIMC

Logic number of the axis that limits the feedrate at the corner, in the block being executed.

Together with the V.G.PARLIMC variable, it may be used to evaluate the behavior of the machining operation at a particular corner where the feedrate decreases too much or is irregular.

V.[ch].G.PARLIMC

Cause that limits the feedrate at the corner in the block being executed.

Special returned values.

This variable returns one of the following values.

Value.	Meaning.
1	Maximum feedrate of the axis.
4	The acceleration at the corner (parameter CORNERACC).
5	The jerk at the corner (parameter CORNERJERK).
8	The chordal error at the corner (CONTERROR).
9	Geometry at the corner (FAST mode).

VARIABLES ASSOCIATED WITH THE FEEDRATE ON THE TOOL PATH.

V.[ch].G.PATHFEED

Theoretical feedrate on the tool path.

V.[ch].G.FREAL

Theoretical feedrate on the tool path.

Use the oscilloscope to compare the real and theoretical feedrates along the path and detect adjustment problems when both are different at specific points. Also, using variables V.G.LINEN and V.G.BLKN, it is possible to associate these feedrate changes with the program blocks or lines where they occur.

Remarks.

This variable takes into account the accelerations and decelerations of the machine. When the axes are stopped, it returns a value of ·0· and when moving it returns the value corresponding to the feedrate type G94/G95. On laser cutting machines, it is recommended to use this variable so the laser power is proportional to the feedrate.

VARIABLES ASSOCIATED WITH THE BLOCK BEING EXECUTED.

(V.)[ch].G.BLKN

Last block executed (number).

If no label has been executed, the variable will return a -1 value.

(V.)[ch].G.LINEN

Number of program blocks executed or prepared.

Remarks.

The variable returns the value of execution or preparation as follows. Reading the variable from the PLC or from the interface returns the number of blocks executed; reading it from the part-program or MDI returns the number of blocks prepared.

VARIABLES ASSOCIATED WITH THE COORDINATES IN THE LOOP.

(V.)[ch].A.IPOPOS.xn
(V.)[ch].A.IPOPOS.sn
(V.)[ch].SP.IPOPOS.sn

Theoretical position value (coordinate) at the output of the interpolator, before the transformation; i.e. in part coordinates.

(V.)[ch].A.ADDMANOF.xn

Distance moved with G201.

(V.)[ch].A.INDPOS.xn
(V.)[ch].A.INDPOS.sn
(V.)[ch].SP.INDPOS.sn

Theoretical coordinate of the independent interpolator.

(V.)[ch].A.FILTERIN.xn
(V.)[ch].A.FILTERIN.sn
(V.)[ch].SP.FILTERIN.sn

Theoretical coordinate of the interpolator before the filter.

(V.)[ch].A.FILTEROUT.xn
(V.)[ch].A.FILTEROUT.sn
(V.)[ch].SP.FILTEROUT.sn

Theoretical coordinate of the interpolator after the filter.

(V.)[ch].A.LOOPTPOS.xn
(V.)[ch].A.LOOPTPOS.sn
(V.)[ch].SP.LOOPTPOS.sn

Theoretical coordinate at the input of the position loop.

(V.)[ch].A.LOOPPOS.xn
(V.)[ch].A.LOOPPOS.sn
(V.)[ch].SP.LOOPPOS.sn

Real coordinate at the input of the position loop.

VARIABLES ASSOCIATED WITH THE VELOCITY IN THE LOOP.

(V.)[ch].A.TFEED.xn
(V.)[ch].A.TFEED.sn
(V.)[ch].SP.TFEED.sn

Instantaneous theoretical speed value at the input of the position loop.

(V.)[ch].A.FEED.xn
(V.)[ch].A.FEED.sn
(V.)[ch].SP.FEED.sn

Instantaneous real speed value at the input of the position loop.

VARIABLES ASSOCIATED WITH THE VELOCITY COMMAND AND THE FEEDBACK.

(V.)[ch].A.POSCMD.xn
(V.)[ch].A.POSCMD.sn
(V.)[ch].SP.POSCMD.sn

Position command for Sercos.

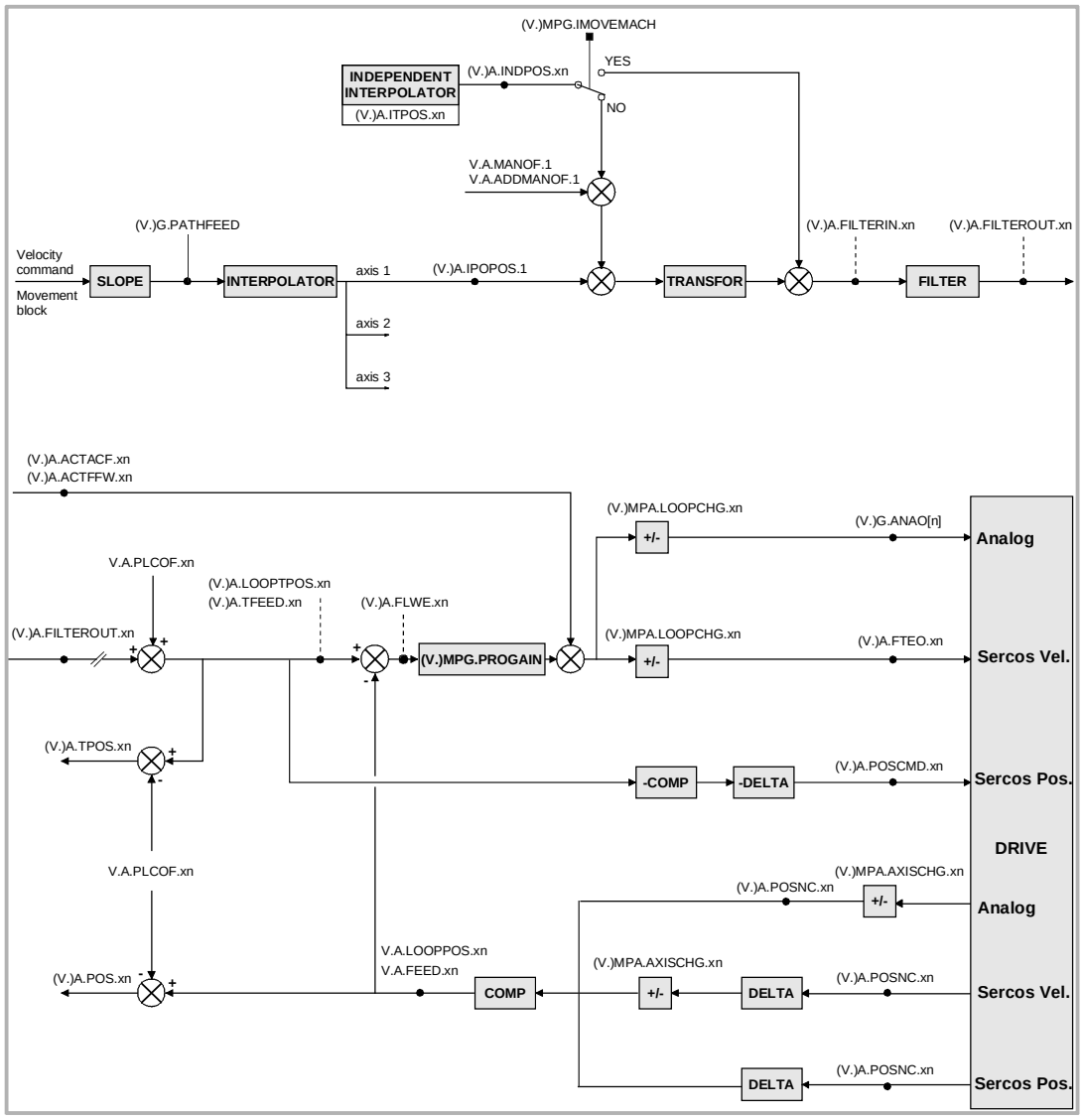
(V.)[ch].A.POSNC.xn
(V.)[ch].A.POSNC.sn
(V.)[ch].SP.POSNC.sn

Position feedback.

(V.)[ch].A.FTEO.xn
(V.)[ch].A.FTEO.sn
(V.)[ch].SP.FTEO.sn

Velocity command for Sercos (in rpm).

9.9.5 The loops and the variables.



COMP	Coordinate compensation due to leadscrew error compensation or to cross compensation.
DELTA	Offset on power-up to maintain the coordinate when it was turned off.

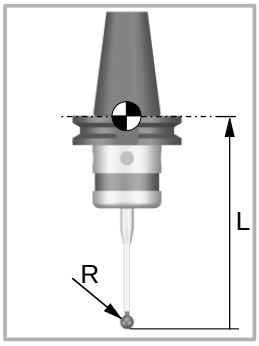
9.10 Calculation of the kinematics dimensions.

The OEM may set up to 6 different kinematics for the same machine. The CNC offers a number of pre-defined kinematics that may be easily configured by machine parameters. Besides these kinematics, the OEM can integrate 6 additional kinematics.

The dimensions to be set by parameter depend on the type of kinematics, set in parameter TYPE. The following sections show, as examples, the methods to calculate the dimensions of a kinematics with the help of a probe or a dial indicator.

9.10.1 Swivel (angular) spindle. Calculation of the dimensions using a probe.

This example calculates the dimensions of a YZ 45º angular spindle, parameter TYPE = 6. Then next example uses a probe and a sample part secured to the table, with its sides aligned with the main axes of the machine.



The machine parameters associated with the probe movement must be set and the probe must be properly calibrated. You need to know the length and radius of the probe to calculate the dimensions of the spindle.

The CNC can have configured two probes. Before any probing move, the CNC must know which is the active probe, or, which is the same, which of the two probes it must attend to. It is selected via part-program or MDI using the instruction #SELECT PROBE.

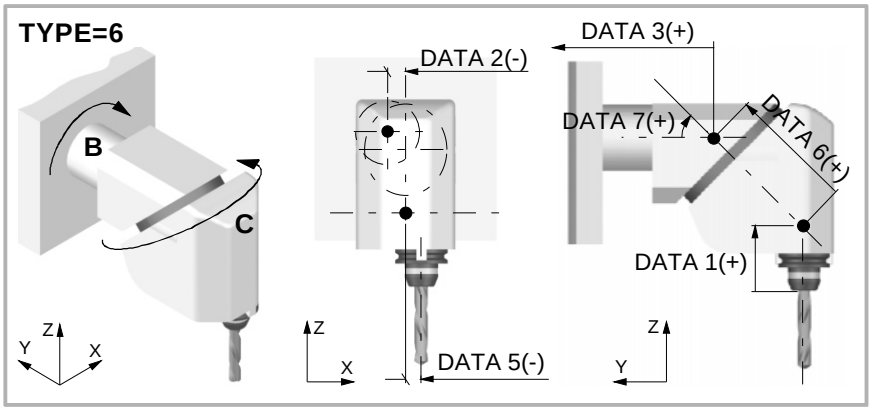


If a probing move is executed without activating the probe, it will not send any signal to the CNC when it makes contact. This can cause the probe to break because the probing move will not be stopped.

Dimensions to calculate at a YZ angular spindle (TYPE=6).

In this type of kinematics the following dimensions must be defined. All the parameters may be set with a positive or negative value. The (+) sign in the illustrations indicates that the direction is assumed as positive.

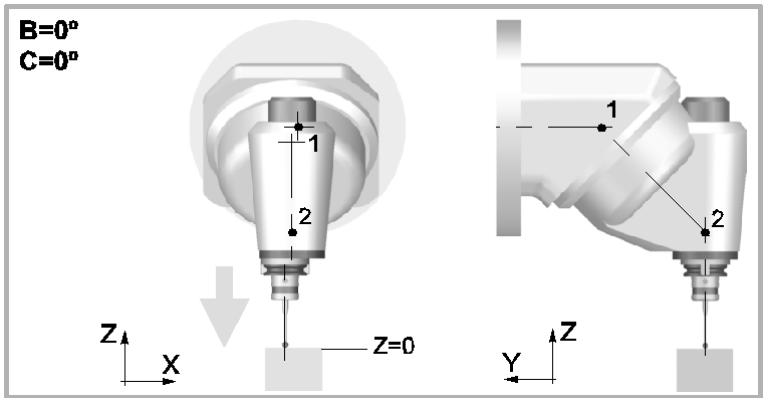
TDATA1	It indicates the distance between the tip of the quill and the secondary rotary axis along the Z axis.
TDATA2	It indicates the distance between the secondary rotary axis and the main axis along the X axis.
TDATA3	It indicates the distance between the secondary rotary axis and the main axis along the Y axis.
TDATA5	It indicates the distance between the tool axis and the secondary rotary axis along the X axis.
TDATA6	It indicates the distance between the tool axis and the secondary rotary axis along the Y axis.
TDATA7	It indicates the angle between the main and secondary rotary axes on rotary spindles (in this case TDATA7 = 45).



Calculation of parameter TDATA11.

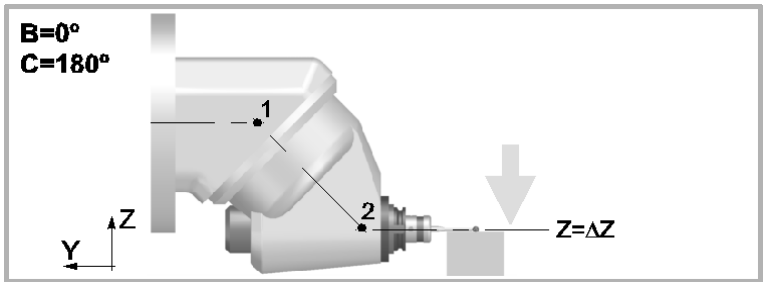
First probing move.

With rotary axes B and C positioned at 0º, make a probing movement on the Z side and preset the position Z = 0.



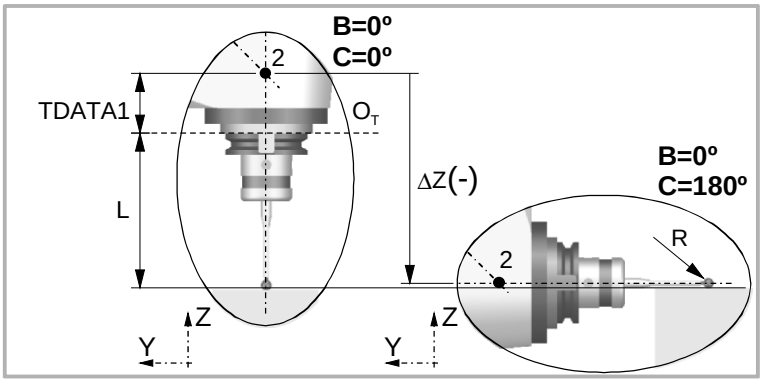
Second probe.

With rotary axis B positioned at 0º and the C axis at 180º, make a probing movement on the same side. Jot down the position value (coordinate) shown at the CNC, appearing in the drawing as ΔZ.



Mathematical calculations.

After changing the position with respect to the center of rotation ·2·, the only unknown data after both probing movements is TDATA1.

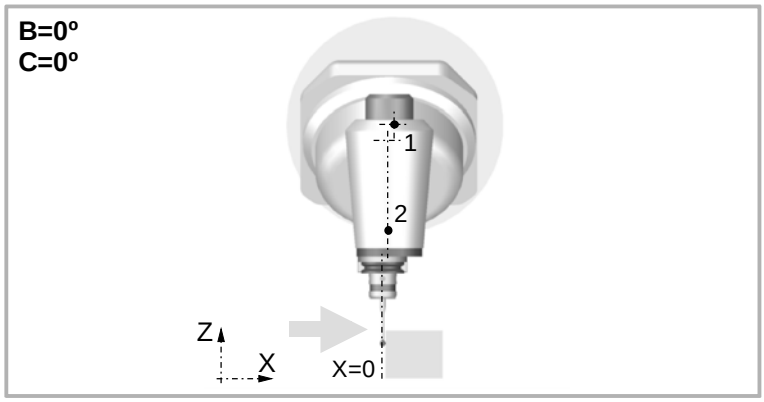


Solution.	
$\|TDATA1\| = \|\Delta Z\| + R - L$	
TDATA1	Dimensions to calculate.
OT	Reference point of the tools.
ΔZ	Coordinate shown by the CNC.
L	Probe length.
R	Probe ball radius.

Calculation of parameter TDATA5.

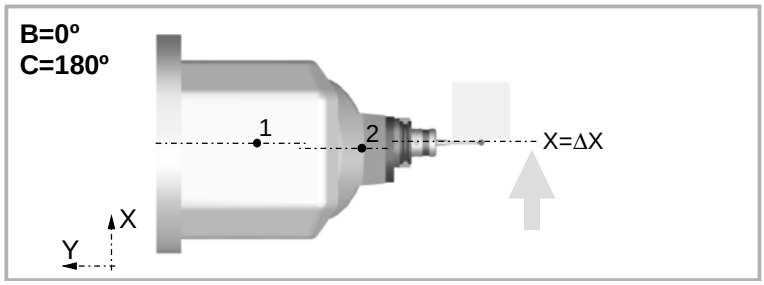
First probing move.

With rotary axes B and C positioned at 0º, make a probing movement on the X side closest to the origin and preset the position X = 0.



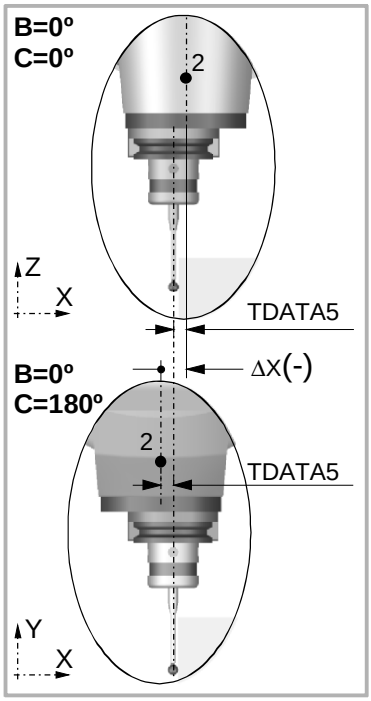
Second probe.

With rotary axis B positioned at 0º and the C axis at 180º, make a probing movement on the same side. Jot down the position value (coordinate) shown at the CNC, appearing in the drawing as ΔX.



Mathematical calculations.

After changing the position with respect to the center of rotation ·2·, the only unknown data after both probing movements is TDATA5.

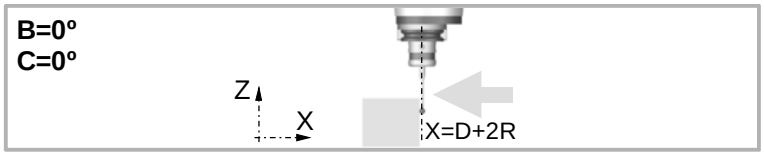


Solution.	
$\|TDATA5\| = 0,5 \times \|\Delta Z\|$	
TDATA5	Dimensions to calculate.
ΔX	Coordinate shown by the CNC.

Calculation of parameter TDATA6.

First probing move.

With the B and C axes positioned at 0º, make a probing movement on the other X side to measure the part whose dimension will be the coordinate shown at the CNC minus the diameter of the probe ball.

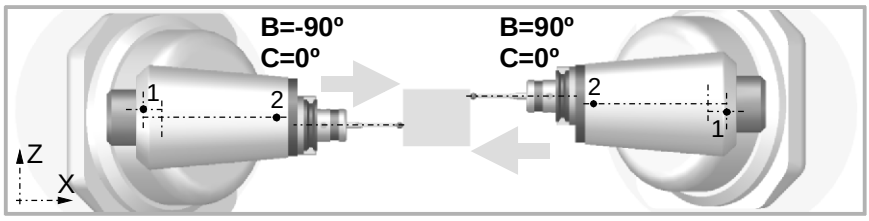


Second probe.

With the C rotary axis positioned at 0º and B at -90º, make a probing movement on the first X side. Jot down the position value (coordinate) shown at the CNC, appearing in the drawing as ΔX'.

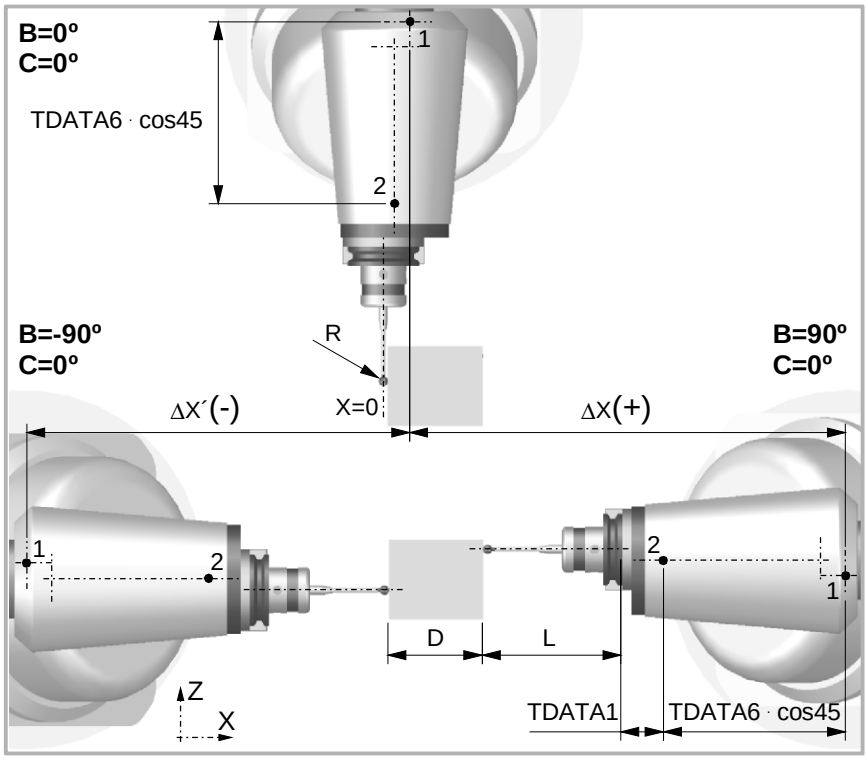
Third probing move.

With the C rotary axis positioned at 0º and B at 90º, make a probing movement on the second X side. Jot down the position value (coordinate) shown at the CNC, appearing in the drawing as ΔX.



Mathematical calculations.

After changing the position with respect to the center of rotation ·1·, the only unknown data after both probing movements is TDATA6.



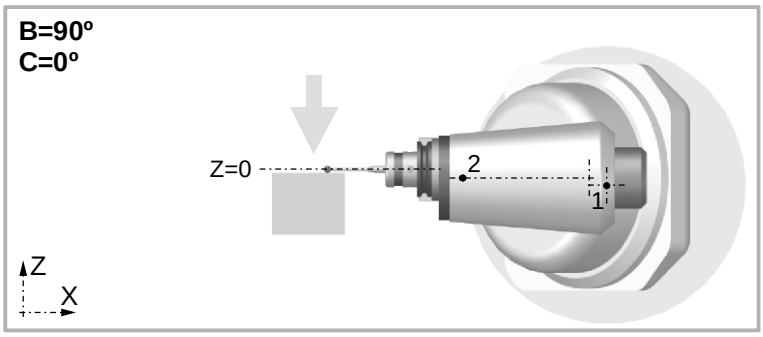
Solution.
$\|TDATA6\| = \frac{\|\Delta X\| + \|\Delta X'\| - D - 2L - 2\|TDATA1\|}{2 \bullet \cos 45}$

TDATA1	Known dimension.
TDATA6	Dimensions to calculate.
D	Length of the part.
ΔX'ΔX	Coordinates shown by the CNC.
L	Probe length.
R	Probe ball radius.

Calculation of parameter TDATA2.

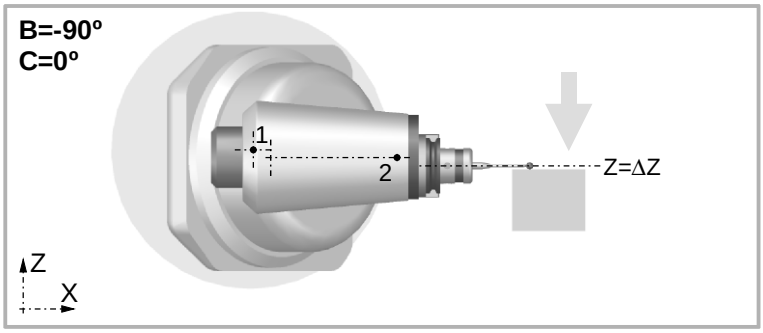
First probing move.

With the C rotary axis positioned at 0º and the B axis at 90º, make a probing movement on the Z side and preset the position Z = 0.



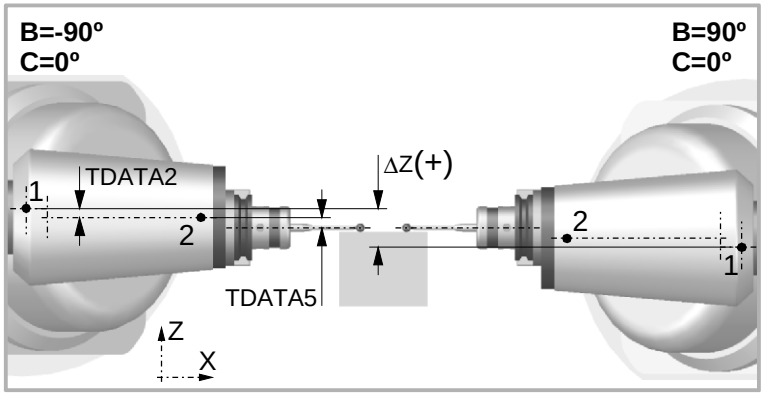
Second probe.

With the C rotary axis positioned at 0º and the B axis at -90º, make a probing movement on the Z side. Jot down the position value (coordinate) shown at the CNC, appearing in the drawing as ΔZ.



Mathematical calculations.

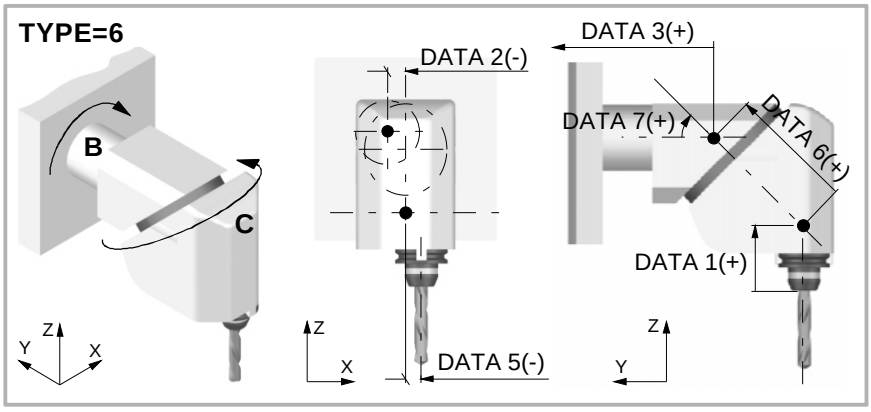
After changing the position with respect to the center of rotation ·1·, the only unknown data after both probing movements is TDATA2.



Solution.	
$\|TDATA2\| = 0,5 \cdot \|\Delta Z\| - \|TDATA5\|$	
TDATA2	Dimensions to calculate.
TDATA5	Known dimension.
ΔZ	Coordinate shown by the CNC.

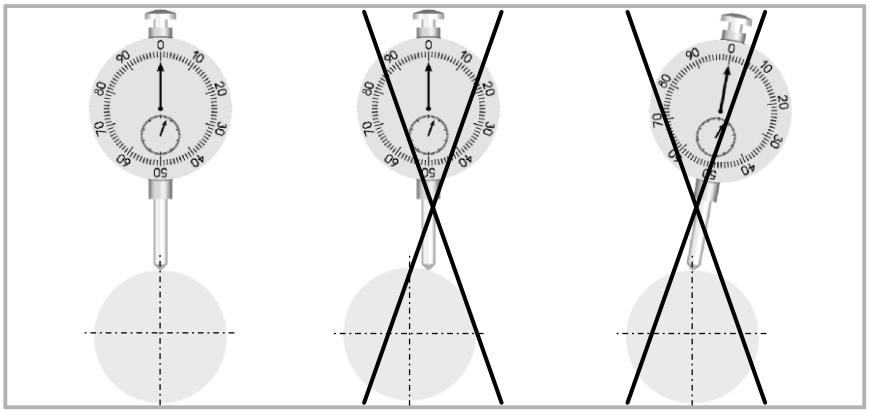
9.10.2 Swivel (angular) spindle. Calculation of the dimensions using a dial indicator.

This example calculates the dimensions of a YZ 45º angular spindle (same as in the previous example), parameter TYPE = 6. The next example uses a dial indicator and a cylinder of known diameter placed in the spindle.



How to use the dial indicator.

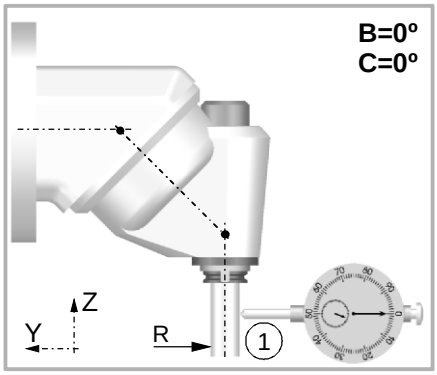
When touching a cylinder with a dial indicator, contact must be made on the outside line of the cylinder. Otherwise, there will be a measuring error when presetting the value of the cylinder value. The dial indicator should be as perpendicular as possible to the tangent of the surface being measured.



A simple method to place the dial indicator on the most external line of the cylinder is to move the dial indicator on the circumference of the cylinder. To do that, preset ·0· on the axis to be moved. When moving the axis, the needle of the dial indicator will rotate in one direction and after passing the most external point of the cylinder, the needle will rotate in the opposite direction. Continue with the movement until the needle of the dial indicator reaches its initial value. The most external point will be in the middle of the distance traveled by the axis.

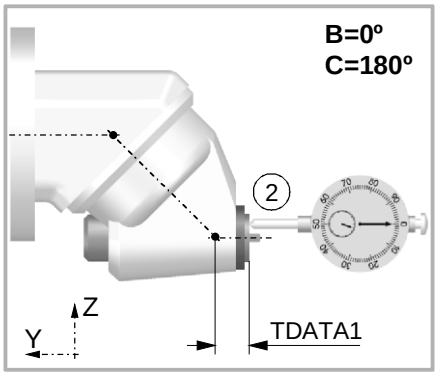
Calculation of parameter TDATA11.

First contact.



1 Position rotary axes B and C at 0º.
2 Place the dial indicator on the table, in the direction of the Y axis.
3 Move the table in the Y axis until the dial indicator and the cylinder touch each other at point 1.
4 Preset Y=R so Y=0 coincides with the shaft of the cylinder placed in the spindle.

Second contact.

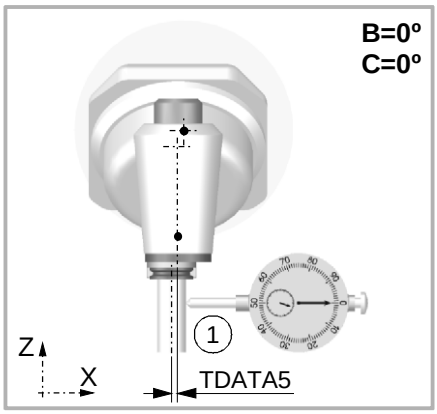


1 Remove the dial indicator, without loosening it and position the rotary C axis at 180º.
2 Touch with the dial indicator at point 2.
3 Jot down the position value (coordinate) shown at the CNC ΔY).

Solution.	
$\|TDATA1\| = \|\Delta Y\|$	
TDATA1	Dimensions to calculate.
ΔY	Coordinate shown by the CNC.

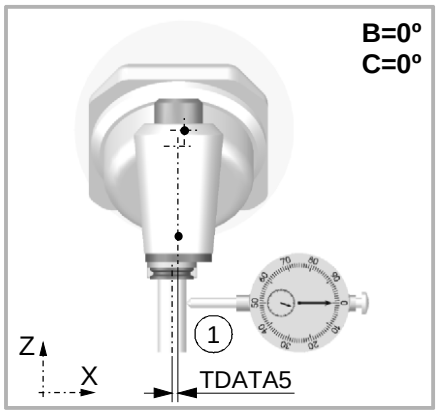
Calculation of parameter TDATA5.

First contact.



1 Position rotary axes B and C at 0º.
2 Place the dial indicator on the table, in the direction of the X axis.
3 Move the table in the X axis until the dial indicator and the cylinder touch each other at point 1.
4 Preset X=0.

Second contact.



1 Remove the dial indicator, without loosening it and position the rotary C axis at 180º.

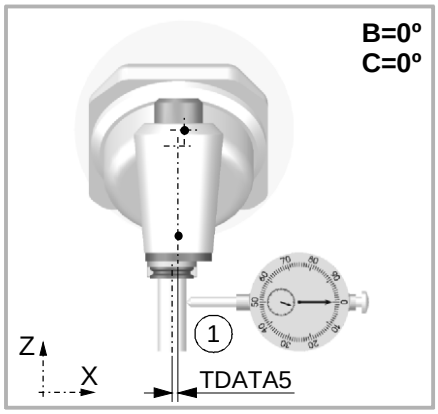
2 Touch with the dial indicator at point 2.

3 Jot down the position value (coordinate) shown at the CNC ΔX).

Solution.	
$\|TDATA5\| = 0,5 \cdot \|\Delta X\|$	
TDATA5	Dimensions to calculate.
ΔX	Coordinate shown by the CNC.

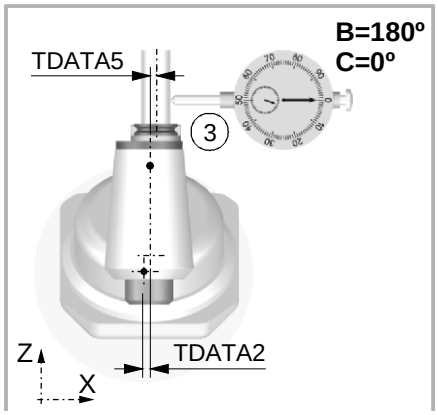
Calculation of parameter TDATA2.

First contact (made when calculating TDATA5).



1 Keep previous preset.

Second contact.



1 Remove the dial indicator, without loosening it and position the rotary C axis at 0º.

2 Position B rotary axis at 0º.

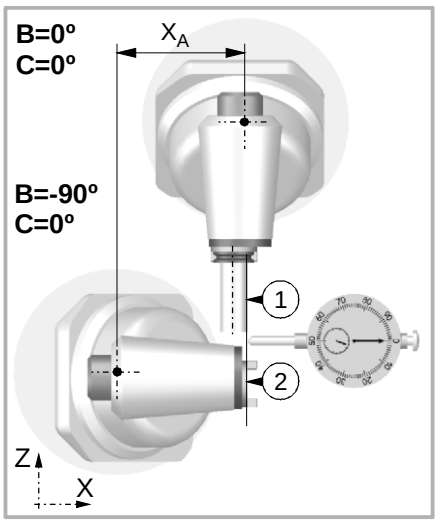
3 Touch with the dial indicator at point 2.

4 Jot down the position value (coordinate) shown at the CNC ΔX).

Solution.	
$\|TDATA2\| = 0,5 \cdot \|\Delta X\| - \|TDATA5\|$	
TDATA1	Dimensions to calculate.
TDATA5	Dimensions to calculate.
ΔX	Coordinate shown by the CNC.

Calculation of parameter TDATA6.

First and second contacts.



1 Position rotary axes B and C at 0º.

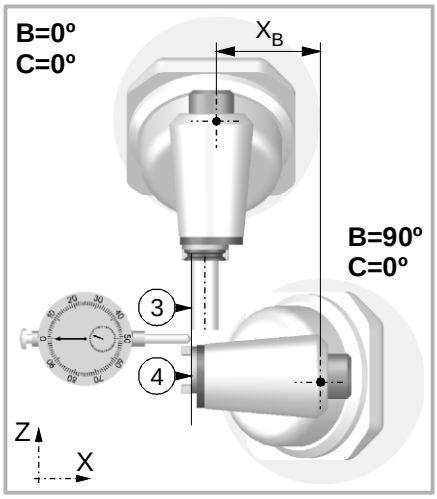
2 Move the table in the X axis until the dial indicator and the cylinder touch each other at point 1.

3 Preset X=R so X=0 coincides with the shaft of the cylinder placed in the spindle.

4 Remove the dial indicator, without loosening it and position the B rotary axis at -90º. Touch with the dial indicator at point 2.

5 Jot down the position value (coordinate) shown at the CNC (Xa).

Third and forth contact.



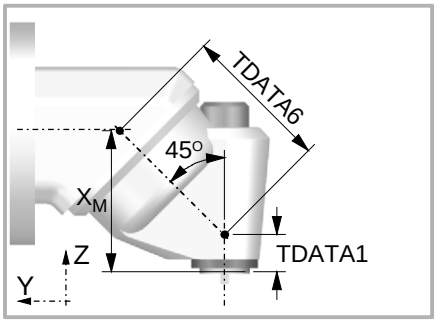
1 Position rotary axes B and C at 0º.

2 Move the table in the X axis until the dial indicator and the cylinder touch each other at point 3.

3 Preset X=R so X=0 coincides with the shaft of the cylinder placed in the spindle.

4 Remove the dial indicator, without loosening it and position the B rotary axis at 90º. Touch with the dial indicator at point 4.

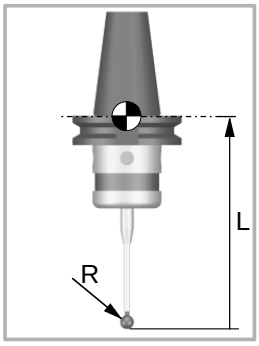
5 Jot down the position value (coordinate) shown at the CNC (Xb).



Solution.	
$\|X_M\| = 0,5 \cdot (\|X_B\| + \|X_A\|)$ $\|TDATA6\| = \sqrt{2} \cdot (\|X_M\| - \|TDATA1\|)$	
TDATA5	Dimensions to calculate.
ΔX	Coordinate shown by the CNC.

9.10.3 Rotary table. Calculation of the dimensions using a probe.

This example calculates the dimensions of an AB rotary table with a moving range for the A axis between -90º and 90º and for the B axis between 0º and -90º, parameter TYPE = 9. The next example uses a probe and a cylinder of known diameter secured to the table.



The machine parameters associated with the probe movement must be set and the probe must be properly calibrated. You need to know the length and radius of the probe to calculate the dimensions of the spindle.

The CNC can have configured two probes. Before any probing move, the CNC must know which is the active probe, or, which is the same, which of the two probes it must attend to. It is selected via part-program or MDI using the instruction #SELECT PROBE.



If a probing move is executed without activating the probe, it will not send any signal to the CNC when it makes contact. This can cause the probe to break because the probing move will not be stopped.

Dimensions to calculate on an AB rotary table (TYPE=9).

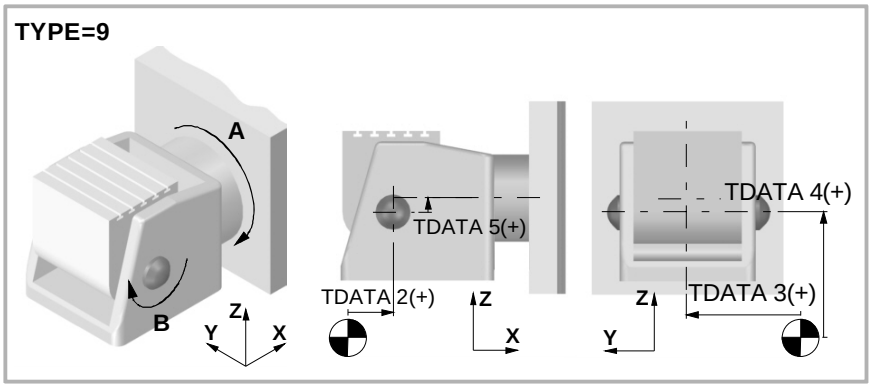
In this type of kinematics the following dimensions must be defined. All the parameters may be set with a positive or negative value. The (+) sign in the illustrations indicates that the direction is assumed as positive.

TDATA2 — It indicates the position of the secondary rotary axis or the intersection with the primary axis along the X axis.

TDATA3 — It indicates the position of the secondary rotary axis or the intersection with the primary axis along the Y axis.

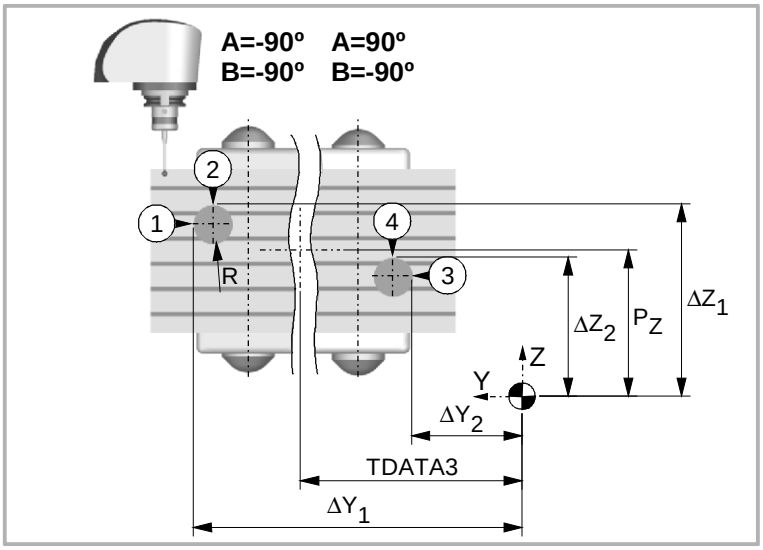
TDATA4 — It indicates the position of the secondary rotary axis or the intersection with the primary axis along the Z axis.

TDATA5 — It indicates the distance between the secondary and the main rotary tables.



Calculation of the coordinates of the A axis.

The drawing shows two different positions that have the same center of the A axis. The coordinate on the Y axis is parameter TDATA3 and the coordinate on the Z axis Pz. Both coordinates are necessary to calculate the rest of the parameters.



First position.

1 Position rotary axes A and B at -90º.

2 Make a probing movement along the Y axis at point 1. Jot down the position value (coordinate) shown at the CNC, appearing in the drawing as ΔY1.

3 Make a probing movement along the Z axis at point 1. Jot down the position value (coordinate) shown at the CNC, appearing in the drawing as ΔZ1.

Second position.

1 Position the A rotary axis at 90º and the B axis at -90º.

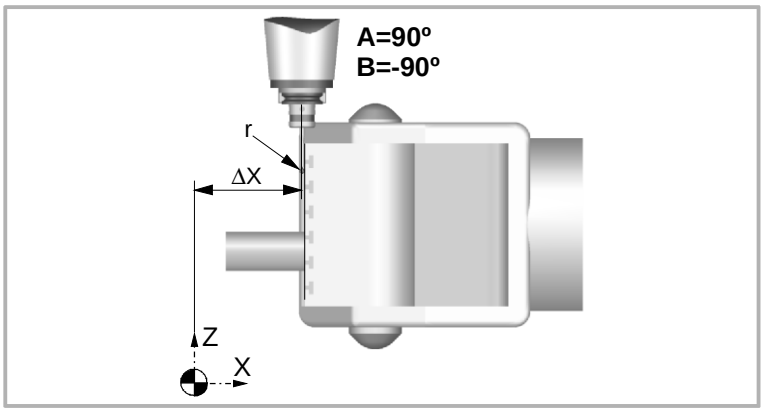
2 Make a probing movement along the Y axis at point 3. Jot down the position value (coordinate) shown at the CNC, appearing in the drawing as ΔY2.

3 Make a probing movement along the Z axis at point 4. Jot down the position value (coordinate) shown at the CNC, appearing in the drawing as ΔZ2.

Mathematical calculations.

Solution.	
$\|TDATA3\| = 0,5 \cdot (\|\Delta Y_1\| + \|\Delta Y_2\|)$ $\|Pz\| = 0,5 \cdot (\|\Delta Z_1\| + \|\Delta Z_2\| - R)$	
TDATA3 Pz	Coordinates of the A axis.
ΔY1, ΔZ1 ΔY2, ΔZ2	Coordinates shown by the CNC.
R	Cylinder radius.

Origin preset on the X axis.



1 Keep previous position, A=90º and B=-90º.

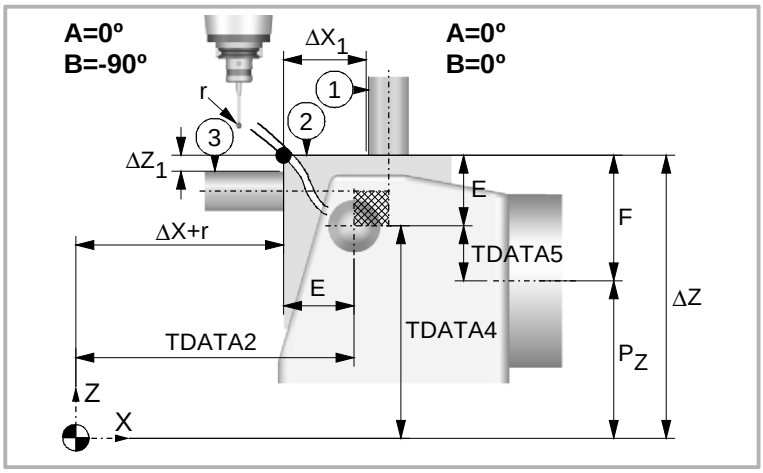
2 Probe on the table, in the direction of the X axis.

3 Jot down the position value (coordinate) shown at the CNC, appearing in the drawing as ΔX.

4 Preset X = -r so X=0 coincides with the surface of the table.

Calculation of the rest of the parameters.

The drawing shows two different positions of the B axis while keeping the position of the A axis.



First position.

1 Position rotary axes A and B at 0º.

2 Make a probing movement along the X axis at point 1. Jot down the position value (coordinate) shown at the CNC, appearing in the drawing as ΔX1.

3 Make a probing movement along the Z axis at point 2. Jot down the position value (coordinate) shown at the CNC, appearing in the drawing as ΔZ.

4 Preset Z=0.

Second position.

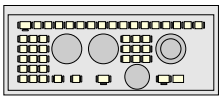
1 Position the A rotary axis at 0º and the B axis at -90º.

2 Make a probing movement along the Z axis at point 3. Jot down the position value (coordinate) shown at the CNC, appearing in the drawing as ΔZ1.

Mathematical calculations.

Solution.	
According to the shaded frame of the drawing. $\|E\| = 0,5 \cdot (\|\Delta X_1\| + \|\Delta Z_1\| + 2R + r)$	
Calculation of parameters. $\|F\| = \|\Delta Z\| - \|Pz\|$ $\|TDATA5\| = \|F\| - \|E\|$ $\|TDATA4\| = \|TDATA4\| - \|Pz\|$ $\|TDATA2\| = \|\Delta X\| + r + \|E\|$	
TDATA2 TDATA4 TDATA5	Parameters to calculate.
Pz	Coordinate of the A axis.
ΔZ, ΔZ1 ΔX, ΔX1	Coordinates shown by the CNC.
R	Cylinder radius.
r	Probe ball radius.
E	Distance from the table surface to the B axis.
F	Distance from the table surface to the A axis.

9.11 Management of several keyboards.

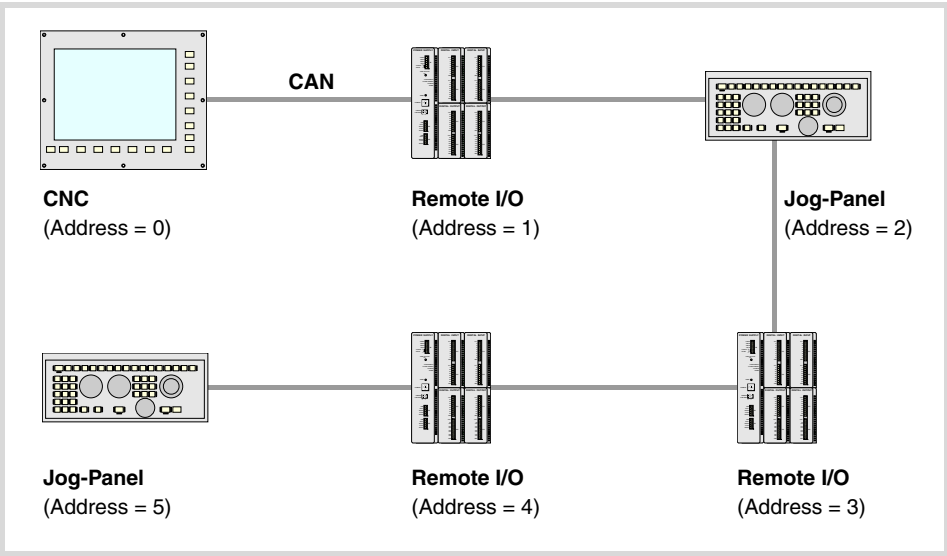


Since the CNC may have several jog panels connected to it, the CNC allows configuring the behavior of each one of them with respect to the channels. A jog panel may be associated with a particular channel (for example, in a TT lathe) or always with the active channel (for example, a machine with two work stations where both jog panels work the same way).

9.11.1 How to configure the feature.

Identify the order of the operator panels in the bus.

The CNC numbers the operator panels following the order (sequence) that they occupy in the CAN bus (Address switch). The first jog panel will be the one with the lowest number and so on.



Address	Element.	Order number.
0	CNC	First operator panel of the system.
1	Remote I/O	
2	Jog-Panel	Second operator panel of the system.
3	Remote I/O	
4	Remote I/O	
5	Jog-Panel	Third operator panel of the system.

Define the number of jog panels and their relationship with the channels.

Machine parameter.	Meaning.
NKEYBD	Number of jog panels.
KEYBD1CH .. KEYBD8CH	List of jog panels of the system.

Parameter NKEYBD sets the number of jog panels of the system. Parameter KEYBDCH shows the table for assigning the jog panels to the channels, where parameter KEYBD1CH corresponds to the first jog panel, KEYBD2CH to the second one and so on.

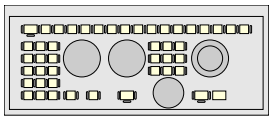
Each jog panel may be assigned to a particular channel or always to the active channel. When a jog panel is assigned to a channel, it is always operative even if the channel is not the active one. When there are several jog panels assigned to the same channel, operations may be carried out from any of them indistinctively.

Configure the jog keys.

Machine parameter.	Meaning.
JOGKEYDEF JOGKEYBD2DEF ... JOGKEYBD8DEF	Tables to configure the jog keys.

These parameters may be used to configure the jog keypad of each panel. Parameter JOGKEYDEF corresponds to the first jog panel, JOGKEYBD2DEF to the second one and so on.

Each parameter contains a table to define the function of each jog key. The CNC always offers 15 parameters; if the jog keypad has fewer keys, the parameters that do not have any keys associated with them will be ignored. The relationship between these parameters and the jog keys is the following.

Hardware module.	Jog keys.
JOG-PANEL 	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

The JOG keypad may consist of the following types of keys. Both types of keys may be defined at the same JOG keypad. To define the behavior of each key, assign to them one of the following values.

Keys.	Meaning.
X+ 7+	Keys to define the axis and the jogging direction. The parameter is set with a value between -1 and +16 (signed). The sign indicates the positive direction (+) or the negative direction (-) and the number corresponds to the logic axis, according to parameter AXISNAME.
X 7	Keys to define the axis to be jogged. The parameter is set with a value between 1 and 16 (unsigned) that corresponds to the logic axis according to parameter AXISNAME.
+ -	Keys to define the direction of the movement. The parameter is set with one of the "+" and "-" values to indicate the moving direction.
	Rapid key. The parameter is set with the "R" value.

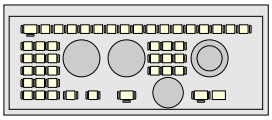
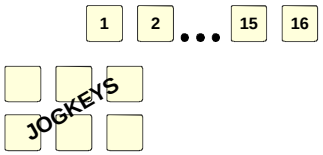
Configure the user keys as jog keys

Machine parameter.	Meaning.
USERKEYDEF USERKEYBD2DEF ... USERKEYBD8DEF	Tables to configure the user keys as jog keys.

These parameters may be used to configure the user keys of each panel as jog keys. Parameter USERKEYDEF corresponds to the first jog panel, USERKEYBD2DEF to the second one and so on.

Each parameter contains a table to define the function of each user key. The CNC always offers 16 parameters; if the jog keypad has fewer keys, the parameters that do not have any

keys associated with them will be ignored. The relationship between these parameters and the user keys is the following.

Hardware module.	User keys.
JOG-PANEL 	

The meaning of this parameter is similar to that of machine parameter JOGKEYBDkbDEF. To define the behavior of each key, assign to them one of the following values:

- They are defined with a value between 1 and 16 (signed) for the keys that define the axis and the direction. The sign indicates the positive direction (+) or the negative direction (-) and the number corresponds to the logic axis, according to parameter AXISNAME.
- They are defined with a value between 1 and 16 (unsigned) for the keys that only define the axis.
- They are defined with the "+" and "-" values for the keys that only define the moving direction.
- They are defined with the "R" value for the rapid key.

The user keys defined this way behave like the jog keys whether they've been defined signed or unsigned and also respect the setting of machine parameter JOGTYPE.

Check the status of the keys.

PLC marks and registers.	Meaning.
KEYBD1 / KEYBD2	These registers are a copy of the map of the keys pressed on the last keyboard used.
KEYBD1_1 / KEYBD2_1 .. KEYBD1_8 / KEYBD2_8	These registers indicate (bit=1) which key has been pressed on each operator panel.

Registers KEYBD1 y KEYBD2 are a copy of the map of the keys pressed on the last keyboard used. These registers indicate which key has been pressed (bit=1). If there is only one keyboard, these registers coincide with KEYBD1_1 and KEYBD2_1. When there are several keyboards, the contents of these registers are not always the same as KEYBD1_1 and KEYBD2_1; therefore, they may be used indistinctively.

Registers KEYBD1_n and KEYBD2_n indicate (bit=1) which key has been pressed on each operator panel. Registers KEYBD1_1 and KEYBD2_1 correspond to the first jog panel, KEYBD1_2 and KEYBD2_2 to the second one and so on.

Disable the keys.

PLC marks and registers.	Meaning.
KEYDIS1 to KEYDIS3	These registers inhibit (bit=1) the keys and the switches on all operator panels at the same time.
KEYDIS1_1 to KEYDIS3_1 .. KEYDIS1_8 to KEYDIS3_8	These registers inhibit (bit=1) the keys and the switches on each operator panel.

Registers KEYDIS1, KEYDIS2 and KEYDIS3 inhibit (bit=1) the keys and the switches on all operator panels at the same time.

Registers KEYDIS1_n, KEYDIS2_n and KEYDIS3_n inhibit (bit=1) the keys and the switches on each operator panel. Registers KEYDIS1_1 to KEYDIS3_1 correspond to the first jog panel, KEYDIS1_2 to KEYDIS3_2 to the second one and so on.

When selecting one of the inhibited positions of the feedrate override, the CNC will take the value of the lowest position allowed. If all of them are inhibited, it will take the value of 0%.

Manage the LED's (lamps) of the keys.

PLC marks and registers.	Meaning.
KEYLED1 / KEYLED2	These registers control the LED's (lamps) of the keys of all the operator panels at the same time.
KEYLED1_1 / KEYLED2_1 .. KEYLED1_8 / KEYLED2_8	These registers control the LED's (lamps) of the keys of each operator panel.

Registers KEYLED1 and KEYLED2 control the LED's (lamps) of the keys of all the operator panels at the same time.

Registers KEYLED1_n and KEYLED2_n control the LED's (lamps) of the keys of each operator panel. Registers KEYLED1_1 and KEYLED2_1 correspond to the first jog panel, KEYLED1_2 and KEYLED2_2 to the second one and so on.

Change the default behavior of the keyboards with respect to the channels.

PLC marks and registers.	Meaning.
KEYBD1CH to KEYBD8CH	These registers may be used to change the default behavior of the keyboards with respect to the channels, set by machine parameters.

These registers may be used to change the default behavior of the keyboards with respect to the channels, set by machine parameters. These registers can associate a jog panel with a particular channel, always with the active channel or restore the configuration defined in the machine parameters.

Value.	Meaning.
0	Configuration defined in the machine parameters.
1	Jog panel assigned to channel 1.
2	Jog panel assigned to channel 2.
3	Jog panel assigned to channel 3.
4	Jog panel assigned to channel 4.
FF	Jog panel assigned to the active channel.

Disable the jog panels.

PLC marks and registers.	Meaning.
PANELOFF PANELOFF1 to PANELOFF8	If the PLC sets this one of these marks high (=1) the CNC disables the corresponding jog panel.

If the PLC activates one of these marks the CNC disables the corresponding jog panel. Each mark may be used to disable the jog panel integrated into the CAN bus; the rest of the elements of the bus are not affected. When the keyboard and the operator panel make up a single element, this mark only disables the jog panel.

There is one mark for each keyboard or jog panel. The PANELOFF1 mark disables the first one of the bus, the PANELOFF2 mark the second one and so on. Both PANELOFF and PANELOFF1 are valid for the first jog panel.

Address	Element	PLC mark
0	CNC	
1	Remote (I/O) group	
2	Jog panel	PANELOFF1

Address	Element	PLC mark
3	Remote (I/O) group	
4	Remote (I/O) group	
5	Jog panel	PANELOFF2

9.11.2 Operation of the jog panels.

When enabling a keyboard (PANELOFF mark) and every time there is a keyboard change (press a key, change a switch, etc.), the channel assumes the status of the keys and the position of the switches of the new keyboard with the active values. If necessary the PLC program must include the proper restrictions to avoid sudden feedrate changes due to different positions of the feedrate override switch on the different keyboards. It is also recommended to manage the user keys using up and down flanks to avoid the possible activation of devices when changing keyboards.

Every time the active keyboard is changed, the new keyboard updates at the PLC the information on all the marks and registers associated with the keyboard itself (KEYBD1_n, KEYBD2_n, KEYDIS1_n a KEYDIS3_n, KEYLED1_n, KEYLED2_n) and the generic ones associated with all the keyboards (KEYBD1, KEYBD2, KEYDIS1 to KEYDIS3, KEYLED1, KEYLED2).

Behavior of the keys and the switches.

The keys from the jog panel.

The keys of each jog panel are only effective for each channel with which the keyboard is associated; whether it is the active channel or not. The following keys have a different behavior.

- The [RESET] key affects the channel associated with the keyboard where the key has been pressed and the channels that belong to its group.
- The keys to select the work mode (automatic, jog, etc.) only affect the active channel.

Feedrate override selector.

If the keyboard is associated with a particular channel, an override change only affects that channel, even if it belongs to a group. If the keyboard is associated with the active channel, an override change affects all the channels of the system at the same time, whether they belong to a group or not.

9.12 Remote OpenPCS (8065 CNC only).

The CNC can work with its own PLC, with a remote OpenPCS or with both. The OpenPCS must be installed at an external PC that may be connected to the CNC through a network or crossed cable. The CNC must be running and with the IEC option active in order to communicate with the OpenPCS and be able to execute the remote PLC program (IEC-61131).

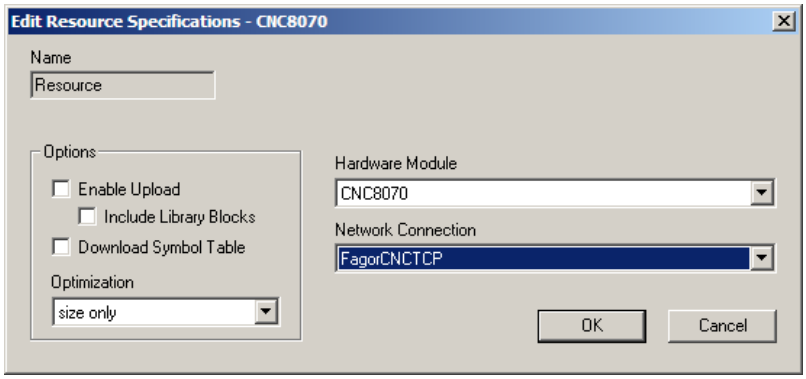
Configure the CNC to use the remote PLC.

Machine parameter PLCTYPE selects the type of PLC to be used; the Fagor one, the standard IEC-61131 or both. The following table shows who is affected by the startup (RUN) and stop (STOP) of the PLC depending on this parameter as well as the status of the mark PLCREADY.

PLCTYPE	RUN		STOP		PLCREADY	
	IEC	Fagor	IEC	Fagor	IEC	Fagor
IEC	Yes	No	Yes	No	ON	- - -
IEC+Fagor	Yes	Yes	Yes	Yes	ON	ON
Fagor	No	Yes	No	Yes	OFF	ON

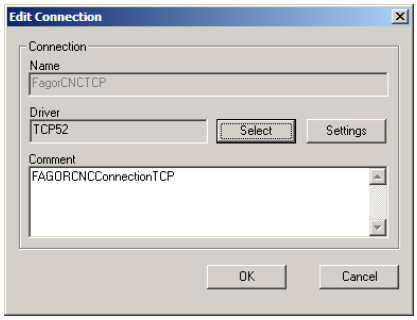
Create the hardware module of the remote OpenPCS.

The information needed by the OpenPCS to generate the proper PLC for the CNC is in the folder c:\cnc8070\fagor\plc\openpcs of the CNC itself. The folder contains a script that must be copied and executed at the work PC. This scripts creates the hardware module for that the PLC program will be compiled for and it also creates a network connection for downloading the program at the CNC, monitor it and debug it.

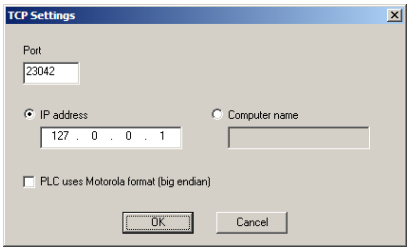


Configure the IP direction of the connection at the hardware module.

The script creates a network connection with a default IP (the loopback IP 127.0.0.1). To connect to the CNC, configure the connection with its IP.



To change the connection, select the FagorCNCTCP on the PLC menu >and press Edit.



In the previous dialog box, press the Settings button to show the connection data. Set the proper IP (that of the CNC).

9.13 Assigning a help text to the graphic softkeys and to the CNC status icon.

The help texts are defined in the "SoftkeyHelper.txt" file and there could be a file per language, they are saved in the "..\MTB\data\Lang" folder. The file is in text format and those texts may be displayed and edited (and thus translated into other languages) using any text editor.

Help text for the graphic softkeys.

The file is divided into sections, one per operating mode, each one of them contains the list of softkeys and the help text associated with it. For example, for the automatic mode.

```
[AUTOMATIC MODE]
AUTO1=Select program
AUTO2=Prg in simulation
AUTO3=Inspection
AUTO4=Quit inspection
AUTO5=Reposition
AUTO6=Reposition block start
AUTO7=Execution mode
AUTO8=Cancel block
AUTO9=Graphics
```

Help text for the CNC status icon.

In the "SoftkeyHelper.txt" file it is also possible to assign a help text to the icons that show the CNC status, at the top of the screen.



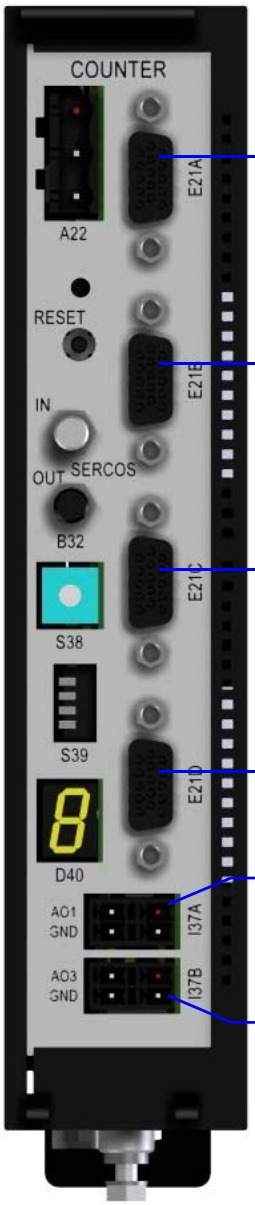
These help texts are defined in the [HMI] section.

```
[HMI]
HMI_NOREADY=
HMI_READY=
HMI_INEXECUTION=
HMI_INTERRUPTED=
HMI_INERROR=
```

9.14 Remote module RCS-S.

The "RCS-S" counter may be used to get extra feedback (counter) inputs and analog outputs (remote I/O). These modules are connected to the central unit via Sercos bus and they are just another bus within the bus. Each module offers the following elements.

Type of input/output	Amount.
Feedback inputs.	4
Analog outputs.	4



Connector ·E21A·.
Feedback input.

Connector ·E21B·.
Feedback input.

Connector ·E21C·.
Feedback input.

Connector ·E21D·.
Feedback input.

Connector ·I37A·.
2 general purpose analog outputs.

Connector ·I37B·.
2 general purpose analog outputs.

9.14.1 Configure the module like as a node of the Sercos bus.

The Sercos counter is, like the drives, a node tye of the Sercos ring. In terms of parameter setting, Sercos counters have dual identification, physical and logic, as described next.

Physical identification of the module inside the bus.

Each counter must be identified by a number (Sercos ID) using the thumbwheel located on its front plate. That identifier must be unique in the ring, the number is sequential from 1 to n and without skipping any and "n" being the total number of Sercos nodes (including drives and counters).

Logic identification of the module inside the bus.

Parameter.	Meaning.
NSERCOUNT	Number of Sercos counter nodes in the bus.
SERCOUNTID	Thumbwheel identifiers associated with Sercos counters.

The SERCOUNTID table of the general machine parameters allows linking the logic number of the Sercos counters (1..8) to the physical number set by the thumbwheel of each node (enter in each position of the Sercos ID table associated with that logic position of the counter).

Although the Sercos ring may have up to 32 nodes, the Sercos counters always occupy positions from 1 to 15.

9.14.2 Configure the analog outputs.

The analog output is configured via machine parameters. Each axis has the following machine parameters to configure it.

Parameter.	Meaning.
ANAOUTTYPE	Type of analog output associated with the axis.
ANAOUTID	Number of the analog output associated with the axis.

Configure the analog outputs.

The velocity command for an analog axis may be taken from an analog output of the remote modules or of a Sercos drive. or of a Sercos counter. It is selected with parameters ANAOUTID and ANAOUTID.

Parameter.	Value.
ANAOUTTYPE	RCS-S module RCS-S (Sercos counter).
ANAOUTID	1 - 4.

Managing the analog command from the Sercos counter.

In this case, parameter ANAOUTID indicates the number of the analog output used for the command. The Sercos counters are numbered according to their logic order (parameter SERCOUNTID). The analog outputs of the first module will be 1 through 4, those of the second module 5 through 8 and so on.

9.14.3 Configuration of the feedback inputs.

The feedback inputs are configured via machine parameters. Each axis has the following machine parameters to configure it.

Parameter.	Meaning.
COUNTERTYPE	Type of feedback input for the axis.
COUNTERID	Number of the feedback input for the axis.

Configure the feedback input.

The axis feedback input may be located in the remote modules, at the free feedback of a Sercos drive, in a Sercos counter or at the local feedback inputs of the central unit. It is selected with parameters COUNTERTYPE and COUNTERID.

Parameter.	Value.
COUNTERTYPE	RCS-S module RCS-S (Sercos counter).
COUNTERID	1 - 32.

Feedback input in Sercos counters.

Parameter COUNTERID indicates the number of the feedback input being used. The Sercos counters are numbered according to their logic order (parameter SERCOUNTID). The feedback inputs of the first module will be 1 through 4, those of the second module 5 through 8 and so on.

9.14.4 Configure the feedback input for a handwheel.

The handwheel is configured via machine parameters of the jog mode. Each handhwel has the following machine parameters to configure it.

Parameter.	Meaning.
COUNTERTYPE	Type of handwheel feedback input.
COUNTERID	Feedback input associated with the handwheel.

Configure the feedback input.

This parameter indicates the location of the handwheel feedback input; at the remote modules, at the keyboard or at the local feedback inputs of the central unit or in a Sercos counter. It is selected with parameters COUNTERTYPE and COUNTERID.

Parameter.	Value.
COUNTERTYPE	RCS-S module RCS-S (Sercos counter).
COUNTERID	1 - 32.

Feedback input in Sercos counters.

Parameter COUNTERID indicates the number of the feedback input being used. The Sercos counters are numbered according to their logic order (parameter SERCOUNTID). The feedback inputs of the first module will be 1 through 4, those of the second module 5 through 8 and so on.

9.14.5 Parameter setting example.

The system has 2 Sercos counters (nodes 3, 5) and 3 Sercos AXD drives (nodes 1, 2, 4). Plus, the system has the following axes.

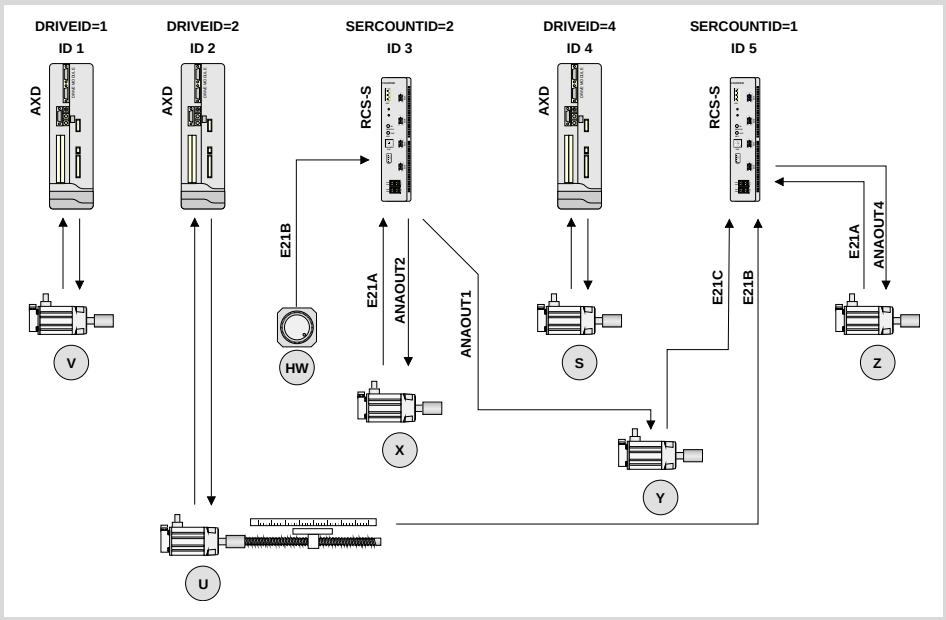
- 3 analog axes X Y Z.
- 1 semi-Sercos axis U.
- 1 Sercos axis V.
- 1 handwheel.
- 1 Sercos spindle S.

The connection is as follows:

NODE 1:	V Sercos axis.	
NODE 2:	U Sercos axis.	
NODE 3:	X analog axis.	E21A / ANAOUT2
	Handwheel 1.	E21B
	Y analog axis.	ANAOUT1
NODE 4:	Sercos spindle S	
NODE 5:	U axis (second feedback)	E21B
	Y analog axis.	E21C
	Z analog axis.	E21A / ANAOUT4

The parameter could be set as follows.

NSERCOUNT	2	Number of Sercos axes in the system.
SERCOUNTID1	5	(No common, but valid)
SERCOUNTID2	3	



Parameters SERCOUNTID1 through SERCOUNTID8 set the order (sequence) of the feedback inputs. Since SERCOUNTID1=5 and SERCOUNTID2=3, the first feedback input will be the first one of node 5.

Axis.	Parameter.	Value.
X axis.	DRIVETYPE COUNTERTYPE COUNTERID ANAOUTTYPE ANAOUTID	Analog. Sercos counter. 5. Sercos counter. 6.
Y axis.	DRIVETYPE COUNTERTYPE COUNTERID ANAOUTTYPE ANAOUTID	Analog. Sercos counter. 3. Sercos counter. 5.
Z axis.	DRIVETYPE COUNTERTYPE COUNTERID ANAOUTTYPE ANAOUTID	Analog. Sercos counter. 1. Sercos counter. 4.
U axis.	DRIVETYPE DRIVEID OPMODEP FBACKSRC COUNTERTYPE COUNTERID	Sercos. 2. Speed. External. Sercos counter. 2.
V axis.	DRIVETYPE DRIVEID	Sercos. 1.
Handwheel	MANPG COUNTERTYPE COUNTERID	1. Sercos counter. 6.
Spindle S	DRIVETYPE DRIVEID	Sercos. 4.

CNC VARIABLES.

10

10.1 Understanding how variables work.

Accessing the variables.

The internal CNC variables may be accessed from the part program, MDI/MDA, PLC and from any application or external interface (for example FGUIM). Each variable must indicate whether it can only be read or read and written.

Access the variables from a part-program. Accessing during execution or during block preparation.

The CNC reads several blocks ahead of the one being executed in order to calculate in advance the path to follow. This prior reading is known as "block preparation".

The CNC checks certain variables during block preparation and others during execution. The variables that use the execution value interrupt block preparation temporarily and the CNC resumes it when it is done reading/writing the variable. Accessing the variables from the PLC or from an external interface never interrupts block preparation.

Be careful with the variables that interrupt block preparation because when they are inserted between machining blocks with compensation may cause undesired profiles. Interrupting block preparation may result in compensated paths different from the one programmed, undesired joints when working with very short moves, etc.

In any case, it is possible to use the #FLUSH instruction to force the evaluation of a variable when it is being executed. This instruction interrupts block preparation in advance, executes the last prepared blocs, synchronizes the preparation and execution of blocks and then goes on with the execution of the program and block preparation.

Accessing the variables. from the PLC. Synchronous or asynchronous access.

PLC access to the variable, both for reading and writing, may be either synchronous or asynchronous. A synchronous access is resolved immediately whereas an asynchronous access takes several PLC cycles to resolve.

The tool variables will be read asynchronously when the tool is neither the active one nor in the magazine. The tool variables will be written asynchronously whether the tool is the active one or not.

Example of how to access asynchronous variables.
Reading of the radius value of offset ·1· of tool ·9· when it is not in the tool magazine.

```
<condition> AND NOT M11 = CNCRD (TM.TORT.[9][1], R11, M11)
```

The PLC activates mark M11 when the operation begins and keeps it active until the operation ends.

```
DFD M11 AND CPS R11 EQ 3 = ···
```

It waits for the consultation to end before evaluating the data.

Example of how to access synchronous variables.

Reading the actual (real) feedrate.

```
<condition> = CNCRD (G.FREAL, R12, M12)
```

The PLC activates mark M12 when the operation begins and keeps it active until the operation ends.

```
CPS R12 GT 2000 = ···
```

There is no need to wait for consulting the data because the synchronous variables are resolved immediately.

It resets the clock enabled by the PLC with the value contained in register R13.

```
<condition> = CNCWR (R13, PLC.TIMER, M13)
```

Accessing the variables. from the PLC. Accessing numeric variables

When the PLC accesses numeric variables that can have decimals, the values will be given in PLC units.

10.1.1 Accessing numeric variables from the PLC.

When the PLC accesses numeric variables that can have decimals, the values will be given as follows.

- The coordinates will be given in ten-thousandths if they are in mm or hundred-thousandths if they are in inches.

Units.	Reading from the PLC.
1 millimeter.	10000.
1 inch.	100000.
1 degree.	10000.

- The feedrate of the axes is given in tenth-thousandths if mm or in hundred-thousandths if inches.

Units.	Reading from the PLC.
1 millimeter/minute.	10000.
1 inch/minute.	100000.
1 degree/minute.	10000.

- The spindle speed will be given in ten-thousandths.

Units.	Reading from the PLC.
G97. 1 rpm.	10000.
G96. 1 meter/minute.	10000.
G96. 1 foot/minute.	10000.
G192. 1 rpm.	10000.
M19. 1 degree/minute.	10000.

- The percentages will be given with the real value, in tenths or in hundredths depending on the variable. If not indicated otherwise, the PLC will read the actual value. If not so, it will indicate if the variable will be read in tenths (x10) or in hundredths (x100).

Units.	Reading from the PLC.
1 %.	1.
1 % (x10).	10.
1 % (x100).	100.

- Time will be given in thousandths.

Units.	Reading from the PLC.
1 second.	1000.

- Voltage will given as follows. The variables associated with the machine parameter table return the actual value (in millivolts). For the rest of the variables (in volts), the reading will appear in ten-thousandths.

Units.	Reading from the PLC.
1 volt.	10000.

10.2 Variables in a single-channel system.

The generic mnemonic associated with the variables is written as follows.

(V.){prefix}.{variable}
(V.){prefijo}.{variable}.{eje/cabezal}

The –V.– indicator.

Programming the -V.- indicator depends on where the variable is used. To access the variables from the part-program or in MDI/MDA mode, the mnemonic begins with the indicator -V.- Accessing the variables from the PLC or from an external interface requires NOT using the -V.- indicator.

All mnemonics in this manual show this indicator as (V.), indicating that it must only be programmed when necessary.

Mnemoni.	Part-program. MDI/MDA mode.	PLC. External interface.
(V.)MPG.NAXIS	V.MPG.NAXIS	MPG.NAXIS

Prefixes of the variables.

The prefix must always be programmed. Prefixes make it possible to easily identify the group the variable belongs to.

Prefix.	Meaning.
A	Axis and/or spindle variables.
C	Canned cycle or subroutine calling parameters.
E	Interface related variables.
G	General variables.
MPA	Variables related to axis and/or spindle machine parameters.
MPG	Variables related to general machine parameters.
MPK	Variables related to kinematic machine parameters.
MPM	Variables related to machine parameters for M functions.
MPMAN	Variables related to machine parameters for JOG mode.
MTB	Variables related to OEM machine parameters.
P	User local variables.
PLC	PLC related variables.
S	User global variables.
SP	Spindle related variables.
TM	Variables related to tools or tool magazines.

Axis and spindle variables.

Axis and spindle variables are identified with the prefix –A.–. When these variables refer to a spindle, they may also be accessed with the prefix –SP.–.

(V.)A.{variable}.{axis/spindle}
(V.)SP.{variable}.{spindle}
(V.)SP.{variable}

The variables of the axis machine parameters (–MPA.– prefix) can also be accessed using the –SP– prefix when referring to a spindle.

(V.)MPA.{variable}.{axis/spindle}
(V.)SP.{variable}.{spindle}
(V.)SP.{variable}

Identifying the axes and the spindles in the variables.

In these variables one must indicate which axis or spindle they refer to. The axis may be referred to by its name or logic number; the spindle may be referred to by its name, logic number or index in the spindle system.

In these variables one must indicate which axis or spindle they refer to. In variables with the prefix –A.– and –MPA.–, the axes and the spindles are identified with their name or logic number. In variables with the prefix –SP.–, the spindles are identified with their name or spindle index. If no spindle is selected in the variables with -SP.- prefix, the variable refers to the master spindle.

Mnemoni.	Meaning when the variable is executed by the part-program, the MDI/MDA mode or the PLC.
V.MPA.variable.Z V.A.variable.Z	Z axis.
V.MPA.variable.S V.A.variable.S V.SP.variable.S	Spindle S.
V.MPA.variable.4 V.A.variable.4	Axis or spindle with logic number ·4·.
V.SP.variable.2	Spindle with index ·2· in the system.
V.SP.variable	Master spindle.

Mnemoni.	Meaning when the variable is executed by an external interface.
V.MPA.variable.Z V.A.variable.Z	Z axis.
V.MPA.variable.S V.A.variable.S V.SP.variable.S	Spindle S.
V.MPA.variable.4 V.A.variable.4	Axis with logic number ·4·.
V.SP.variable.2	Spindle with index ·2· in the system.
V.SP.variable	Master spindle.

The logic number of the axes is determined by the order in which they have been defined in the machine parameter table (AXISNAME). The first axis of the table will be logic axis -1- and so on.

The logic number of the spindles is determined by the order in which they have been defined in the machine parameter table (NAXIS + SPDLNAME). The logic numbering of the spindles continues from the last logic axis; hence, in a 5-axis system, the first spindle of the table will be logic spindle ·6· and so on.

The index of a spindle in the system is determined by the order in which they have been defined in the machine parameter table (SPDLNAME). The index of the first spindle of the table will be ·1· and so on.

AXISNAME	SPDLNAME	Logic order.	Index of the spindle in the system.
AXISNAME 1		Logic number 1.	
AXISNAME 2		Logic number 2.	
AXISNAME 3		Logic number 3.	
AXISNAME 4		Logic number 4.	
AXISNAME 5		Logic number 5.	
	SPDLNAME 1	Logic number 6.	Index 1.
	SPDLNAME 2	Logic number 7.	Index 2.

Variables of the master spindle.

In a multi-spindle system, the master spindle is the main spindle, the one receiving the commands when no specific spindle is mentioned. In a single-spindle channel, that will be its master spindle.

The variables of the master spindle are identified with the prefix –SP.– but without indicating the spindle. These variables may be used to access the data of the master spindle without knowing its name or logic number. These variables are meant for displaying data and programming cycles.

10.3 Variables in a multi-channel system.

The generic mnemonic associated with the variables is written as follows.

(V.)[channel].{prefix}.{variable}
(V.)[canal].{prefix}.{variable}.{axis/spindle}

The –V.– indicator.

Programming the -V.- indicator depends on where the variable is used. To access the variables from the part-program or in MDI/MDA mode, the mnemonic begins with the indicator -V.- Accessing the variables from the PLC or from an external interface requires NOT using the -V.- indicator.

All mnemonics in this manual show this indicator as (V.), indicating that it must only be programmed when necessary.

Mnemoni.	Part-program. MDI/MDA mode.	PLC. External interface.
(V.)[2].MPG.NAXIS	V.[2].MPG.NAXIS	[2].MPG.NAXIS

Programming the channel.

Programming the channel makes it possible to access the variables of the channel itself or another channel from a channel. The first channel is identified with the number 1, "0" is not a valid number.

Programming the channel number is optional; if not programmed, it works as follows depending on who executes the variable. The following table does not apply to the axis and spindle variables.

Where they are executed.	Meaning when no channel has been programmed.
Part-program. MDI/MDA mode.	Channel that is executing the variable.
PLC	First channel or main channel.
External interface.	Active channel.

Prefixes of the variables.

The prefix must always be programmed. Prefixes make it possible to easily identify the group the variable belongs to.

Prefix.	Meaning.
A	Axis and/or spindle variables.
C	Canned cycle or subroutine calling parameters.
E	Interface related variables.
G	General variables.
MPA	Variables related to axis and/or spindle machine parameters.
MPG	Variables related to general machine parameters.
MPK	Variables related to kinematic machine parameters.
MPM	Variables related to machine parameters for M functions.
MPMAN	Variables related to machine parameters for JOG mode.
MTB	Variables related to OEM machine parameters.
P	User local variables.
PLC	PLC related variables.

Prefix.	Meaning.
S	User global variables.
SP	Spindle related variables.
TM	Variables related to tools or tool magazines.

Axis and spindle variables.

Axis and spindle variables are identified with the prefix –A.–. When these variables refer to a spindle, they may also be accessed with the prefix –SP.–.

(V.)[channel].A.{variable}.{axis/spindle}
(V.)[channel].SP.{variable}.{spindle}
(V.)[channel].SP.{variable}

The variables of the axis machine parameters (–MPA.– prefix) can also be accessed using the –SP– prefix when referring to a spindle.

(V.)[channel].MPA.{variable}.{axis/spindle}
(V.)[channel].SP.{variable}.{spindle}
(V.)[channel].SP.{variable}

Identifying the axes and the spindles in the variables.

In these variables one must indicate which axis or spindle they refer to. In variables with the prefix –A.– and –MPA.–, the axes and the spindles are identified with their name, logic number or index in the channel. In variables with the prefix –SP.–, the spindles are identified with their name, index in the channel or spindle index. If no spindle is selected in the variables with -SP.- prefix, the variable refers to the master spindle.

Mnemoni.	Meaning when the variable is executed by the part-program, the MDI/MDA mode or the PLC.
V.MPA.variable.Z V.A.variable.Z	Z axis.
V.MPA.variable.S V.A.variable.S V.SP.variable.S	Spindle S.
V.MPA.variable.4 V.A.variable.4	Axis or spindle with logic number ·4·.
V.[2].MPA.variable.4 V.[2].A.variable.4	Axis with index ·4· in the channel ·2·.
V.SP.variable.2	Spindle with index ·2· in the system.
V.[2].SP.variable.1	Spindle with index ·1· in the channel ·2·.
V.SP.variable	Master spindle of the channel. If the variable is executed by the PLC, spindle master of the first channel.
V.[2].SP.variable	Master spindle of channel ·2·.

Mnemoni.	Meaning when the variable is executed by an external interface.
V.MPA.variable.Z V.A.variable.Z	Z axis.
V.MPA.variable.S V.A.variable.S V.SP.variable.S	Spindle S.
V.MPA.variable.4 V.A.variable.4	Axis with logic number ·4·.
V.[2].MPA.variable.4 V.[2].A.variable.4	Axis with index ·4· in the channel ·2·.
V.SP.variable.2	Spindle with index ·2· in the active channel.
V.[2].SP.variable.1	Spindle with index ·1· in the channel ·2·.
V.SP.variable	Master spindle of the active channel.
V.[2].SP.variable	Master spindle of channel ·2·.

When referring to the axis or spindle by its name, programming the channel they are in is not a determining factor; thus, programming them in this case is irrelevant. When programming the channel, if the axis or spindle is not in it, its programming is ignored.

The logic number of the axes is determined by the order in which they have been defined in the machine parameter table (AXISNAME). The first axis of the table will be logic axis -1- and so on.

The logic number of the spindles is determined by the order in which they have been defined in the machine parameter table (NAXIS + SPDLNAME). The logic numbering of the spindles continues from the last logic axis; hence, in a 5-axis system, the first spindle of the table will be logic spindle ·6· and so on.

The index of a spindle in the system is determined by the order in which they have been defined in the machine parameter table (SPDLNAME). The index of the first spindle of the table will be ·1· and so on.

AXISNAME	SPDLNAME	Logic order.	Index of the spindle in the system.
AXISNAME 1		Logic number 1.	
AXISNAME 2		Logic number 2.	
AXISNAME 3		Logic number 3.	
AXISNAME 4		Logic number 4.	
AXISNAME 5		Logic number 5.	
	SPDLNAME 1	Logic number 6.	Index 1.
	SPDLNAME 2	Logic number 7.	Index 2.

The index of an axis in the channel is determined by the order in which they have been defined in the machine parameter table (CHAXISNAME). The index of the first axis of the table will be ·1· and so on.

The index of a spindle in the channel is determined by the order in which they have been defined in the machine parameter table (CHSPDLNAME). The index of the first spindle of the table will be ·1· and so on.

CHAXISNAME CHSPDLNAME	Index of the axis in the channel.	Index of the spindle in the channel.
CHAXISNAME 1	Index 1.	
CHAXISNAME 2	Index 2.	
CHAXISNAME 3	Index 3.	
CHSPDLNAME 1		Index 1.
CHSPDLNAME 2		Index 2.

Variables of the master spindle.

In a multi-spindle system, the master spindle is the main spindle of the channel, the one receiving the commands when no specific spindle is mentioned. Each channel has a master spindle. In a single-spindle channel, that will always be the master spindle.

The variables of the master spindle are identified with the prefix –SP.– but without indicating the spindle. These variables may be used to access the data of the master spindle without knowing its name or logic number. These variables are meant for displaying data and programming cycles.

Programming the channel number is optional; if not programmed, it works as follows depending on who executes the variable.

Where they are executed.	Meaning when no channel has been programmed.
Part-program. MDI/MDA mode.	Channel that is executing the variable.
PLC	First channel or main channel.
External interface.	Active channel.

10.4 Variables related to general machine parameters.

CHANNEL CONFIGURATION.

(V.)MPG.NCHANNEL

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Number of CNC channels.

```
V.MPG.NCHANNEL
```

CONFIGURING THE AXES OF THE SYSTEM

(V.)MPG.NAXIS

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Number of axes governed by the CNC (spindles not included).

```
V.MPG.NAXIS
```

(V.)MPG.AXISNAMEn

Variable to be read via interface.

Name of logic axis "n".

Syntax.

Replace the "n" letter with the axis logic number.

```
MPG.AXISNAME2          Axis with logic number ·2·.
```

Remarks.

The logic number of the axes is determined by the order in which the axes have been defined in the machine parameter table. The first axis of the table will be logic axis -1- and so on.

CONFIGURATION OF A TANDEM SYSTEM.

(V.)MPG.TMASTERAXIS[nb]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Tandem pair [nb]. Logic number of the master axis/spindle.

If no axis has been defined, the variable will return a ·0· value.

Syntax.

·nb· Number of the tandem pair

```
V.MPG.TMASTERAXIS[2]          Second tandem pair.
```

(V.)MPG.TSLAVEAXIS[nb]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Tandem pair [nb]. Logic number of the slave axis/spindle.

If no axis has been defined, the variable will return a ·0· value.

Syntax.

·nb· Number of the tandem pair

V.MPG.TSLAVEAXIS[2]	Second tandem pair.

(V.)MPG.TORQDIST[nb]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Tandem pair [nb]. Torque distribution (percentage required from the master motor).

By torque distribution, we mean the torque percentage supplied by each motor to obtain the total necessary torque in the tandem. This variable shows the percentage of the total torque to be supplied by the master axis. The difference between this value and 100% will be the percentage to be supplied by the slave motor.

Syntax.

·nb· Number of the tandem pair

V.MPG.TORQDIST[2]	Second tandem pair.

Remarks.

The reading from the PLC will be given in hundredths (x100); i.e. if the parameter value is ·10·, the PLC reading will return a value of ·1000·.

(V.)MPG.PRELOAD[nb]

Variable that can only be read from the program, PLC and interface.
Variable that may be modified from the environment of the oscilloscope and setup.
The variable returns the execution value; reading it interrupts block preparation.

Tandem pair [nb]. Preload between both motors.

Preload is the torque difference to be applied between the master motor and the slave motor. The preload sets a traction between both motors in order to eliminate the backlash when the tandem is in rest position. This variable shows the percentage of the rated torque of the master motor that is applied as preload.

If the variable returns a ·0· value, it means that preload is disabled.

Syntax.

·nb· Number of the tandem pair

V.MPG.PRELOAD[2]	Second tandem pair.

Remarks.

The reading from the PLC will be given in hundredths (x100); i.e. if the parameter value is ·10·, the PLC reading will return a value of ·1000·.

(V.)MPG.PRELFITI[nb]

Variable that can only be read from the program, PLC and interface.
Variable that may be modified from the environment of the oscilloscope and setup.
The variable returns the execution value; reading it interrupts block preparation.

Tandem pair [nb]. Filter time to apply the preload.

This filter sets the time during which preload is applied gradually. If the variable returns a ·0· value, it means that the filter is disabled.

Syntax.

·nb· Number of the tandem pair

V.MPG.PRELFITI[2]	Second tandem pair.

(V.)MPG.TPROGAIN[nb]

Variable that can only be read from the program, PLC and interface.
Variable that may be modified from the environment of the oscilloscope and setup.
The variable returns the execution value; reading it interrupts block preparation.

Tandem pair [nb]. Proportional gain (Kp) for the tandem.

The proportional controller generates an output proportional to the torque error between the two motors. If the variable returns a ·0· value, it means that no proportional gain is applied.

Syntax.

·nb· Number of the tandem pair

V.MPG.TPROGAIN[2]	Second tandem pair.

Remarks.

The reading from the PLC will be given in hundredths (x100); i.e. if the parameter value is ·10·, the PLC reading will return a value of ·1000·.

(V.)MPG.TINTIME[nb]

Variable that can only be read from the program, PLC and interface.
Variable that may be modified from the environment of the oscilloscope and setup.
The variable returns the execution value; reading it interrupts block preparation.

Tandem pair [nb]. Integral gain (Ki) for the tandem.

The integral controller generates an output proportional to the integral of the torque error between the two motors. If the variable returns a ·0· value, it means that no integral gain is applied.

Syntax.

·nb· Number of the tandem pair

V.MPG.TINTIME[2]	Second tandem pair.

(V.)MPG.TCOMPLIM[nb]

Variable that can only be read from the program, PLC and interface.
Variable that may be modified from the environment of the oscilloscope and setup.
The variable returns the execution value; reading it interrupts block preparation.

Tandem pair [nb]. Compensation limit.

Syntax.

·nb· Number of the tandem pair

V.MPG.TCOMPLIM[2]	Second tandem pair.

Remarks.

The reading from the PLC will be given in hundredths (x100); i.e. if the parameter value is ·10·, the PLC reading will return a value of ·1000·.

CONFIGURATION OF A GANTRY AXIS..

(V.)MPG.MASTERAXIS[nb]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Gantry axis [nb]. Logic number of the master axis.

If no axis has been defined, the variable will return a ·0· value.

Syntax.

·nb· Number of the gantry pair

V.MPG.MASTERAXIS[2]	Second gantry pair.

(V.)MPG.SLAVEAXIS[nb]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Gantry axis [nb]. Logic number of the slave axis.

If no axis has been defined, the variable will return a ·0· value.

Syntax.

·nb· Number of the gantry pair

V.MPG.SLAVEAXIS[2]	Second gantry pair.

(V.)MPG.WARNCOUPE[nb]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Gantry axis [nb]. Difference between the following errors of both axes for issuing a warning.

This variable shows the maximum difference allowed between the following errors of both axes for issuing a warning.

Syntax.

·nb· Number of the gantry pair

V.MPG.WARNCOUPE[2]	Second gantry pair.

(V.)MPG.MAXCOUPE[nb]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Gantry axis [nb]. Maximum difference allowed between the following errors of both axes.

This variable shows the maximum difference allowed between the following errors of both axes.

Syntax.

·nb· Number of the gantry pair

V.MPG.MAXCOUPE[2]	Second gantry pair.

(V.)MPG.DIFFCOMP[nb]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Gantry axis [nb]. Compensate for the coordinate (position) difference between the two axes after G74.

Syntax.

·nb· Number of the gantry pair

V.MPG.DIFFCOMP[2]	Second gantry pair.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

(V.)MPG.MAXDIFF[nb]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Gantry axis [nb]. Maximum position value (coordinate) difference between both axes to compensate for.

Syntax.

·nb· Number of the gantry pair

V.MPG.MAXDIFF[2]	Second gantry pair.

CONFIGURATION OF A MULTI-AXIS GROUP.

(V.)MPG.MULNGROUP

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Number of multi-axis groups in the system.

V.MPG.MULNGROUP

(V.)MPG.MULNAXIS[nb]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Number of axes and/or spindles making up the multi-axis group.

Syntax.

·nb· Multi-axis group number.

V.MPG.MULNAXIS[2]	Second multi-axis group.

(V.)MPG.MULAXISNAMExn[nb]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Name of the axes and/or spindles that make up the multi-axis group.

Syntax.

·nb· Multi-axis group number.

·xn· Name of the axis and/or spindle within the multi-axis group.

V.MPG.MULAXISNAME4[2]	Fourth axis of the second multi-axis group.

Values of the variable.

The values returned by this variable are encoded as follows.

X=10	X1=11	X2=12	X3=13	X4=14	... X9=19
Y=20	Y1=21	Y2=22	Y3=23	Y4=24	... Y9=29
Z=30	Z1=31	Z2=32	Z3=33	Z4=34	... Z9=39
U=40	U1=41	U2=42	U3=43	U4=44	... U9=49

V=50	V1=51	V2=52	V3=53	V4=54	... V9=59
W=60	W1=61	W2=62	W3=63	W4=64	... W9=69
A=70	A1=71	A2=72	A3=73	A4=74	... A9=79
B=80	B1=81	B2=82	B3=83	B4=84	... B9=89
C=90	C1=91	C2=92	C3=93	C4=94	... C9=99
S=100	S1=101	S2=102	S3=103	S4=104	... S9=109

CONFIGURING THE SPINDLES OF THE SYSTEM

(V.)MPG.NSPDL

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Number of spindles governed by the CNC.

V.MPG.NSPDL

(V.)MPG.SPDLNAMEn

Variable to be read via interface.

Name of logic spindle "n".

Syntax.

Replace the "n" letter with the spindle logic number.

MPG.SPDLNAME2	Spindle with logic number ·2·.

Remarks.

The logic number of the spindles is determined by the order in which they have been defined in the machine parameter table. The logic numbering of the spindles continues from the last logic axis; hence, in a 5-axis system, the first spindle of the table will be logic spindle ·6· and so on.

TIME SETTING (SYSTEM).

(V.)MPG.LOOPTIME

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

CNC cycle (loop) time in milliseconds.

V.MPG.LOOPTIME

(V.)MPG.PRGFREQ

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Frequency of the PLC's PRG module (in cycles).

The frequency of the module means how often (every how many CNC cycles) a full scan of the PLC program is executed.

V.MPG.PRGFREQ

SERCOS BUS CONFIGURATION

(V.)MPG.SERBRATE

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Sercos transmission speed

V.MPG.SERBRATE

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	4 Mbps.
1	2 Mbps.
2	16 Mbps.
3	8 Mbps.

(V.)MPG.SERPOWSE

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Sercos optical power.

V.MPG.SERPOWSE

MECHATROLINK BUS CONFIGURATION.

(V.)MPG.MLINK

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Mechatrolink mode.

V.MPG.MLINK

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Mlink-I
2	Mlink-II

(V.)MPG.DATASIZE
Telegram size in Mlink-II mode.

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Telegram size in Mlink-II mode.

V.MPG.DATASIZE

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	17 bytes.
1	32 bytes.

CAN BUS CONFIGURATION.

(V.)MPG.CANMODE

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

CAN bus protocol.

V.MPG.CANMODE

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	CANfagor protocol.
1	CANopen protocol.

(V.)MPG.CANLENGTH

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

CANfagor bus cable length (in meters)

V.MPG.CANLENGTH

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.	Value.	Meaning.
0	Up to 20 meters.	7	Up to 90 meters.
1	Up to 30 meters.	8	Up to 100 meters.
2	Up to 40 meters.	9	Up to 110 meters.
3	Up to 50 meters.	10	Up to 120 meters.
4	Up to 60 meters.	11	Up to 130 meters.
5	Up to 70 meters.	12	More than 130 meters.
6	Up to 80 meters.		

SERIAL LINE CONFIGURATION.

(V.)MPG.RSTYPE

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Type of serial line.

V.MPG.RSTYPE

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
1	RS232.
2	RS485.
3	RS422.

DEFAULT CONDITIONS (SYTEM).

(V.)MPG.INCHES

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Default measuring units.

V.MPG.INCHES

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Millimeters.
1	Inches.

ARITHMETIC PARAMETERS.

(V.)MPG.MAXLOCP

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Upper limit for local arithmetic parameters.

V.MPG.MAXLOCP

(V.)MPG.MINLOCP

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Lower limit for local arithmetic parameters.

V.MPG.MINLOCP

(V.)MPG.MAXGLBP

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Upper limit for global arithmetic parameters.

V.MPG.MAXGLBP

(V.)MPG.MINGLBP

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Lower limit for global arithmetic parameters.

V.MPG.MINGLBP

(V.)MPG.ROPARMIN

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Lower limit for global read-only arithmetic parameters.

The variable will return a ·0· if no range has been defined or the range is wrong.

V.MPG.ROPARMIN

(V.)MPG.ROPARMAX

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Upper limit for global read-only arithmetic parameters.

The variable will return a ·0· if no range has been defined or the range is wrong.

V.MPG.ROPARMAX

(V.)MPG.MAXCOMP

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Upper limit for common arithmetic parameters.

V.MPG.MAXCOMP

(V.)MPG.MINCOMP

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Lower limit for common arithmetic parameters.

V.MPG.MINCOMP

(V.)MPG.BKUPCUP

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Number of common non-volatile arithmetic parameters.

V.MPG.BKUPCUP

CROSS COMPENSATION TABLE.

(V.)MPG.MOVAXIS[tbl]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Cross compensation table [tbl]. Logic number of the master axis.

The variable will return a ·0· value if the table has not been defined.

Syntax.

·tbl· Table number.

V.MPG.MOVAXIS[3]	Third cross compensation table.

(V.)MPG.COMPAXIS[tbl]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Cross compensation table [tbl]. Logic number of the compensated axis.

The variable will return a ·0· value if the table has not been defined.

Syntax.

·tbl· Table number.

V.MPG.COMPAXIS[3]	Third cross compensation table.

(V.)MPG.NPCROSS[tbl]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Cross compensation table [tbl]. Number of points in the table.

Syntax.

·tbl· Table number.

V.MPG.NPCROSS[3]	Third cross compensation table.

(V.)MPG.TYPCROSS[tbl]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Cross compensation table [tbl]. Type of compensation (type of coordinates).

Syntax.

·tbl· Table number.

V.MPG.TYPCROSS[3]	Third cross compensation table.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Compensation is applied using real coordinates.
1	Compensation is applied using theoretical coordinates.

(V.)MPG.BIDIR[tbl]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Cross compensation table [tbl]. Bi-directional compensation.

When using bi-directional compensation, the table permits using a different compensation for each moving direction. If the compensation is not bidirectional, it applies the same compensation in both directions.

Syntax.

·tbl· Table number.

V.MPG.BIDIR[3]	Third cross compensation table.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

(V.)MPG.REFNEED[tbl]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Cross compensation table [tbl]. Both axes must be homed in order to apply the compensation.

Syntax.

·tbl· Table number.

V.MPG.REFNEED[3]	Third cross compensation table.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

(V.)MPG.POSITION[tbl][pt]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Cross compensation table [tbl]; point [pt] Position of the master axis.

Syntax.

·tbl· Table number.

·pt· Point in the table.

V.MPG.POSITION[3][14]	Point 14 of the third cross compensation table.

(V.)MPG.POSERROR[tbl][pt]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Cross compensation table [tbl]; point [pt] Amount of error to compensate when moving in the positive direction.

If there is no bi-directional, it indicates the amount of error to compensate in both directions.

Syntax.

·tbl· Table number.

·pt· Point in the table.

V.MPG.POSERROR[3][14]	Point 14 of the third cross compensation table.

(V.)MPG.NEGERROR[tbl][pt]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Cross compensation table [tbl]; point [pt] Amount of error to compensate when moving in the negative direction.

Syntax.

·tbl· Table number.

·pt· Point in the table.

V.MPG.NEGERROR[3][14]	Point 14 of the third cross compensation table.

VOLUMETRIC COMPENSATION TABLES.

(V.)MPG.VCOMPAXIS1[tbl]
(V.)MPG.VCOMPAXIS2[tbl]
(V.)MPG.VCOMPAXIS3[tbl]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Volumetric compensation table [tbl]. Axis to be compensated.

Syntax.

·tbl· Table number.

V.MPG.COMPAXIS2[1]	Second axis to be compensated in the first volumetric compensation table.

(V.)MPG.VCOMPFILE[tbl]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Volumetric compensation table [tbl]. File containing volumetric compensation data.

Syntax.

·tbl· Table number.

V.MPG.VCOMPFILE[1]	File for defining the first volumetric compensation.

EXECUTION TIMES.

(V.)MPG.MINAENDW

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Minimum duration of the AUXEND signal (in milliseconds).

V.MPG.MINAENDW

(V.)MPG.REFTIME

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Estimated home searching time (in milliseconds).

V.MPG.REFTIME

(V.)MPG.HTIME

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Estimated time for an "H" function (in milliseconds).

V.MPG.HTIME

(V.)MPG.DTIME

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Estimated time for an "D" function (in milliseconds).

V.MPG.DTIME

(V.)MPG.TTIME

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Estimated time for an "T" function (in milliseconds).

V.MPG.TTIME

NUMBERING OF THE DIGITAL INPUTS AND OUTPUTS.

(V.)MPG.NDIMOD

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Total number of digital input modules.

V.MPG.NDIMOD

This variable indicates the number of these modules connected to the same CAN bus. In remote modules with CANopen protocol each double module of digital inputs and outputs counts as two.

(V.)MPG.NDOMOD

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Total number of digital output modules.

V.MPG.NDOMOD

This variable indicates the number of these modules connected to the same CAN bus. In remote modules with CANopen protocol each double module of digital inputs and outputs counts as two.

(V.)MPG.DIMODADDR[nb]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Base address of the digital input modules.

Syntax.

·nb· Module number.

V.MPG.DIMODADDR[4]	Fourth digital input module.

(V.)MPG.DOMODADDR[nb]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Base address of the digital output modules.

Syntax.

·nb· Module number.

V.MPG.DOMODADDR[4]	Fourth digital output module.

NUMBERING OF ANALOG INPUTS FOR TEMPERATURE SENSORS PT100.

(V.)MPG.NPT100

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Total number of active PT100 inputs.

V.MPG.NPT100

(V.)MPG.PT100[nb]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Analog input associated with the PT100 input.

Syntax.

·nb· PT100 input number.

V.MPG.NPT100[3]	Third PT100 input.

PROBE SETTING.

(V.)MPG.PROBE

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

There is a probe.

V.MPG.PROBE

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

(V.)MPG.PROBETYPE1

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Probe type 1, depending on where it is connected.

V.MPG.PROBETYPE1

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Remote probe.
1	Local probe.

(V.)MPG.PROBETYPE2

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Probe type 2, depending on where it is connected.

V.MPG.PROBETYPE2

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Remote probe.
1	Local probe.

(V.)MPG.PRBDI1

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Number of the input associated with probe 1.

For remote probes, this parameter indicates the digital input number; for local probes, it indicates the local probe input number.

The variable will return a ·0· value if no digital input has been defined.

V.MPG.PRBDI1

(V.)MPG.PRBDI2

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Number of the input associated with probe 2.

For remote probes, this parameter indicates the digital input number; for local probes, it indicates the local probe input number.

The variable will return a ·0· value if no digital input has been defined.

V.MPG.PRBDI2

(V.)MPG.PRBPULSE1

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Logic level to activate probe 1.

V.MPG.PRBPULSE1

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Low logic level (0 V).
1	High logic level (5 V / 24 V).

(V.)MPG.PRBPULSE2

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Logic level to activate probe 2.

V.MPG.PRBPULSE2

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Low logic level (0 V).
1	High logic level (5 V / 24 V).

SHARED PLC MEMORY.

(V.)MPG.PLCDATASIZE

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Size (in bytes) of the PLC's shared data area.

V.MPG.PLCDATASIZE

MANAGEMENT OF LOCAL I/O.

(V.)MPG.NLOCOUT

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Number of local digital outputs.

V.MPG.NLOCOUT

(V.)MPG.EXPSCHK

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Activate the 24 V monitoring at the local digital outputs.

V.MPG.EXPSCHK

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

SYNCHRONIZED SWITCHING.

(V.)MPG.SWTOUTPUT

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Local digital output associated with synchronized switching.

V.MPG.SWTOUTPUT

(V.)MPG.SWTDELAY

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Delay of the device associated with synchronized switching.

V.MPG.SWTDELAY

PWM (PULSE-WIDTH MODULATION).

(V.)MPG.PWMOUTPUT

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Local digital output associated with the PWM.

(V.)MPG.PWMOUTPUT

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	PWM missing.
1	Local output 1 (pin LI/O1).
2	Local output 2 (pin LI/O2).

(V.)MPG.PWMCANCEL

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Cancel the PWM after an M30 or a reset.

(V.)MPG.PWMCANCEL

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

BACKUP OF NON-VOLATILE DATA.

(V.)MPG.BKUPREG

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Number of non-volatile PLC registers.

V.MPG.BKUPREG

(V.)MPG.BKUPCOUN

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Number of non-volatile PLC counters.

V.MPG.BKUPCOUN

TOOL OFFSET AND WEAR.

(V.)MPG.TOOLOFSG

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Sign criteria for tool offsets and tool wear.

The offsets are used to define the tool dimensions in each axis. The dimensions of the turning tools are defined using these offsets; either these offsets or tool length and radius may be used for the dimensions of the rest of the tools.

V.MPG.TOOLOFSG

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Positive.
1	Negative.

SPINDLE SYNCHRONIZATION.

(V.)MPG.SYNCCANCEL

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Cancel the spindle synchronization.

V.MPG.SYNCCANCEL

This parameter indicates whether the CNC cancels spindle synchronization or not after executing M02, M30 or after an error or reset.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

DEFINE THE NUMBER OF JOG PANELS AND THEIR RELATIONSHIP WITH THE CHANNELS.

(V.)MPG.NKEYBD

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Number of jog panels.

V.MPG.NKEYBD

(V.)MPG.KEYBDCH[jog]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel to which the jog panel is assigned.

Syntax.

·jog· Number of jog panel.

V.MPG.KEYBDCH[2] Jog panel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Active channel.
1	Channel ·1·.
2	Channel ·2·.
3	Channel ·3·.
4	Channel ·4·.

Remarks.

The CNC numbers the operator panels following the order (sequence) that they occupy in the CAN bus (Address switch). The first jog panel will be the one with the lowest number and so on.

PLC TYPE.

(V.)MPG.PLCTYPE

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

PLC type.

V.MPG.PLCTYPE

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	IEC.
1	IEC+Fagor.
2	Fagor.

RENAME THE AXES AND THE SPINDLES.

(V.)MPG.RENAMECANCEL

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Cancel the name change for the axes and the spindles.

V.MPG.RENAMECANCEL

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

ZERO OFFSETS.

(V.)MPG.FINEORG

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Fine definition of zero offsets.

V.MPG.FINEORG

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

REMOTE MODULE RCS-S (SERCOS COUNTER).

(V.)MPG.NSERCOUNT

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Number of RCS-S modules in the bus.

V.MPG.NSERCOUNT

(V.)MPG.SERCOUNTID[n]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Thumbwheel identifiers associated with Sercos counters.

Syntax.

·num· RCS-S module number.

V.MPG.SERCOUNTID[2] Module RCS-S ·2·.

10.5 Variables related to the machine parameters of the channels.

CHANNEL CONFIGURATION.

(V.)[ch].MPG.GROUPID

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Group the channel belongs to.

If the variable returns a ·0· value, it means that the channel is not associated with any group.

Syntax.

·ch· Channel number.

V.[2].MPG.GROUPID	Channel ·2·.

(V.)[ch].MPG.CHTYPE

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Channel type.

A channel may be governed from the CNC, from the PLC or from both. Channels governed by the PLC are not displayed in automatic, jog or edisimu modes. The tables can be accessed.

Syntax.

·ch· Channel number.

V.[2].MPG.CHTYPE	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	CNC channel.
1	PLC channel.
2	CNC and PLC channel.

(V.)[ch].MPG.HIDDENCH

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Hidden channel.

Hidden channels are not displayed and cannot be selected.

Syntax.

·ch· Channel number.

V.[2].MPG.HIDDENCH	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

CONFIGURING THE AXES OF THE CHANNEL.

(V.)[ch].MPG.CHNAXIS

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Number of the channel axes without including spindles.

It is possible to change the configuration of the axes of a channel via part-program, (defining a new configuration, adding or removing axes) using the instructions #SET AX, #FREE AX and #CALL AX.

Syntax.

·ch· Channel number.

V.[2].MPG.CHNAXIS	Channel ·2·.

(V.)[ch].MPG.CHAXISNAMEn

Variable to be read via interface.

Channel [ch]. Name the "n" axis of the channel.

Syntax.

·ch· Channel number.

·n· Index of the axis in the channel.

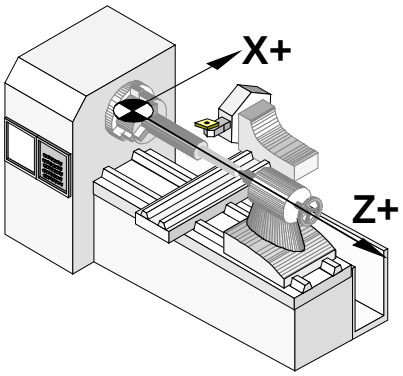
[2].MPG.CHAXISNAME4	Axis with index ·1· in the channel ·2·.

(V.)[ch].MPG.GEOCONFIG

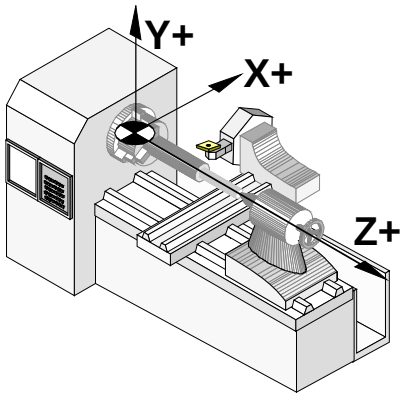
Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Geometrical configuration of the axes of the channel.

On the lathe model, the geometric configuration of the axes may be a "trihedron" type or a "plane" type.



Configuration of "plane" type axes.



Configuration of trihedron type axes.

Syntax.

·ch· Channel number.

V.[2].MPG.GEOCONFIG	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Configuration of "plane" type axes.
1	Configuration of "Trihedron" type axes.

Remarks.

Configuration.	Configuration properties.
"Trihedron" type.	In this configuration, there are three axes forming a Cartesian XYZ type trihedron like on a milling machine. There may be more axes besides those forming the trihedron. With this configuration, the planes behave in the same way as on a milling machine except that the usual work plane will be G18 (if it has been configured like that).
"plane" type.	In this configuration, there are two axes forming the usual work plane. There may be more axes, but they cannot be part of the trihedron; there must be auxiliary, rotary, etc. With this configuration, the work plane is always G18 and will be formed by the first two axes defined in the channel. In this configuration, the second axis of the channel is considered as longitudinal axis. If the X (first) and Z (second) axes have been defined, the work plane will be the ZX (Z as abscissa and X as ordinate) and Z the longitudinal axis. Tool length compensation is applied on this longitudinal axis when using milling tools. With lathe tools, tool length compensation is applied on all the axes where a tool offset has been defined.

CONFIGURING THE SPINDLES OF THE CHANNEL.

(V.)[ch].MPG.CHNSPDL

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Number of spindles of the channel.

It is possible to change the configuration of the spindles of a channel via part-program, (defining a new configuration, adding or remove spindles) using the instructions #SET SP, #FREE SP and #CALL SP.

Syntax.

·ch· Channel number.

V.[2].MPG.CHSPDL	Channel ·2·.

(V.)[ch].MPG.CHSPDLNAMEn

Variable to be read via interface.

Channel [ch]. Name the "n" spindle of the channel.

Syntax.

·ch· Channel number.

·n· Index of the spindle in the channel.

[2].MPG.CHSPDLNAME1	Spindle with index ·1· in the channel ·2·.

CONFIGURATION OF THE C AXIS.

(V.)[ch].MPG.CAXNAME

Variable to be read via interface.

Channel [ch]. Name of the axis working as "C" axis (by default).

When setting more than one C axis, use the program instruction #CAX to indicate the one that is active. Only one C axis may be active in each channel.

Syntax.

·ch· Channel number.

[2].MPG.CAXNAME	Channel ·2·.

(V.)[ch].MPG.ALIGNC

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. "C" axis alignment for diametrical machining.

This parameter indicates whether the C axis must be aligned for diameter machining (ALIGNC = Yes) or the tool can machine the whole surface diametrically in a single fixturing operation (ALIGNC = NO).

Syntax.

·ch· Channel number.

V.[2].MPG.ALIGNC	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

TIME SETTING (CHANNEL).

(V.)[ch].MPG.PREPFREQ

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Maximum number of blocks to prepare per cycle.

Syntax.

·ch· Channel number.

V.[2].MPG.PREPFREQ	Channel ·2·.

(V.)[ch].MPG.ANTIME

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Anticipation time.

Syntax.

·ch· Channel number.

V.[2].MPG.ANTIME	Channel ·2·.

CONFIGURATION OF THE HSC MODE (CHANNEL).

(V.)[ch].MPG.FEEDAVRG

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Calculating feedrate average.

Syntax.

·ch· Channel number.

V.[2].MPG.FEEDAVRG	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

(V.)[ch].MPG.SMOOTHFREQ

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Smoothing frequency in the interpolation.

Syntax.

·ch· Channel number.

V.[2].MPG.SMOOTHFREQ	Channel ·2·.

(V.)[ch].MPG.CORNER

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Maximum angle of the corner to machine it in square corner mode.

Syntax.

·ch· Channel number.

V.[2].MPG.CORNER	Channel ·2·.

(V.)[ch].MPG.HSCFILTFREQ

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Filter frequency (CONTERROR mode).

Syntax.

·ch· Channel number.

V.[2].MPG.HSCFILTFREQ	Channel ·2·.

(V.)[ch].MPG.FASTFACTOR

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Default feedrate percentage (FAST mode).

Syntax.

·ch· Channel number.

V.[2].MPG.FASTFACTOR	Channel ·2·.

(V.)[ch].MPG.FTIMELIM

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Time difference permitted in feedrate interpolation (FAST mode).

Syntax.

·ch· Channel number.

V.[2].MPG.FTIMELIM	Channel ·2·.

(V.)[ch].MPG.MINCORFEED

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Minimum feedrate at the corners.

Syntax.

·ch· Channel number.

V.[2].MPG.MINCORFEED	Channel ·2·.

(V.)[ch].MPG.FSMOOTHFREQ

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Smoothing frequency in the interpolation (FAST mode).

Syntax.

·ch· Channel number.

V.[2].MPG.FSMOOTHFREQ	Channel ·2·.

(V.)[ch].MPG.FASTFILTFREQ

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Filter frequency (FAST mode).

Syntax.

·ch· Channel number.

V.[2].MPG.FASTFILTFREQ	Channel ·2·.

(V.)[ch].MPG.FREQRES

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. First resonance frequency of the machine.

Syntax.

·ch· Channel number.

V.[2].MPG.FREQRES	Channel ·2·.

(V.)[ch].MPG.SOFTFREQ

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Path filter frequency for profiles of linear acceleration.

Syntax.

·ch· Channel number.

V.[2].MPG.SOFTFREQ	Channel ·2·.

(V.)[ch].MPG.HSCROUND

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Default value of the maximum path error in HSC.

Syntax.

·ch· Channel number.

V.[2].MPG.HSCROUND	Channel ·2·.

(V.)[ch].MPG.SURFFILFREQ

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Axis filter frequency (SURFACE mode).

Syntax.

·ch· Channel number.

V.[2].MPG.SURFFILFREQ	Channel ·2·.

(V.)[ch].MPG.HSCDEFAULMODE

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Default mode when programming #HSC ON.

Syntax.

·ch· Channel number.

V.[2].MPG.HSCDEFAULMODE	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	SURFACE
1	CONTERROR
2	FAST

VIRTUAL TOOL AXIS.

(V.)[ch].MPG.VIRTAXISNAME

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Name of the virtual tool axis.

Syntax.

·ch· Channel number.

V.[2].MPG.VIRTAXISNAME	Channel ·2·.

Values of the variable.

The values returned by this variable are encoded as follows.

X=10	X1=11	X2=12	X3=13	X4=14	... X9=19
Y=20	Y1=21	Y2=22	Y3=23	Y4=24	... Y9=29
Z=30	Z1=31	Z2=32	Z3=33	Z4=34	... Z9=39

U=40	U1=41	U2=42	U3=43	U4=44	... U9=49
V=50	V1=51	V2=52	V3=53	V4=54	... V9=59
W=60	W1=61	W2=62	W3=63	W4=64	... W9=69
A=70	A1=71	A2=72	A3=73	A4=74	... A9=79
B=80	B1=81	B2=82	B3=83	B4=84	... B9=89
C=90	C1=91	C2=92	C3=93	C4=94	... C9=99

(V.)[ch].MPG.VIRTAXCANCEL

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Cancel the virtual tool axis after an M30 and Reset.

Syntax.

·ch· Channel number.

```
V.[2].MPG.VIRTAXCANCEL          Channel ·2·.
```

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

DEFAULT CONDITIONS (CHANNEL).

(V.)[ch].MPG.KINID

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Default kinematics number.

The channel assumes the default value on power-up, after executing an M02 or M30 or after a Reset. To select another kinematics from the part-program, use the #KIN ID instruction.

Syntax.

·ch· Channel number.

```
V.[2].MPG.KINID                 Channel ·2·.
```

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	The CNC restores the last kinematics active.
1··6	Default kinematics number.
255	There is no default kinematics.

(V.)[ch].MPG.CSCANCEL

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Cancel the inclined plane on start-up.

This parameter indicates whether on start-up the CNC cancels the inclined plane (#CS/#ACS) that was active when the CNC was turned off.

Syntax.

·ch· Channel number.

V.[2].MPG.CSCANCEL	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

(V.)[ch].MPG.LINKCANCEL

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Cancel axis coupling by default.

The channel assumes the default value after executing an M02 or M30, after an emergency or a Reset. To couple axes from the part-program, use the #LINK instruction.

Syntax.

·ch· Channel number.

V.[2].MPG.LINKCANCEL	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

(V.)[ch].MPG.MIRRORCANCEL

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Cancel mirror image (G11/G12/G13/G14) after M30 and reset.

Syntax.

·ch· Channel number.

V.[2].MPG.MIRRORCANCEL	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

(V.)[ch].MPG.SLOPETYPE

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Default acceleration type.

It indicates the type of acceleration applied by default in automatic movements. When working in manual (JOG) mode, the CNC always applies linear acceleration.

The channel assumes the default value on power-up, after executing an M02 or M30 or after a Reset. To select another acceleration from the part-program, use the #SLOPE instruction.

Syntax.

·ch· Channel number.

V.[2].MPG.SLOPETYPE	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Linear acceleration.
1	Trapezoidal acceleration.
2	Square sine (bell shaped) acceleration.

(V.)[ch].MPG.IPLANE

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Main work plane (G17/G18) by default.

The channel assumes the default value on power-up, after executing an M02 or M30 or after a Reset. To change the work plane via part-program, use function G17, G18, G19 or G20.

Syntax.

·ch· Channel number.

V.[2].MPG.IPLANE	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	G17.
1	G18.

(V.)[ch].MPG.ISYSTEM

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Default type of coordinates (G90/G91).

The coordinates of a point may be defined either in absolute coordinates (G90) referred to part zero or in incremental coordinates (G91) referred to the current position.

The channel assumes the default value on power-up, after executing an M02 or M30 or after a Reset. To change the type of coordinates via part-program, use function G90 or G91.

Syntax.

·ch· Channel number.

V.[2].MPG.ISYSTEM	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	G90.
1	G91.

(V.)[ch].MPG.IMOVE

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Default type of movement (G0/G1).

The channel assumes the default value on power-up, after executing an M02 or M30 or after a Reset. To change the type of movement via part-program, use function G0 or G1.

Syntax.

·ch· Channel number.

V.[2].MPG.IMOVE	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	G00.
1	G01.

(V.)[ch].MPG.IFEED

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Default type of feedrate (G94/G95).

The channel assumes the default value on power-up, after executing an M02 or M30 or after a Reset. To change the type of feedrate via part-program, use function G93 or G94.

Syntax.

·ch· Channel number.

V.[2].MPG.IFEED	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	G94.
1	G95.

(V.)[ch].MPG.FPRMAN

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Function G95 admitted in jog mode.

Syntax.

·ch· Channel number.

V.[2].MPG.FPRMAN	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

(V.)[ch].MPG.LCOMPTYP

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Keep the longitudinal axis when changing planes (G17/G18/G19).

Syntax.

·ch· Channel number.

V.[2].MPG.LCOMPTYP	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

(V.)[ch].MPG.ICORNER

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Default type of corner (G5/G7/G50).

The channel assumes the default value on power-up, after executing an M02 or M30 or after a Reset. To change the type of corner via part-program, use function G5, G7 or G50.

Syntax.

·ch· Channel number.

V.[2].MPG.ICORNER	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	G50.
1	G05.
2	G07.

(V.)[ch].MPG.IRCOMP

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Tool radius compensation mode (G136/G137) by default

The channel assumes the default value on power-up, after executing an M02 or M30 or after a Reset. To change the type of tool radius compensation via part-program, use function G136 or G137.

Syntax.

·ch· Channel number.

V.[2].MPG.IRCOMP	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	G136.
1	G137.

(V.)[ch].MPG.COMPCANCEL

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. How to cancel tool radius compensation.

This parameter indicates whether tool radius compensation is canceled in the first motion block, even if the plane axes are not involved, or if it requires a movement of the plane axes.

Syntax.

·ch· Channel number.

V.[2].MPG.COMPCANCEL	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Without moving the plane axes.
1	Moving the plane axes.

(V.)[ch].MPG.ROUNDTYPE

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Default rounding type in G5.

The rounding may be executed by limiting the chordal error or the feedrate. The chordal error (#ROUNDPAR [1]) defines the maximum deviation allowed between the programmed point and the resulting profile. The feedrate (#ROUNDPAR [2]) defines the percentage of the active feedrate to be used for machining.

The channel assumes the default value on power-up, after executing an M02 or M30 or after a Reset. To change the type of rounding via program, use the #ROUNDPAR instruction.

Syntax.

·ch· Channel number.

V.[2].MPG.ROUNDTYPE	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Chordal error.
1	Percentage of feedrate.

(V.)[ch].MPG.MAXROUND

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Maximum rounding error in G5.

Syntax.

·ch· Channel number.

V.[2].MPG.MAXROUND	Channel ·2·.

(V.)[ch].MPG.ROUNDFEED

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Percentage of feedrate in G5.

Syntax.

·ch· Channel number.

V.[2].MPG.ROUNDFEED	Channel ·2·.

ARC CENTER CORRECTION.

(V.)[ch].MPG.CIRINERR

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Maximum absolute radius error permitted.

Syntax.

·ch· Channel number.

V.[2].MPG.CIRINERR	Channel ·2·.

(V.)[ch].MPG.CIRINFACT

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Maximum relative radius error permitted.

The relative error is given as a percentage over the radius.

Syntax.

·ch· Channel number.

V.[2].MPG.CIRINFACT	Channel ·2·.

Remarks.

The reading from the PLC will be given in tenths (x10); i.e. if the parameter value is ·10·, the PLC reading will return a value of ·100·.

BEHAVIOR OF THE FEEDRATE AND THE FEEDRATE OVERRIDE.

(V.)[ch].MPG.MAXOVR

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Maximum override (%) permitted.

Syntax.

·ch· Channel number.

V.[2].MPG.MAXOVR	Channel ·2·.

Remarks.

The reading from the PLC will be given in tenths (x10); i.e. if the parameter value is ·10·, the PLC reading will return a value of ·100·.

(V.)[ch].MPG.RAPIDOVR

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Override acts in G00 (from 0 to 100%).

This parameter indicates whether the feedrate % may be modified (between 0% and 100%) or not when working in G0; if not possible, it will stay fixed at 100%.

Regardless of the value assigned to this parameter, the override always attends to the 0% position and never acts over 100%. It is always possible to change the % of feedrate when moving in jog mode.

Syntax.

·ch· Channel number.

V.[2].MPG.RAPIDOVR	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

(V.)[ch].MPG.FEEDND

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Apply the programmed feedrate to all the axes of the channel

This parameter indicates whether the programmed feedrate is applied to all the axes of the channel or only to the main axes. If it is only applied to the main axes, the rest of the axes move at their corresponding feedrate to end the movement of them all at the same time.

Syntax.

·ch· Channel number.

V.[2].MPG.FEEDND	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No. The programmed feedrate is only applied to the main axes.
1	Yes. The programmed feedrate is applied to all the axes of the channel.

OVERRIDE OF THE DYNAMICS FOR HSC.

(V.)[ch].MPG.MINDYNOVR

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Minimum override for the dynamics in HSC.

Syntax.

·ch· Channel number.

V.[2].MPG.MINDYNOVR	Channel ·2·.

(V.)[ch].MPG.MAXDYNOVR

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Maximum override for the dynamics in HSC.

Syntax.

·ch· Channel number.

V.[2].MPG.MAXDYNOVR	Channel ·2·.

(V.)[ch].MPG.STEPDYNOVR

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Override step for the dynamics in HSC.

Syntax.

·ch· Channel number.

V.[2].MPG.STEPDYNOVR	Channel ·2·.

MOVEMENT OF THE INDEPENDENT AXES.

(V.)[ch].MPG.IMOVEMACH

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Movement of the independent axis referred to machine coordinates.

Syntax.

·ch· Channel number.

V.[2].MPG.IMOVEMACH	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

(V.)[ch].MPG.XFITOIND

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. The transfer inhibit of the channel affects the independent axes.

This parameter indicates whether the transfer inhibit of the channel (_XFERINH mark) affects the independent axis or not.

Syntax.

·ch· Channel number.

V.[2].MPG.XFITOIND	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

DEFINITION OF THE SUBROUTINES.

(V.)[ch].MPG.TOOLSUB

Variable to be read via interface.

Channel [ch]. Subroutine associated with "T".

Syntax.

·ch· Channel number.

[2].MPG.TOOLSUB	Channel ·2·.

(V.)[ch].MPG.REFPSUB

Variable to be read via interface.

Channel [ch]. Subroutine associated with function G74.

Syntax.

·ch· Channel number.

[2].MPG.REFPSUB	Channel ·2·.

(V.)[ch].MPG.OEMSUB1
..
(V.)[ch].MPG.OEMSUB10

Variable to be read via interface.

Channel [ch]. Subroutines associated with G180 through G189.

Syntax.

·ch· Channel number.

[2].MPG.OEMSUB1	Channel ·2·.

(V.)[ch].MPG.OEMSUB11
..
(V.)[ch].MPG.OEMSUB30

Variable to be read via interface.

Channel [ch]. Subroutines associated with G380 through G399.

Syntax.

·ch· Channel number.

[2].MPG.OEMSUB11	Channel ·2·.

(V.)[ch].MPG.INT1SUB
..
(V.)[ch].MPG.INT4SUB

Variable to be read via interface.

Channel [ch]. Interruption subroutines.

Syntax.

·ch· Channel number.

[2].MPG.INT1SUB	Channel ·2·.

(V.)[ch].MPG.SUBPATH

Variable to be read via interface.

Channel [ch]. Path of the program subroutines.

Syntax.

·ch· Channel number.

[2].MPG.SUBPATH	Channel ·2·.

TABLETOP PROBE POSITION.

(V.)[ch].MPG.PRB1MIN

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Minimum probe coordinate (abscissa axis).

Syntax.

·ch· Channel number.

V.[2].MPG.PRB1MIN	Channel ·2·.

(V.)[ch].MPG.PRB1MAX

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Maximum probe coordinate (abscissa axis).

Syntax.

·ch· Channel number.

V.[2].MPG.PRB1MAX	Channel ·2·.

(V.)[ch].MPG.PRB2MIN

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Minimum probe coordinate (ordinate axis).

Syntax.

·ch· Channel number.

V.[2].MPG.PRB2MIN	Channel ·2·.

(V.)[ch].MPG.PRB2MAX

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Maximum probe coordinate (ordinate axis).

Syntax.

·ch· Channel number.

V.[2].MPG.PRB2MAX	Channel ·2·.

(V.)[ch].MPG.PRB3MIN

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Minimum probe coordinate (axis perpendicular to the plane).

Syntax.

·ch· Channel number.

V.[2].MPG.PRB3MIN	Channel ·2·.

(V.)[ch].MPG.PRB3MAX

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Maximum probe coordinate (axis perpendicular to the plane).

Syntax.

·ch· Channel number.

V.[2].MPG.PRB3MAX	Channel ·2·.

BLOCK SEARCH.

(V.)[ch].MPG.FUNPLC

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Send the M, H, S functions to the PLC during block search.

Syntax.

·ch· Channel number.

V.[2].MPG.FUNPLC	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

INTERRUPTION SUBROUTINES.

(V.)[ch].MPG.SUBINTSTOP

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Execute interruption subroutines while the program is stopped or while no program is in execution.

Syntax.

·ch· Channel number.

V.[2].MPG.SUBINTSTOP	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

MACHINING FEEDRATE.

(V.)[ch].MPG.MAXFEED

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Maximum machining feedrate.

If the variable returns a ·0· value, the machining feedrate is not limited, the CNC assumes for all the movements the one set in machine parameter G00FEED as the maximum feedrate.

Syntax.

·ch· Channel number.

```
V.[2].MPG.MAXFEED                Channel ·2·.
```

(V.)[ch].MPG.DEFAULTFEED

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Assume MAXFEED for movements in G1/G2/G3 without active feedrate

If parameter MAXFEED = 0, the variable will always return a 0 value.

Syntax.

·ch· Channel number.

```
V.[2].MPG.DEFAULTFEED            Channel ·2·.
```

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

RAPID TRAVERSE FOR THE AUTOMATIC MODE.

(V.)[ch].MPG.RAPIDEN

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. It may be used to enable the rapid traverse for the automatic mode while executing a program.

Syntax.

·ch· Channel number.

```
V.[2].MPG.RAPIDEN                Channel ·2·.
```

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Disabled. Rapid traverse is not available for the automatic mode.
1	EXRAPID or rapid key. To activate the rapid feed, just activate the PLC mark EXRAPID or press the "rapid" key of the jog panel.
2	EXRAPID and rapid key. To activate the rapid feed, just activate the PLC mark EXRAPID and press the "rapid" key of the jog panel.

(V.)[ch].MPG.FRAPIDEN

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Rapid traverse in the channel when rapid traverse for the automatic mode is active (parameter RAPIDEN). If the parameter is set with a "0" value, the feedrate is not limited.

This parameter does not the movements programmed in G00 or the threads. Movements in G0 are carried out at the feedrate set in parameter G00FEED. The threads are executed at the programmed feedrate.

Syntax.

·ch· Channel number.

V.[2].MPG.FRAPIDEN	Channel ·2·.

Remarks.

Rapid traverse cannot exceed the value set in axis parameters G00FEED and FRAPIDEN or the maximum feedrate set by PLC (variable (V.)PLC.G00FEED). Rapid traverse cannot exceed the value set in axis parameter MAXFEED of the channel and the active feedrate set by PLC (variable (V.)PLC.F).

MAXIMUM ACCELERATION AND JERK ON THE TOOL PATH.

(V.)[ch].MPG.MAXACCEL

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Maximum acceleration on the machining path.

Syntax.

·ch· Channel number.

V.[2].MPG.MAXACCEL	Channel ·2·.

(V.)[ch].MPG.MAXJERK

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Maximum jerk on the machining path.

Syntax.

·ch· Channel number.

V.[2].MPG.MAXJERK	Channel ·2·.

"RETRACE" FUNCTION.

(V.)[ch].MPG.RETRACAC

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Permit enabling the "retrace" function

Syntax.

·ch· Channel number.

V.[2].MPG.RETRACAC	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

(V.)[ch].MPG.NRETBLK

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Maximum number of blocks allowed for the "retrace" function.

Syntax.

·ch· Channel number.

V.[2].MPG.NRETBLK	Channel ·2·.

(V.)[ch].MPG.RETMFUNC

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Treatment of the M functions with the "retrace" function.

This parameter sets the behavior of the "retrace" function when executing M functions. When the CNC finds an M function, it can either ignore it and keep executing blocks in retrace or cancel the "retrace" function.

Syntax.

·ch· Channel number.

V.[2].MPG.RETMFUNC	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Ignore the M function and go on.
1	Turn the retrace function off.

Remarks.

This parameter does not affect the following "M" functions.

- Functions M00 and M01 are always executed; they are sent to the PLC and [CYCLE START] must be pressed to resume execution in retrace.
- Functions M03 and M04 are always ignored; the CNC does not start the spindle nor does it change its turning direction.
- Function M05 cancels the "retrace" function; the CNC does not stop the spindle.

MASTER SPINDLE.

(V.)[ch].MPG.MASTERSPDL

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Master spindle maintained.

This parameter indicates whether the master spindle of a channel maintains its master condition or not after executing M02, M30 or after an emergency or reset or restarting the CNC.

Syntax.

·ch· Channel number.

V.[2].MPG.MASTERSPDL	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Temporary.
1	Maintained.

10.6 Variables related to axis and spindle machine parameters.

BELONGING TO THE CHANNEL.

(V.)[ch].MPA.AXISEXCH.xn
(V.)[ch].MPA.AXISEXCH.sn
(V.)[ch].SP.AXISEXCH.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Channel change permission.

This variable shows whether it is possible for the axis or spindle to change channels via part-program and, if so, whether the change is temporary or permanent; in other words, whether the change is maintained after an M02, M30 or a reset.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.AXISEXCH.Z	Z axis.
V.MPA.AXISEXCH.S	Spindle S.
V.SP.AXISEXCH.S	Spindle S.
V.SP.AXISEXCH	Master spindle.
V.MPA.AXISEXCH.4	Axis or spindle with logic number ·4·.
V.[2].MPA.AXISEXCH.1	Axis with index ·1· in the channel ·2·.
V.SP.AXISEXCH.2	Spindle with index ·2· in the system.
V.[2].SP.AXISEXCH.1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	It is not possible to change the axis or the spindle of the channel.
1	The change is temporary.
2	The change is permanent.

TYPE OF AXIS AND DRIVE.

(V.)[ch].MPA.AXISTYPE.xn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary and linear axes.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Type of axis.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.MPA.AXISTYPE.Z	Z axis.
V.MPA.AXISTYPE.4	Axis with logic number ·4·.
V.[2].MPA.AXISTYPE.1	Axis with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
1	Linear axis.
2	Rotary axis.

(V.)[ch].MPA.DRIVETYPE.xn
(V.)[ch].MPA.DRIVETYPE.sn
(V.)[ch].SP.DRIVETYPE.sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for an analog, position-Sercos, velocity-Sercos and Mechatrolink drive.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Type of drive.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.DRIVETYPE.Z	Z axis.
V.MPA.DRIVETYPE.S	Spindle S.
V.SP.DRIVETYPE.S	Spindle S.
V.SP.DRIVETYPE	Master spindle.
V.MPA.DRIVETYPE.4	Axis or spindle with logic number ·4·.
V.[2].MPA.DRIVETYPE.1	Axis with index ·1· in the channel ·2·.
V.SP.DRIVETYPE.2	Spindle with index ·2· in the system.
V.[2].SP.DRIVETYPE.1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
1	Analog drive.
2	Sercos drive.
16	Simulated drive.
32	Mechatrolink drive.

(V.)[ch].MPA.POSUNITS.xn
(V.)[ch].MPA.POSUNITS.sn
(V.)[ch].SP.POSUNITS.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for a position-Sercos, velocity-Servos and Mechatrolink drive.
The variable returns the value of block preparation.

Channel [ch]. Units systgen used by the feedback system.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.POSUNITS.Z	Z axis.
V.MPA.POSUNITS.S	Spindle S.
V.SP.POSUNITS.S	Spindle S.
V.SP.POSUNITS	Master spindle.
V.MPA.POSUNITS.4	Axis or spindle with logic number ·4·.
V.[2].MPA.POSUNITS.1	Axis with index ·1· in the channel ·2·.
V.SP.POSUNITS.2	Spindle with index ·2· in the system.
V.[2].SP.POSUNITS.1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Measurement (mm or degrees).
1	Pulses.

CONFIGURING A SERCOS DRIVE.

(V.)[ch].MPA.DRIVEID.xn
(V.)[ch].MPA.DRIVEID.sn
(V.)[ch].SP.DRIVEID.sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for a position-Sercos, velocity-Servos and Mechatrolink drive.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Address (node) of the drive.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.DRIVEID.Z	Z axis.
V.MPA.DRIVEID.S	Spindle S.
V.SP.DRIVEID.S	Spindle S.
V.SP.DRIVEID	Master spindle.
V.MPA.DRIVEID.4	Axis or spindle with logic number ·4·.
V.[2].MPA.DRIVEID.1	Axis with index ·1· in the channel ·2·.
V.SP.DRIVEID.2	Spindle with index ·2· in the system.
V.[2].SP.DRIVEID.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.OPMODEP.xn
(V.)[ch].MPA.OPMODEP.sn
(V.)[ch].SP.OPMODEP.sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for a position-Sercos, velocity-Servos and Mechatrolink drive.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Main operating mode of the Sercos drive or Mechatrolink device.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.OPMODEP.Z	Z axis.
V.MPA.OPMODEP.S	Spindle S.
V.SP.OPMODEP.S	Spindle S.
V.SP.OPMODEP	Master spindle.
V.MPA.OPMODEP.4	Axis or spindle with logic number ·4·.
V.[2].MPA.OPMODEP.1	Axis with index ·1· in the channel ·2·.
V.SP.OPMODEP.2	Spindle with index ·2· in the system.
V.[2].SP.OPMODEP.1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Sercos drive or Mechatrolink servo. Position command.
1	Sercos drive or Mechatrolink servo. Velocity command.
2	Inverter Mechatrolink.

(V.)[ch].MPA.FBACKSRC.xn
(V.)[ch].MPA.FBACKSRC.sn
(V.)[ch].SP.FBACKSRC.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for position-Sercos drives and velocity-Sercos drives.
The variable returns the value of block preparation.

Channel [ch]. Type of feedback.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.FBACKSRC.Z	Z axis.
V.MPA.FBACKSRC.S	Spindle S.
V.SP.FBACKSRC.S	Spindle S.
V.SP.FBACKSRC	Master spindle.
V.MPA.FBACKSRC.4	Axis or spindle with logic number ·4·.
V.[2].MPA.FBACKSRC.1	Axis with index ·1· in the channel ·2·.
V.SP.FBACKSRC.2	Spindle with index ·2· in the system.
V.[2].SP.FBACKSRC.1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Internal feedback (motor feedback).
1	External feedback (direct feedback).
2	Combined feedback (internal + external).

(V.)[ch].MPA.FBACKDIFF.xn
(V.)[ch].MPA.FBACKDIFF.sn
(V.)[ch].SP.FBACKDIFF.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for position Sercos drives.
The variable returns the value of block preparation.

Channel [ch]. Maximum difference between feedbacks.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.FBACKDIFF.Z	Z axis.
V.MPA.FBACKDIFF.S	Spindle S.
V.SP.FBACKDIFF.S	Spindle S.
V.SP.FBACKDIFF	Master spindle.
V.MPA.FBACKDIFF.4	Axis or spindle with logic number ·4·.
V.[2].MPA.FBACKDIFF.1	Axis with index ·1· in the channel ·2·.
V.SP.FBACKDIFF.2	Spindle with index ·2· in the system.
V.[2].SP.FBACKDIFF.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.FBMIXTIME.xn
(V.)[ch].MPA.FBMIXTIME.sn
(V.)[ch].SP.FBMIXTIME.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for position Sercos drives.
The variable returns the value of block preparation.

Channel [ch]. Time constant for combined feedback.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.FBMIXTIME.Z	Z axis.
V.MPA.FBMIXTIME.S	Spindle S.
V.SP.FBMIXTIME.S	Spindle S.
V.SP.FBMIXTIME	Master spindle.
V.MPA.FBMIXTIME.4	Axis or spindle with logic number ·4·.
V.[2].MPA.FBMIXTIME.1	Axis with index ·1· in the channel ·2·.
V.SP.FBMIXTIME.2	Spindle with index ·2· in the system.
V.[2].SP.FBMIXTIME.1	Spindle with index ·1· in the channel ·2·.

HIRTH AXIS CONFIGURATION.

(V.)[ch].MPA.HIRTH.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Hirth toothed axis.

A Hirth axis is the one that can only be positioned at positions multiple of a given value.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.MPA.HIRTH.Z	Z axis.
V.MPA.HIRTH.4	Axis with logic number ·4·.
V.[2].MPA.HIRTH.1	Axis with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

(V.)[ch].MPA.HPITCH.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Hirth axis pitch.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.MPA.HPITCH.Z	Z axis.
V.MPA.HPITCH.4	Axis with logic number ·4·.
V.[2].MPA.HPITCH.1	Axis with index ·1· in the channel ·2·.

AXIS CONFIGURATION FOR LATHE TYPE MACHINES.

(V.)[ch].MPA.FACEAXIS.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for linear axes.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Cross axis on a lathe.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.MPA.FACEAXIS.Z	Z axis.
V.MPA.FACEAXIS.4	Axis with logic number ·4·.
V.[2].MPA.FACEAXIS.1	Axis with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

(V.)[ch].MPA.LONGAXIS.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for linear axes.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Longitudinal axis on a lathe.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.MPA.LONGAXIS.Z	Z axis.
V.MPA.LONGAXIS.4	Axis with logic number ·4·.
V.[2].MPA.LONGAXIS.1	Axis with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

CONFIGURATION OF THE ROTARY AXES.

(V.)[ch].MPA.AXISMODE.xn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary axes.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Working method with the rotary axis.

This variable indicates how the rotary axis will behave in relation to the number of turns and position display.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.MPA.AXISMODE.Z	Z axis.
V.MPA.AXISMODE.4	Axis with logic number ·4·.
V.[2].MPA.AXISMODE.1	Axis with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	"Module" type rotary axis.
1	"Linearlike" type rotary axis.

(V.)[ch].MPA.UNIDIR.xn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary axes.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Unidirectional rotation.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.MPA.UNIDIR.Z	Z axis.
V.MPA.UNIDIR.4	Axis with logic number ·4·.
V.[2].MPA.UNIDIR.1	Axis with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

(V.)[ch].MPA.SHORTESTWAY.xn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary axes.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Positioning via shortest way.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.MPA.SHORTESTWAY.Z	Z axis.
V.MPA.SHORTESTWAY.4	Axis with logic number ·4·.
V.[2].MPA.SHORTESTWAY.1	Axis with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

MODULE CONFIGURATION (ROTARY AXES AND SPINDLE).

(V.)[ch].MPA.MODCOMP.xn
(V.)[ch].MPA.MODCOMP.sn
(V.)[ch].SP.MODCOMP.sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary axes and spindles.
Variable valid for analog drives and velocity-Sercos drives.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Module compensation.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.MODCOMP.Z	Z axis.
V.MPA.MODCOMP.S	Spindle S.
V.SP.MODCOMP.S	Spindle S.
V.SP.MODCOMP	Master spindle.
V.MPA.MODCOMP.4	Axis or spindle with logic number ·4·.
V.[2].MPA.MODCOMP.1	Axis with index ·1· in the channel ·2·.
V.SP.MODCOMP.2	Spindle with index ·2· in the system.
V.[2].SP.MODCOMP.1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

CONFIGURATION OF THE C AXIS.

(V.)[ch].MPA.CAXIS.xn
(V.)[ch].MPA.CAXIS.sn
(V.)[ch].SP.CAXIS.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary axes and spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Possibility of working as C axis.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.CAXIS.Z	Z axis.
V.MPA.CAXIS.S	Spindle S.
V.SP.CAXIS.S	Spindle S.
V.SP.CAXIS	Master spindle.
V.MPA.CAXIS.4	Axis or spindle with logic number ·4·.

V.[2].MPA.CAXIS.1	Axis with index ·1· in the channel ·2·.
V.SP.CAXIS.2	Spindle with index ·2· in the system.
V.[2].SP.CAXIS.1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

(V.)[ch].MPA.CAXSET.xn
(V.)[ch].MPA.CAXSET.sn
(V.)[ch].SP.CAXSET.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary axes and spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Set of parameters for working as C axis.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.CAXSET.Z	Z axis.
V.MPA.CAXSET.S	Spindle S.
V.SP.CAXSET.S	Spindle S.
V.SP.CAXSET	Master spindle.
V.MPA.CAXSET.4	Axis or spindle with logic number ·4·.
V.[2].MPA.CAXSET.1	Axis with index ·1· in the channel ·2·.
V.SP.CAXSET.2	Spindle with index ·2· in the system.
V.[2].SP.CAXSET.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.PERCAX.xn
(V.)[ch].MPA.PERCAX.sn
(V.)[ch].SP.PERCAX.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary axes and spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. C axis kept active after end of program, emergency or reset.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.PERCAX.Z	Z axis.
V.MPA.PERCAX.S	Spindle S.
V.SP.PERCAX.S	Spindle S.
V.SP.PERCAX	Master spindle.
V.MPA.PERCAX.4	Axis or spindle with logic number ·4·.
V.[2].MPA.PERCAX.1	Axis with index ·1· in the channel ·2·.
V.SP.PERCAX.2	Spindle with index ·2· in the system.
V.[2].SP.PERCAX.1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

CONFIGURATION OF THE SPINDLE.

(V.)[ch].MPA.AUTOGEAR.sn
(V.)[ch].SP.AUTOGEAR.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Automatic gear change.

Syntax.

·ch· Channel number.

·sn· Name, logic number or index of the spindle.

V.MPA.AUTOGEAR.S	Spindle S.
V.SP.AUTOGEAR.S	Spindle S.
V.SP.AUTOGEAR	Master spindle.
V.MPA.AUTOGEAR.4	Spindle with logic number ·4·.
V.SP.AUTOGEAR.2	Spindle with index ·2· in the system.
V.[2].SP.AUTOGEAR.1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

(V.)[ch].MPA.LOSPDLIM.sn
(V.)[ch].SP.LOSPDLIM.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Lower "rpm OK" percentage.

Syntax.

·ch· Channel number.

·sn· Name, logic number or index of the spindle.

V.MPA.LOSPDLIM.S	Spindle S.
V.SP.LOSPDLIM.S	Spindle S.
V.SP.LOSPDLIM	Master spindle.
V.MPA.LOSPDLIM.4	Spindle with logic number ·4·.
V.SP.LOSPDLIM.2	Spindle with index ·2· in the system.
V.[2].SP.LOSPDLIM.1	Spindle with index ·1· in the channel ·2·.

Remarks.

The reading from the PLC will be given in tenths (x10); i.e. if the parameter value is ·10·, the PLC reading will return a value of ·100·.

(V.)[ch].MPA.UPSPDLIM.sn
(V.)[ch].SP.UPSPDLIM.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Upper "rpm OK" percentage.

Syntax.

·ch· Channel number.

·sn· Name, logic number or index of the spindle.

V.MPA.UPSPDLIM.S	Spindle S.
V.SP.UPSPDLIM.S	Spindle S.
V.SP.UPSPDLIM	Master spindle.
V.MPA.UPSPDLIM.4	Spindle with logic number ·4·.
V.SP.UPSPDLIM.2	Spindle with index ·2· in the system.
V.[2].SP.UPSPDLIM.1	Spindle with index ·1· in the channel ·2·.

Remarks.

The reading from the PLC will be given in tenths (x10); i.e. if the parameter value is ·10·, the PLC reading will return a value of ·100·.

(V.)[ch].MPA.SPDLTIME.sn
(V.)[ch].SP.SPDLTIME.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Estimated time for an S function.

Syntax.

·ch· Channel number.

·sn· Name, logic number or index of the spindle.

V.MPA.SPDLTIME.S	Spindle S.
V.SP.SPDLTIME.S	Spindle S.
V.SP.SPDLTIME	Master spindle.
V.MPA.SPDLTIME.4	Spindle with logic number ·4·.
V.SP.SPDLTIME.2	Spindle with index ·2· in the system.
V.[2].SP.SPDLTIME.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.SPDLSTOP.sn
(V.)[ch].SP.SPDLSTOP.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Functions M2 and M30, an error or a reset stop the spindle.

Syntax.

·ch· Channel number.

·sn· Name, logic number or index of the spindle.

V.MPA.SPDLSTOP.S	Spindle S.
V.SP.SPDLSTOP.S	Spindle S.
V.SP.SPDLSTOP	Master spindle.
V.MPA.SPDLSTOP.4	Spindle with logic number ·4·.
V.SP.SPDLSTOP.2	Spindle with index ·2· in the system.
V.[2].SP.SPDLSTOP.1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

(V.)[ch].MPA.SREVM05.sn
(V.)[ch].SP.SREVM05.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. With G84, the spindle must be stopped to change the turning direction.

Syntax.

·ch· Channel number.

·sn· Name, logic number or index of the spindle.

V.MPA.SREVM05.S	Spindle S.
V.SP.SREVM05.S	Spindle S.
V.SP.SREVM05	Master spindle.
V.MPA.SREVM05.4	Spindle with logic number ·4·.
V.SP.SREVM05.2	Spindle with index ·2· in the system.
V.[2].SP.SREVM05.1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

(V.)[ch].MPA.M19SPDLEREV.sn
(V.)[ch].SP.M19SPDLEREV.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for spindles.
Variable valid for analog drives.
The variable returns the value of block preparation.

Channel [ch]. The SPDLEREV mark (reverse turning direction) affects the spindle in M19.

Syntax.

·ch· Channel number.

·sn· Name, logic number or index of the spindle.

V.MPA.M19SPDLEREV.S	Spindle S.
V.SP.M19SPDLEREV.S	Spindle S.
V.SP.M19SPDLEREV	Master spindle.
V.MPA.M19SPDLEREV.4	Spindle with logic number ·4·.
V.SP.M19SPDLEREV.2	Spindle with index ·2· in the system.
V.[2].SP.M19SPDLEREV.1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

(V.)[ch].MPA.STEPOVR.sn
(V.)[ch].SP.STEPOVR.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Spindle override step.

Syntax.

·ch· Channel number.

·sn· Name, logic number or index of the spindle.

V.MPA.STEPOVR.S	Spindle S.
V.SP.STEPOVR.S	Spindle S.
V.SP.STEPOVR	Master spindle.
V.MPA.STEPOVR.4	Spindle with logic number ·4·.
V.SP.STEPOVR.2	Spindle with index ·2· in the system.
V.[2].SP.STEPOVR.1	Spindle with index ·1· in the channel ·2·.

Remarks.

The reading from the PLC will be given in tenths (x10); i.e. if the parameter value is ·10·, the PLC reading will return a value of ·100·.

(V.)[ch].MPA.MINOVR.sn
(V.)[ch].SP.MINOVR.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Minimum override (%) permitted for the spindle.

Syntax.

·ch· Channel number.

·sn· Name, logic number or index of the spindle.

V.MPA.MINOVR.S	Spindle S.
V.SP.MINOVR.S	Spindle S.
V.SP.MINOVR	Master spindle.

V.MPA.MINOVR.4	Spindle with logic number ·4·.
V.SP.MINOVR.2	Spindle with index ·2· in the system.
V.[2].SP.MINOVR.1	Spindle with index ·1· in the channel ·2·.

Remarks.

The reading from the PLC will be given in tenths (x10); i.e. if the parameter value is ·10·, the PLC reading will return a value of ·100·.

(V.)[ch].MPA.MAXOVR.sn
(V.)[ch].SP.MAXOVR.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Maximum override (%) permitted for the spindle.

Syntax.

·ch· Channel number.

·sn· Name, logic number or index of the spindle.

V.MPA.MAXOVR.S	Spindle S.
V.SP.MAXOVR.S	Spindle S.
V.SP.MAXOVR	Master spindle.
V.MPA.MAXOVR.4	Spindle with logic number ·4·.
V.SP.MAXOVR.2	Spindle with index ·2· in the system.
V.[2].SP.MAXOVR.1	Spindle with index ·1· in the channel ·2·.

Remarks.

The reading from the PLC will be given in tenths (x10); i.e. if the parameter value is ·10·, the PLC reading will return a value of ·100·.

SYNCHRONIZATION OF AXES AND SPINDLES.

(V.)[ch].MPA.SYNCSET.xn
(V.)[ch].SP.SYNCSET.sn
(V.)[ch].SP.SYNCSET.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary axes and spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Set of parameters for synchronization.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.SYNCSET.Z	Z axis.
V.MPA.SYNCSET.S	Spindle S.
V.SP.SYNCSET.S	Spindle S.
V.SP.SYNCSET	Master spindle.
V.MPA.SYNCSET.4	Axis or spindle with logic number ·4·.
V.[2].MPA.SYNCSET.1	Axis with index ·1· in the channel ·2·.
V.SP.SYNCSET.2	Spindle with index ·2· in the system.
V.[2].SP.SYNCSET.1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Do not force the parameter set.
1 to 4	Parameter set.

(V.)[ch].MPA.DSYNCVELW.xn
(V.)[ch].SP.DSYNCVELW.sn
(V.)[ch].SP.DSYNCVELW.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Velocity synchronization window.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.DSYNCVELW.Z	Z axis.
V.MPA.DSYNCVELW.S	Spindle S.
V.SP.DSYNCVELW.S	Spindle S.
V.SP.DSYNCVELW	Master spindle.
V.MPA.DSYNCVELW.4	Axis or spindle with logic number ·4·.
V.[2].MPA.DSYNCVELW.1	Axis with index ·1· in the channel ·2·.
V.SP.DSYNCVELW.2	Spindle with index ·2· in the system.
V.[2].SP.DSYNCVELW.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.DSYNCPOSW.xn
(V.)[ch].SP.DSYNCPOSW.sn
(V.)[ch].SP.DSYNCPOSW.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Position synchronization window.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.DSYNCPOSW.Z	Z axis.
V.MPA.DSYNCPOSW.S	Spindle S.
V.SP.DSYNCPOSW.S	Spindle S.
V.SP.DSYNCPOSW	Master spindle.
V.MPA.DSYNCPOSW.4	Axis or spindle with logic number ·4·.
V.[2].MPA.DSYNCPOSW.1	Axis with index ·1· in the channel ·2·.
V.SP.DSYNCPOSW.2	Spindle with index ·2· in the system.
V.[2].SP.DSYNCPOSW.1	Spindle with index ·1· in the channel ·2·.

SOFTWARE AXIS LIMITS.

(V.)[ch].MPA.POSLIMIT.xn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary and linear axes.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Positive software limit.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.MPA.POSLIMIT.Z	Z axis.
V.MPA.POSLIMIT.4	Axis with logic number ·4·.
V.[2].MPA.POSLIMIT.1	Axis with index ·1· in the channel ·2·.

(V.)[ch].MPA.NEGLIMIT.xn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary and linear axes.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Negative software limit.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.MPA.NEGLIMIT.Z	Z axis.
V.MPA.NEGLIMIT.4	Axis with logic number ·4·.
V.[2].MPA.NEGLIMIT.1	Axis with index ·1· in the channel ·2·.

(V.)[ch].MPA.SWLIMITTOL.xn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary and linear axes.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Software limit tolerance.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.MPA.SWLIMITTOL.Z	Z axis.
V.MPA.SWLIMITTOL.4	Axis with logic number ·4·.
V.[2].MPA.SWLIMITTOL.1	Axis with index ·1· in the channel ·2·.

OVERRIDE CHANGE WHILE THREADING.

(V.)[ch].MPA.THREADOVR.sn
(V.)[ch].SP.THREADOVR.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Maximum variation allowed for the override while threading.

Syntax.

·ch· Channel number.

·sn· Name, logic number or index of the spindle.

V.MPA.THREADOVR.S	Spindle S.
V.SP.THREADOVR.S	Spindle S.
V.SP.THREADOVR	Master spindle.
V.MPA.THREADOVR.4	Spindle with logic number ·4·.
V.SP.THREADOVR.2	Spindle with index ·2· in the system.
V.[2].SP.THREADOVR.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.OVRFILTER.sn
(V.)[ch].SP.OVRFILTER.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Time to make the override change effective.

Syntax.

·ch· Channel number.

·sn· Name, logic number or index of the spindle.

V.MPA.OVRFILTER.S	Spindle S.
V.SP.OVRFILTER.S	Spindle S.
V.SP.OVRFILTER	Master spindle.
V.MPA.OVRFILTER.4	Spindle with logic number ·4·.
V.SP.OVRFILTER.2	Spindle with index ·2· in the system.
V.[2].SP.OVRFILTER.1	Spindle with index ·1· in the channel ·2·.

RUNAWAY PROTECTION AND TENDENCY TEST.

(V.)[ch].MPA.TENDENCY.xn
(V.)[ch].SP.TENDENCY.sn
(V.)[ch].SP.TENDENCY.sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drives.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Activation of tendency test.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.TENDENCY.Z	Z axis.
V.MPA.TENDENCY.S	Spindle S.
V.SP.TENDENCY.S	Spindle S.
V.SP.TENDENCY	Master spindle.
V.MPA.TENDENCY.4	Axis or spindle with logic number ·4·.
V.[2].MPA.TENDENCY.1	Axis with index ·1· in the channel ·2·.
V.SP.TENDENCY.2	Spindle with index ·2· in the system.
V.[2].SP.TENDENCY.1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

(V.)[ch].MPA.TENDTIME.xn
(V.)[ch].SP.TENDTIME.sn
(V.)[ch].SP.TENDTIME.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drives.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Time to detect axis runaway.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.TENDTIME.Z	Z axis.
V.MPA.TENDTIME.S	Spindle S.
V.SP.TENDTIME.S	Spindle S.
V.SP.TENDTIME	Master spindle.
V.MPA.TENDTIME.4	Axis or spindle with logic number ·4·.
V.[2].MPA.TENDTIME.1	Axis with index ·1· in the channel ·2·.
V.SP.TENDTIME.2	Spindle with index ·2· in the system.
V.[2].SP.TENDTIME.1	Spindle with index ·1· in the channel ·2·.

PLC OFFSET.

(V.)[ch].MPA.PLCOINC.xn
(V.)[ch].MPA.PLCOINC.sn
(V.)[ch].SP.PLCOINC.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. PLC offset increment per cycle.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.PLCOINC.Z	Z axis.
V.MPA.PLCOINC.S	Spindle S.
V.SP.PLCOINC.S	Spindle S.
V.SP.PLCOINC	Master spindle.
V.MPA.PLCOINC.4	Axis or spindle with logic number ·4·.
V.[2].MPA.PLCOINC.1	Axis with index ·1· in the channel ·2·.
V.SP.PLCOINC.2	Spindle with index ·2· in the system.
V.[2].SP.PLCOINC.1	Spindle with index ·1· in the channel ·2·.

DWELL FOR DEAD AXES.

(V.)[ch].MPA.DWELL.xn
(V.)[ch].MPA.DWELL.sn
(V.)[ch].SP.DWELL.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Dwell for dead axes.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.DWELL.Z	Z axis.
V.MPA.DWELL.S	Spindle S.
V.SP.DWELL.S	Spindle S.
V.SP.DWELL	Master spindle.
V.MPA.DWELL.4	Axis or spindle with logic number ·4·.
V.[2].MPA.DWELL.1	Axis with index ·1· in the channel ·2·.
V.SP.DWELL.2	Spindle with index ·2· in the system.
V.[2].SP.DWELL.1	Spindle with index ·1· in the channel ·2·.

PROGRAMMING IN RADIUS OR DIAMETERS.

(V.)[ch].MPA.DIAMPROG.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for linear axes.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Programming in diameters.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.MPA.DIAMPROG.Z	Z axis.
V.MPA.DIAMPROG.4	Axis with logic number ·4·.
V.[2].MPA.DIAMPROG.1	Axis with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

HOME SEARCH.

(V.)[ch].MPA.REFDIREC.xn
(V.)[ch].MPA.REFDIREC.sn
(V.)[ch].SP.REFDIREC.sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Direction of the search.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.REFDIREC.Z	Z axis.
V.MPA.REFDIREC.S	Spindle S.
V.SP.REFDIREC.S	Spindle S.
V.SP.REFDIREC	Master spindle.
V.MPA.REFDIREC.4	Axis or spindle with logic number ·4·.
V.[2].MPA.REFDIREC.1	Axis with index ·1· in the channel ·2·.
V.SP.REFDIREC.2	Spindle with index ·2· in the system.
V.[2].SP.REFDIREC.1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Movement in the negative direction.
1	Movement in the positive direction.

(V.)[ch].MPA.DECINPUT.xn
(V.)[ch].MPA.DECINPUT.sn
(V.)[ch].SP.DECINPUT.sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. The axis/spindle has a home switch.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.DECINPUT.Z	Z axis.
V.MPA.DECINPUT.S	Spindle S.
V.SP.DECINPUT.S	Spindle S.
V.SP.DECINPUT	Master spindle.
V.MPA.DECINPUT.4	Axis or spindle with logic number ·4·.
V.[2].MPA.DECINPUT.1	Axis with index ·1· in the channel ·2·.
V.SP.DECINPUT.2	Spindle with index ·2· in the system.
V.[2].SP.DECINPUT.1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

(V.)[ch].MPA.REFINI.sn
(V.)[ch].SP.REFINI.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Home search in the first movement

Syntax.

·ch· Channel number.

·sn· Name, logic number or index of the spindle.

V.MPA.REFINI.S	Spindle S.
V.SP.REFINI.S	Spindle S.
V.SP.REFINI	Master spindle.
V.MPA.REFINI.4	Spindle with logic number ·4·.
V.SP.REFINI.2	Spindle with index ·2· in the system.
V.[2].SP.REFINI.1	Spindle with index ·1· in the channel ·2·.

CONFIGURATION OF THE PROBING MOVEMENT.

(V.)[ch].MPA.PROBEAXIS.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. The axis can participate in probing movements.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.MPA.PROBEAXIS.Z	Z axis.
V.MPA.PROBEAXIS.4	Axis with logic number ·4·.
V.[2].MPA.PROBEAXIS.1	Axis with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

(V.)[ch].MPA.PROBERANGE.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Maximum braking distance.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.MPA.PROBERANGE.Z	Z axis.
V.MPA.PROBERANGE.4	Axis with logic number ·4·.
V.[2].MPA.PROBERANGE.1	Axis with index ·1· in the channel ·2·.

(V.)[ch].MPA.PROBEFEED.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Maximum probing feedrate.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.MPA.PROBEFEED.Z	Z axis.
V.MPA.PROBEFEED.4	Axis with logic number ·4·.
V.[2].MPA.PROBEFEED.1	Axis with index ·1· in the channel ·2·.

(V.)[ch].MPA.PROBEDELAY.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Delay for the "probe 1" signal.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.MPA.PROBEDELAY.Z	Z axis.
V.MPA.PROBEDELAY.4	Axis with logic number ·4·.
V.[2].MPA.PROBEDELAY.1	Axis with index ·1· in the channel ·2·.

(V.)[ch].MPA.PROBEDELAY2.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Delay for the "probe 2" signal.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.MPA.PROBEDELAY2.Z	Z axis.
V.MPA.PROBEDELAY2.4	Axis with logic number ·4·.
V.[2].MPA.PROBEDELAY2.1	Axis with index ·1· in the channel ·2·.

REPOSITIONING OF THE AXES IN TOOL INSPECTION.

(V.)[ch].MPA.REPOSFEED.xn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary and linear axes.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Maximum repositioning feedrate.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.MPA.REPOSFEED.Z	Z axis.
V.MPA.REPOSFEED.4	Axis with logic number ·4·.
V.[2].MPA.REPOSFEED.1	Axis with index ·1· in the channel ·2·.

CONFIGURATION OF THE INDEPENDENT AXIS.

(V.)[ch].MPA.POSFEED.xn
(V.)[ch].MPA.POSFEED.sn
(V.)[ch].SP.POSFEED.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Positioning feedrate

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.POSFEED.Z	Z axis.
V.MPA.POSFEED.S	Spindle S.
V.SP.POSFEED.S	Spindle S.
V.SP.POSFEED	Master spindle.
V.MPA.POSFEED.4	Axis or spindle with logic number ·4·.

V.[2].MPA.POSFEED.1	Axis with index ·1· in the channel ·2·.
V.SP.POSFEED.2	Spindle with index ·2· in the system.
V.[2].SP.POSFEED.1	Spindle with index ·1· in the channel ·2·.

CONFIGURE THE MAXIMUM SAFETY LIMIT FOR THE FEEDRATE AND FOR THE SPEED.

(V.)[ch].MPA.FLIMIT.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Maximum safety limit for axis feedrate.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.MPA.FLIMIT.Z	Z axis.
V.MPA.FLIMIT.4	Axis with logic number ·4·.
V.[2].MPA.FLIMIT.1	Axis with index ·1· in the channel ·2·.

(V.)[ch].MPA.SLIMIT.sn
(V.)[ch].SP.SLIMIT.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Maximum safety limit for spindle speed.

Syntax.

·ch· Channel number.

·sn· Name, logic number or index of the spindle.

V.MPA.SLIMIT.S	Spindle S.
V.SP.SLIMIT.S	Spindle S.
V.SP.SLIMIT	Master spindle.
V.MPA.SLIMIT.4	Spindle with logic number ·4·.
V.SP.SLIMIT.2	Spindle with index ·2· in the system.
V.[2].SP.SLIMIT.1	Spindle with index ·1· in the channel ·2·.

JOG MODE. CONTINUOUS JOG.

(V.)[ch].MPA.JOGFEED.xn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary and linear axes.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Feedrate in continuous jog.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.MPA.JOGFEED.Z	Z axis.
V.MPA.JOGFEED.4	Axis with logic number ·4·.
V.[2].MPA.JOGFEED.1	Axis with index ·1· in the channel ·2·.

(V.)[ch].MPA.JOGRAPFEED.xn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary and linear axes.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Rapid feedrate in continuous jog.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.MPA.JOGRAPFEED.Z	Z axis.
V.MPA.JOGRAPFEED.4	Axis with logic number ·4·.
V.[2].MPA.JOGRAPFEED.1	Axis with index ·1· in the channel ·2·.

(V.)[ch].MPA.MAXMANFEED.xn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary and linear axes.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Maximum feedrate in continuous jog.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.MPA.MAXMANFEED.Z	Z axis.
V.MPA.MAXMANFEED.4	Axis with logic number ·4·.
V.[2].MPA.MAXMANFEED.1	Axis with index ·1· in the channel ·2·.

(V.)[ch].MPA.MAXMANACC.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Maximum acceleration in continuous JOG.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.MPA.MAXMANACC.Z	Z axis.
V.MPA.MAXMANACC.4	Axis with logic number ·4·.
V.[2].MPA.MAXMANACC.1	Axis with index ·1· in the channel ·2·.

JOG MODE. INCREMENTAL JOG.

(V.)[ch].MPA.INCJOGDIST[pos].xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Distance to move in incremental jog, [pos] dial position.

pos=1 for position ·1·, pos=2 for position ·10· and so on.

Syntax.

·ch· Channel number.

·pos· Position of the switch on the operator panel.

·xn· Name, logic number or index of the axis..

V.MPA.INCJOGDIST[4].Z	Z axis.
V.MPA.INCJOGDIST[4].4	Axis with logic number ·4·.
V.[2].MPA.INCJOGDIST[4].1	Axis with index ·1· in the channel ·2·.

(V.)[ch].MPA.INCJOGFEED[pos].xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Feedrate in incremental jog, [pos] dial position.

pos=1 for position ·1·, pos=2 for position ·10· and so on.

Syntax.

·ch· Channel number.

·pos· Position of the switch on the operator panel for incremental jog.

·xn· Name, logic number or index of the axis.

V.MPA.INCJOGFEED[4].Z	Z axis.
V.MPA.INCJOGFEED[4].4	Axis with logic number ·4·.
V.[2].MPA.INCJOGFEDD[4].1	Axis with index ·1· in the channel ·2·.

JOG MODE. HANDWHEELS.

(V.)[ch].MPA.MPGRESOL[pos].xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Resolution of the handwheel, in the [pos] position.

pos=1 for position ·1·, pos=2 for position ·10· and pos=3 for ·100·.

Syntax.

·ch· Channel number.

·pos· Position of the switch on the operator panel for the handwheel mode.

·xn· Name, logic number or index of the axis.

V.MPA.MPGRESOL[2].Z	Z axis.
V.MPA.MPGRESOL[2].4	Axis with logic number ·4·.
V.[2].MPA.MPGRESOL[2].1	Axis with index ·1· in the channel ·2·.

(V.)[ch].MPA.MPGFILTER.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Filter time for the handwheel.

Syntax.

·ch· Channel number.

·pos· Position of the switch on the operator panel for the handwheel mode.

·xn· Name, logic number or index of the axis.

V.MPA.MPGFILTER[2].Z	Z axis.
V.MPA.MPGFILTER[2].4	Axis with logic number ·4·.
V.[2].MPA.MPGFILTER[2].1	Axis with index ·1· in the channel ·2·.

JOG MODE. MANUAL INTERVENTION.

(V.)[ch].MPA.MANPOSSW.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Maximum positive travel with G201.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.MPA.MANPOSSW.Z	Z axis.
V.MPA.MANPOSSW.4	Axis with logic number ·4·.
V.[2].MPA.MANPOSSW.1	Axis with index ·1· in the channel ·2·.

(V.)[ch].MPA.MANNEGSW.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Maximum negative travel with G201.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.MPA.MANNEGSW.Z	Z axis.
V.MPA.MANNEGSW.4	Axis with logic number ·4·.
V.[2].MPA.MANNEGSW.1	Axis with index ·1· in the channel ·2·.

(V.)[ch].MPA.MANFEEDP.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Maximum % of jog feedrate in G201.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.MPA.MANFEEDP.Z	Z axis.
V.MPA.MANFEEDP.4	Axis with logic number ·4·.
V.[2].MPA.MANFEEDP.1	Axis with index ·1· in the channel ·2·.

(V.)[ch].MPA.IPOFEEDP.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Maximum % of execution feedrate in G201.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.MPA.IPOFEEDP.Z	Z axis.
V.MPA.IPOFEEDP.4	Axis with logic number ·4·.
V.[2].MPA.IPOFEEDP.1	Axis with index ·1· in the channel ·2·.

(V.)[ch].MPA.MANACCP.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Maximum % of jog acceleration in G201.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.MPA.MANACCP.Z	Z axis.
V.MPA.MANACCP.4	Axis with logic number ·4·.
V.[2].MPA.MANACCP.1	Axis with index ·1· in the channel ·2·.

(V.)[ch].MPA.IPOACCP.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Maximum % of execution acceleration in G201.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.MPA.IPOACCP.Z	Z axis.
V.MPA.IPOACCP.4	Axis with logic number ·4·.
V.[2].MPA.IPOACCP.1	Axis with index ·1· in the channel ·2·.

LEADSCREW ERROR COMPENSATION.

(V.)[ch].MPA.LSCRWCOMP.xn
(V.)[ch].MPA.LSCRWCOMP.sn
(V.)[ch].SP.LSCRWCOMP.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Leadscrew error compensation.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.LSCRWCOMP.Z	Z axis.
V.MPA.LSCRWCOMP.S	Spindle S.
V.SP.LSCRWCOMP.S	Spindle S.
V.SP.LSCRWCOMP	Master spindle.
V.MPA.LSCRWCOMP.4	Axis or spindle with logic number ·4·.
V.[2].MPA.LSCRWCOMP.1	Axis with index ·1· in the channel ·2·.
V.SP.LSCRWCOMP.2	Spindle with index ·2· in the system.
V.[2].SP.LSCRWCOMP.1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

(V.)[ch].MPA.NPOINTS.xn
(V.)[ch].MPA.NPOINTS.sn
(V.)[ch].SP.NPOINTS.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Number of points in the table.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.NPOINTS.Z	Z axis.
V.MPA.NPOINTS.S	Spindle S.
V.SP.NPOINTS.S	Spindle S.
V.SP.NPOINTS	Master spindle.
V.MPA.NPOINTS.4	Axis or spindle with logic number ·4·.
V.[2].MPA.NPOINTS.1	Axis with index ·1· in the channel ·2·.
V.SP.NPOINTS.2	Spindle with index ·2· in the system.
V.[2].SP.NPOINTS.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.TYPLSCRW.xn
(V.)[ch].MPA.TYPLSCRW.sn
(V.)[ch].SP.TYPLSCRW.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Type of compensation (type of coordinates).

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.TYPLSCRW.Z	Z axis.
V.MPA.TYPLSCRW.S	Spindle S.
V.SP.TYPLSCRW.S	Spindle S.
V.SP.TYPLSCRW	Master spindle.
V.MPA.TYPLSCRW.4	Axis or spindle with logic number ·4·.
V.[2].MPA.TYPLSCRW.1	Axis with index ·1· in the channel ·2·.
V.SP.TYPLSCRW.2	Spindle with index ·2· in the system.
V.[2].SP.TYPLSCRW.1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Compensation is applied using real coordinates.
1	Compensation is applied using theoretical coordinates.

(V.)[ch].MPA.BIDIR.xn
(V.)[ch].MPA.BIDIR.sn
(V.)[ch].SP.BIDIR.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Bi-directional compensation.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.BIDIR.Z	Z axis.
V.MPA.BIDIR.S	Spindle S.
V.SP.BIDIR.S	Spindle S.
V.SP.BIDIR	Master spindle.
V.MPA.BIDIR.4	Axis or spindle with logic number ·4·.
V.[2].MPA.BIDIR.1	Axis with index ·1· in the channel ·2·.
V.SP.BIDIR.2	Spindle with index ·2· in the system.
V.[2].SP.BIDIR.1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

(V.)[ch].MPA.REFNEED.xn
(V.)[ch].MPA.REFNEED.sn
(V.)[ch].SP.REFNEED.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. The axis must be homed in order to apply the compensation.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.REFNEED.Z	Z axis.
V.MPA.REFNEED.S	Spindle S.
V.SP.REFNEED.S	Spindle S.
V.SP.REFNEED	Master spindle.
V.MPA.REFNEED.4	Axis or spindle with logic number ·4·.
V.[2].MPA.REFNEED.1	Axis with index ·1· in the channel ·2·.
V.SP.REFNEED.2	Spindle with index ·2· in the system.
V.[2].SP.REFNEED.1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

(V.)[ch].MPA.POSITION[pt].xn
(V.)[ch].MPA.POSITION[pt].sn
(V.)[ch].SP.POSITION[pt].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Master axis position for point [pt]

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

·pt· Point in the table.

V.MPA.POSITION[13].Z	Z axis.
V.MPA.POSITION[13].S	Spindle S.
V.SP.POSITION[13].S	Spindle S.
V.SP.POSITION[13]	Master spindle.

V.MPA.POSITION[13].4	Axis or spindle with logic number ·4·.
V.[2].MPA.POSITION[13].1	Axis with index ·1· in the channel ·2·.
V.SP.POSITION[13].2	Spindle with index ·2· in the system.
V.[2].SP.POSITION[13].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.POSERROR[pt].xn
(V.)[ch].MPA.POSERROR[pt].sn
(V.)[ch].SP.POSERROR[pt].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Amount of error at point [pt] in the positive direction

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

·pt· Point in the table.

V.MPA.POSERROR[13].Z	Z axis.
V.MPA.POSERROR[13].S	Spindle S.
V.SP.POSERROR[13].S	Spindle S.
V.SP.POSERROR[13]	Master spindle.
V.MPA.POSERROR[13].4	Axis or spindle with logic number ·4·.
V.[2].MPA.POSERROR[13].1	Axis with index ·1· in the channel ·2·.
V.SP.POSERROR[13].2	Spindle with index ·2· in the system.
V.[2].SP.POSERROR[13].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.NEGERROR[pt].xn
(V.)[ch].MPA.NEGERROR[pt].sn
(V.)[ch].SP.NEGERROR[pt].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Amount of error at point [pt] in the negative direction.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

·pt· Point in the table.

V.MPA.NEGERROR[13].Z	Z axis.
V.MPA.NEGERROR[13].S	Spindle S.
V.SP.NEGERROR[13].S	Spindle S.
V.SP.NEGERROR[13]	Master spindle.
V.MPA.NEGERROR[13].4	Axis or spindle with logic number ·4·.
V.[2].MPA.NEGERROR[13].1	Axis with index ·1· in the channel ·2·.
V.SP.NEGERROR[13].2	Spindle with index ·2· in the system.
V.[2].SP.NEGERROR[13].1	Spindle with index ·1· in the channel ·2·.

FILTERS TO ELIMINATE FREQUENCIES.

(V.)[ch].MPA.ORDER[nb].xn
(V.)[ch].MPA.ORDER[nb].sn
(V.)[ch].SP.ORDER[nb].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Filter order.

Syntax.

·ch· Channel number.

·nb· Filter number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.ORDER[3].Z	Z axis.
V.MPA.ORDER[3].S	Spindle S.
V.SP.ORDER[3].S	Spindle S.
V.SP.ORDER[3]	Master spindle.
V.MPA.ORDER[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.ORDER[3].1	Axis with index ·1· in the channel ·2·.
V.SP.ORDER[3].2	Spindle with index ·2· in the system.
V.[2].SP.ORDER[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.TYPE[nb].xn
(V.)[ch].MPA.TYPE[nb].sn
(V.)[ch].SP.TYPE[nb].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Type of filter.

Syntax.

·ch· Channel number.

·nb· Filter number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.TYPE[3].Z	Z axis.
V.MPA.TYPE[3].S	Spindle S.
V.SP.TYPE[3].S	Spindle S.
V.SP.TYPE[3]	Master spindle.
V.MPA.TYPE[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.TYPE[3].1	Axis with index ·1· in the channel ·2·.
V.SP.TYPE[3].2	Spindle with index ·2· in the system.
V.[2].SP.TYPE[3].1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Low passing filter.
1	Anti-resonance Filter.
2	FAGOR low passing filter.

(V.)[ch].MPA.FREQUENCY[nb].xn
(V.)[ch].MPA.FREQUENCY[nb].sn
(V.)[ch].SP.FREQUENCY[nb].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Cutoff or center frequency

Syntax.

·ch· Channel number.

·nb· Filter number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.FREQUENCY[3].Z	Z axis.
V.MPA.FREQUENCY[3].S	Spindle S.
V.SP.FREQUENCY[3].S	Spindle S.
V.SP.FREQUENCY[3]	Master spindle.
V.MPA.FREQUENCY[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.FREQUENCY[3].1	Axis with index ·1· in the channel ·2·.
V.SP.FREQUENCY[3].2	Spindle with index ·2· in the system.
V.[2].SP.FREQUENCY[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.NORBWIDTH[nb].xn
(V.)[ch].MPA.NORBWIDTH[nb].sn
(V.)[ch].SP.NORBWIDTH[nb].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Normal bandwidth.

Syntax.

·ch· Channel number.

·nb· Filter number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.NORBWIDTH[3].Z	Z axis.
V.MPA.NORBWIDTH[3].S	Spindle S.
V.SP.NORBWIDTH[3].S	Spindle S.
V.SP.NORBWIDTH[3]	Master spindle.
V.MPA.NORBWIDTH[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.NORBWIDTH[3].1	Axis with index ·1· in the channel ·2·.
V.SP.NORBWIDTH[3].2	Spindle with index ·2· in the system.
V.[2].SP.NORBWIDTH[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.SHARE[nb].xn
(V.)[ch].MPA.SHARE[nb].sn
(V.)[ch].SP.SHARE[nb].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Percentage of signal going through the filter.

Syntax.

·ch· Channel number.

·nb· Filter number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.SHARE[3].Z	Z axis.
V.MPA.SHARE[3].S	Spindle S.
V.SP.SHARE[3].S	Spindle S.
V.SP.SHARE[3]	Master spindle.
V.MPA.SHARE[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.SHARE[3].1	Axis with index ·1· in the channel ·2·.
V.SP.SHARE[3].2	Spindle with index ·2· in the system.
V.[2].SP.SHARE[3].1	Spindle with index ·1· in the channel ·2·.

PARAMETER SETS.

(V.)[ch].MPA.NPARSETS.xn
(V.)[ch].MPA.NPARSETS.sn
(V.)[ch].SP.NPARSETS.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Number of parameter sets available.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.NPARSETS.Z	Z axis.
V.MPA.NPARSETS.S	Spindle S.
V.SP.NPARSETS.S	Spindle S.
V.SP.NPARSETS	Master spindle.
V.MPA.NPARSETS.4	Axis or spindle with logic number ·4·.
V.[2].MPA.NPARSETS.1	Axis with index ·1· in the channel ·2·.
V.SP.NPARSETS.2	Spindle with index ·2· in the system.
V.[2].SP.NPARSETS.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.DEFAULTSET.xn
(V.)[ch].MPA.DEFAULTSET.sn
(V.)[ch].SP.DEFAULTSET.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Default parameter set on power-up.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.DEFAULTSET.Z	Z axis.
V.MPA.DEFAULTSET.S	Spindle S.
V.SP.DEFAULTSET.S	Spindle S.
V.SP.DEFAULTSET	Master spindle.
V.MPA.DEFAULTSET.4	Axis or spindle with logic number ·4·.
V.[2].MPA.DEFAULTSET.1	Axis with index ·1· in the channel ·2·.
V.SP.DEFAULTSET.2	Spindle with index ·2· in the system.
V.[2].SP.DEFAULTSET.1	Spindle with index ·1· in the channel ·2·.

10.7 Variables related to the sets of machine parameters.

FEEDBACK RESOLUTION.

(V.)[ch].MPA.PITCH[set].xn
(V.)[ch].MPA.PITCH[set].sn
(V.)[ch].SP.PITCH[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Leadscrew pitch.

Depending on the type of feedback, this parameter means the following.

- On a linear axis with a rotary encoder and leadscrew, it defines the leadscrew pitch.
- On a linear axis with a linear encoder (scale), it defines the pitch of the scale.
- On a rotary axis, it sets the number of degrees per turn of the encoder.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.PITCH[3].Z	Z axis.
V.MPA.PITCH[3].S	Spindle S.
V.SP.PITCH[3].S	Spindle S.
V.SP.PITCH[3]	Master spindle.
V.MPA.PITCH[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.PITCH[3].1	Axis with index ·1· in the channel ·2·.
V.SP.PITCH[3].2	Spindle with index ·2· in the system.
V.[2].SP.PITCH[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.INPUTREV[set].xn
(V.)[ch].MPA.INPUTREV[set].sn
(V.)[ch].SP.INPUTREV[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Gear ratio; turns of the motor shaft.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.INPUTREV[3].Z	Z axis.
V.MPA.INPUTREV[3].S	Spindle S.
V.SP.INPUTREV[3].S	Spindle S.
V.SP.INPUTREV[3]	Master spindle.
V.MPA.INPUTREV[3].4	Axis or spindle with logic number ·4·.

V.[2].MPA.INPUTREV[3].1	Axis with index ·1· in the channel ·2·.
V.SP.INPUTREV[3].2	Spindle with index ·2· in the system.
V.[2].SP.INPUTREV[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.OUTPUTREV[set].xn
(V.)[ch].MPA.OUTPUTREV[set].sn
(V.)[ch].SP.OUTPUTREV[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Gear ratio; turns of the machine shaft.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.OUTPUTREV[3].Z	Z axis.
V.MPA.OUTPUTREV[3].S	Spindle S.
V.SP.OUTPUTREV[3].S	Spindle S.
V.SP.OUTPUTREV[3]	Master spindle.
V.MPA.OUTPUTREV[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.OUTPUTREV[3].1	Axis with index ·1· in the channel ·2·.
V.SP.OUTPUTREV[3].2	Spindle with index ·2· in the system.
V.[2].SP.OUTPUTREV[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.NPULSES[set].xn
(V.)[ch].MPA.NPULSES[set].sn
(V.)[ch].SP.NPULSES[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drives and velocity-Sercos drives.
The variable returns the execution value; reading it interrupts block preparation.

Number of encoder pulses.

With linear encoders (scales), this parameter will be set to ·0·. When using a gear reduction on the axis, the whole assembly must be taken into account when defining the number of pulses per turn.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.NPULSES[3].Z	Z axis.
V.MPA.NPULSES[3].S	Spindle S.
V.SP.NPULSES[3].S	Spindle S.
V.SP.NPULSES[3]	Master spindle.
V.MPA.NPULSES[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.NPULSES[3].1	Axis with index ·1· in the channel ·2·.
V.SP.NPULSES[3].2	Spindle with index ·2· in the system.
V.[2].SP.NPULSES[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.PITCH2[set].xn
(V.)[ch].MPA.PITCH2[set].sn
(V.)[ch].SP.PITCH2[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for position-Sercos drives and velocity-Sercos drives.
The variable returns the execution value; reading it interrupts block preparation.

Leadscrew pitch (external feedback).

Depending on the type of feedback, this parameter means the following.

- On a linear axis with a rotary encoder and leadscrew, it defines the leadscrew pitch.
- On a linear axis with a linear encoder (scale), it defines the pitch of the scale.
- On a rotary axis, it sets the number of degrees per turn of the encoder.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.PITCH2[3].Z	Z axis.
V.MPA.PITCH2[3].S	Spindle S.
V.SP.PITCH2[3].S	Spindle S.
V.SP.PITCH2[3]	Master spindle.
V.MPA.PITCH2[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.PITCH2[3].1	Axis with index ·1· in the channel ·2·.
V.SP.PITCH2[3].2	Spindle with index ·2· in the system.
V.[2].SP.PITCH2[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.INPUTREV2[set].xn
(V.)[ch].MPA.INPUTREV2[set].sn
(V.)[ch].SP.INPUTREV2[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for position-Sercos drives and velocity-Sercos drives.
The variable returns the execution value; reading it interrupts block preparation.

Gear ratio; turns of the motor shaft (external feedback).

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.INPUTREV2[3].Z	Z axis.
V.MPA.INPUTREV2[3].S	Spindle S.
V.SP.INPUTREV2[3].S	Spindle S.
V.SP.INPUTREV2[3]	Master spindle.
V.MPA.INPUTREV2[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.INPUTREV2[3].1	Axis with index ·1· in the channel ·2·.
V.SP.INPUTREV2[3].2	Spindle with index ·2· in the system.
V.[2].SP.INPUTREV2[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.OUTPUTREV2[set].xn
(V.)[ch].MPA.OUTPUTREV2[set].sn
(V.)[ch].SP.OUTPUTREV2[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for position-Sercos drives and velocity-Sercos drives.
The variable returns the execution value; reading it interrupts block preparation.

Gear ratio; turns of the machine shaft (external feedback).

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.OUTPUTREV2[3].Z	Z axis.
V.MPA.OUTPUTREV2[3].S	Spindle S.
V.SP.OUTPUTREV2[3].S	Spindle S.
V.SP.OUTPUTREV2[3]	Master spindle.
V.MPA.OUTPUTREV2[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.OUTPUTREV2[3].1	Axis with index ·1· in the channel ·2·.
V.SP.OUTPUTREV2[3].2	Spindle with index ·2· in the system.
V.[2].SP.OUTPUTREV2[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.NPULSES2[set].xn
(V.)[ch].MPA.NPULSES2[set].sn
(V.)[ch].SP.NPULSES2[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for position-Sercos drives and velocity-Sercos drives.
The variable returns the execution value; reading it interrupts block preparation.

Number of pulses per turn of the encoder (external feedback).

With linear encoders (scales), this parameter will be set to ·0·. When using a gear reduction on the axis, the whole assembly must be taken into account when defining the number of pulses per turn.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.NPULSES2[3].Z	Z axis.
V.MPA.NPULSES2[3].S	Spindle S.
V.SP.NPULSES2[3].S	Spindle S.
V.SP.NPULSES2[3]	Master spindle.
V.MPA.NPULSES2[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.NPULSES2[3].1	Axis with index ·1· in the channel ·2·.
V.SP.NPULSES2[3].2	Spindle with index ·2· in the system.
V.[2].SP.NPULSES2[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.SINMAGNI[set].xn
(V.)[ch].MPA.SINMAGNI[set].sn
(V.)[ch].SP.SINMAGNI[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Multiplying factor for sinusoidal feedback signal.

For square feedback signals, it will be set to ·0·; the CNC applies a x4 factor.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.SINMAGNI[3].Z	Z axis.
V.MPA.SINMAGNI[3].S	Spindle S.
V.SP.SINMAGNI[3].S	Spindle S.
V.SP.SINMAGNI[3]	Master spindle.
V.MPA.SINMAGNI[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.SINMAGNI[3].1	Axis with index ·1· in the channel ·2·.
V.SP.SINMAGNI[3].2	Spindle with index ·2· in the system.
V.[2].SP.SINMAGNI[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.ABSFEEDBACK[set].xn
(V.)[ch].MPA.ABSFEEDBACK[set].sn
(V.)[ch].SP.ABSFEEDBACK[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Absolute feedback system.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.ABDFEEDBACK[3].Z	Z axis.
V.MPA.ABDFEEDBACK[3].S	Spindle S.
V.SP.ABDFEEDBACK[3].S	Spindle S.
V.SP.ABDFEEDBACK[3]	Master spindle.
V.MPA.ABDFEEDBACK[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.ABDFEEDBACK[3].1	Axis with index ·1· in the channel ·2·.
V.SP.ABDFEEDBACK[3].2	Spindle with index ·2· in the system.
V.[2].SP.ABDFEEDBACK[3].1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

(V.)[ch].MPA.FBACKAL[set].xn
(V.)[ch].MPA.FBACKAL[set].sn
(V.)[ch].SP.FBACKAL[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drives.
The variable returns the execution value; reading it interrupts block preparation.

Activate feedback alarm.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.FBACKAL[3].Z	Z axis.
V.MPA.FBACKAL[3].S	Spindle S.
V.SP.FBACKAL[3].S	Spindle S.
V.SP.FBACKAL[3]	Master spindle.
V.MPA.FBACKAL[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.FBACKAL[3].1	Axis with index ·1· in the channel ·2·.
V.SP.FBACKAL[3].2	Spindle with index ·2· in the system.
V.[2].SP.FBACKAL[3].1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

LOOP SETTING.

(V.)[ch].MPA.LOOPCH[set].xn
(V.)[ch].MPA.LOOPCH[set].sn
(V.)[ch].SP.LOOPCH[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Velocity command sign change.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.LOOPCH[3].Z	Z axis.
V.MPA.LOOPCH[3].S	Spindle S.
V.SP.LOOPCH[3].S	Spindle S.
V.SP.LOOPCH[3]	Master spindle.
V.MPA.LOOPCH[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.LOOPCH[3].1	Axis with index ·1· in the channel ·2·.
V.SP.LOOPCH[3].2	Spindle with index ·2· in the system.
V.[2].SP.LOOPCH[3].1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

(V.)[ch].MPA.AXISCH[set].xn
(V.)[ch].MPA.AXISCH[set].sn
(V.)[ch].SP.AXISCH[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Feedback sign change.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.AXISCH[3].Z	Z axis.
V.MPA.AXISCH[3].S	Spindle S.
V.SP.AXISCH[3].S	Spindle S.
V.SP.AXISCH[3]	Master spindle.
V.MPA.AXISCH[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.AXISCH[3].1	Axis with index ·1· in the channel ·2·.
V.SP.AXISCH[3].2	Spindle with index ·2· in the system.
V.[2].SP.AXISCH[3].1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

(V.)[ch].MPA.INPOSW[set].xn
(V.)[ch].MPA.INPOSW[set].sn
(V.)[ch].SP.INPOSW[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified from the environment of the oscilloscope and setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

In-position zone.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.INPOSW[3].Z	Z axis.
V.MPA.INPOSW[3].S	Spindle S.
V.SP.INPOSW[3].S	Spindle S.
V.SP.INPOSW[3]	Master spindle.
V.MPA.INPOSW[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.INPOSW[3].1	Axis with index ·1· in the channel ·2·.
V.SP.INPOSW[3].2	Spindle with index ·2· in the system.
V.[2].SP.INPOSW[3].1	Spindle with index ·1· in the channel ·2·.

BACKLASH COMPENSATION IN MOVEMENT REVERSAL.

(V.)[ch].MPA.BACKLASH[set].xn
(V.)[ch].MPA.BACKLASH[set].sn
(V.)[ch].SP.BACKLASH[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified from the environment of the oscilloscope and setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Amount of backlash to be compensated.

With linear encoders (scales), this parameter will be set to ·0·.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.BACKLASH[3].Z	Z axis.
V.MPA.BACKLASH[3].S	Spindle S.
V.SP.BACKLASH[3].S	Spindle S.
V.SP.BACKLASH[3]	Master spindle.
V.MPA.BACKLASH[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.BACKLASH[3].1	Axis with index ·1· in the channel ·2·.
V.SP.BACKLASH[3].2	Spindle with index ·2· in the system.
V.[2].SP.BACKLASH[3].1	Spindle with index ·1· in the channel ·2·.

BACKLASH COMPENSATION IN MOVEMENT REVERSALS WITH AN ADDITIONAL COMMAND PULSE.

(V.)[ch].MPA.BAKANOUT[set].xn
(V.)[ch].MPA.BAKANOUT[set].sn
(V.)[ch].SP.BAKANOUT[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified from the environment of the oscilloscope and setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Additional velocity command pulse.

- With digital drives, the additional velocity command is given in rpm.
- With an analog drive, the additional velocity command is given in units of the D/A converter with an integer within ± 32767. - 10 V for a value of -32767 and +10V for a value of 32767.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.BAKANOUT[3].Z	Z axis.
V.MPA.BAKANOUT[3].S	Spindle S.
V.SP.BAKANOUT[3].S	Spindle S.
V.SP.BAKANOUT[3]	Master spindle.
V.MPA.BAKANOUT[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.BAKANOUT[3].1	Axis with index ·1· in the channel ·2·.
V.SP.BAKANOUT[3].2	Spindle with index ·2· in the system.
V.[2].SP.BAKANOUT[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.BAKTIME[set].xn
(V.)[ch].MPA.BAKTIME[set].sn
(V.)[ch].SP.BAKTIME[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified from the environment of the oscilloscope and setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Duration of the additional velocity command pulse.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.BAKTIME[3].Z	Z axis.
V.MPA.BAKTIME[3].S	Spindle S.
V.SP.BAKTIME[3].S	Spindle S.
V.SP.BAKTIME[3]	Master spindle.
V.MPA.BAKTIME[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.BAKTIME[3].1	Axis with index ·1· in the channel ·2·.
V.SP.BAKTIME[3].2	Spindle with index ·2· in the system.
V.[2].SP.BAKTIME[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.ACTBAKAN[set].xn
(V.)[ch].MPA.ACTBAKAN[set].sn
(V.)[ch].SP.ACTBAKAN[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Application of the additional velocity command pulse.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.ACTBAKAN[3].Z	Z axis.
V.MPA.ACTBAKAN[3].S	Spindle S.
V.SP.ACTBAKAN[3].S	Spindle S.
V.SP.ACTBAKAN[3]	Master spindle.
V.MPA.ACTBAKAN[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.ACTBAKAN[3].1	Axis with index ·1· in the channel ·2·.
V.SP.ACTBAKAN[3].2	Spindle with index ·2· in the system.
V.[2].SP.ACTBAKAN[3].1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	In G02 / G03 positioning.
1	Always.

(V.)[ch].MPA.PEAKDISP[set].xn
(V.)[ch].MPA.PEAKDISP[set].sn
(V.)[ch].SP.PEAKDISP[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Backlash peak cuttoff distance.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.PEAKDISP[3].Z	Z axis.
V.MPA.PEAKDISP[3].S	Spindle S.
V.SP.PEAKDISP[3].S	Spindle S.
V.SP.PEAKDISP[3]	Master spindle.
V.MPA.PEAKDISP[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.PEAKDISP[3].1	Axis with index ·1· in the channel ·2·.
V.SP.PEAKDISP[3].2	Spindle with index ·2· in the system.
V.[2].SP.PEAKDISP[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.REVEHYST[set].xn
(V.)[ch].MPA.REVEHYST[set].sn
(V.)[ch].SP.REVEHYST[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Hysteresis for applying the additional command pulse in movement reversals.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.REVEHYST[3].Z	Z axis.
V.MPA.REVEHYST[3].S	Spindle S.
V.SP.REVEHYST[3].S	Spindle S.
V.SP.REVEHYST[3]	Master spindle.
V.MPA.REVEHYST[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.REVEHYST[3].1	Axis with index ·1· in the channel ·2·.
V.SP.REVEHYST[3].2	Spindle with index ·2· in the system.
V.[2].SP.REVEHYST[3].1	Spindle with index ·1· in the channel ·2·.

ADJUSTMENT OF RAPID TRAVERSE G00 AND MAXIMUM SPEED.

(V.)[ch].MPA.G00FEED[set].xn
(V.)[ch].MPA.G00FEED[set].sn
(V.)[ch].SP.G00FEED[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified from the environment of the oscilloscope and setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Feedrate in G00.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.G00FEED[3].Z	Z axis.
V.MPA.G00FEED[3].S	Spindle S.
V.SP.G00FEED[3].S	Spindle S.
V.SP.G00FEED[3]	Master spindle.
V.MPA.G00FEED[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.G00FEED[3].1	Axis with index ·1· in the channel ·2·.
V.SP.G00FEED[3].2	Spindle with index ·2· in the system.
V.[2].SP.G00FEED[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.MAXFEED[set].xn
(V.)[ch].MPA.MAXFEED[set].sn
(V.)[ch].SP.MAXFEED[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Channel [ch]. Maximum axis machining feedrate.

If the variable returns a ·0· value, the machining feedrate is not limited, the CNC assumes for all the movements the one set in machine parameter G00FEED as the maximum feedrate.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.MAXFEED[3].Z	Z axis.
V.MPA.MAXFEED[3].S	Spindle S.
V.SP.MAXFEED[3].S	Spindle S.
V.SP.MAXFEED[3]	Master spindle.
V.MPA.MAXFEED[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.MAXFEED[3].1	Axis with index ·1· in the channel ·2·.
V.SP.MAXFEED[3].2	Spindle with index ·2· in the system.
V.[2].SP.MAXFEED[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.MAXVOLT[set].xn
(V.)[ch].MPA.MAXVOLT[set].sn
(V.)[ch].SP.MAXVOLT[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified from the environment of the oscilloscope and setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drives and velocity-Sercos drives.
The variable returns the execution value; reading it interrupts block preparation.

Velocity command to reach G00FEED

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.MAXVOLT[3].Z	Z axis.
V.MPA.MAXVOLT[3].S	Spindle S.
V.SP.MAXVOLT[3].S	Spindle S.
V.SP.MAXVOLT[3]	Master spindle.
V.MPA.MAXVOLT[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.MAXVOLT[3].1	Axis with index ·1· in the channel ·2·.
V.SP.MAXVOLT[3].2	Spindle with index ·2· in the system.
V.[2].SP.MAXVOLT[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.MAXFREQ[set].sn
(V.)[ch].SP.MAXFREQ[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified from the environment of the oscilloscope and setup.
Variable valid for spindles.
Variable valid for Mechatrolink drive.
The variable returns the execution value; reading it interrupts block preparation.

This is the frequency the CNC must provide for the spindle to reach the maximum speed set by parameter G00FEED.

Syntax.

·ch· Channel number.

·set· Parameter set.

·sn· Name, logic number or index of the spindle.

V.MPA.MAXFREQ[3].S	Spindle S.
V.SP.MAXFREQ[3].S	Spindle S.
V.SP.MAXFREQ[3]	Master spindle.
V.MPA.MAXFREQ[3].4	Spindle with logic number ·4·.
V.SP.MAXFREQ[3].2	Spindle with index ·2· in the system.
V.[2].SP.MAXFREQ[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.MAXRPM[set].xn
(V.)[ch].MPA.MAXRPM[set].sn
(V.)[ch].SP.MAXRPM[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for Mechatrolink drive.
The variable returns the execution value; reading it interrupts block preparation.

Maximum motor rpm.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.MAXRPM[3].Z	Z axis.
V.MPA.MAXRPM[3].S	Spindle S.
V.SP.MAXRPM[3].S	Spindle S.
V.SP.MAXRPM[3]	Master spindle.
V.MPA.MAXRPM[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.MAXRPM[3].1	Axis with index ·1· in the channel ·2·.
V.SP.MAXRPM[3].2	Spindle with index ·2· in the system.
V.[2].SP.MAXRPM[3].1	Spindle with index ·1· in the channel ·2·.

RAPID TRAVERSE FOR THE AUTOMATIC MODE.

(V.)[ch].MPA.FRAPIDEN[set].xn
(V.)[ch].MPA.FRAPIDEN[set].sn
(V.)[ch].SP.FRAPIDEN[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Rapid traverse of the axis when the rapid traverse for the automatic mode is active (parameter RAPIDEN). If the parameter is set with a "0" value, the feedrate is not limited.

This parameter does not the movements programmed in G00 or the threads. Movements in G0 are carried out at the feedrate set in parameter G00FEED. The threads are executed at the programmed feedrate.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.FRAPIDEN[3].Z	Z axis.
V.MPA.FRAPIDEN[3].S	Spindle S.
V.SP.FRAPIDEN[3].S	Spindle S.
V.SP.FRAPIDEN[3]	Master spindle.
V.MPA.FRAPIDEN[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.FRAPIDEN[3].1	Axis with index ·1· in the channel ·2·.
V.SP.FRAPIDEN[3].2	Spindle with index ·2· in the system.
V.[2].SP.FRAPIDEN[3].1	Spindle with index ·1· in the channel ·2·.

Remarks.

The rapid traverse cannot exceed the value set in axis parameters G00FEED or FRAPIDEN of the channel or the maximum feedrate set by PLC (variable (V.)PLC.G00FEED). Rapid traverse cannot exceed the value set in axis parameter MAXFEED of the channel and the active feedrate set by PLC (variable (V.)PLC.F).

Variable.	**Machine parameters.**		**Rapid feed.**	
(V.)PLC.G00FEED	G00FEED (axis)	FRAPIDEN (axis)	G00	G01, G02, ···
0	10000	0	10000	10000
0	10000	6000	10000	6000
4000	10000	6000	4000	4000
7000	10000	6000	7000	6000
12000	10000	6000	10000	6000

GAIN SETTING.

(V.)[ch].MPA.PROGAIN[set].xn
(V.)[ch].MPA.PROGAIN[set].sn
(V.)[ch].SP.PROGAIN[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified from the environment of the oscilloscope and setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Proportional gain.

It sets the following error (difference between the theoretical instantaneous position and the actual - real - axis position) for a particular feedrate.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.PROGAIN[3].Z	Z axis.
V.MPA.PROGAIN[3].S	Spindle S.
V.SP.PROGAIN[3].S	Spindle S.
V.SP.PROGAIN[3]	Master spindle.
V.MPA.PROGAIN[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.PROGAIN[3].1	Axis with index ·1· in the channel ·2·.
V.SP.PROGAIN[3].2	Spindle with index ·2· in the system.
V.[2].SP.PROGAIN[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.FFWTYPE[set].xn
(V.)[ch].MPA.FFWTYPE[set].sn
(V.)[ch].SP.FFWTYPE[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Pre-control type

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.FFWTYPE[3].Z	Z axis.
V.MPA.FFWTYPE[3].S	Spindle S.
V.SP.FFWTYPE[3].S	Spindle S.
V.SP.FFWTYPE[3]	Master spindle.
V.MPA.FFWTYPE[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.FFWTYPE[3].1	Axis with index ·1· in the channel ·2·.
V.SP.FFWTYPE[3].2	Spindle with index ·2· in the system.
V.[2].SP.FFWTYPE[3].1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Without pre-control.
1	Feed forward.
2	AC-forward
3	Feed-Forward and AC-Forward.

(V.)[ch].MPA.FFGAIN[set].xn
(V.)[ch].MPA.FFGAIN[set].sn
(V.)[ch].SP.FFGAIN[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified from the environment of the oscilloscope and setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Percentage of Feed-Forward in automatic.

It sets the part of the velocity commmand that is proportional to the programmed feedrate. The rest will be proportional to the following error.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.FFGAIN[3].Z	Z axis.
V.MPA.FFGAIN[3].S	Spindle S.
V.SP.FFGAIN[3].S	Spindle S.
V.SP.FFGAIN[3]	Master spindle.
V.MPA.FFGAIN[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.FFGAIN[3].1	Axis with index ·1· in the channel ·2·.
V.SP.FFGAIN[3].2	Spindle with index ·2· in the system.
V.[2].SP.FFGAIN[3].1	Spindle with index ·1· in the channel ·2·.

Remarks.

The reading from the PLC will be given in hundredths (x100); i.e. if the parameter value is ·10·, the PLC reading will return a value of ·1000·.

Although the parameter may be set with up to 4 decimals, only 2 decimals will be used when reading the variable.

(V.)[ch].MPA.MANFFGAIN[set].xn
(V.)[ch].MPA.MANFFGAIN[set].sn
(V.)[ch].SP.MANFFGAIN[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified from the environment of the oscilloscope and setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Percentage of Feed-Forward in JOG

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.MANFFGAIN[3].Z	Z axis.
V.MPA.MANFFGAIN[3].S	Spindle S.
V.SP.MANFFGAIN[3].S	Spindle S.
V.SP.MANFFGAIN[3]	Master spindle.
V.MPA.MANFFGAIN[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.MANFFGAIN[3].1	Axis with index ·1· in the channel ·2·.
V.SP.MANFFGAIN[3].2	Spindle with index ·2· in the system.
V.[2].SP.MANFFGAIN[3].1	Spindle with index ·1· in the channel ·2·.

Remarks.

The reading from the PLC will be given in hundredths (x100); i.e. if the parameter value is ·10·, the PLC reading will return a value of ·1000·.

Although the parameter may be set with up to 4 decimals, only 2 decimals will be used when reading the variable.

(V.)[ch].MPA.ACFWFACTOR[set].xn
(V.)[ch].MPA.ACFWFACTOR[set].sn
(V.)[ch].SP.ACFWFACTOR[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified from the environment of the oscilloscope and setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drives and velocity-Sercos drives.
The variable returns the execution value; reading it interrupts block preparation.

Acceleration time constant.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.ACFWFACTOR[3].Z	Z axis.
V.MPA.ACFWFACTOR[3].S	Spindle S.
V.SP.ACFWFACTOR[3].S	Spindle S.
V.SP.ACFWFACTOR[3]	Master spindle.
V.MPA.ACFWFACTOR[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.ACFWFACTOR[3].1	Axis with index ·1· in the channel ·2·.
V.SP.ACFWFACTOR[3].2	Spindle with index ·2· in the system.
V.[2].SP.ACFWFACTOR[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.ACFGAIN[set].xn
(V.)[ch].MPA.ACFGAIN[set].sn
(V.)[ch].SP.ACFGAIN[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified from the environment of the oscilloscope and setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Percentage of Feed-Forward in automatic.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.ACFGAIN[3].Z	Z axis.
V.MPA.ACFGAIN[3].S	Spindle S.
V.SP.ACFGAIN[3].S	Spindle S.
V.SP.ACFGAIN[3]	Master spindle.
V.MPA.ACFGAIN[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.ACFGAIN[3].1	Axis with index ·1· in the channel ·2·.
V.SP.ACFGAIN[3].2	Spindle with index ·2· in the system.
V.[2].SP.ACFGAIN[3].1	Spindle with index ·1· in the channel ·2·.

Remarks.

The reading from the PLC will be given in tenths (x10); i.e. if the parameter value is ·10·, the PLC reading will return a value of ·100·.

Although the parameter may be set with up to 4 decimals, only the first decimal will be used when reading the variable.

(V.)[ch].MPA.MANACFGAIN[set].xn
(V.)[ch].MPA.MANACFGAIN[set].sn
(V.)[ch].SP.MANACFGAIN[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified from the environment of the oscilloscope and setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Percentage of Feed-Forward in jog mode.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.MANACFGAIN[3].Z	Z axis.
V.MPA.MANACFGAIN[3].S	Spindle S.
V.SP.MANACFGAIN[3].S	Spindle S.
V.SP.MANACFGAIN[3]	Master spindle.
V.MPA.MANACFGAIN[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.MANACFGAIN[3].1	Axis with index ·1· in the channel ·2·.
V.SP.MANACFGAIN[3].2	Spindle with index ·2· in the system.
V.[2].SP.MANACFGAIN[3].1	Spindle with index ·1· in the channel ·2·.

Remarks.

The reading from the PLC will be given in tenths (x10); i.e. if the parameter value is ·10·, the PLC reading will return a value of ·100·.

Although the parameter may be set with up to 4 decimals, only the first decimal will be used when reading the variable.

LINEAR ACCELERATION.

(V.)[ch].MPA.LACC1[set].xn
(V.)[ch].MPA.LACC1[set].sn
(V.)[ch].SP.LACC1[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified from the environment of the oscilloscope and setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Acceleration of the first section.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.LACC1[3].Z	Z axis.
V.MPA.LACC1[3].S	Spindle S.
V.SP.LACC1[3].S	Spindle S.
V.SP.LACC1[3]	Master spindle.
V.MPA.LACC1[3].4	Axis or spindle with logic number ·4·.

V.[2].MPA.LACC1[3].1	Axis with index ·1· in the channel ·2·.
V.SP.LACC1[3].2	Spindle with index ·2· in the system.
V.[2].SP.LACC1[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.LACC2[set].xn
(V.)[ch].MPA.LACC2[set].sn
(V.)[ch].SP.LACC2[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified from the environment of the oscilloscope and setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Acceleration of the second section.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.LACC2[3].Z	Z axis.
V.MPA.LACC2[3].S	Spindle S.
V.SP.LACC2[3].S	Spindle S.
V.SP.LACC2[3]	Master spindle.
V.MPA.LACC2[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.LACC2[3].1	Axis with index ·1· in the channel ·2·.
V.SP.LACC2[3].2	Spindle with index ·2· in the system.
V.[2].SP.LACC2[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.LFEED[set].xn
(V.)[ch].SP.LFEED[set].sn
(V.)[ch].SP.LFEED[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified from the environment of the oscilloscope and setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Change speed.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.LFEED[3].Z	Z axis.
V.MPA.LFEED[3].S	Spindle S.
V.SP.LFEED[3].S	Spindle S.
V.SP.LFEED[3]	Master spindle.
V.MPA.LFEED[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.LFEED[3].1	Axis with index ·1· in the channel ·2·.
V.SP.LFEED[3].2	Spindle with index ·2· in the system.
V.[2].SP.LFEED[3].1	Spindle with index ·1· in the channel ·2·.

TRAPEZOIDAL AND SQUARE SINE ACCELERATION.

(V.)[ch].MPA.ACCEL[set].xn
(V.)[ch].MPA.ACCEL[set].sn
(V.)[ch].SP.ACCEL[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified from the environment of the oscilloscope and setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Acceleration.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.ACCEL[3].Z	Z axis.
V.MPA.ACCEL[3].S	Spindle S.
V.SP.ACCEL[3].S	Spindle S.
V.SP.ACCEL[3]	Master spindle.
V.MPA.ACCEL[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.ACCEL[3].1	Axis with index ·1· in the channel ·2·.
V.SP.ACCEL[3].2	Spindle with index ·2· in the system.
V.[2].SP.ACCEL[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.DECEL[set].xn
(V.)[ch].MPA.DECEL[set].sn
(V.)[ch].SP.DECEL[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified from the environment of the oscilloscope and setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Deceleration.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.DECEL[3].Z	Z axis.
V.MPA.DECEL[3].S	Spindle S.
V.SP.DECEL[3].S	Spindle S.
V.SP.DECEL[3]	Master spindle.
V.MPA.DECEL[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.DECEL[3].1	Axis with index ·1· in the channel ·2·.
V.SP.DECEL[3].2	Spindle with index ·2· in the system.
V.[2].SP.DECEL[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.ACCJERK[set].xn
(V.)[ch].MPA.ACCJERK[set].sn
(V.)[ch].SP.ACCJERK[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified from the environment of the oscilloscope and setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Acceleration Jerk.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.ACCJERK[3].Z	Z axis.
V.MPA.ACCJERK[3].S	Spindle S.
V.SP.ACCJERK[3].S	Spindle S.
V.SP.ACCJERK[3]	Master spindle.
V.MPA.ACCJERK[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.ACCJERK[3].1	Axis with index ·1· in the channel ·2·.
V.SP.ACCJERK[3].2	Spindle with index ·2· in the system.
V.[2].SP.ACCJERK[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.DECJERK[set].xn
(V.)[ch].MPA.DECJERK[set].sn
(V.)[ch].SP.DECJERK[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified from the environment of the oscilloscope and setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Deceleration Jerk.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.DECJERK[3].Z	Z axis.
V.MPA.DECJERK[3].S	Spindle S.
V.SP.DECJERK[3].S	Spindle S.
V.SP.DECJERK[3]	Master spindle.
V.MPA.DECJERK[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.DECJERK[3].1	Axis with index ·1· in the channel ·2·.
V.SP.DECJERK[3].2	Spindle with index ·2· in the system.
V.[2].SP.DECJERK[3].1	Spindle with index ·1· in the channel ·2·.

ENABLE SPECIFIC ACCELERATION VALUES FOR MOVEMENTS IN G0.

(V.)[ch].MPA.G0ACDCJERK[set].xn
(V.)[ch].MPA.G0ACDCJERK[set].sn
(V.)[ch].SP.G0ACDCJERK[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

More parameters if G0.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.G0ACDCJERK[3].Z	Z axis.
V.MPA.G0ACDCJERK[3].S	Spindle S.
V.SP.G0ACDCJERK[3].S	Spindle S.
V.SP.G0ACDCJERK[3]	Master spindle.
V.MPA.G0ACDCJERK[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.G0ACDCJERK[3].1	Axis with index ·1· in the channel ·2·.
V.SP.G0ACDCJERK[3].2	Spindle with index ·2· in the system.
V.[2].SP.G0ACDCJERK[3].1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

LINEAR ACCELERATION (G0 MOVEMENTS).

(V.)[ch].MPA.LACC1G0[set].xn
(V.)[ch].MPA.LACC1G0[set].sn
(V.)[ch].SP.LACC1G0[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified from the environment of the oscilloscope and setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Acceleration of the first section (G0 movements).

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.LACC1G0[3].Z	Z axis.
V.MPA.LACC1G0[3].S	Spindle S.
V.SP.LACC1G0[3].S	Spindle S.

V.SP.LACC1G0[3]	Master spindle.
V.MPA.LACC1G0[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.LACC1G0[3].1	Axis with index ·1· in the channel ·2·.
V.SP.LACC1G0[3].2	Spindle with index ·2· in the system.
V.[2].SP.LACC1G0[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.LACC2G0[set].xn
(V.)[ch].MPA.LACC2G0[set].sn
(V.)[ch].SP.LACC2G0[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified from the environment of the oscilloscope and setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Acceleration of the second section (G0 movements).

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.LACC2G0[3].Z	Z axis.
V.MPA.LACC2G0[3].S	Spindle S.
V.SP.LACC2G0[3].S	Spindle S.
V.SP.LACC2G0[3]	Master spindle.
V.MPA.LACC2G0[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.LACC2G0[3].1	Axis with index ·1· in the channel ·2·.
V.SP.LACC2G0[3].2	Spindle with index ·2· in the system.
V.[2].SP.LACC2G0[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.LFEEDG0[set].xn
(V.)[ch].SP.LFEEDG0[set].sn
(V.)[ch].SP.LFEEDG0[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified from the environment of the oscilloscope and setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Acceleration changing speed (G0 movements).

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.LFEEDG0[3].Z	Z axis.
V.MPA.LFEEDG0[3].S	Spindle S.
V.SP.LFEEDG0[3].S	Spindle S.
V.SP.LFEEDG0[3]	Master spindle.
V.MPA.LFEEDG0[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.LFEEDG0[3].1	Axis with index ·1· in the channel ·2·.
V.SP.LFEEDG0[3].2	Spindle with index ·2· in the system.
V.[2].SP.LFEEDG0[3].1	Spindle with index ·1· in the channel ·2·.

TRAPEZOIDAL AND SQUARE SINE ACCELERATION (G0 MOVEMENTS).

(V.)[ch].MPA.ACCELG0[set].xn
(V.)[ch].MPA.ACCELG0[set].sn
(V.)[ch].SP.ACCELG0[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified from the environment of the oscilloscope and setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Acceleration (G0 movements).

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.ACCELG0[3].Z	Z axis.
V.MPA.ACCELG0[3].S	Spindle S.
V.SP.ACCELG0[3].S	Spindle S.
V.SP.ACCELG0[3]	Master spindle.
V.MPA.ACCELG0[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.ACCELG0[3].1	Axis with index ·1· in the channel ·2·.
V.SP.ACCELG0[3].2	Spindle with index ·2· in the system.
V.[2].SP.ACCELG0[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.DECELG0[set].xn
(V.)[ch].MPA.DECELG0[set].sn
(V.)[ch].SP.DECELG0[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified from the environment of the oscilloscope and setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Deceleration (G0 movements).

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.DECELG0[3].Z	Z axis.
V.MPA.DECELG0[3].S	Spindle S.
V.SP.DECELG0[3].S	Spindle S.
V.SP.DECELG0[3]	Master spindle.
V.MPA.DECELG0[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.DECELG0[3].1	Axis with index ·1· in the channel ·2·.
V.SP.DECELG0[3].2	Spindle with index ·2· in the system.
V.[2].SP.DECELG0[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.ACCJERKG0[set].xn
(V.)[ch].MPA.ACCJERKG0[set].sn
(V.)[ch].SP.ACCJERKG0[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified from the environment of the oscilloscope and setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Acceleration jerk (G0 movements).

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.ACCJERKG0[3].Z	Z axis.
V.MPA.ACCJERKG0[3].S	Spindle S.
V.SP.ACCJERKG0[3].S	Spindle S.
V.SP.ACCJERKG0[3]	Master spindle.
V.MPA.ACCJERKG0[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.ACCJERKG0[3].1	Axis with index ·1· in the channel ·2·.
V.SP.ACCJERKG0[3].2	Spindle with index ·2· in the system.
V.[2].SP.ACCJERKG0[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.DECJERKG0[set].xn
(V.)[ch].MPA.DECJERKG0[set].sn
(V.)[ch].SP.DECJERKG0[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified from the environment of the oscilloscope and setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Deceleration jerk (G0 movements).

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.DECJERKG0[3].Z	Z axis.
V.MPA.DECJERKG0[3].S	Spindle S.
V.SP.DECJERKG0[3].S	Spindle S.
V.SP.DECJERKG0[3]	Master spindle.
V.MPA.DECJERKG0[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.DECJERKG0[3].1	Axis with index ·1· in the channel ·2·.
V.SP.DECJERKG0[3].2	Spindle with index ·2· in the system.
V.[2].SP.DECJERKG0[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.CORNERACC[set].xn
(V.)[ch].MPA.CORNERACC[set].sn
(V.)[ch].SP.CORNERACC[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Maximum acceleration permitted at the corners.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.CORNERACC[3].Z	Z axis.
V.MPA.CORNERACC[3].S	Spindle S.
V.SP.CORNERACC[3].S	Spindle S.
V.SP.CORNERACC[3]	Master spindle.
V.MPA.CORNERACC[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.CORNERACC[3].1	Axis with index ·1· in the channel ·2·.
V.SP.CORNERACC[3].2	Spindle with index ·2· in the system.
V.[2].SP.CORNERACC[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.CURVACC[set].xn
(V.)[ch].MPA.CURVACC[set].sn
(V.)[ch].SP.CURVACC[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Maximum contouring acceleration permitted.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.CURVACC[3].Z	Z axis.
V.MPA.CURVACC[3].S	Spindle S.
V.SP.CURVACC[3].S	Spindle S.
V.SP.CURVACC[3]	Master spindle.
V.MPA.CURVACC[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.CURVACC[3].1	Axis with index ·1· in the channel ·2·.
V.SP.CURVACC[3].2	Spindle with index ·2· in the system.
V.[2].SP.CURVACC[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.CORNERJERK[set].xn
(V.)[ch].MPA.CORNERJERK[set].sn
(V.)[ch].SP.CORNERJERK[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Maximum Jerk permitted at the corners.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.CORNERJERK[3].Z	Z axis.
V.MPA.CORNERJERK[3].S	Spindle S.
V.SP.CORNERJERK[3].S	Spindle S.
V.SP.CORNERJERK[3]	Master spindle.
V.MPA.CORNERJERK[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.CORNERJERK[3].1	Axis with index ·1· in the channel ·2·.
V.SP.CORNERJERK[3].2	Spindle with index ·2· in the system.
V.[2].SP.CORNERJERK[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.CURVJERK[set].xn
(V.)[ch].MPA.CURVJERK[set].sn
(V.)[ch].SP.CURVJERK[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Maximum contouring Jerk permitted.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.CURVJERK[3].Z	Z axis.
V.MPA.CURVJERK[3].S	Spindle S.
V.SP.CURVJERK[3].S	Spindle S.
V.SP.CURVJERK[3]	Master spindle.
V.MPA.CURVJERK[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.CURVJERK[3].1	Axis with index ·1· in the channel ·2·.
V.SP.CURVJERK[3].2	Spindle with index ·2· in the system.
V.[2].SP.CURVJERK[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.FASTACC[set].xn
(V.)[ch].MPA.FASTACC[set].sn
(V.)[ch].SP.FASTACC[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Maximum acceleration permitted (FAST mode).

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.FASTACC[3].Z	Z axis.
V.MPA.FASTACC[3].S	Spindle S.
V.SP.FASTACC[3].S	Spindle S.
V.SP.FASTACC[3]	Master spindle.
V.MPA.FASTACC[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.FASTACC[3].1	Axis with index ·1· in the channel ·2·.
V.SP.FASTACC[3].2	Spindle with index ·2· in the system.
V.[2].SP.FASTACC[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.MAXERROR[set].xn
(V.)[ch].MPA.MAXERROR[set].sn
(V.)[ch].SP.MAXERROR[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Maximum axis position error in HSC when working outside the plane/trihedron.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.MAXERROR[3].Z	Z axis.
V.MPA.MAXERROR[3].S	Spindle S.
V.SP.MAXERROR[3].S	Spindle S.
V.SP.MAXERROR[3]	Master spindle.
V.MPA.MAXERROR[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.MAXERROR[3].1	Axis with index ·1· in the channel ·2·.
V.SP.MAXERROR[3].2	Spindle with index ·2· in the system.
V.[2].SP.MAXERROR[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.CONTERROR[set].xn
(V.)[ch].MPA.CONTERROR[set].sn
(V.)[ch].SP.CONTERROR[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Tolerance per axis for smoothing the generated n-dimensional path.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.CONTERROR[3].Z	Z axis.
V.MPA.CONTERROR[3].S	Spindle S.
V.SP.CONTERROR[3].S	Spindle S.
V.SP.CONTERROR[3]	Master spindle.
V.MPA.CONTERROR[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.CONTERROR[3].1	Axis with index ·1· in the channel ·2·.
V.SP.CONTERROR[3].2	Spindle with index ·2· in the system.
V.[2].SP.CONTERROR[3].1	Spindle with index ·1· in the channel ·2·.

HOME SEARCH.

(V.)[ch].MPA.I0TYPE[set].xn
(V.)[ch].MPA.I0TYPE[set].sn
(V.)[ch].SP.I0TYPE[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Reference mark (I0) type

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.I0TYPE[3].Z	Z axis.
V.MPA.I0TYPE[3].S	Spindle S.
V.SP.I0TYPE[3].S	Spindle S.
V.SP.I0TYPE[3]	Master spindle.
V.MPA.I0TYPE[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.I0TYPE[3].1	Axis with index ·1· in the channel ·2·.
V.SP.I0TYPE[3].2	Spindle with index ·2· in the system.
V.[2].SP.I0TYPE[3].1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Incremental (Not distance-coded)
1	Increasing distance-coded reference mark.
2	Decreasing distance-coded reference mark.

(V.)[ch].MPA.REFVALUE[set].xn
(V.)[ch].MPA.REFVALUE[set].sn
(V.)[ch].SP.REFVALUE[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Home position.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.REFVALUE[3].Z	Z axis.
V.MPA.REFVALUE[3].S	Spindle S.
V.SP.REFVALUE[3].S	Spindle S.
V.SP.REFVALUE[3]	Master spindle.
V.MPA.REFVALUE[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.REFVALUE[3].1	Axis with index ·1· in the channel ·2·.
V.SP.REFVALUE[3].2	Spindle with index ·2· in the system.
V.[2].SP.REFVALUE[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.REFSHIFT[set].xn
(V.)[ch].MPA.REFSHIFT[set].sn
(V.)[ch].SP.REFSHIFT[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Offset of the reference point (home)

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.REFSHIFT[3].Z	Z axis.
V.MPA.REFSHIFT[3].S	Spindle S.
V.SP.REFSHIFT[3].S	Spindle S.
V.SP.REFSHIFT[3]	Master spindle.
V.MPA.REFSHIFT[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.REFSHIFT[3].1	Axis with index ·1· in the channel ·2·.
V.SP.REFSHIFT[3].2	Spindle with index ·2· in the system.
V.[2].SP.REFSHIFT[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.REFFEED1[set].xn
(V.)[ch].MPA.REFFEED1[set].sn
(V.)[ch].SP.REFFEED1[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Fast home searching feedrate.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.REFFEED1[3].Z	Z axis.
V.MPA.REFFEED1[3].S	Spindle S.
V.SP.REFFEED1[3].S	Spindle S.
V.SP.REFFEED1[3]	Master spindle.
V.MPA.REFFEED1[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.REFFEED1[3].1	Axis with index ·1· in the channel ·2·.
V.SP.REFFEED1[3].2	Spindle with index ·2· in the system.
V.[2].SP.REFFEED1[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.REFFEED2[set].xn
(V.)[ch].MPA.REFFEED2[set].sn
(V.)[ch].SP.REFFEED2[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Slow home searching feedrate.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.REFFEED2[3].Z	Z axis.
V.MPA.REFFEED2[3].S	Spindle S.
V.SP.REFFEED2[3].S	Spindle S.
V.SP.REFFEED2[3]	Master spindle.
V.MPA.REFFEED2[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.REFFEED2[3].1	Axis with index ·1· in the channel ·2·.
V.SP.REFFEED2[3].2	Spindle with index ·2· in the system.
V.[2].SP.REFFEED2[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.REFPULSE[set].xn
(V.)[ch].MPA.REFPULSE[set].sn
(V.)[ch].SP.REFPULSE[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Type of I0 pulse.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.REFPULSE[3].Z	Z axis.
V.MPA.REFPULSE[3].S	Spindle S.
V.SP.REFPULSE[3].S	Spindle S.
V.SP.REFPULSE[3]	Master spindle.
V.MPA.REFPULSE[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.REFPULSE[3].1	Axis with index ·1· in the channel ·2·.
V.SP.REFPULSE[3].2	Spindle with index ·2· in the system.
V.[2].SP.REFPULSE[3].1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Negative pulse.
1	Positive pulse.

(V.)[ch].MPA.POSINREF[set].xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Home search moving the axis to the reference point.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

V.MPA.POSINREF[3].Z	Z axis.
V.MPA.POSINREF[3].4	Axis with logic number ·4·.
V.[2].MPA.POSINREF[3].1	Axis with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

(V.)[ch].MPA.ABSOFF[set].xn
(V.)[ch].MPA.ABSOFF[set].sn
(V.)[ch].SP.ABSOFF[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Offset with respect to the distance-coded ref. mark.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.ABSOFF[3].Z	Z axis.
V.MPA.ABSOFF[3].S	Spindle S.
V.SP.ABSOFF[3].S	Spindle S.
V.SP.ABSOFF[3]	Master spindle.
V.MPA.ABSOFF[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.ABSOFF[3].1	Axis with index ·1· in the channel ·2·.
V.SP.ABSOFF[3].2	Spindle with index ·2· in the system.
V.[2].SP.ABSOFF[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.EXTMULT[set].xn
(V.)[ch].MPA.EXTMULT[set].sn
(V.)[ch].SP.EXTMULT[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drives.
The variable returns the value of block preparation.

External factor for distance-coded reference marks.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.EXTMULT[3].Z	Z axis.
V.MPA.EXTMULT[3].S	Spindle S.
V.SP.EXTMULT[3].S	Spindle S.
V.SP.EXTMULT[3]	Master spindle.
V.MPA.EXTMULT[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.EXTMULT[3].1	Axis with index ·1· in the channel ·2·.
V.SP.EXTMULT[3].2	Spindle with index ·2· in the system.
V.[2].SP.EXTMULT[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.I0CODDI1[set].xn
(V.)[ch].MPA.I0CODDI1[set].sn
(V.)[ch].SP.I0CODDI1[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Pitch between 2 fixed distance-coded reference marks.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.I0CODDI1[3].Z	Z axis.
V.MPA.I0CODDI1[3].S	Spindle S.
V.SP.I0CODDI1[3].S	Spindle S.
V.SP.I0CODDI1[3]	Master spindle.
V.MPA.I0CODDI1[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.I0CODDI1[3].1	Axis with index ·1· in the channel ·2·.
V.SP.I0CODDI1[3].2	Spindle with index ·2· in the system.
V.[2].SP.I0CODDI1[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.I0CODDI2[set].xn
(V.)[ch].MPA.I0CODDI2[set].sn
(V.)[ch].SP.I0CODDI2[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Pitch between 2 variable distance-coded reference marks.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.I0CODDI2[3].Z	Z axis.
V.MPA.I0CODDI2[3].S	Spindle S.
V.SP.I0CODDI2[3].S	Spindle S.
V.SP.I0CODDI2[3]	Master spindle.
V.MPA.I0CODDI2[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.I0CODDI2[3].1	Axis with index ·1· in the channel ·2·.
V.SP.I0CODDI2[3].2	Spindle with index ·2· in the system.
V.[2].SP.I0CODDI2[3].1	Spindle with index ·1· in the channel ·2·.

FOLLOWING ERROR.

(V.)[ch].MPA.FLWEMONITOR[set].xn
(V.)[ch].MPA.FLWEMONITOR[set].sn
(V.)[ch].SP.FLWEMONITOR[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Type of monitoring of the following error (axis lag).

The CNC offers two types of monitoring of the following error (axis lag). The "standard" monitoring type performs a continuous supervision of the following error whereas the "linear" monitoring performs a dynamic supervision

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.FLWEMONITOR[3].Z	Z axis.
V.MPA.FLWEMONITOR[3].S	Spindle S.
V.SP.FLWEMONITOR[3].S	Spindle S.
V.SP.FLWEMONITOR[3]	Master spindle.
V.MPA.FLWEMONITOR[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.FLWEMONITOR[3].1	Axis with index ·1· in the channel ·2·.
V.SP.FLWEMONITOR[3].2	Spindle with index ·2· in the system.
V.[2].SP.FLWEMONITOR[3].1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No monitoring.
1	Standard monitoring.
2	Linear monitoring.

(V.)[ch].MPA.MINFLWE[set].xn
(V.)[ch].MPA.MINFLWE[set].sn
(V.)[ch].SP.MINFLWE[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Maximum following error when stopped.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.MINFLWE[3].Z	Z axis.
V.MPA.MINFLWE[3].S	Spindle S.
V.SP.MINFLWE[3].S	Spindle S.
V.SP.MINFLWE[3]	Master spindle.
V.MPA.MINFLWE[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.MINFLWE[3].1	Axis with index ·1· in the channel ·2·.
V.SP.MINFLWE[3].2	Spindle with index ·2· in the system.
V.[2].SP.MINFLWE[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.MAXFLWE[set].xn
(V.)[ch].MPA.MAXFLWE[set].sn
(V.)[ch].SP.MAXFLWE[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Maximum following error when moving.

With "standard" monitoring, this variable indicates the maximum amount of following error allowed when the axis is moving; with "linear" monitoring, it indicates when dynamic supervision begins (starting at which amount of following error).

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.MAXFLWE[3].Z	Z axis.
V.MPA.MAXFLWE[3].S	Spindle S.
V.SP.MAXFLWE[3].S	Spindle S.
V.SP.MAXFLWE[3]	Master spindle.
V.MPA.MAXFLWE[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.MAXFLWE[3].1	Axis with index ·1· in the channel ·2·.
V.SP.MAXFLWE[3].2	Spindle with index ·2· in the system.
V.[2].SP.MAXFLWE[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.FEDYNFAC[set].xn
(V.)[ch].MPA.FEDYNFAC[set].sn
(V.)[ch].SP.FEDYNFAC[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Percentage of following error deviation allowed.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.FEDYNFAC[3].Z	Z axis.
V.MPA.FEDYNFAC[3].S	Spindle S.
V.SP.FEDYNFAC[3].S	Spindle S.
V.SP.FEDYNFAC[3]	Master spindle.
V.MPA.FEDYNFAC[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.FEDYNFAC[3].1	Axis with index ·1· in the channel ·2·.
V.SP.FEDYNFAC[3].2	Spindle with index ·2· in the system.
V.[2].SP.FEDYNFAC[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.ESTDELAY[set].xn
(V.)[ch].MPA.ESTDELAY[set].sn
(V.)[ch].SP.ESTDELAY[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified from the environment of the oscilloscope and setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Following error delay.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.ESTDELAY[3].Z	Z axis.
V.MPA.ESTDELAY[3].S	Spindle S.
V.SP.ESTDELAY[3].S	Spindle S.
V.SP.ESTDELAY[3]	Master spindle.
V.MPA.ESTDELAY[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.ESTDELAY[3].1	Axis with index ·1· in the channel ·2·.
V.SP.ESTDELAY[3].2	Spindle with index ·2· in the system.
V.[2].SP.ESTDELAY[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.INPOMAX[set].xn
(V.)[ch].MPA.INPOMAX[set].sn
(V.)[ch].SP.INPOMAX[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Time to get in position.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.INPOMAX[3].Z	Z axis.
V.MPA.INPOMAX[3].S	Spindle S.
V.SP.INPOMAX[3].S	Spindle S.
V.SP.INPOMAX[3]	Master spindle.
V.MPA.INPOMAX[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.INPOMAX[3].1	Axis with index ·1· in the channel ·2·.
V.SP.INPOMAX[3].2	Spindle with index ·2· in the system.
V.[2].SP.INPOMAX[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.INPOTIME[set].xn
(V.)[ch].MPA.INPOTIME[set].sn
(V.)[ch].SP.INPOTIME[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Minimum time to stay in position.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.INPOTIME[3].Z	Z axis.
V.MPA.INPOTIME[3].S	Spindle S.
V.SP.INPOTIME[3].S	Spindle S.

V.SP.INPOTIME[3]	Master spindle.
V.MPA.INPOTIME[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.INPOTIME[3].1	Axis with index ·1· in the channel ·2·.
V.SP.INPOTIME[3].2	Spindle with index ·2· in the system.
V.[2].SP.INPOTIME[3].1	Spindle with index ·1· in the channel ·2·.

AXIS LUBRICATION.

(V.)[ch].MPA.DISTLUBRI[set].xn
(V.)[ch].MPA.DISTLUBRI[set].sn
(V.)[ch].SP.DISTLUBRI[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Distance to move to lubricate the axis.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.DISTLUBRI[3].Z	Z axis.
V.MPA.DISTLUBRI[3].S	Spindle S.
V.SP.DISTLUBRI[3].S	Spindle S.
V.SP.DISTLUBRI[3]	Master spindle.
V.MPA.DISTLUBRI[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.DISTLUBRI[3].1	Axis with index ·1· in the channel ·2·.
V.SP.DISTLUBRI[3].2	Spindle with index ·2· in the system.
V.[2].SP.DISTLUBRI[3].1	Spindle with index ·1· in the channel ·2·.

MODULE CONFIGURATION (ROTARY AXES AND SPINDLE).

(V.)[ch].MPA.MODUPLIM[set].xn
(V.)[ch].MPA.MODUPLIM[set].sn
(V.)[ch].SP.MODUPLIM[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary axes and spindles.
Variable valid for analog drives and velocity-Sercos drives.
The variable returns the execution value; reading it interrupts block preparation.

Upper limit of the module.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.MODUPLIM[3].Z	Z axis.
V.MPA.MODUPLIM[3].S	Spindle S.
V.SP.MODUPLIM[3].S	Spindle S.
V.SP.MODUPLIM[3]	Master spindle.

V.MPA.MODUPLIM[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.MODUPLIM[3].1	Axis with index ·1· in the channel ·2·.
V.SP.MODUPLIM[3].2	Spindle with index ·2· in the system.
V.[2].SP.MODUPLIM[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.MODLOWLIM[set].xn
(V.)[ch].MPA.MODLOWLIM[set].sn
(V.)[ch].SP.MODLOWLIM[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary axes and spindles.
Variable valid for analog drives and velocity-Sercos drives.
The variable returns the execution value; reading it interrupts block preparation.

Lower limit of the module.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.MODLOWLIM[3].Z	Z axis.
V.MPA.MODLOWLIM[3].S	Spindle S.
V.SP.MODLOWLIM[3].S	Spindle S.
V.SP.MODLOWLIM[3]	Master spindle.
V.MPA.MODLOWLIM[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.MODLOWLIM[3].1	Axis with index ·1· in the channel ·2·.
V.SP.MODLOWLIM[3].2	Spindle with index ·2· in the system.
V.[2].SP.MODLOWLIM[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.MODNROT[set].xn
(V.)[ch].MPA.MODNROT[set].sn
(V.)[ch].SP.MODNROT[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary axes and spindles.
Variable valid for analog drives and velocity-Sercos drives.
The variable returns the execution value; reading it interrupts block preparation.

Module error. Number of turns.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.MODNROT[3].Z	Z axis.
V.MPA.MODNROT[3].S	Spindle S.
V.SP.MODNROT[3].S	Spindle S.
V.SP.MODNROT[3]	Master spindle.
V.MPA.MODNROT[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.MODNROT[3].1	Axis with index ·1· in the channel ·2·.
V.SP.MODNROT[3].2	Spindle with index ·2· in the system.
V.[2].SP.MODNROT[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.MODERR[set].xn
(V.)[ch].MPA.MODERR[set].sn
(V.)[ch].SP.MODERR[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified during setup.
Variable valid for rotary axes and spindles.
Variable valid for analog drives and velocity-Sercos drives.
The variable returns the execution value; reading it interrupts block preparation.

Module error. Number of increments.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.MODERR[3].Z	Z axis.
V.MPA.MODERR[3].S	Spindle S.
V.SP.MODERR[3].S	Spindle S.
V.SP.MODERR[3]	Master spindle.
V.MPA.MODERR[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.MODERR[3].1	Axis with index ·1· in the channel ·2·.
V.SP.MODERR[3].2	Spindle with index ·2· in the system.
V.[2].SP.MODERR[3].1	Spindle with index ·1· in the channel ·2·.

SPINDLE SPEED.

(V.)[ch].MPA.SZERO[set].sn
(V.)[ch].SP.SZERO[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for spindles.
Variable valid for position-Sercos drives and velocity-Sercos drives.
The variable returns the value of block preparation.

Speed considered "0 rpm".

Syntax.

·ch· Channel number.

·set· Parameter set.

·sn· Name, logic number or index of the spindle.

V.MPA.SZERO[3].S	Spindle S.
V.SP.SZERO[3].S	Spindle S.
V.SP.SZERO[3]	Master spindle.
V.MPA.SZERO[3].4	Spindle with logic number ·4·.
V.SP.SZERO[3].2	Spindle with index ·2· in the system.
V.[2].SP.SZERO[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.POLARM3[set].sn
(V.)[ch].SP.POLARM3[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for spindles.
Variable valid for velocity-Sercos drives.
The variable returns the value of block preparation.

Sign of the velocity command for M3.

Syntax.

·ch· Channel number.

·set· Parameter set.

·sn· Name, logic number or index of the spindle.

V.MPA.POLARM3[3].S	Spindle S.
V.SP.POLARM3[3].S	Spindle S.
V.SP.POLARM3[3]	Master spindle.
V.MPA.POLARM3[3].4	Spindle with logic number ·4·.
V.SP.POLARM3[3].2	Spindle with index ·2· in the system.
V.[2].SP.POLARM3[3].1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Negative.
1	Positive.

(V.)[ch].MPA.POLARM4[set].sn
(V.)[ch].SP.POLARM4[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for spindles.
Variable valid for velocity-Sercos drives.
The variable returns the value of block preparation.

Sign of the velocity command for M4.

Syntax.

·ch· Channel number.

·set· Parameter set.

·sn· Name, logic number or index of the spindle.

V.MPA.POLARM4[3].S	Spindle S.
V.SP.POLARM4[3].S	Spindle S.
V.SP.POLARM4[3]	Master spindle.
V.MPA.POLARM4[3].4	Spindle with logic number ·4·.
V.SP.POLARM4[3].2	Spindle with index ·2· in the system.
V.[2].SP.POLARM4[3].1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Negative.
1	Positive.

ANALOG COMMAND SETTING.

(V.)[ch].MPA.SERVOOFF[set].xn
(V.)[ch].MPA.SERVOOFF[set].sn
(V.)[ch].SP.SERVOOFF[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified from the environment of the oscilloscope and setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drives.
The variable returns the execution value; reading it interrupts block preparation.

Offset compensation.

The velocity command is given in D/A converter units which could be any integer within ±32767 where a value of ±32767 corresponds to ±10V.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.SERVOOFF[3].Z	Z axis.
V.MPA.SERVOOFF[3].S	Spindle S.
V.SP.SERVOOFF[3].S	Spindle S.
V.SP.SERVOOFF[3]	Master spindle.
V.MPA.SERVOOFF[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.SERVOOFF[3].1	Axis with index ·1· in the channel ·2·.
V.SP.SERVOOFF[3].2	Spindle with index ·2· in the system.
V.[2].SP.SERVOOFF[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.MINANOUT[set].xn
(V.)[ch].MPA.MINANOUT[set].sn
(V.)[ch].SP.MINANOUT[set].sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified from the environment of the oscilloscope and setup.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drives.
The variable returns the execution value; reading it interrupts block preparation.

Minimum velocity command.

The velocity command is given in D/A converter units which could be any integer within ±32767 where a value of ±32767 corresponds to ±10V.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.MINANOUT[3].Z	Z axis.
V.MPA.MINANOUT[3].S	Spindle S.
V.SP.MINANOUT[3].S	Spindle S.
V.SP.MINANOUT[3]	Master spindle.
V.MPA.MINANOUT[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.MINANOUT[3].1	Axis with index ·1· in the channel ·2·.
V.SP.MINANOUT[3].2	Spindle with index ·2· in the system.
V.[2].SP.MINANOUT[3].1	Spindle with index ·1· in the channel ·2·.

NUMBER OF THE ANALOG OUTPUT AND OF THE FEEDBACK INPUT ASSOCIATED WITH THE AXIS.

(V.)[ch].MPA.ANAOUTTYPE[set].xn
(V.)[ch].MPA.ANAOUTTYPE[set].sn
(V.)[ch].SP.ANAOUTTYPE[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drives.
The variable returns the value of block preparation.

Type of analog output associated with the axis.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.ANAOUTTYPE[3].Z	Z axis.
V.MPA.ANAOUTTYPE[3].S	Spindle S.
V.SP.ANAOUTTYPE[3].S	Spindle S.
V.SP.ANAOUTTYPE[3]	Master spindle.
V.MPA.ANAOUTTYPE[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.ANAOUTTYPE[3].1	Axis with index ·1· in the channel ·2·.
V.SP.ANAOUTTYPE[3].2	Spindle with index ·2· in the system.
V.[2].SP.ANAOUTTYPE[3].1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	The analog output is in the remote modules.
1	The analog output is in a Sercos drive.
2	The analog output is in a RCS-S module.

(V.)[ch].MPA.ANAOUTID[set].xn
(V.)[ch].MPA.ANAOUTID[set].sn
(V.)[ch].SP.ANAOUTID[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drives.
The variable returns the value of block preparation.

Number of the analog output associated with the axis.

The velocity command for an analog axis may be taken from an analog output of the remote modules or of the Sercos drive.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.ANAOUTID[3].Z	Z axis.
V.MPA.ANAOUTID[3].S	Spindle S.
V.SP.ANAOUTID[3].S	Spindle S.

V.SP.ANAOUTID[3]	Master spindle.
V.MPA.ANAOUTID[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.ANAOUTID[3].1	Axis with index ·1· in the channel ·2·.
V.SP.ANAOUTID[3].2	Spindle with index ·2· in the system.
V.[2].SP.ANAOUTID[3].1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
1 - 16	The analog command is taken from the remote modules. The variable returns the number of the analog output.
101 -132 201 - 232	The analog command is taken from a Sercos drive. The first digit indicates the analog output to be used (1 or 2) and the other two digits indicate the logic address of the drive (1 through 32).
1 - 32	The analog command is taken from an RCS-S module.. The variable returns the number of the analog output.

(V.)[ch].MPA.COUNTERTYPE[set].xn
(V.)[ch].MPA.COUNTERTYPE[set].sn
(V.)[ch].SP.COUNTERTYPE[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drives and velocity-Sercos drives.
The variable returns the value of block preparation.

Type of feedback input for the axis.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.COUNTERTYPE[3].Z	Z axis.
V.MPA.COUNTERTYPE[3].S	Spindle S.
V.SP.COUNTERTYPE[3].S	Spindle S.
V.SP.COUNTERTYPE[3]	Master spindle.
V.MPA.COUNTERTYPE[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.COUNTERTYPE[3].1	Axis with index ·1· in the channel ·2·.
V.SP.COUNTERTYPE[3].2	Spindle with index ·2· in the system.
V.[2].SP.COUNTERTYPE[3].1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Remote feedback input.
1	Local feedback input.
2	Feedback input of a Sercos drive.
3	Feedback input of an RCS-S module.

(V.)[ch].MPA.COUNTERID[set].xn
(V.)[ch].MPA.COUNTERID[set].sn
(V.)[ch].SP.COUNTERID[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drives and velocity-Sercos drives.
The variable returns the value of block preparation.

Number of the feedback input associated with the axis.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.COUNTERID[3].Z	Z axis.
V.MPA.COUNTERID[3].S	Spindle S.
V.SP.COUNTERID[3].S	Spindle S.
V.SP.COUNTERID[3]	Master spindle.
V.MPA.COUNTERID[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.COUNTERID[3].1	Axis with index ·1· in the channel ·2·.
V.SP.COUNTERID[3].2	Spindle with index ·2· in the system.
V.[2].SP.COUNTERID[3].1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
1 - 40	Number of the remote feedback input.
1 - 32	Address of the Sercos drive (always the second feedback input).
1 - 2	Number of the local feedback input.
1 - 32	Number of the RCS-S module input.

SET THE DRIVE ASSOCIATED WITH THE AXES OF A MULTI-AXIS GROUP.

(V.)[ch].MPA.DRIVESET[set].xn
(V.)[ch].MPA.DRIVESET[set].sn
(V.)[ch].SP.DRIVESET[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for position-Sercos drives and velocity-Sercos drives.
The variable returns the value of block preparation.

At the axes that belong to a multi-axis group, parameter DRIVESET indicates the set to be activated at the drive after a set or gear change at the CNC (G112 and M41 to M44).

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.DRIVESET[3].Z	Z axis.
V.MPA.DRIVESET[3].S	Spindle S.
V.SP.DRIVESET[3].S	Spindle S.

V.SP.DRIVESET[3]	Master spindle.
V.MPA.DRIVESET[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.DRIVESET[3].1	Axis with index ·1· in the channel ·2·.
V.SP.DRIVESET[3].2	Spindle with index ·2· in the system.
V.[2].SP.DRIVESET[3].1	Spindle with index ·1· in the channel ·2·.

FEEDBACK TYPE OF THE RCS-S MODULE.

(V.)[ch].MPA.FEEDBACKTYPE[set].xn
(V.)[ch].MPA.FEEDBACKTYPE[set].sn
(V.)[ch].SP.FEEDBACKTYPE[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for velocity-Sercos drives.
The variable returns the value of block preparation.

Feedback type of the RCS-S module.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.FEEDBACKTYPE[3].Z	Z axis.
V.MPA.FEEDBACKTYPE[3].S	Spindle S.
V.SP.FEEDBACKTYPE[3].S	Spindle S.
V.SP.FEEDBACKTYPE[3]	Master spindle.
V.MPA.FEEDBACKTYPE[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.FEEDBACKTYPE[3].1	Axis with index ·1· in the channel ·2·.
V.SP.FEEDBACKTYPE[3].2	Spindle with index ·2· in the system.
V.[2].SP.FEEDBACKTYPE[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.SSITYPE[set].xn
(V.)[ch].MPA.SSITYPE[set].sn
(V.)[ch].SP.SSITYPE[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for velocity-Sercos drives.
The variable returns the value of block preparation.

Feedback type connected to the corresponding feedback.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.SSITYPE[3].Z	Z axis.
V.MPA.SSITYPE[3].S	Spindle S.
V.SP.SSITYPE[3].S	Spindle S.
V.SP.SSITYPE[3]	Master spindle.
V.MPA.SSITYPE[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.SSITYPE[3].1	Axis with index ·1· in the channel ·2·.
V.SP.SSITYPE[3].2	Spindle with index ·2· in the system.
V.[2].SP.SSITYPE[3].1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Fagor LA
1	Fagor GA SA SVA
2	Fagor HA-27-D200
3	Fagor HA-23-D90 SA-23-D90 SA-23-D170
4	ABSIND (inductosyn LIN+ABS)
5	ABSIND (inductosyn ROT+ABS)
6	ABSIND (resolver)
7	ABSIND (inductosyn LIN)
8	ABSIND (inductosyn ROT)
9	G user
10	User

(V.)[ch].MPA.SSICLKFREQ[set].xn
(V.)[ch].MPA.SSICLKFREQ[set].sn
(V.)[ch].SP.SSICLKFREQ[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for velocity-Sercos drives.
The variable returns the value of block preparation.

SSI communication frequency.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.SSICLKFREQ[3].Z	Z axis.
V.MPA.SSICLKFREQ[3].S	Spindle S.
V.SP.SSICLKFREQ[3].S	Spindle S.
V.SP.SSICLKFREQ[3]	Master spindle.
V.MPA.SSICLKFREQ[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.SSICLKFREQ[3].1	Axis with index ·1· in the channel ·2·.
V.SP.SSICLKFREQ[3].2	Spindle with index ·2· in the system.
V.[2].SP.SSICLKFREQ[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.SSIDATALENGTH[set].xn
(V.)[ch].MPA.SSIDATALENGTH[set].sn
(V.)[ch].SP.SSIDATALENGTH[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for velocity-Sercos drives.
The variable returns the value of block preparation.

Number of bits of the SSI transmission that make up the position value (coordinate).

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.SSIDATALENGTH[3].Z	Z axis.
V.MPA.SSIDATALENGTH[3].S	Spindle S.
V.SP.SSIDATALENGTH[3].S	Spindle S.
V.SP.SSIDATALENGTH[3]	Master spindle.
V.MPA.SSIDATALENGTH[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.SSIDATALENGTH[3].1	Axis with index ·1· in the channel ·2·.
V.SP.SSIDATALENGTH[3].2	Spindle with index ·2· in the system.
V.[2].SP.SSIDATALENGTH[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.SSIPACKFORMAT[set].xn
(V.)[ch].MPA.SSIPACKFORMAT[set].sn
(V.)[ch].SP.SSIPACKFORMAT[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for velocity-Sercos drives.
The variable returns the value of block preparation.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.SSIPACKFORMAT[3].Z	Z axis.
V.MPA.SSIPACKFORMAT[3].S	Spindle S.
V.SP.SSIPACKFORMAT[3].S	Spindle S.
V.SP.SSIPACKFORMAT[3]	Master spindle.
V.MPA.SSIPACKFORMAT[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.SSIPACKFORMAT[3].1	Axis with index ·1· in the channel ·2·.
V.SP.SSIPACKFORMAT[3].2	Spindle with index ·2· in the system.
V.[2].SP.SSIPACKFORMAT[3].1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Data.
1	Data-CRC.
2	CRC-Data.
3	Data-Alarm.
4	Alarm-Data.
5	Data-CRC-Alarm
6	Alarm-Data-CRC.
7	Data-Alarm-CRC.
8	Alarm-CRC-Data.
9	CRC-Data-Alarm.
10	CRC-Alarm-Data.

(V.)[ch].MPA.SSICRCTYPE[set].xn
(V.)[ch].MPA.SSICRCTYPE[set].sn
(V.)[ch].SP.SSICRCTYPE[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for velocity-Sercos drives.
The variable returns the value of block preparation.

Tipo de CRC.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.SSICRCTYPE[3].Z	Z axis.
V.MPA.SSICRCTYPE[3].S	Spindle S.
V.SP.SSICRCTYPE[3].S	Spindle S.
V.SP.SSICRCTYPE[3]	Master spindle.
V.MPA.SSICRCTYPE[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.SSICRCTYPE[3].1	Axis with index ·1· in the channel ·2·.
V.SP.SSICRCTYPE[3].2	Spindle with index ·2· in the system.
V.[2].SP.SSICRCTYPE[3].1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	"Do not calculate CRC"
1	Fagor checksum.
2	INDUCTOSYN checksum.

(V.)[ch].MPA.SSICRCBITS[set].xn
(V.)[ch].MPA.SSICRCBITS[set].sn
(V.)[ch].SP.SSICRCBITS[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for velocity-Sercos drives.
The variable returns the value of block preparation.

Number of CRC bits of the SSI transmission to check that the transmission is valid.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.SSICRCBITS[3].Z	Z axis.
V.MPA.SSICRCBITS[3].S	Spindle S.
V.SP.SSICRCBITS[3].S	Spindle S.
V.SP.SSICRCBITS[3]	Master spindle.
V.MPA.SSICRCBITS[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.SSICRCBITS[3].1	Axis with index ·1· in the channel ·2·.
V.SP.SSICRCBITS[3].2	Spindle with index ·2· in the system.
V.[2].SP.SSICRCBITS[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.SSISTARTBITS[set].xn
(V.)[ch].MPA.SSISTARTBITS[set].sn
(V.)[ch].SP.SSISTARTBITS[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for velocity-Sercos drives.
The variable returns the value of block preparation.

Número de bits de start.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.SSISTARTBITS[3].Z	Z axis.
V.MPA.SSISTARTBITS[3].S	Spindle S.
V.SP.SSISTARTBITS[3].S	Spindle S.
V.SP.SSISTARTBITS[3]	Master spindle.
V.MPA.SSISTARTBITS[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.SSISTARTBITS[3].1	Axis with index ·1· in the channel ·2·.
V.SP.SSISTARTBITS[3].2	Spindle with index ·2· in the system.
V.[2].SP.SSISTARTBITS[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.SSIALARMBITS[set].xn
(V.)[ch].MPA.SSIALARMBITS[set].sn
(V.)[ch].SP.SSIALARMBITS[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for velocity-Sercos drives.
The variable returns the value of block preparation.

Number of alarm bits.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.SSIALARMBITS[3].Z	Z axis.
V.MPA.SSIALARMBITS[3].S	Spindle S.
V.SP.SSIALARMBITS[3].S	Spindle S.
V.SP.SSIALARMBITS[3]	Master spindle.
V.MPA.SSIALARMBITS[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.SSIALARMBITS[3].1	Axis with index ·1· in the channel ·2·.
V.SP.SSIALARMBITS[3].2	Spindle with index ·2· in the system.
V.[2].SP.SSIALARMBITS[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.SSIALARMLEVEL[set].xn
(V.)[ch].MPA.SSIALARMLEVEL[set].sn
(V.)[ch].SP.SSIALARMLEVEL[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for velocity-Sercos drives.
The variable returns the value of block preparation.

Level of the alarm bits to issue an error.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.SSIALARMLEVEL[3].Z	Z axis.
V.MPA.SSIALARMLEVEL[3].S	Spindle S.
V.SP.SSIALARMLEVEL[3].S	Spindle S.
V.SP.SSIALARMLEVEL[3]	Master spindle.
V.MPA.SSIALARMLEVEL[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.SSIALARMLEVEL[3].1	Axis with index ·1· in the channel ·2·.
V.SP.SSIALARMLEVEL[3].2	Spindle with index ·2· in the system.
V.[2].SP.SSIALARMLEVEL[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.SSIDATAMODE[set].xn
(V.)[ch].MPA.SSIDATAMODE[set].sn
(V.)[ch].SP.SSIDATAMODE[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for velocity-Sercos drives.
The variable returns the value of block preparation.

Transmission mode.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.SSIDATAMODE[3].Z	Z axis.
V.MPA.SSIDATAMODE[3].S	Spindle S.
V.SP.SSIDATAMODE[3].S	Spindle S.
V.SP.SSIDATAMODE[3]	Master spindle.
V.MPA.SSIDATAMODE[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.SSIDATAMODE[3].1	Axis with index ·1· in the channel ·2·.
V.SP.SSIDATAMODE[3].2	Spindle with index ·2· in the system.
V.[2].SP.SSIDATAMODE[3].1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	The LSB (Least Significant Bit) is the first one.
1	The MSB (Most Significant Bit) is the first one.

(V.)[ch].MPA.STARTDELAY[set].xn
(V.)[ch].MPA.STARTDELAY[set].sn
(V.)[ch].SP.STARTDELAY[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for velocity-Sercos drives.
The variable returns the value of block preparation.

Number of clocks to wait between the first down flank and the first up flank.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.STARTDELAY[3].Z	Z axis.
V.MPA.STARTDELAY[3].S	Spindle S.
V.SP.STARTDELAY[3].S	Spindle S.
V.SP.STARTDELAY[3]	Master spindle.
V.MPA.STARTDELAY[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.STARTDELAY[3].1	Axis with index ·1· in the channel ·2·.
V.SP.STARTDELAY[3].2	Spindle with index ·2· in the system.
V.[2].SP.STARTDELAY[3].1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].MPA.SSIRESOL[set].xn
(V.)[ch].MPA.SSIRESOL[set].sn
(V.)[ch].SP.SSIRESOL[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for velocity-Sercos drives.
The variable returns the value of block preparation.

Digital counting (feedback) resolution.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.SSIRESOL[3].Z	Z axis.
V.MPA.SSIRESOL[3].S	Spindle S.
V.SP.SSIRESOL[3].S	Spindle S.
V.SP.SSIRESOL[3]	Master spindle.
V.MPA.SSIRESOL[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.SSIRESOL[3].1	Axis with index ·1· in the channel ·2·.
V.SP.SSIRESOL[3].2	Spindle with index ·2· in the system.
V.[2].SP.SSIRESOL[3].1	Spindle with index ·1· in the channel ·2·.

(V.)MPMAN.COUNTERTYPE[hw]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Handwheel [hw]. Type of handwheel feedback input.

Syntax.

·hw· Handwheel number.

V.MPMAN.COUNTERTYPE[1]	Handwheel ·1·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Handwheel connected to the remote feedback modules.
1	Handwheel connected to the keyboards.
2	Handwheel connected to the local feedback inputs.
3	Handwheel connected to an RCS-S module.

(V.)MPMAN.COUNTERID[hw]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Handwheel [hw]. Feedback input associated with the handwheel.

Syntax.

·hw· Handwheel number.

V.MPMAN.COUNTERID[1]	Handwheel ·1·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
-1 -2 -3	Handwheel connected to the first keyboard.
-4 -5 -6	Handwheel connected to the second keyboard.
-7 -8 -9	Handwheel connected to the third keyboard.
1 ·· 40	Feedback input of the remote modules.
1 ·· 2	Local feedback input.
1 ·· 32	Feedback input of the RCS-S module.

DELAY ESTIMATE AT THE DRIVE.

(V.)[ch].MPA.AXDELAY[set].xn
(V.)[ch].MPA.AXDELAY[set].sn
(V.)[ch].SP.AXDELAY[set].sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the value of block preparation.

Parameter AXDELAY is an estimate of the drive's delay when applying the velocity command sent by the CNC.

Syntax.

·ch· Channel number.

·set· Parameter set.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.AXDELAY[3].Z	Z axis.
V.MPA.AXDELAY[3].S	Spindle S.
V.SP.AXDELAY[3].S	Spindle S.
V.SP.AXDELAY[3]	Master spindle.
V.MPA.AXDELAY[3].4	Axis or spindle with logic number ·4·.
V.[2].MPA.AXDELAY[3].1	Axis with index ·1· in the channel ·2·.
V.SP.AXDELAY[3].2	Spindle with index ·2· in the system.
V.[2].SP.AXDELAY[3].1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
Standard.	For Fagor drives, that do not require any delay compensation.
0.	The CNC does not apply delay compensation at the drive.
1 to 127.	Delay cycles at the drive. The CNC automatically compensates for the delay differences between the axes of the channel.

10.8 Variables related to machine parameters for JOG mode.

HANDWHEEL CONFIGURATION.

(V.)MPMAN.NMPG

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Number of handwheels connected to the CNC.

V.MPMAN.NMPG

(V.)MPMAN.COUNTERTYPE[hw]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Handwheel [hw]. Type of handwheel feedback input.

Syntax.

·hw· Handwheel number.

V.MPMAN.COUNTERTYPE[1] Handwheel ·1·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Handwheel connected to the remote feedback modules.
1	Handwheel connected to the keyboards.
2	Handwheel connected to the local feedback inputs.
3	Handwheel connected to an RCS-S module.

(V.)MPMAN.COUNTERID[hw]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Handwheel [hw]. Feedback input associated with the handwheel.

Syntax.

·hw· Handwheel number.

V.MPMAN.COUNTERID[1] Handwheel ·1·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
-1 -2 -3	Handwheel connected to the first keyboard.
-4 -5 -6	Handwheel connected to the second keyboard.
-7 -8 -9	Handwheel connected to the third keyboard.
1 ·· 40	Feedback input of the remote modules.
1 ·· 2	Local feedback input.
1 ·· 32	Feedback input of the RCS-S module.

(V.)MPMAN.MPGAXIS[hw]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Handwheel [hw]. Logic number of the axis associated with the handwheel.

If the variable returns a ·0· value, it means that it is a general handwheel that may be used to move any axis.

Syntax.

·hw· Handwheel number.

V.MPMAN.MPGAXIS[1]	Handwheel ·1·.

Remarks.

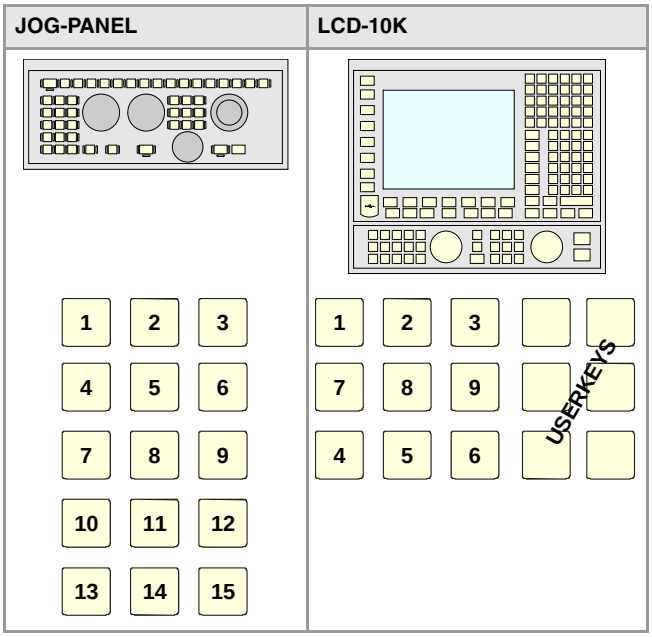
The logic number of the axes is determined by the order in which the axes have been defined in the machine parameter table. The first axis of the table will be logic axis -1- and so on.

CONFIGURE THE JOG KEYS.

(V.)MPMAN.JOGKEYDEF[jk]
(V.)MPMAN.JOGKEY2DEF[jk]
..
(V.)MPMAN.JOGKEY8DEF[jk]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Jog key [jk]. Axis and moving direction.



Syntax.

·jk· Jog key number.

V.MPMAN.JOGKEYDEF[11]	Jog key ·11· of first keyboard.
V.MPMAN.JOGKEY3DEF[11]	Jog key ·11· of third keyboard.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	The key has no function assigned to it.
1 .. 16	The key is assigned to the logic axis 1, 2, .., 16 for movements in the positive direction.
-1 .. -16	The key is assigned to the logic axis 1, 2, .., 16 for movements in the negative direction.
101 .. 116	The key is assigned to the logic axis 1, 2, .., 16.
300	The key is assigned to the rapid movement.
301	The key is assigned to the movement in the positive direction.
302	The key is assigned to the movement in the negative direction.

Remarks.

The logic number of the axes is determined by the order in which the axes have been defined in the machine parameter table. The first axis of the table will be logic axis -1- and so on.

(V.)MPMAN.JOGTYPE

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Behavior of the JOG keys.

V.MPMAN.JOGTYPE

Values of the variable.

This variable returns one of the following values.

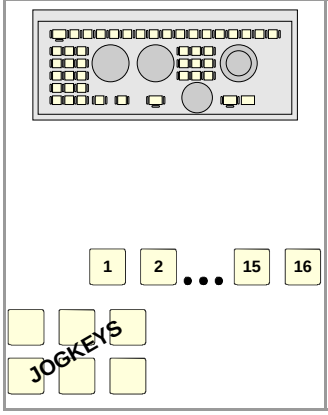
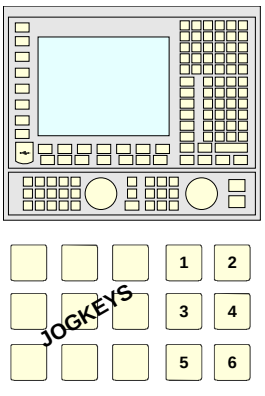
Value.	Meaning.
0	Pressed axis. The axis will move while both keys are pressed, the axis key and the direction key.
1	Selected axis. Pressing the axis key selects the axis. The axis will move while the direction key is kept pressed.

CONFIGURE THE USER KEYS AS JOG KEYS

(V.)MPMAN.USERKEYDEF[uk]
(V.)MPMAN.USERKEY2DEF[uk]
..
(V.)MPMAN.USERKEY8DEF[uk]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

User key [uk] as jog key.

JOG-PANEL	LCD-10K
	

Syntax.

·uk· User key number.

V.MPMAN.USERKEYDEF[7]	User key ·7· of first keyboard.
V.MPMAN.USERKEY3DEF[7]	User key ·7· of third keyboard.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	The key has no function assigned to it.
1 ·· 16	The key is assigned to the logic axis 1, 2, ··, 16 for movements in the positive direction.
-1 ·· -16	The key is assigned to the logic axis 1, 2, ··, 16 for movements in the negative direction.
101 ·· 116	The key is assigned to the logic axis 1, 2, ··, 16.
300	The key is assigned to the rapid movement.
301	The key is assigned to the movement in the positive direction.
302	The key is assigned to the movement in the negative direction.

Remarks.

The logic number of the axes is determined by the order in which the axes have been defined in the machine parameter table. The first axis of the table will be logic axis -1- and so on.

(V.)MPMAN.HBLS

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

This parameter indicates whether there is a portable panel HBLS connected to the CNC through the serial line or not.

V.MPMAN.HBLS

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

10.9 Variables related to machine parameters for M functions.

(V.)MPM.MTABLESIZE

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

"M" function table. Number of table elements.

V.MPM.MTABLESIZE	

(V.)MPM.MNUM[pos]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Position [pos] of the "M" function table. "M" function number.

If the variable returns a ·-1· value, it means that no "M" function has been defined in that position.

Syntax.

·pos· Position inside the "M" function table.

V.MPM.MNUM[12]	Position ·12· of the "M" function table.

(V.)MPM.SYNCHTYPE[pos]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Position [pos] of the "M" function table. Type of synchronism.

Since the M functions may be programmed together with the movement of the axes, in the same block, it must be indicated when the function is to be sent out to the PLC and when it will be checked that it has been already executed (synchronization). The M functions may be sent and/or synchronized before or after the movement.

Syntax.

·pos· Position inside the "M" function table.

V.MPM.SYNCHTYPE[12]	Position ·12· of the "M" function table.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Without synchronization.
2	The M function is both sent to the PLC and synchronized before the movement.
4	The M function is sent to the PLC before the movement and synchronized after the movement.
8	The M function is both sent to the PLC and synchronized after the movement.

(V.)MPM.MPROGNAME[pos]

Variable to be read via interface.

Position [pos] of the "M" function table. Name of the associated subroutine.

MPM.MPROGNAME[12]	Position ·12· of the "M" function table.

(V.)MPM.MTIME[pos]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Position [pos] of the "M" function table. Estimated execution time (in milliseconds).

Syntax.

·pos· Position inside the "M" function table.

V.MPM.MTIME[12]	Position ·12· of the "M" function table.

(V.)MPM.MPLC[pos]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Position [pos] of the "M" function table. Send the M function to the PLC during block search.

Syntax.

·pos· Position inside the "M" function table.

V.MPM.MPLC[12]	Position ·12· of the "M" function table.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

10.10 Variables related to kinematic machine parameters.

KINEMATICS CONFIGURATION.

(V.)MPK.NKIN

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Kinematics table. Number of kinematics defined.

V.MPK.NKIN

(V.)MPK.TYPE[kin]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Kinematics [kin]. Kinematics type.

Syntax.

·kin· Number of the kinematics.

V.MPK.TYPE[3] Kinematics ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
1	Orthogonal or spherical spindle head YX.
2	Orthogonal or spherical spindle head ZX.
3	Orthogonal or spherical spindle head XY.
4	Orthogonal or spherical spindle head ZY.
5	Swivel (angular) spindle XZ.
6	Swivel (angular) spindle YZ.
7	Swivel (angular) spindle ZX.
8	Swivel (angular) spindle ZY.
9	Rotary table AB.
10	Rotary table AC.
11	Rotary table BA.
12	Rotary table BC.
13	Spindle - table AB.
14	Spindle - table AC.
15	Spindle - table BA.
16	Spindle - table BC.
17	Orthogonal spindle with three rotary axes ABA.
18	Orthogonal spindle with three rotary axes ACA.
19	Orthogonal spindle with three rotary axes ACB.
20	Orthogonal spindle with three rotary axes BAB.
21	Orthogonal spindle with three rotary axes BCA.
22	Orthogonal spindle with three rotary axes BCB.
23	Orthogonal spindle with three rotary axes CAB.
24	Orthogonal spindle with three rotary axes CBA.
41	"C" axis. Machining on the face of the part when ALIGNC = YES.

Value.	Meaning.
42	"C" axis. Machining on the face of the part when ALIGNC = NO.
43	"C" axis. Machining of the turning side of the part.
100 ·· 105	OEM Kinematics.

(V.)MPK.TDATAkin[nb]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Kinematics [kin]. Value of parameter TDATA [nb].

Syntax.

·kin· Number of the kinematics.

·nb· Parameter number.

V.MPK.TDATA2[34]	Kinematics ·2·. Value of parameter TDATA34.

(V.)MPK.TDATA_Ikin[nb]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Kinematics [kin]. Value of parameter TDATA_I [nb].

Syntax.

·kin· Number of the kinematics.

·nb· Parameter number.

V.MPK.TDATA2[23]	Kinematics ·2·. Value of parameter TDATA_I23.

(V.)MPK.NKINAX[kin]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Kinematics [kin]. Number of axes of the kinematics.

Syntax.

·kin· Number of the kinematics.

V.MPK.NKINAX[2]	Kinematics ·2·.

(V.)MPK.PARAM_D_SIZE[kin]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Kinematics [kin]. Number of parameters in decimal format

Syntax.

·kin· Number of the kinematics.

V.MPK.PARAM_D_SIZE[2]	Kinematics ·2·.

(V.)MPK.PARAM_I_SIZE[kin]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Kinematics [kin]. Number of parameters in integer format.

Syntax.

·kin· Number of the kinematics.

V.MPK.PARAM_I_SIZE[2]	Kinematics ·2·.

(V.)MPK.AUXCTE_SIZE[kin]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Kinematics [kin]. Size of the area of auxiliary variables.

Syntax.

·kin· Number of the kinematics.

V.MPK.AUXCTE_SIZE[2]	Kinematics ·2·.

(V.)MPK.KINDATA_SIZE[kin]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Kinematics [kin]. Size of the area for general purpose data.

Syntax.

·kin· Number of the kinematics.

V.MPK.KINDATA_SIZE[2]	Kinematics ·2·.

CONFIGURATION OF ANGULAR TRANSFORMATIONS.

(V.)MPK.NANG

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Number of angular transformations defined.

V.MPK.NANG

(V.)MPK.ANGAXNA[ang]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Angular transformation [ang]. Logic number of the angular axis.

Syntax.

·ang· Number of the angular transformation.

V.MPK.ANGAXNA[2]	Angular transformation ·2·.

Remarks.

The logic number of the axes is determined by the order in which the axes have been defined in the machine parameter table. The first axis of the table will be logic axis -1- and so on.

(V.)MPK.ORTAXNA[ang]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Angular transformation [ang]. Logic number of the orthogonal axis.

Syntax.

·ang· Number of the angular transformation.

V.MPK.ORTAXNA[2]	Angular transformation ·2·.

Remarks.

The logic number of the axes is determined by the order in which the axes have been defined in the machine parameter table. The first axis of the table will be logic axis -1- and so on.

(V.)MPK.ANGANTR[ang]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Angular transformation [ang]. Angle between the Cartesian axis and the incline axis.

Syntax.

·ang· Number of the angular transformation.

V.MPK.ORTAXNA[2]	Angular transformation ·2·.

(V.)MPK.OFFANGAX[ang]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Angular transformation [ang]. Offset of the origin of the angular transformation.

Syntax.

·ang· Number of the angular transformation.

V.MPK.OFFANGAX[2]	Angular transformation ·2·.

10.11 Variables related to machine parameters for the tool magazine.

(V.)TM.NTOOLMZ

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Number of tool magazines.

V.TM.NTOOLMZ

(V.)TM.MZGROUND

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Ground tools are permitted (manual load).

V.TM.MZGROUND

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

(V.)TM.MZSIZE[mz]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Tool magazine [mz]. Size of the tool magazine (number of pockets).

Syntax.

·mz· Type of tool magazine.

V.TM.MZSIZE[2] Magazine ·2·.

(V.)TM.MZRANDOM[mz]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Tool magazine [mz]. Random magazine.

In a random tool magazine, the tools can occupy any position. In a non-random tool magazine, the tools always occupy the same position.

Syntax.

·mz· Type of tool magazine.

V.TM.MZRANDOM[2] Magazine ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	It is not a random tool magazine.
1	It is a random tool magazine.

(V.)TM.MZTYPE[mz]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Tool magazine [mz]. Type of magazine.

Syntax.

·mz· Type of tool magazine.

V.TM.MZTYPE[2]	Magazine ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
1	Asynchronous.
2	Synchronous.
3	Turret
4	Synchronous with 2 arms.
5	Synchronous with 1 arm.

(V.)TM.MZCYCLIC[mz]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Tool magazine [mz]. Cyclic tool changer.

A cyclic tool changer requires a tool change command (function M06) after searching a tool and before searching the next one. In a non-cyclic magazine, it is possible to perform several consecutive tool searches without necessarily having to change the tool.

Syntax.

·mz· Type of tool magazine.

V.TM.MZCYCLIC[2]	Magazine ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	It is not a cyclic tool magazine.
1	It is a cyclic tool magazine.

(V.)TM.MZOPTIMIZED[mz]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Tool magazine [mz]. Tool management.

When programming several T's in a row without M06, a magazine with optimized management only looks for the tool to be changed; a magazine without optimized management looks for all the tools.

Syntax.

·mz· Type of tool magazine.

V.TM.MZOPTIMIZED[2]	Magazine ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	The tool magazine has no optimized management.
1	The tool magazine has optimized management.

(V.)TM.MZRESPECTSIZE[mz]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Tool magazine [mz]. In the random magazine, look for pockets of the same size.

Syntax.

·mz· Type of tool magazine.

V.TM.MZRESPECTSIZE[2]	Magazine ·2·.

(V.)TM.MZM6ALONE[mz]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Tool magazine [mz]. Action after executing an M6 without having selected a tool.

Syntax.

·mz· Type of tool magazine.

V.TM.MZM6ALONE[2]	Magazine ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No action is carried out.
1	The CNC issues a warning.
2	The CNC issues an error message.

10.12 Variables related to OEM machine parameters.

GENERIC OEM PARAMETERS.

(V.)MTB.SIZE

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Number of OEM parameters.

V.MPB.SIZE

(V.)MTB.P[i]

Variable that can be read and written via program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Value of the OEM parameter [nb]

Syntax.

·nb· Parameter number.

V.MTB.P[10]	Value of the OEM parameter P10.

Remarks.

When reading this variable from the PLC, it truncates the decimal side. If the value of the parameter is 54.9876, when read from the PLC, it will return the value 54.

Machine parameter.	Reading from the PLC. MTB.P[0]	Reading from the PLC. V.MTB.PF[0]
P0 = 54.9876	54	549876
P0 = -34.1234	-34	-341234

It must be borne in mind that reading and writing these variables interrupts block preparation affecting program execution time. If the value of the parameter is not going to be changed during execution, it is recommended to read the MTB variables at the beginning of the program using arithmetic parameters (local or global) and use the global ones throughout the program.

(V.)MTB.PF[i]

Variable that can be read and written via program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Value of the OEM parameter [nb] Value per 10000.

Syntax.

·nb· Parameter number.

V.MTB.PF[10]	Value of the OEM parameter P10.

Remarks.

When reading this variable from the PLC, it returns the value in ten-thousandths. If the value of the parameter is 54.9876, when read from the PLC, it will return the value 549876.

Machine parameter.	Reading from the PLC. MTB.P[0]	Reading from the PLC. V.MTB.PF[0]
P0 = 54.9876	54	549876
P0 = -34.1234	-34	-341234

It must be borne in mind that reading and writing these variables interrupts block preparation affecting program execution time. If the value of the parameter is not going to be changed during execution, it is recommended to read the MTB variables at the beginning of the

program using arithmetic parameters (local or global) and use the global ones throughout the program.

READING DRIVE VARIABLES.

(V.)DRV.SIZE

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Number of variables to be consulted at the drive.

V.DRV.SIZE

(V.)DRV.name
(V.)DRV.name.xn
(V.)DRV.name.sn

Variable that can be read and written via program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Variables defined in parameter DRIVEVAR that grant access to the variables and/or parameters of the digital drives.

- Sercos. DRV variables can access the variables of the drive. These variables may be read-only or read and written.
- Mechatrolink. DRV variables can access the variables and/or parameters of the drive. The variables of the drive can only be read, whereas the parameters may be read or written.

Syntax.

·name· Mnemonic name set at the machine parameters.

·xn· Axis name.

·sn· Spindle name.

V.DRV.FEED	Value of the variable defined as FEED.
V.DRV.AXISFEED.Z	Value of the variable defined as AXISFEED for the Z axis.
V.DRV.AXISFEED.S	Value of the variable defined as AXISFEED for the spindle S.

10.13 Variables associated with the status and resources of the PLC.

PLC STATUS.

(V.)PLC.STATUS

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

PLC status.

V.PLC.STATUS

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	PLC stopped.
1	PLC running.

PLC RESOURCES.

(V.)PLC.I[nb]

Variable that can be read and written via program and interface.
The variable returns the execution value; reading it interrupts block preparation.

Status of PLC digital input [nb].

Syntax.

·nb· Number of the digital input.

V.PLC.I[122]	Status of PLC digital input ·122·.

(V.)PLC.O[nb]

Variable that can be read and written via program and interface.
The variable returns the execution value; reading it interrupts block preparation.

Status of PLC digital output [nb].

Syntax.

·nb· Number of the digital output.

V.PLC.O[243]	Status of PLC digital output ·243·.

(V.)PLC.LI[nb]

Variable that can be read and written via program and interface.
The variable returns the execution value; reading it interrupts block preparation.

Status of PLC local digital input [nb].

Syntax.

·nb· Number of the digital input.

V.PLC.LI[2]	Status of PLC local digital input ·2·.

(V.)PLC.LO[nb]

Variable that can be read and written via program and interface.
The variable returns the execution value; reading it interrupts block preparation.

Status of PLC local digital output [nb].

Syntax.

·nb· Number of the digital output.

V.PLC.LO[3]	Status of PLC local digital output ·3·.

(V.)PLC.M[nb]

Variable that can be read and written via program and interface.
The variable returns the execution value; reading it interrupts block preparation.

Status of PLC mark [nb].

Syntax.

·nb· Mark number.

V.PLC.M[111]	Status of PLC mark ·111·.

(V.)PLC.R[nb]

Variable that can be read and written via program and interface.
The variable returns the execution value; reading it interrupts block preparation.

Status of PLC register [nb].

Syntax.

·nb· Register number.

V.PLC.R[200]	Status of PLC register ·200·.

(V.)PLC.T[nb]

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.

Status of PLC timer [nb]

Syntax.

·nb· Timer number.

V.PLC.T[8]	Status of PLC timer ·8·.

(V.)PLC.C[nb]

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.

Status of PLC counter [nb].

Syntax.

·nb· Counter number.

V.PLC.C[16]	Status of PLC counter ·16·.

PLC MESSAGES.

(V.)PLC.MSG[msg]

Variable that can be read and written via program and interface.
The variable returns the execution value; reading it interrupts block preparation.

Status of PLC message [msg].

Syntax.

·msg· Message number.

V.PLC.MSG[87]	Status of message 87.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Message off.
1	Message on.

(V.)PLC.PRIORMSG

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

Active message with the highest priority (the one with the lowest number among the active ones).

V.PLC.PRIORMSG

(V.)PLC.EMERGMSG

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

Active emerging message (the one shown at full screen).

V.PLC.EMERGMSG

PLC ERRORS.

(V.)PLC.ERR[err]

Variable that can be read and written via program and interface.
The variable returns the execution value; reading it interrupts block preparation.

Status of PLC error [err].

Syntax.

·err· Error number.

V.PLC.ERR[62]	Status of error 62.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Error off.
1	Error on.

(V.)PLC.PRIORERR

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

Active error with the highest priority (the one with the lowest number among the active ones).

V.PLC.PRIORERR

PLC CLOCKS.

(V.)PLC.TIMER

Variable that can be read and written via program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

Value of the freely available clock (in seconds).

```
V.PLC.TIMER
```

This variable may be used to consult and/or modify the timing of the clock. Value in seconds.

Remarks.

The PLC "TIMER" is enabled or disabled with the PLC mark TIMERON. This clock is timing when TIMERON=1.

(V.)PLC.CLKnb

Variable that can be read and written via program and interface.
The variable returns the execution value; reading it interrupts block preparation.

Status of PLC clock nb.

```
V.PLC.CLK128
```

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Clock off.
1	Clock on.

10.14 PLC consulting logic signals; general.

(V.)PLC.CNCREADY

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The CNC activates the mark when it is not in an error state.

V.PLC.CNCREADY

(V.)PLC.READY

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The channel activates the mark when it is not in an error state.

V.PLC.READYC1	Channel ·1·.
V.PLC.READYC2	Channel ·2·.
V.PLC.READYC3	Channel ·3·.
V.PLC.READYC4	Channel ·4·.

(V.)PLC.SERCOSRDY

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The CNC activates the mark when the Sercos ring has been initialized properly.

V.PLC.SERCOSRDY

(V.)PLC.START

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The channel activates this mark when the [START] key has been pressed.

V.PLC.START	Channel ·1·.
V.PLC.STARTC1	Channel ·1·.
V.PLC.STARTC2	Channel ·2·.
V.PLC.STARTC3	Channel ·3·.
V.PLC.STARTC4	Channel ·4·.

(V.)PLC.RESETOUT

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

When pressing the [RESET] key or when the PLC activates the RESETIN mark, the CNC assumes the initial conditions and activates the RESETOUT mark.

V.PLC.RESETOUT	Channel ·1·.
V.PLC.RESETOUTC1	Channel ·1·.
V.PLC.RESETOUTC2	Channel ·2·.
V.PLC.RESETOUTC3	Channel ·3·.
V.PLC.RESETOUTC4	Channel ·4·.

(V.)PLC.FHOUT

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The CNC channel activates this mark when the execution of the part-program is interrupted.

V.PLC.FHOUT	Channel ·1·.
V.PLC.FHOUTC1	Channel ·1·.
V.PLC.FHOUTC2	Channel ·2·.
V.PLC.FHOUTC3	Channel ·3·.
V.PLC.FHOUTC4	Channel ·4·.

(V.)PLC._ALARM

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The channel turns this mark off when there is an alarm or an emergency in the channel.

V.PLC._ALARM	Channel ·1·.
V.PLC._ALARMC1	Channel ·1·.
V.PLC._ALARMC2	Channel ·2·.
V.PLC._ALARMC3	Channel ·3·.
V.PLC._ALARMC4	Channel ·4·.

(V.)PLC.MANUAL

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The channel activates the mark when the jog mode is selected.

V.PLC.MANUAL	Channel ·1·.
V.PLC.MANUALC1	Channel ·1·.
V.PLC.MANUALC2	Channel ·2·.
V.PLC.MANUALC3	Channel ·3·.
V.PLC.MANUALC4	Channel ·4·.

(V.)PLC.AUTOMAT

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The channel activates the mark when the automatic mode is selected.

V.PLC.AUTOMAT	Channel ·1·.
V.PLC.AUTOMATC1	Channel ·1·.
V.PLC.AUTOMATC2	Channel ·2·.
V.PLC.AUTOMATC3	Channel ·3·.
V.PLC.AUTOMATC4	Channel ·4·.

(V.)PLC.MDI

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The channel activates the mark when the MDI/MDA mode is selected.

V.PLC.MDI	Channel ·1·.
V.PLC.MDIC1	Channel ·1·.
V.PLC.MDIC2	Channel ·2·.
V.PLC.MDIC3	Channel ·3·.
V.PLC.MDIC4	Channel ·4·.

(V.)PLC.SBOUT

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The channel activates the mark when the single-block mode is selected.

V.PLC.SBOUT	Channel ·1·.
V.PLC.SBOUTC1	Channel ·1·.
V.PLC.SBOUTC2	Channel ·2·.
V.PLC.SBOUTC3	Channel ·3·.
V.PLC.SBOUTC4	Channel ·4·.

(V.)PLC.INCYCLE

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The channel activates the mark when it is executing a block or moving an axis.

V.PLC.INCYCLE	Channel ·1·.
V.PLC.INCYCLEC1	Channel ·1·.
V.PLC.INCYCLEC2	Channel ·2·.
V.PLC.INCYCLEC3	Channel ·3·.
V.PLC.INCYCLEC4	Channel ·4·.

(V.)PLC.RAPID

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The channel activates the mark when it is executing a rapid positioning (G00).

V.PLC.RAPID	Channel ·1·.
V.PLC.RAPIDC1	Channel ·1·.
V.PLC.RAPIDC2	Channel ·2·.
V.PLC.RAPIDC3	Channel ·3·.
V.PLC.RAPIDC4	Channel ·4·.

(V.)PLC.ZERO

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The channel activates this mark when searching home (G74).

V.PLC.ZERO	Channel ·1·.
V.PLC.ZEROC1	Channel ·1·.
V.PLC.ZEROC2	Channel ·2·.
V.PLC.ZEROC3	Channel ·3·.
V.PLC.ZEROC4	Channel ·4·.

(V.)PLC.PROBE

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The channel activates the mark when it is executing a probing movement (G100).

V.PLC.PROBE	Channel ·1·.
V.PLC.PROBEC1	Channel ·1·.
V.PLC.PROBEC2	Channel ·2·.
V.PLC.PROBEC3	Channel ·3·.
V.PLC.PROBEC4	Channel ·4·.

(V.)PLC.THREAD

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The channel activates the mark when it is executing an electronic threading (G33).

V.PLC.THREAD	Channel ·1·.
V.PLC.THREADC1	Channel ·1·.
V.PLC.THREADC2	Channel ·2·.
V.PLC.THREADC3	Channel ·3·.
V.PLC.THREADC4	Channel ·4·.

(V.)PLC.TAPPING

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The channel activates the mark when it is executing a tapping canned cycle.

V.PLC.TAPPING	Channel ·1·.
V.PLC.TAPPINGC1	Channel ·1·.
V.PLC.TAPPINGC2	Channel ·2·.
V.PLC.TAPPINGC3	Channel ·3·.
V.PLC.TAPPINGC4	Channel ·4·.

(V.)PLC.RIGID

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The channel activates the mark when it is executing a rigid tapping cycle (G63).

V.PLC.RIGID	Channel ·1·.
V.PLC.RIGIDC1	Channel ·1·.
V.PLC.RIGIDC2	Channel ·2·.
V.PLC.RIGIDC3	Channel ·3·.
V.PLC.RIGIDC4	Channel ·4·.

(V.)PLC.CSS

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The channel activates this mark when constant surface speed (G96) is active.

V.PLC.CSS	Channel ·1·.
V.PLC.CSSC1	Channel ·1·.
V.PLC.CSSC2	Channel ·2·.
V.PLC.CSSC3	Channel ·3·.
V.PLC.CSSC4	Channel ·4·.

(V.)PLC.INTEREND

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The channel activates the mark when the theoretical movement of the axes is completed.

V.PLC.INTEREND	Channel ·1·.
V.PLC.INTERENDC1	Channel ·1·.
V.PLC.INTERENDC2	Channel ·2·.
V.PLC.INTERENDC3	Channel ·3·.
V.PLC.INTERENDC4	Channel ·4·.

(V.)PLC.INPOSI

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The channel activates the mark when the axes are in position. This mark also remains active during the movement of independent axes.

V.PLC.INPOSI	Channel ·1·.
V.PLC.INPOSIC1	Channel ·1·.
V.PLC.INPOSIC2	Channel ·2·.
V.PLC.INPOSIC3	Channel ·3·.
V.PLC.INPOSIC4	Channel ·4·.

(V.)PLC.SPN1
(V.)PLC.SPN2
(V.)PLC.SPN3
(V.)PLC.SPN4
(V.)PLC.SPN5

(V.)PLC.SPN6
(V.)PLC.SPN7

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The channel indicates in these registers which spindle of the channel the M functions pending execution are addressed to.

There is a register for each channel. The mnemonics for each channel are the following. Here is an example of the mnemonics for SPN1; it is the same for the rest of the registers.

V.PLC.SPN1C1	Channel ·1·.
V.PLC.SPN1C2	Channel ·2·.
V.PLC.SPN1C3	Channel ·3·.
V.PLC.SPN1C4	Channel ·4·.

(V.)PLC.MFUN1
(V.)PLC.MFUN2
(V.)PLC.MFUN3
(V.)PLC.MFUN4
(V.)PLC.MFUN5
(V.)PLC.MFUN6
(V.)PLC.MFUN7

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The channel indicates in these registers the H functions pending execution.

There is a register for each channel. The mnemonics for each channel are the following. Here is an example of the mnemonics for MFUN1; it is the same for the rest of the registers.

V.PLC.MFUN1	Channel ·1·.
V.PLC.MFUN1C1	Channel ·1·.
V.PLC.MFUN1C2	Channel ·2·.
V.PLC.MFUN1C3	Channel ·3·.
V.PLC.MFUN1C4	Channel ·4·.

(V.)PLC.HFUN1
(V.)PLC.HFUN2
(V.)PLC.HFUN3
(V.)PLC.HFUN4
(V.)PLC.HFUN5
(V.)PLC.HFUN6
(V.)PLC.HFUN7

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The channel indicates in these registers the H functions pending execution.

There is a register for each channel. The mnemonics for each channel are the following. Here is an example of the mnemonics for HFUN1; it is the same for the rest of the registers.

V.PLC.HFUN1	Channel ·1·.
V.PLC.HFUN1C1	Channel ·1·.
V.PLC.HFUN1C2	Channel ·2·.
V.PLC.HFUN1C3	Channel ·3·.
V.PLC.HFUN1C4	Channel ·4·.

(V.)PLC.MSTROBE

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The channel activates this mark to indicate to the PLC that it must execute the M functions indicated in MFUN1 through MFUN7.

V.PLC.MSTROBE	Channel ·1·.
V.PLC.MSTROBEC1	Channel ·1·.
V.PLC.MSTROBEC2	Channel ·2·.
V.PLC.MSTROBEC3	Channel ·3·.
V.PLC.MSTROBEC4	Channel ·4·.

(V.)PLC.HSTROBE

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The channel activates this mark to indicate to the PLC that it must execute the H functions indicated in HFUN1 through HFUN7.

V.PLC.HSTROBE	Channel ·1·.
V.PLC.HSTROBEC1	Channel ·1·.
V.PLC.HSTROBEC2	Channel ·2·.
V.PLC.HSTROBEC3	Channel ·3·.
V.PLC.HSTROBEC4	Channel ·4·.

(V.)PLC.SFUN1
(V.)PLC.SFUN2
(V.)PLC.SFUN3
(V.)PLC.SFUN4

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The channel indicates in these registers the speed programmed in each spindle.

V.PLC.SFUN1	Spindle ·1·.
V.PLC.SFUN2	Spindle ·2·.
V.PLC.SFUN3	Spindle ·3·.
V.PLC.SFUN4	Spindle ·4·.

(V.)PLC.SSTROBE

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The channel activates this mark to indicate to the PLC that a new spindle speed has been selected in registers SFUN1 through SFUN4.

V.PLC.SSTROBE	Channel ·1·.
V.PLC.SSTROBE1	Channel ·1·.
V.PLC.SSTROBE2	Channel ·2·.
V.PLC.SSTROBE3	Channel ·3·.
V.PLC.SSTROBE4	Channel ·4·.

(V.)PLC.DM00
(V.)PLC.DM01
(V.)PLC.DM02
(V.)PLC.DM06
(V.)PLC.DM08

(V.)PLC.DM09
(V.)PLC.DM30

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The CNC indicates in these marks the status of the M functions. This mark is active if the M function is active.

Each one of functions M00, M01, M02, M06, M08, M09, M30 has a mark for each channel. Here is an example of the mnemonics for DM00; it is the same for the rest of the marks (DM01, DM02, DM06, DM08, DM09, DM30).

V.PLC.DM00	Channel ·1·.
V.PLC.DM00C1	Channel ·1·.
V.PLC.DM00C2	Channel ·2·.
V.PLC.DM00C3	Channel ·3·.
V.PLC.DM00C4	Channel ·4·.

(V.)PLC.DM03
(V.)PLC.DM04
(V.)PLC.DM05
(V.)PLC.DM19
(V.)PLC.DM41
(V.)PLC.DM42
(V.)PLC.DM43
(V.)PLC.DM44

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The CNC indicates in these marks the status of the spindle M functions. This mark is active if the M function is active.

Each one of functions M03, M04, M05, M19, M41, M42, M43, M44 has a mark for each spindle. Here is an example of the mnemonics for DM03; it is the same for the rest of the marks (DM04, DM05, DM19, DM41, DM42, DM43, DM44).

V.PLC.DM03	Spindle ·1·.
V.PLC.DM03SP1	Spindle ·1·.
V.PLC.DM03SP2	Spindle ·2·.
V.PLC.DM03SP3	Spindle ·3·.
V.PLC.DM03SP4	Spindle ·4·.

(V.)PLC.BLKSEARCH

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The channel activates the mark when the block-search mode is active.

V.PLC.BLKSEARCH	Channel ·1·.
V.PLC.BLKSEARCHC1	Channel ·1·.
V.PLC.BLKSEARCHC2	Channel ·2·.
V.PLC.BLKSEARCHC3	Channel ·3·.
V.PLC.BLKSEARCHC4	Channel ·4·.

(V.)PLC.ADVINPOS

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The channel activates a certain amount of time before the axes reach position. The amount of time is set by parameter ANTIME.

V.PLC.ADVINPOS	Channel ·1·.
V.PLC.ADVINPOSC1	Channel ·1·.
V.PLC.ADVINPOSC2	Channel ·2·.
V.PLC.ADVINPOSC3	Channel ·3·.
V.PLC.ADVINPOSC4	Channel ·4·.

(V.)PLC.CAXIS

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The CNC channel sets this mark when there is a spindle working as C axis. This mark is kept active while any of the functions #CAX, #FACE or #CYL are kept active.

V.PLC.CAXIS	Channel ·1·.
V.PLC.CAXISC1	Channel ·1·.
V.PLC.CAXISC2	Channel ·2·.
V.PLC.CAXISC3	Channel ·3·.
V.PLC.CAXISC4	Channel ·4·.

(V.)PLC.FREE

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The channel activates the mark when it can accept a block sent with CNCEX.

V.PLC.FREEC1	Channel ·1·.
V.PLC.FREEC2	Channel ·2·.
V.PLC.FREEC3	Channel ·3·.
V.PLC.FREEC4	Channel ·4·.

(V.)PLC.WAITOUT

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The channel activates the mark when it is waiting for a synchronism signal.

V.PLC.WAITOUTC1	Channel ·1·.
V.PLC.WAITOUTC2	Channel ·2·.
V.PLC.WAITOUTC3	Channel ·3·.
V.PLC.WAITOUTC4	Channel ·4·.

(V.)PLC.MMCWDG

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The CNC activates this mark when the operating system is locked up.

V.PLC.MMCWDG

(V.)PLC.RETRAEND

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The channel activates this mark to cancel the retrace function.

V.PLC.RETRAENDC1	Channel ·1·.
V.PLC.RETRAENDC2	Channel ·2·.
V.PLC.RETRAENDC3	Channel ·3·.
V.PLC.RETRAENDC4	Channel ·4·.

(V.)PLC.TANGACTIV

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The channel activates the mark when a tangential control is active.

V.PLC.TANGACTIVC1	Channel ·1·.
V.PLC.TANGACTIVC2	Channel ·2·.
V.PLC.TANGACTIVC3	Channel ·3·.
V.PLC.TANGACTIVC4	Channel ·4·.

(V.)PLC.PSWSET

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The CNC activates this mark when there is an OEM password.

V.PLC.PSWSET

(V.)PLC.DINDISTC1
(V.)PLC.DINDISTC2
(V.)PLC.DINDISTC3
(V.)PLC.DINDISTC4

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

These marks are related to the dynamic machining distribution between channels (#DINDIST instruction), for the option to distribute passes between channels. During the roughing operation of the cycle, the CNC channel activates these marks to indicate which channel has the cycle been programmed in and which are the channels involved in the distribution of the passes. During the finishing operation, the CNC channel cancels all these marks.

(V.)PLC.DINDISTC1
(V.)PLC.DINDISTC2
(V.)PLC.DINDISTC3
(V.)PLC.DINDISTC4

(V.)PLC.SERPLCAC

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

This mark is related to the change of the work set or of the parameter set of a Sercos drive (variable (V.)[ch].A.SETGE.xn). The CNC activates the mark to indicate that the requested change is being carried out.

V.PLC.SERPLCAC

(V.)PLC.OVERTEMP

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

This mark indicates the temperature status of the CNC. This mark is deactivated while the CNC temperature is correct. When the CNC temperature exceeds the maximum allowed (60 ºC, 140 ºF), the CNC activates this mark and issues a warning to indicate this situation. The CNC will deactivate this mark when the temperature of the unit goes below the maximum allowed. The CNC checks its temperature every minute.

V.PLC.OVERTEMP

(V.)PLC.MLINKRDY

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The CNC activates the mark when the Mechatrolink bus has been initialized properly.

V.PLC.MLINKRDY

10.15 PLC consulting logic signals; axes and spindles.

(V.)PLC.ENABLExn
(V.)PLC.ENABLEsn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The CNC activates this mark to allow moving the axis or the spindle.

Syntax.

·xn· Name or logic number of the axis.

·sn· Name or logic number of the spindle.

V.PLC.ENABLEX	X axis.
V.PLC.ENABLES	Spindle S.
V.PLC.ENABLE3	Axis or spindle with logic number ·3·.

(V.)PLC.DIRxn
(V.)PLC.DIRsn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The CNC turns this mark on when moving the axis in the negative direction and turns it off when moving the axis in the positive direction. When the axis is stopped, the mark keeps its last value.

Syntax.

·xn· Name or logic number of the axis.

·sn· Name or logic number of the spindle.

V.PLC.DIRX	X axis.
V.PLC.DIRS	Spindle S.
V.PLC.DIR3	Axis or spindle with logic number ·3·.

(V.)PLC.REFPOINxn
(V.)PLC.REFPOINsn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The CNC activates this mark after a home search.

Syntax.

·xn· Name or logic number of the axis.

·sn· Name or logic number of the spindle.

V.PLC.REFPOINX	X axis.
V.PLC.REFPOINS	Spindle S.
V.PLC.REFPOIN3	Axis or spindle with logic number ·3·.

(V.)PLC.DRSTAFxn
(V.)PLC.DRSTAFsn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The CNC uses these marks to indicate the drive status.

Syntax.

·xn· Name or logic number of the axis.

·sn· Name or logic number of the spindle.

V.PLC.DRSTAFX	X axis.
V.PLC.DRSTAFS	Spindle S.
V.PLC.DRSTAF3	Axis or spindle with logic number ·3·.

(V.)PLC.DRSTASxn
(V.)PLC.DRSTASsn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The CNC uses these marks to indicate the drive status.

Syntax.

·xn· Name or logic number of the axis.

·sn· Name or logic number of the spindle.

V.PLC.DRSTASX	X axis.
V.PLC.DRSTASS	Spindle S.
V.PLC.DRSTAS3	Axis or spindle with logic number ·3·.

(V.)PLC.INPOSxn
(V.)PLC.INPOSsn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The CNC activates the mark when the axis or the spindle is in position.

Syntax.

·xn· Name or logic number of the axis.

·sn· Name or logic number of the spindle.

V.PLC.INPOSX	X axis.
V.PLC.INPOSS	Spindle S.
V.PLC.INPOS3	Axis or spindle with logic number ·3·.

(V.)PLC.LUBRxn
(V.)PLC.LUBRsn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The CNC activates the mark when the axis or the spindle must be lubricated.

Syntax.

·xn· Name or logic number of the axis.

·sn· Name or logic number of the spindle.

V.PLC.LUBRX	X axis.
V.PLC.LUBRS	Spindle S.
V.PLC.LUBR3	Axis or spindle with logic number ·3·.

(V.)PLC.HIRTHONxn
(V.)PLC.HIRTHONsn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The CNC activates the mark when the axis or the spindle works as Hirth axis.

Syntax.

·xn· Name or logic number of the axis.

·sn· Name or logic number of the spindle.

V.PLC.HIRTHX	X axis.
V.PLC.HIRTHS	Spindle S.
V.PLC.HIRTH3	Axis or spindle with logic number ·3·.

(V.)PLC.MATCHxn
(V.)PLC.MATCHsn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The CNC activates the mark when the Hirth axis or spindle is positioned properly.

Syntax.

·xn· Name or logic number of the axis.

·sn· Name or logic number of the spindle.

V.PLC.MATCHX	X axis.
V.PLC.MATCHS	Spindle S.
V.PLC.MATCH3	Axis or spindle with logic number ·3·.

(V.)PLC.PARKxn
(V.)PLC.PARKsn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The CNC activates the mark when it is parking the axis or the spindle.

Syntax.

·xn· Name or logic number of the axis.

·sn· Name or logic number of the spindle.

V.PLC.PARKX	X axis.
V.PLC.PARKS	Spindle S.
V.PLC.PARK3	Axis or spindle with logic number ·3·.

(V.)PLC.UNPARKxn
(V.)PLC.UNPARKsn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The CNC activates the mark when it is unparking the axis or the spindle.

Syntax.

·xn· Name or logic number of the axis.

·sn· Name or logic number of the spindle.

V.PLC.UNPARKX	X axis.
V.PLC.UNPARKS	Spindle S.
V.PLC.UNPARK3	Axis or spindle with logic number ·3·.

(V.)PLC.ACTFBACKxn
(V.)PLC.ACTFBACKsn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

In systems with external+internal feedback, the CNC turns this mark on when it is using the external feedback and turns it off when it is using the internal feedback.

Syntax.

·xn· Name or logic number of the axis.

·sn· Name or logic number of the spindle.

V.PLC.ACTFBACKX	X axis.
V.PLC.ACTFBACKS	Spindle S.
V.PLC.ACTFBACK3	Axis or spindle with logic number ·3·.

(V.)PLC.TANGACTxn
(V.)PLC.TANGACTsn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The CNC activates the mark when tangential control is active on the axis or spindle.

Syntax.

·xn· Name or logic number of the axis.

·sn· Name or logic number of the spindle.

V.PLC.TANGACTX	X axis.
V.PLC.TANGACTS	Spindle S.
V.PLC.TANGACT3	Axis or spindle with logic number ·3·.

(V.)PLC.LOPENxn
(V.)PLC.LOPENsn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The CNC activates this mark to inform the PLC that the axis position loop is open.

Syntax.

·xn· Name or logic number of the axis.

·sn· Name or logic number of the spindle.

V.PLC.LOPENX	X axis.
V.PLC.LOPENS	Spindle S.
V.PLC.LOPEN3	Axis or spindle with logic number ·3·.

(V.)PLC.MAXDIFFxn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.

Gantry axis [nb]. The CNC cannot correct the position difference between the master axis and the slave axis.

10.16 PLC consulting logic signals; spindles.

(V.)PLC.REVOK

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The spindle turns this mark on when it reaches the programmed rpm.

The mark is also on when the spindle is stopped (M05) or in position (M19, G63).

V.PLC.REVOK	Spindle ·1·.
V.PLC.REVOK1	Spindle ·1·.
V.PLC.REVOK2	Spindle ·2·.
V.PLC.REVOK3	Spindle ·3·.
V.PLC.REVOK4	Spindle ·4·.

(V.)PLC.SYNCMASTER

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The master spindle activates this mark when it has a spindle synchronized using #SYNC.

V.PLC.SYNCHRON1	Spindle ·1·.
V.PLC.SYNCHRON2	Spindle ·2·.
V.PLC.SYNCHRON3	Spindle ·3·.
V.PLC.SYNCHRON4	Spindle ·4·.

(V.)PLC.SYNCHRON

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The slave spindle activates this mark at the beginning of a synchronization using #SYNC.

V.PLC.SYNCHRON1	Spindle ·1·.
V.PLC.SYNCHRON2	Spindle ·2·.
V.PLC.SYNCHRON3	Spindle ·3·.
V.PLC.SYNCHRON4	Spindle ·4·.

(V.)PLC.SYNCHRONP

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The slave spindle activates this mark at the beginning of a synchronization in position.

V.PLC.SYNCHRONP1	Spindle ·1·.
V.PLC.SYNCHRONP2	Spindle ·2·.
V.PLC.SYNCHRONP3	Spindle ·3·.
V.PLC.SYNCHRONP4	Spindle ·4·.

(V.)PLC.SYNCSPEED

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The slave spindle activates this mark when it is synchronized in velocity.

V.PLC.SYNCSPEED1	Spindle ·1·.
V.PLC.SYNCSPEED2	Spindle ·2·.
V.PLC.SYNCSPEED3	Spindle ·3·.
V.PLC.SYNCSPEED4	Spindle ·4·.

(V.)PLC.SYNCPOSI

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The slave spindle activates this mark when it is synchronized in position.

V.PLC.SYNCPOSI1	Spindle ·1·.
V.PLC.SYNCPOSI2	Spindle ·2·.
V.PLC.SYNCPOSI3	Spindle ·3·.
V.PLC.SYNCPOSI4	Spindle ·4·.

(V.)PLC.GEAROK

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The spindle activates this mark when the parameter set selected at the CNC and at the PLC are the same.

V.PLC.GEAROK	Spindle ·1·.
V.PLC.GEAROK1	Spindle ·1·.
V.PLC.GEAROK2	Spindle ·2·.
V.PLC.GEAROK3	Spindle ·3·.
V.PLC.GEAROK4	Spindle ·4·.

10.17 PLC consulting logic signals; independent interpolator.

(V.)PLC.IBUSYxn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The interpolator is active when there is an instruction pending execution.

Syntax.

·xn· Name or logic number of the axis.

V.PLC.IBUSYX	X axis.
V.PLC.IBUSY3	Axis with logic number ·3·.

(V.)PLC.IFREExn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The interpolator turns this mark on when it is ready to accept a motion block.

Syntax.

·xn· Name or logic number of the axis.

V.PLC.IFREEX	X axis.
V.PLC.IFREE3	Axis with logic number ·3·.

(V.)PLC.IFHOUTxn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The interpolator turns this mark on when the execution is interrupted.

Syntax.

·xn· Name or logic number of the axis.

V.PLC.IFHOUTX	X axis.
V.PLC.IFHOUT3	Axis with logic number ·3·.

(V.)PLC.IENDxn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The interpolator turns this mark on when the axis movement has finished and it has reached its the final position.

Syntax.

·xn· Name or logic number of the axis.

V.PLC.IENDX	X axis.
V.PLC.IEND3	Axis with logic number ·3·.

(V.)PLC.ISYNCxn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The interpolator turns this mark on when the axis or the cam has reached synchronism.

Syntax.

·xn· Name or logic number of the axis.

V.PLC.ISYNCX	X axis.
V.PLC.ISYNC3	Axis with logic number ·3·.

10.18 PLC consulting logic signals; tool manager.

(V.)PLC.TMOPERATION

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The manager indicates in this register the type of operation to be carried out by the PLC.

V.PLC.TMOPERATION	Channel ·1·.
V.PLC.TMOPERATIONC1	Channel ·1·.
V.PLC.TMOPERATIONC2	Channel ·2·.
V.PLC.TMOPERATIONC3	Channel ·3·.
V.PLC.TMOPERATIONC4	Channel ·4·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Do nothing.
1	Take a tool from the magazine and insert it in the spindle.
2	Leave the spindle tool in the magazine.
3	Insert a ground tool in the spindle.
4	Leave the spindle tool on the ground.
5	Leave the spindle tool in the magazine and take another one from the magazine.
6	Leave the spindle tool in the magazine and take another one from ground.
7	Leave the spindle tool on the ground and take another one from the magazine.
8	Leave the spindle tool on the ground and take another one from ground.
9	Take a ground tool to the magazine going through the spindle.
10	Take a tool from the magazine and leave on the ground going through the spindle.
11	Orient the magazine.
12	Leave the spindle tool in the magazine and take another one from the same magazine. Special for a synchronous magazine in the following cases: • Non-random having a tool changer arm with two claws. • Random when having special tools.
13	Orienting two magazines
14	Leave the spindle tool in a magazine and take another one from another magazine.

(V.)PLC.TMOPSTROBE

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The manager turns this mark on to let the PLC know that it must execute the operation indicated by TMOPERATION.

V.PLC.TMOPSTROBE	Channel ·1·.
V.PLC.TMOPSTROBEC1	Channel ·1·.
V.PLC.TMOPSTROBEC2	Channel ·2·.
V.PLC.TMOPSTROBEC3	Channel ·3·.
V.PLC.TMOPSTROBEC4	Channel ·4·.

(V.)PLC.LEAVEPOS

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The manager indicates in this register the magazine position to leave the tool.

V.PLC.LEAVEPOS	Magazine ·1·.
V.PLC.LEAVEPOSMZ1	Magazine ·1·.
V.PLC.LEAVEPOSMZ2	Magazine ·2·.
V.PLC.LEAVEPOSMZ3	Magazine ·3·.
V.PLC.LEAVEPOSMZ4	Magazine ·4·.

(V.)PLC.TAKEPOS

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The manager indicates in this register the magazine position of the tool to be picked up.

V.PLC.TAKEPOS	Magazine ·1·.
V.PLC.TAKEPOSMZ1	Magazine ·1·.
V.PLC.TAKEPOSMZ2	Magazine ·2·.
V.PLC.TAKEPOSMZ3	Magazine ·3·.
V.PLC.TAKEPOSMZ4	Magazine ·4·.

(V.)PLC.NEXTPOS

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The manager indicates in this register the magazine position of the next tool.

V.PLC.NEXTPOS	Magazine ·1·.
V.PLC.NEXTPOSMZ1	Magazine ·1·.
V.PLC.NEXTPOSMZ2	Magazine ·2·.
V.PLC.NEXTPOSMZ3	Magazine ·3·.
V.PLC.NEXTPOSMZ4	Magazine ·4·.

(V.)PLC.TWORNOUT

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The manager turns this mark on when it has rejected the tool.

V.PLC.TWONRNOUT	Channel ·1·.
V.PLC.TWONRNOUTC1	Channel ·1·.
V.PLC.TWONRNOUTC2	Channel ·2·.
V.PLC.TWONRNOUTC3	Channel ·3·.
V.PLC.TWONRNOUTC4	Channel ·4·.

(V.)PLC.TMINEM

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The manager activates the mark when it is in an error state.

V.PLC.TMINEM	Magazine ·1·.
V.PLC.TMINEMZ1	Magazine ·1·.
V.PLC.TMINEMZ2	Magazine ·2·.
V.PLC.TMINEMZ3	Magazine ·3·.
V.PLC.TMINEMZ4	Magazine ·4·.

(V.)PLC.MZID

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The manager indicates in this register the magazine that has the requested tool. When two magazines are involved in a tool change, the lower portion of this register indicates the destination magazine for the tool and the higher portion the source magazine for the tool.

V.PLC.MZID	Channel ·1·.
V.PLC.MZIDC1	Channel ·1·.
V.PLC.MZIDC2	Channel ·2·.
V.PLC.MZIDC3	Channel ·3·.
V.PLC.MZIDC4	Channel ·4·.

10.19 PLC consulting logic signals; keys.

(V.)PLC.KEYBD1
(V.)PLC.KEYBD2

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

These registers are a copy of the map of the keys pressed on the last keyboard used. These registers indicate which key has been pressed (bit=1).

V.PLC.KEYBD1

V.PLC.KEYBD2

(V.)PLC.KEYBD1_1
(V.)PLC.KEYBD2_1
..
(V.)PLC.KEYBD1_8
(V.)PLC.KEYBD2_8

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

These registers indicate (bit=1) which key has been pressed on each operator panel. Registers KEYBD1_1 and KEYBD2_1 correspond to the first jog panel, KEYBD1_2 and KEYBD2_2 to the second one and so on.

V.PLC.KEYBD1_1

V.PLC.KEYBD2_1

10.20 PLC modifiable logic signals; general.

(V.)PLC._EMERGEN

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

If the PLC turns this mark off, the channel stops the axes and spindles and issues an error message.

While the mark is off, the channel does not allow executing programs and aborts attempt to move the axes or start the spindle.

V.PLC._EMERGEN	Channel ·1·.
V.PLC._EMERGENC1	Channel ·1·.
V.PLC._EMERGENC2	Channel ·2·.
V.PLC._EMERGENC3	Channel ·3·.
V.PLC._EMERGENC4	Channel ·4·.

(V.)PLC._STOP

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

If the PLC turns this mark off, the channel stops the execution of the program, but keeps the spindles turning. The status of this mark does not affect the independent axes.

V.PLC._STOP	Channel ·1·.
V.PLC._STOPC1	Channel ·1·.
V.PLC._STOPC2	Channel ·2·.
V.PLC._STOPC3	Channel ·3·.
V.PLC._STOPC4	Channel ·4·.

(V.)PLC._XFERINH

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

If the PLC turns this mark off, the prevents the execution of the next block, but it lets complete the execution of the current block.

V.PLC._XFERINH	Channel ·1·.
V.PLC._XFERINHC1	Channel ·1·.
V.PLC._XFERINHC2	Channel ·2·.
V.PLC._XFERINHC3	Channel ·3·.
V.PLC._XFERINHC4	Channel ·4·.

(V.)PLC._FEEDHOL

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

If the PLC turns this mark off, the channel stops the axes, but keeps the spindles turning. The status of this mark does not affect the independent axes.

V.PLC._FEEDHOL	Channel ·1·.
V.PLC._FEEDHOLC1	Channel ·1·.
V.PLC._FEEDHOLC2	Channel ·2·.
V.PLC._FEEDHOLC3	Channel ·3·.
V.PLC._FEEDHOLC4	Channel ·4·.

(V.)PLC.CYSTART

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

If the PLC turns this mark on, the execution of the part-program begins.

V.PLC.CYSTART	Channel ·1·.
V.PLC.CYSTARTC1	Channel ·1·.
V.PLC.CYSTARTC2	Channel ·2·.
V.PLC.CYSTARTC3	Channel ·3·.
V.PLC.CYSTARTC4	Channel ·4·.

(V.)PLC.SBLOCK

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

If the PLC turns this mark on, the channel activates the single-block mode.

V.PLC.SBLOCK	Channel ·1·.
V.PLC.SBLOCKC1	Channel ·1·.
V.PLC.SBLOCKC2	Channel ·2·.
V.PLC.SBLOCKC3	Channel ·3·.
V.PLC.SBLOCKC4	Channel ·4·.

(V.)PLC.MANRAPID

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

If the PLC turns this mark on, the CNC selects the rapid traverse for jog movements.

V.PLC.MANRAPID	Channel ·1·.
V.PLC.MANRAPIDC1	Channel ·1·.
V.PLC.MANRAPIDC2	Channel ·2·.
V.PLC.MANRAPIDC3	Channel ·3·.
V.PLC.MANRAPIDC4	Channel ·4·.

(V.)PLC.OVRCAN

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

If the PLC turns this mark on, the channel applies 100% of the feedrate in all work mode.

V.PLC.OVRCAN	Channel ·1·.
V.PLC.OVRCANC1	Channel ·1·.
V.PLC.OVRCANC2	Channel ·2·.
V.PLC.OVRCANC3	Channel ·3·.
V.PLC.OVRCANC4	Channel ·4·.

(V.)PLC.LATCHM

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

With this mark, it is possible to select how the JOG keys will work in JOG mode.

If this mark is off, the axes will move while the corresponding JOG key kept pressed. If this mark is on, the axes will move from the instant the JOG key is pressed until the software limits

are reached or the [STOP] key is pressed or another JOG key is pressed (in this case the new axis will start moving).

V.PLC.LATCHM

(V.)PLC.RESETIN

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

If the PLC turns this mark on, the channel assumes the initial conditions.

V.PLC.RESETIN	Channel ·1·.
V.PLC.RESETINC1	Channel ·1·.
V.PLC.RESETINC2	Channel ·2·.
V.PLC.RESETINC3	Channel ·3·.
V.PLC.RESETINC4	Channel ·4·.

(V.)PLC.AUXEND

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The PLC uses this mark when executing functions S and M with synchronization.

V.PLC.AUXEND	Channel ·1·.
V.PLC.AUXENDC1	Channel ·1·.
V.PLC.AUXENDC2	Channel ·2·.
V.PLC.AUXENDC3	Channel ·3·.
V.PLC.AUXENDC4	Channel ·4·.

(V.)PLC.BLKSKIP1

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

If the PLC turns this mark on, the channel considers the block-skip condition.

V.PLC.BLKSKIP1	Channel ·1·.
V.PLC.BLKSKIP1C1	Channel ·1·.
V.PLC.BLKSKIP1C2	Channel ·2·.
V.PLC.BLKSKIP1C3	Channel ·3·.
V.PLC.BLKSKIP1C4	Channel ·4·.

(V.)PLC.M01STOP

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

If the PLC turns this mark on, the channel considers the block-skips.

V.PLC.M01STOP	Channel ·1·.
V.PLC.M01STOPC1	Channel ·1·.
V.PLC.M01STOPC2	Channel ·2·.
V.PLC.M01STOPC3	Channel ·3·.
V.PLC.M01STOPC4	Channel ·4·.

(V.)PLC.TIMERON

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

If the PLC turns this mark on, the CNC enables the freely available clock.

V.PLC.TIMERON

(V.)PLC.PLCREADY

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

If the PLC turns this mark off, it interrupts the execution of the PLC program and issues an error message.

V.PLC.PLCREADY

(V.)PLC.NOWAIT

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The PLC turns this mark on to cancel the channel synchronizations programmed with #WAIT.

V.PLC.NOWAITC1	Channel ·1·.
V.PLC.NOWAITC2	Channel ·2·.
V.PLC.NOWAITC3	Channel ·3·.
V.PLC.NOWAITC4	Channel ·4·.

(V.)PLC.DISCROSS1
(V.)PLC.DISCROSS2
(V.)PLC.DISCROSS3
(V.)PLC.DISCROSS4
(V.)PLC.DISCROSS5
(V.)PLC.DISCROSS6
(V.)PLC.DISCROSS7
(V.)PLC.DISCROSS8
(V.)PLC.DISCROSS9

Variable that can only be read from the program and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The PLC this mark on to disable the cross compensation table.

V.PLC.DISCROSS1	Cross compensation table ·1·.
V.PLC.DISCROSS2	Cross compensation table ·2·.

(V.)PLC.PLCABORT

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

If the PLC turns this mark on, the channel aborts the CNCEX command launched from the PLC. This mark does not set the initial conditions in the channel and keeps the history.

V.PLC.PLCABORT	Channel ·1·.
V.PLC.PLCABORTC1	Channel ·1·.
V.PLC.PLCABORTC2	Channel ·2·.
V.PLC.PLCABORTC3	Channel ·3·.
V.PLC.PLCABORTC4	Channel ·4·.

(V.)PLC.NEXTMPGAXIS

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

Every time the PLC turns this mark on, the CNC selects an axis to be moved with the handwheel.

V.PLC.NEXTMPGAXIS

(V.)PLC.PANELOFF1
(V.)PLC.PANELOFF2
(V.)PLC.PANELOFF3
(V.)PLC.PANELOFF4
(V.)PLC.PANELOFF5
(V.)PLC.PANELOFF6
(V.)PLC.PANELOFF7
(V.)PLC.PANELOFF8

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

If the PLC activates one of these marks the CNC disables the corresponding jog panel.

V.PLC.PANELOFF1	Disable operator panel ·1·.
V.PLC.PANELOFF2	Disable operator panel ·2·.

(V.)PLC.SYNC

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The PLC indicates in this register the spindle to be used for synchronization.

The channel uses this spindle with function G33 to thread a particular spindle and with G95 to program the feedrate according to a particular spindle.

V.PLC.SYNC1	Channel ·1·.
V.PLC.SYNC2	Channel ·2·.
V.PLC.SYNC3	Channel ·3·.
V.PLC.SYNC4	Channel ·4·.

(V.)PLC.RETRACE

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

If the PLC turns this mark on while a program is being executed, the channel turns the retrace function on.

V.PLC.RETRACEC1	Channel ·1·.
V.PLC.RETRACEC2	Channel ·2·.
V.PLC.RETRACEC3	Channel ·3·.
V.PLC.RETRACEC4	Channel ·4·.

(V.)PLC.PRGABORT

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

If the PLC turns this mark on, the channel aborts the execution of the program but without affecting the spindle, initializes the program history and resumes the execution in the point indicated by the #ABORT instruction active in the part-program.

V.PLC.PRGABORT	Channel ·1·.
V.PLC.PRGABORTC1	Channel ·1·.
V.PLC.PRGABORTC2	Channel ·2·.
V.PLC.PRGABORTC3	Channel ·3·.
V.PLC.PRGABORTC4	Channel ·4·.

(V.)PLC.CNCOFF

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

If the PLC turns this mark on, the CNC starts the shut-down sequence.

V.PLC.CNCOFF

(V.)PLC.INHIBITMPG1
...
(V.)PLC.INHIBITMPG12

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

If the PLC turns one of these marks on, it disables the corresponding handwheel. The PLC has a mark for each handwheel, the INHIBITMPG1 mark disables the first handwheel, the INHIBITMPG2 mark the second one and so on.

V.PLC.INHIBITMPG1	Handwheel ·1·.
V.PLC.INHIBITMPG2	Handwheel ·2·.

(V.)PLC.EXRAPID

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

If the PLC activates this mark, the CNC channel enables rapid traverse during the execution of a program for the programmed movements. The behavior of this mark depends on how parameter RAPIDEN has been set.

V.PLC.EXRAPIDC1	Channel ·1·.
V.PLC.EXRAPIDC2	Channel ·2·.
V.PLC.EXRAPIDC3	Channel ·3·.
V.PLC.EXRAPIDC4	Channel ·4·.

(V.)PLC.KEYBD1CH
..
(V.)PLC.KEYBD8CH

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

These registers may be used to change the default behavior of the keyboards with respect to the channels, set by machine parameters.

V.PLC.KEYBD1CH	Operator panel ·1·.
V.PLC.KEYBD2CH	Operator panel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Configuration defined in the machine parameters.
1	Jog panel assigned to channel 1.
2	Jog panel assigned to channel 2.
3	Jog panel assigned to channel 3.
4	Jog panel assigned to channel 4.
FF	Jog panel assigned to the active channel.

(V.)PLC.VOLCOMP1
..
(V.)PLC.VOLCOMP4

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

If the PLC activates one of these marks, the CNC activates the corresponding volumetric compensation.

V.PLC.VOLCOMP1	Volumetric compensation ·1·.

(V.)PLC.QWERTYOFF1
..
(V.)PLC.QWERTYOFF8

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

If the PLC turns one of these marks on, it disables the corresponding alphanumeric keyboard.

V.PLC.QWERTYOFF1	Disable keyboard ·1·.
V.PLC.QWERTYOFF2	Disable keyboard ·2·.

(V.)PLC.FLIMITAC

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

If the PLC activates this mark, the CNC activates the feedrate safety limits (parameter FLIMIT) on all the axes of the system.

V.PLC.FLIMITAC

(V.)PLC.FLIMITACCH

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

If the PLC activates this mark, the CNC activates the feedrate safety limits (parameter FLIMIT) on all the axes of the channel.

V.PLC.FLIMITAC	Channel ·1·.
V.PLC.FLIMITACCH1	Channel ·1·.
V.PLC.FLIMITACCH2	Channel ·2·.
V.PLC.FLIMITACCH3	Channel ·3·.
V.PLC.FLIMITACCH4	Channel ·4·.

(V.)PLC.SLIMITAC

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

If the PLC activates this mark, the CNC activates the speed safety limits (parameter SLIMIT) on all the spindles of the system.

V.PLC.SLIMITAC

(V.)PLC.INT1
..
(V.)PLC.INT4

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

If the PLC activates one of these marks, the channel executes the corresponding interruption subroutine.

V.PLC.INT1	Channel ·1·.
V.PLC.INT1C1	Channel ·1·.
V.PLC.INT1C2	Channel ·2·.
V.PLC.INT1C3	Channel ·3·.
V.PLC.INT1C4	Channel ·4·.

10.21 PLC modifiable logic signals; axes and spindles.

(V.)PLC.LIMITPOSxn
(V.)PLC.LIMITPOSsn

Variable that can be read and written from the interface and read from the program and from the PLC.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The PLC must turn this mark on to indicate that the axis or the spindle has exceeded the positive travel limit.

Syntax.

·xn· Name or logic number of the axis.

·sn· Name or logic number of the spindle.

V.PLC.LIMITPOSX	X axis.
V.PLC.LIMITPOSS	Spindle S.
V.PLC.LIMITPOS3	Axis or spindle with logic number ·3·.

(V.)PLC.LIMITNEGxn
(V.)PLC.LIMITNEGsn

Variable that can be read and written from the interface and read from the program and from the PLC.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The PLC must turn this mark on to indicate that the axis or the spindle has exceeded the negative travel limit.

Syntax.

·xn· Name or logic number of the axis.

·sn· Name or logic number of the spindle.

V.PLC.LIMITNEGX	X axis.
V.PLC.LIMITNEGS	Spindle S.
V.PLC.LIMITNEG3	Axis or spindle with logic number ·3·.

(V.)PLC.DECELxn
(V.)PLC.DECELsn

Variable that can be read and written from the interface and read from the program and from the PLC.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The PLC must turn this mark on to indicate that the home switch is pressed.

Syntax.

·xn· Name or logic number of the axis.

·sn· Name or logic number of the spindle.

V.PLC.DECELX	X axis.
V.PLC.DECELS	Spindle S.
V.PLC.DECEL3	Axis or spindle with logic number ·3·.

(V.)PLC.INHIBITxn
(V.)PLC.INHIBITsn

Variable that can be read and written from the interface and read from the program and from the PLC.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

If the PLC turns this mark on, the CNC inhibits any movement of the axis or spindle.

For independent axes and electronic cam, if the PLC turns this mark on, it interrupts the synchronization movement switching to zero speed. The system waits for the signal to deactivate before resuming the execution and the movement from the interruption point.

Syntax.

·xn· Name or logic number of the axis.

·sn· Name or logic number of the spindle.

V.PLC.INHIBITX	X axis.
V.PLC.INHIBITS	Spindle S.
V.PLC.INHIBIT3	Axis or spindle with logic number ·3·.

(V.)PLC.AXISPOSxn
(V.)PLC.AXISPOSsn

Variable that can be read and written from the interface and read from the program and from the PLC.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

If the PLC turns this mark on while the CNC is in jog mode, the CNC moves the axis or the spindle in the positive direction.

Syntax.

·xn· Name or logic number of the axis.

·sn· Name or logic number of the spindle.

V.PLC.AXISPOSX	X axis.
V.PLC.AXISPOSS	Spindle S.
V.PLC.AXISPOS3	Axis or spindle with logic number ·3·.

(V.)PLC.AXISNEGxn
(V.)PLC.AXISNEGsn

Variable that can be read and written from the interface and read from the program and from the PLC.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

If the PLC turns this mark on while the CNC is in jog mode, the CNC moves the axis or the spindle in the positive direction.

Syntax.

·xn· Name or logic number of the axis.

·sn· Name or logic number of the spindle.

V.PLC.AXISNEGX	X axis.
V.PLC.AXISNEGS	Spindle S.
V.PLC.AXISNEG3	Axis or spindle with logic number ·3·.

(V.)PLC.SERVOxnON
(V.)PLC.SERVOsnON

Variable that can be read and written from the interface and read from the program and from the PLC.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The PLC must turn this mark on to allow moving the axis or the spindle.

Syntax.

·xn· Name or logic number of the axis.

·sn· Name or logic number of the spindle.

V.PLC.SERVOXON	X axis.
V.PLC.SERVOSON	Spindle S.
V.PLC.SERVO3ON	Axis or spindle with logic number ·3·.

(V.)PLC.DROxn
(V.)PLC.DROsn

Variable that can be read and written from the interface and read from the program and from the PLC.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The PLC must turn this mark on for the axis or the spindle to work in DRO mode.

Syntax.

·xn· Name or logic number of the axis.

·sn· Name or logic number of the spindle.

V.PLC.DROX	X axis.
V.PLC.DROS	Spindle S.
V.PLC.DRO3	Axis or spindle with logic number ·3·.

(V.)PLC.SPENAxn
(V.)PLC.SPENAsn

Variable that can be read and written from the interface and read from the program and from the PLC.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The PLC must turn this mark on to enable the speed enable signal of the drive.

Syntax.

·xn· Name or logic number of the axis.

·sn· Name or logic number of the spindle.

V.PLC.SPENAX	X axis.
V.PLC.SPENAS	Spindle S.
V.PLC.SPENA3	Axis or spindle with logic number ·3·.

(V.)PLC.DRENAxn
(V.)PLC.DRENAsn

Variable that can be read and written from the interface and read from the program and from the PLC.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The PLC must turn this mark on to enable the drive enable signal of the drive.

Syntax.

·xn· Name or logic number of the axis.

·sn· Name or logic number of the spindle.

V.PLC.DRENAX	X axis.
V.PLC.DRENAS	Spindle S.
V.PLC.DRENA3	Axis or spindle with logic number ·3·.

(V.)PLC.LIMxnOFF
(V.)PLC.LIMsnOFF

Variable that can be read and written from the interface and read from the program and from the PLC.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

If the PLC turns this mark on, the CNC ignores the software travel limits.

Syntax.

·xn· Name or logic number of the axis.

·sn· Name or logic number of the spindle.

V.PLC.LIMXOFF	X axis.
V.PLC.LIMSOFF	Spindle S.
V.PLC.LIM3OFF	Axis or spindle with logic number ·3·.

(V.)PLC.PARKEDxn
(V.)PLC.PARKEDsn

Variable that can be read and written from the interface and read from the program and from the PLC.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The PLC turns this mark on when the axis or the spindle is parked.

Syntax.

·xn· Name or logic number of the axis.

·sn· Name or logic number of the spindle.

V.PLC.PARKEDX	X axis.
V.PLC.PARKEDS	Spindle S.
V.PLC.PARKED3	Axis or spindle with logic number ·3·.

(V.)PLC.LUBRENAxn
(V.)PLC.LUBRENAsn

Variable that can be read and written from the interface and read from the program and from the PLC.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The PLC turns this mark on to enable the lubrication of the axis or the spindle.

Syntax.

·xn· Name or logic number of the axis.

·sn· Name or logic number of the spindle.

V.PLC.LUBRENAX	X axis.
V.PLC.LUBRENAS	Spindle S.
V.PLC.LUBRENA3	Axis or spindle with logic number ·3·.

(V.)PLC.LUBROKxn
(V.)PLC.LUBROKsn

Variable that can be read and written from the interface and read from the program and from the PLC.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The PLC turns this mark on to indicate that the axis lubrication is done.

Syntax.

·xn· Name or logic number of the axis.

·sn· Name or logic number of the spindle.

V.PLC.LUBROKX	X axis.
V.PLC.LUBROKS	Spindle S.
V.PLC.LUBROK3	Axis or spindle with logic number ·3·.

(V.)PLC.DIFFCOMPxn
(V.)PLC.DIFFCOMPsn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The PLC uses this mark on Gantry axes to correct the position difference between the two axes.

Syntax.

·xn· Name or logic number of the axis.

·sn· Name or logic number of the spindle.

V.PLC.DIFFCOMPX	X axis.
V.PLC.DIFFCOMPS	Spindle S.
V.PLC.DIFFCOMP3	Axis or spindle with logic number ·3·.

(V.)PLC.FBACKSELxn
(V.)PLC.FBACKSELsn

Variable that can be read and written from the interface and read from the program and from the PLC.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

In systems with external+internal feedback, the PLC turns this mark on to use the external feedback and turns it off to use the internal feedback.

Syntax.

·xn· Name or logic number of the axis.

·sn· Name or logic number of the spindle.

V.PLC.FBACKSELX	X axis.
V.PLC.FBACKSELS	Spindle S.
V.PLC.FBACKSEL3	Axis or spindle with logic number ·3·.

(V.)PLC.DEADxn
(V.)PLC.DEADsn

Variable that can be read and written from the program and interface and read from the PLC.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

In systems with dead axes, the PLC uses this mark to indicate to the CNC how to handle the path joints when a dead axis is involved.

Syntax.

·xn· Name or logic number of the axis.

·sn· Name or logic number of the spindle.

V.PLC.DEADX	X axis.
V.PLC.DEADS	Spindle S.
V.PLC.DEAD3	Axis or spindle with logic number ·3·.

(V.)PLC.SWITCHxn
(V.)PLC.SWITCHsn

Variable that can be read and written from the program and interface and read from the PLC.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

When the system has multi-axis groups, this mark may be used to toggle between the various axes or spindles of the group.

Syntax.

·xn· Name or logic number of the axis.

·sn· Name or logic number of the spindle.

V.PLC.SWITCHX	X axis.
V.PLC.SWITCHS	Spindle S.
V.PLC.SWITCH3	Axis or spindle with logic number ·3·.

(V.)PLC.TANDEMOFFxn
(V.)PLC.TANDEMOFFsn

Variable that can be read and written from the program and interface and read from the PLC.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

This mark may be used to temporarily decouple (unslave) the loop of the axes or spindles involved in the tandem so they can be move separately.

Syntax.

·xn· Name or logic number of the axis.

·sn· Name or logic number of the spindle.

V.PLC.TANDEMOFFX	X axis.
V.PLC.TANDEMOFFS	Spindle S.
V.PLC.TANDEMOFF3	Axis or spindle with logic number ·3·.

10.22 PLC modifiable logic signals; spindles.

(V.)PLC.GEAR1
(V.)PLC.GEAR2
(V.)PLC.GEAR3
(V.)PLC.GEAR4

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The PLC must turn on the mark corresponding to the selected speed range.

There is a mark for each spindle. The mnemonics for each channel are the following. Here is an example of the mnemonics for GEAR1; it is the same for the rest of the registers.

V.PLC.GEAR1	Spindle ·1·.
V.PLC.GEAR1SP1	Spindle ·1·.
V.PLC.GEAR1SP2	Spindle ·2·.
V.PLC.GEAR1SP3	Spindle ·3·.
V.PLC.GEAR1SP4	Spindle ·4·.

(V.)PLC.PLCCNTL

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The PLC must turn this mark on when the spindle is controlled by the PLC.

V.PLC.PLCCNTL	Spindle ·1·.
V.PLC.PLCCNTL1	Spindle ·1·.
V.PLC.PLCCNTL2	Spindle ·2·.
V.PLC.PLCCNTL3	Spindle ·3·.
V.PLC.PLCCNTL4	Spindle ·4·.

(V.)PLC.SANALOG

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

When the spindle is controlled by the PLC, the PLC must indicate in this register the velocity command to be applied to the spindle.

V.PLC.SANALOG	Spindle ·1·.
V.PLC.SANALOG1	Spindle ·1·.
V.PLC.SANALOG2	Spindle ·2·.
V.PLC.SANALOG3	Spindle ·3·.
V.PLC.SANALOG4	Spindle ·4·.

(V.)PLC.SPDLEREV

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

If the PLC turns this mark on, the CNC inverts the spindle turning direction.

V.PLC.SPDLEREV	Spindle ·1·.
V.PLC.SPDLEREV1	Spindle ·1·.
V.PLC.SPDLEREV2	Spindle ·2·.
V.PLC.SPDLEREV3	Spindle ·3·.
V.PLC.SPDLEREV4	Spindle ·4·.

(V.)PLC.PLCM3
(V.)PLC.PLCM4
(V.)PLC.PLCM5

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The PLC turns this mark on to let the CNC know that it must execute the corresponding M function at the indicated spindle.

V.PLC.PLCM3	Spindle ·1·.
V.PLC.PLCM3SP1	Spindle ·1·.
V.PLC.PLCM3SP2	Spindle ·2·.
V.PLC.PLCM3SP3	Spindle ·3·.
V.PLC.PLCM3SP4	Spindle ·4·.

(V.)PLC.SLIMITACSPDL

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

There is a mark for each spindle. The mnemonics for each spindle are the following:

SLIMITACSPDL1 (can also be programmed as SLIMITACSPDL)

SLIMITACSPDL2 SLIMITACSPDL3 SLIMITACSPDL4

If the PLC activates this mark, the CNC activates the speed safety limits (parameter SLIMIT) on the indicated spindle.

V.PLC.SLIMITACSPDL	Spindle ·1·.
V.PLC.SLIMITACSPDL1	Spindle ·1·.
V.PLC.SLIMITACSPDL2	Spindle ·2·.
V.PLC.SLIMITACSPDL3	Spindle ·3·.
V.PLC.SLIMITACSPDL4	Spindle ·4·.

10.23 PLC modifiable logic signals; independent interpolator.

(V.)PLC._IXFERINHxn

If the PLC deactivates this mark, the movements of the independent axis remain waiting for the PLC to activate this mark again.

Syntax.

·xn· Name or logic number of the axis.

V.PLC._IXFERINHX	X axis.
V.PLC._IXFERINH3	Axis with logic number ·3·.

(V.)PLC.IRESETxn

Variable that can be read and written from the program and interface and read from the PLC.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

If the PLC turns this mark on, the independent interpolator interrupts the instruction being executed and eliminates the instructions pending execution.

Syntax.

·xn· Name or logic number of the axis.

V.PLC.IRESETX	X axis.
V.PLC.IRESET3	Axis with logic number ·3·.

(V.)PLC.IABORTxn

Variable that can be read and written from the program and interface and read from the PLC.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

If the PLC turns this mark on, the independent interpolator interrupts the positioning block being executed (if any) eliminating also the rest of the positioning blocks pending execution.

Syntax.

·xn· Name or logic number of the axis.

V.PLC.IABORTX	X axis.
V.PLC.IABORT3	Axis with logic number ·3·.

10.24 PLC modifiable logic signals; tool manager.

(V.)PLC.SETTMEM

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The PLC must turn this mark on to activate the emergency of the tool manager.

V.PLC.SETTMEM	Magazine ·1·.
V.PLC.SETTMEMZ1	Magazine ·1·.
V.PLC.SETTMEMZ2	Magazine ·2·.
V.PLC.SETTMEMZ3	Magazine ·3·.
V.PLC.SETTMEMZ4	Magazine ·4·.

(V.)PLC.RESTMEM

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The PLC must turn this mark on to cancel the emergency of the tool manager.

V.PLC.RESTMEM	Magazine ·1·.
V.PLC.RESTMEMZ1	Magazine ·1·.
V.PLC.RESTMEMZ2	Magazine ·2·.
V.PLC.RESTMEMZ3	Magazine ·3·.
V.PLC.RESTMEMZ4	Magazine ·4·.

(V.)PLC.CUTTINGON

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The PLC must turn this mark on to indicate that the tool is machining (cutting).

V.PLC.CUTTINGON	Channel ·1·.
V.PLC.CUTTINGON1	Channel ·1·.
V.PLC.CUTTINGON2	Channel ·2·.
V.PLC.CUTTINGON3	Channel ·3·.
V.PLC.CUTTINGON4	Channel ·4·.

(V.)PLC.TREJECT

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The PLC must turn this mark on to reject the tool.

V.PLC.TREJECT	Channel ·1·.
V.PLC.TREJECTC1	Channel ·1·.
V.PLC.TREJECTC2	Channel ·2·.
V.PLC.TREJECTC3	Channel ·3·.
V.PLC.TREJECTC4	Channel ·4·.

(V.)PLC.MZTOCH1

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The PLC must turn this mark on after taking the tool from the magazine to the claw 1 of the tool changer arm.

V.PLC.MZTOCH1	Magazine ·1·.
V.PLC.MZTOCH1MZ1	Magazine ·1·.
V.PLC.MZTOCH1MZ2	Magazine ·2·.
V.PLC.MZTOCH1MZ3	Magazine ·3·.
V.PLC.MZTOCH1MZ4	Magazine ·4·.

(V.)PLC.CH1TOSPDL

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The PLC must turn this mark on after taking the tool from the claw 1 of the tool changer arm to the spindle.

V.PLC.CH1TOSPDL	Magazine ·1·.
V.PLC.CH1TOSPDLMZ1	Magazine ·1·.
V.PLC.CH1TOSPDLMZ2	Magazine ·2·.
V.PLC.CH1TOSPDLMZ3	Magazine ·3·.
V.PLC.CH1TOSPDLMZ4	Magazine ·4·.

(V.)PLC.SPDLTOCH1

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The PLC must turn this mark on after taking the tool from the spindle to the claw 1 of the tool changer arm.

V.PLC.SPDLTOCH1	Magazine ·1·.
V.PLC.SPDLTOCH1MZ1	Magazine ·1·.
V.PLC.SPDLTOCH1MZ2	Magazine ·2·.
V.PLC.SPDLTOCH1MZ3	Magazine ·3·.
V.PLC.SPDLTOCH1MZ4	Magazine ·4·.

(V.)PLC.SPDLTOCH2

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The PLC must turn this mark on after taking the tool from the spindle to the claw 2 of the tool changer arm.

V.PLC.SPDLTOCH1	Magazine ·1·.
V.PLC.SPDLTOCH2MZ1	Magazine ·1·.
V.PLC.SPDLTOCH2MZ2	Magazine ·2·.
V.PLC.SPDLTOCH2MZ3	Magazine ·3·.
V.PLC.SPDLTOCH2MZ4	Magazine ·4·.

(V.)PLC.CH1TOMZ

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The PLC must turn this mark on after taking the tool from the claw 1 of the tool changer arm to the magazine.

V.PLC.CH1TOMZ	Magazine ·1·.
V.PLC.CH1TOMZ1	Magazine ·1·.
V.PLC.CH1TOMZ2	Magazine ·2·.
V.PLC.CH1TOMZ3	Magazine ·3·.
V.PLC.CH1TOMZ4	Magazine ·4·.

(V.)PLC.CH2TOMZ

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The PLC must turn this mark on after taking the tool from the claw 2 of the tool changer arm to the magazine.

V.PLC.CH2TOMZ	Magazine ·1·.
V.PLC.CH2TOMZ1	Magazine ·1·.
V.PLC.CH2TOMZ2	Magazine ·2·.
V.PLC.CH2TOMZ3	Magazine ·3·.
V.PLC.CH2TOMZ4	Magazine ·4·.

(V.)PLC.SPDLTOGR

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The PLC must turn this mark on after taking the tool from the spindle to ground.

V.PLC.SPDLTOGR	Channel ·1·.
V.PLC.SPDLTOGRC1	Channel ·1·.
V.PLC.SPDLTOGRC2	Channel ·2·.
V.PLC.SPDLTOGRC3	Channel ·3·.
V.PLC.SPDLTOGRC4	Channel ·4·.

(V.)PLC.GRTOSPDL

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The PLC must turn this mark on after taking the tool from the ground to the spindle.

V.PLC.GRTOSPDL	Channel ·1·.
V.PLC.GRTOSPDLC1	Channel ·1·.
V.PLC.GRTOSPDLC2	Channel ·2·.
V.PLC.GRTOSPDLC3	Channel ·3·.
V.PLC.GRTOSPDLC4	Channel ·4·.

(V.)PLC.MZTOSPDL

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The PLC must turn this mark on after taking the tool from the magazine to the spindle.

V.PLC.MZTOSPDL	Magazine ·1·.
V.PLC.MZTOSPDLMZ1	Magazine ·1·.
V.PLC.MZTOSPDLMZ2	Magazine ·2·.
V.PLC.MZTOSPDLMZ3	Magazine ·3·.
V.PLC.MZTOSPDLMZ4	Magazine ·4·.

(V.)PLC.SPDLTOMZ

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The PLC must turn this mark on after taking the tool from the spindle to the magazine.

V.PLC.SPDLTOMZ	Magazine ·1·.
V.PLC.SPDLTOMZ1	Magazine ·1·.
V.PLC.SPDLTOMZ2	Magazine ·2·.
V.PLC.SPDLTOMZ3	Magazine ·3·.
V.PLC.SPDLTOMZ4	Magazine ·4·.

(V.)PLC.MZROT

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The PLC must turn this mark on to rotate the turret.

V.PLC.MZROT	Magazine ·1·.
V.PLC.MZROTMZ1	Magazine ·1·.
V.PLC.MZROTMZ2	Magazine ·2·.
V.PLC.MZROTMZ3	Magazine ·3·.
V.PLC.MZROTMZ4	Magazine ·4·.

(V.)PLC.TCHANGEOK

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The PLC must turn this mark on when the tool change is done.

V.PLC.TCHANGEOK	Magazine ·1·.
V.PLC.TCHANGEOKMZ1	Magazine ·1·.
V.PLC.TCHANGEOKMZ2	Magazine ·2·.
V.PLC.TCHANGEOKMZ3	Magazine ·3·.
V.PLC.TCHANGEOKMZ4	Magazine ·4·.

(V.)PLC.MZPOS

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

The PLC must indicate the current magazine position in this register.

V.PLC.MZPOS	Magazine ·1·.
V.PLC.MZPOSMZ1	Magazine ·1·.
V.PLC.MZPOSMZ2	Magazine ·2·.
V.PLC.MZPOSMZ3	Magazine ·3·.
V.PLC.MZPOSMZ4	Magazine ·4·.

10.25 PLC modifiable logic signals; keys.

(V.)PLC.KEYLED1
(V.)PLC.KEYLED2

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

These registers control the LED's (lamps) of the keys of all the operator panels at the same time.

V.PLC.KEYLED1
V.PLC.KEYLED2

(V.)PLC.KEYLED1_1
(V.)PLC.KEYLED2_1
..
(V.)PLC.KEYLED1_8
(V.)PLC.KEYLED2_8

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

These registers control the LED's (lamps) of the keys of each operator panel. Registers KEYLED1_1 and KEYLED2_1 correspond to the first jog panel, KEYLED1_2 and KEYLED2_2 to the second one and so on.

V.PLC.KEYLED1_1
V.PLC.KEYLED2_1

(V.)PLC.KEYDIS1
(V.)PLC.KEYDIS2
(V.)PLC.KEYDIS3

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

These registers inhibit (bit=1) the keys and the switches on all operator panels at the same time.

V.PLC.KEYDIS1
V.PLC.KEYDIS2
V.PLC.KEYDIS3

(V.)PLC.KEYDIS1_1
(V.)PLC.KEYDIS2_1
(V.)PLC.KEYDIS3_1
..
(V.)PLC.KEYDIS1_8
(V.)PLC.KEYDIS2_8
(V.)PLC.KEYDIS3_8

Variable that can be read and written from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

These registers inhibit (bit=1) the keys and the switches on each operator panel. Registers KEYDIS1_1 to KEYDIS3_1 correspond to the first jog panel, KEYDIS1_2 to KEYDIS3_2 to the second one and so on.

V.PLC.KEYDIS1_1
V.PLC.KEYDIS2_1
V.PLC.KEYDIS3_1

10.26 Variables related to the machine configuration.

AXES AND SPINDLES OF THE SYSTEM.

(V.)G.GAXISNAMEn

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

Name of logic axis "n".

Syntax.

Replace the "n" letter with the axis logic number.

V.G.GAXISNAME2	Axis with logic number ·2·.

Values of the variable.

The values returned by this variable are encoded as follows.

X=10	X1=11	X2=12	X3=13	X4=14	... X9=19
Y=20	Y1=21	Y2=22	Y3=23	Y4=24	... Y9=29
Z=30	Z1=31	Z2=32	Z3=33	Z4=34	... Z9=39
U=40	U1=41	U2=42	U3=43	U4=44	... U9=49
V=50	V1=51	V2=52	V3=53	V4=54	... V9=59
W=60	W1=61	W2=62	W3=63	W4=64	... W9=69
A=70	A1=71	A2=72	A3=73	A4=74	... A9=79
B=80	B1=81	B2=82	B3=83	B4=84	... B9=89
C=90	C1=91	C2=92	C3=93	C4=94	... C9=99

Remarks.

The logic number of the axes is determined by the order in which the axes have been defined in the machine parameter table. The first axis of the table will be logic axis -1- and so on.

When some axes are parked, it is a good idea to know which axes are available. This variable indicates which axes are available, if an axis is not available, this variable returns a "?".

(V.)G.GSPDLNAMEn

Variable that can only be read from the program, PLC and interface.
The variable returns the value of execution or preparation depending on the spindle.
Report variable (to be used from the scripts).

Name of logic spindle "n".

Syntax.

Replace the "n" letter with the spindle logic number.

V.G.GSPDLNAME2	Spindle with logic number ·2·.

Values of the variable.

The values returned by this variable are encoded as follows.

S=100	S1=101	S2=102	S3=103	S4=104	... S9=109

Remarks.

The variable returns the value of execution or preparation as follows. If the spindle belongs to the channel requesting the variable, it returns the preparation value; if the spindle belongs to a different channel, the variable returns the execution value and interrupts block preparation.

The logic number of the spindles is determined by the order in which they have been defined in the machine parameter table. The logic numbering of the spindles continues from the last logic axis; hence, in a 5-axis system, the first spindle of the table will be logic spindle ·6· and so on.

(V.)[ch].A.ACTCH.xn
(V.)[ch].A.ACTCH.sn
(V.)[ch].SP.ACTCH.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the value of execution or preparation depending on the axis or spindle.

Current channel of the axis or of the spindle.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.ACTCH.Z	Z axis.
V.A.ACTCH.S	Spindle S.
V.SP.ACTCH.S	Spindle S.
V.SP.ACTCH	Master spindle.
V.A.ACTCH.4	Axis or spindle with logic number ·4·.
V.[2].A.ACTCH.1	Axis with index ·1· in the channel ·2·.
V.SP.ACTCH.2	Spindle with index ·2· in the system.
V.[2].SP.ACTCH.1	Spindle with index ·1· in the channel ·2·.

Remarks.

The variable returns the value of execution or preparation as follows. If the spindle belongs to the channel requesting the variable, it returns the preparation value; if the axis or spindle belongs to a different channel, the variable returns the execution value and interrupts block preparation.

(V.)[ch].A.ACTIVSET.xn
(V.)[ch].A.ACTIVSET.sn
(V.)[ch].SP.ACTIVSET.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the value of execution or preparation depending on the axis or spindle.

Parameter set active at the axis or spindle.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.ACTIVSET.Z	Z axis.
V.A.ACTIVSET.S	Spindle S.
V.SP.ACTIVSET.S	Spindle S.
V.SP.ACTIVSET	Master spindle.
V.A.ACTIVSET.4	Axis or spindle with logic number ·4·.
V.[2].A.ACTIVSET.1	Axis with index ·1· in the channel ·2·.
V.SP.ACTIVSET.2	Spindle with index ·2· in the system.
V.[2].SP.ACTIVSET.1	Spindle with index ·1· in the channel ·2·.

Remarks.

The variable returns the value of execution or preparation as follows. If the spindle belongs to the channel requesting the variable, it returns the preparation value; if the axis or spindle belongs to a different channel, the variable returns the execution value and interrupts block preparation.

CHANNELS, AXES AND SPINDLES.

(V.)G.NUMCH

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Number of channels.

V.G.NUMCH

(V.)[ch].G.AXIS

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.
Report variable (to be used from the scripts).

Number of axes of the channel.

Syntax.

·ch· Channel number.

V.[2].G.AXIS	Channel ·2·.

(V.)[ch].G.NAXIS

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.
Report variable (to be used from the scripts).

Number of axes of the channel including the empty positions of the yielded axes.

Syntax.

·ch· Channel number.

V.[2].G.NAXIS	Channel ·2·.

(V.)[ch].G.NSPDL

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.
Report variable (to be used from the scripts).

Number of spindles of the channel.

Syntax.

·ch· Channel number.

V.[2].G.NSPDL	Channel ·2·.

(V.)[ch].G.AXISCH

Variable to be read via interface.
Report variable (to be used from the scripts).

Axes of the system that belong to the channel.

Syntax.

·ch· Channel number.

```
[2].G.NSPDL                    Channel ·2·.
```

Values of the variable.

The variable returns a 32-bit value where each bit represents one axis; the least significant bit corresponds to the axis with the lowest logic number. Each bit indicates whether the axis belongs to the channel (bit = 1) or not (bit = 0).

AXISNAME.	Channel ·1·.	Channel ·2·.	Reading the variables.
X (Logic axis ·1·.) Y (Logic axis ·2·.) Z (Logic axis ·3·.) X2 (Logic axis ·4·.) Y2 (Logic axis ·5·.) Z2 (Logic axis ·6·.)	X Y Z	X2 Y2 Z2	[1].G.AXISCH = $7 [2].G.AXISCH = $38

(V.)[ch].G.AXISNAMEn

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.
Report variable (to be used from the scripts).

Name of the axis with n index in the channel.

Syntax.

·ch· Channel number.

```
V.[2].G.AXISNAME1              Channel ·2·.
```

Values of the variable.

The values returned by this variable are encoded as follows.

X=10	X1=11	X2=12	X3=13	X4=14	... X9=19
Y=20	Y1=21	Y2=22	Y3=23	Y4=24	... Y9=29
Z=30	Z1=31	Z2=32	Z3=33	Z4=34	... Z9=39
U=40	U1=41	U2=42	U3=43	U4=44	... U9=49
V=50	V1=51	V2=52	V3=53	V4=54	... V9=59
W=60	W1=61	W2=62	W3=63	W4=64	... W9=69
A=70	A1=71	A2=72	A3=73	A4=74	... A9=79
B=80	B1=81	B2=82	B3=83	B4=84	... B9=89
C=90	C1=91	C2=92	C3=93	C4=94	... C9=99

Remarks.

When some axes are parked, it is a good idea to know which axes are available. This variable indicates which axes are available, if an axis is not available, this variable returns a "?".

(V.)[ch].G.SPDLNAMEn

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.
Report variable (to be used from the scripts).

Name of the spindle with n index in the channel.

Syntax.

·ch· Channel number.

```
V.[2].G.SPDLNAME1              Channel ·2·.
```

Values of the variable.

The values returned by this variable are encoded as follows.

S=100 S1=101 S2=102 S3=103 S4=104 ... S9=109

(V.)[ch].G.MASTERSP

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Logic number of the master spindle of the channel.

Syntax.

·ch· Channel number.

V.[2].G.MASTERSP	Channel ·2·.

TRAVEL LIMITS OF LINEAR AND ROTARY AXES.

(V.)[ch].G.SOFTLIMIT

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

Software limits reached.

Syntax.

·ch· Channel number.

V.[2].G.SOFTLIMIT	Channel ·2·.

(V.)[ch].A.POSLIMIT.xn
(V.)[ch].A.NEGLIMIT.xn

Variable that can be read and written from the program and read from the PLC and from the interface.
Variable valid for rotary and linear axes.
The variable returns the value of block preparation.

Positive and negative software limit.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.A.POSLIMIT.Z	Z axis.
V.A.POSLIMIT.4	Axis with logic number ·4·.
V.[2].A.POSLIMIT.1	Axis with index ·1· in the channel ·2·.

Remarks.

These variables correspond to the limits set by machine parameters. When modifying these variables, the CNC assumes those values as the new limits from then on.

These variables keep their value after a Reset, they are initialized with the values of the machine parameters when validating them and when turning the CNC on.

(V.)[ch].A.RTPOSLIMIT.xn
(V.)[ch].A.RTNEGLIMIT.xn

Variable that can be read and written via program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.

Second positive and negative software limit.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.A.RTPOSLIMIT.Z	Z axis.
V.A.RTPOSLIMIT.4	Axis with logic number ·4·.
V.[2].A.RTPOSLIMIT.1	Axis with index ·1· in the channel ·2·.

Remarks.

There are two software limits; the CNC always applies the most restrictive one.

These variables keep their value after a Reset, they are initialized with the maximum value possible when turning the CNC on.

KINETIMATICS DIMENSIONS.

(V.)[ch].A.HEADOF.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the execution or preparation value.

Dimension in each axis of the kinematics.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.A.HEADOF.Z	Z axis.
V.A.HEADOF.4	Axis with logic number ·4·.
V.[2].A.HEADOF.1	Axis with index ·1· in the channel ·2·.

Remarks.

The variable returns the value of execution or preparation as follows. If the axis is consulted through its logic number, the variable always returns the preparation value. In the rest of the cases, if the axis belongs to the channel requesting the variable, it returns the preparation value; if the axis belongs to a different channel, the variable returns the execution value and interrupts block preparation.

This variable returns the resulting measurement of the active kinematics on that axis. It may be a particular value of TDATA (kinematics table) or the combination of several of them depending on the type of kinematics.

CHANGE THE TURNING DIRECTION ASSIGNED TO M3 AND M4.

(V.)[ch].A.POLARITY.sn
(V.)[ch].SP.POLARITY.sn

Variable that can be read and written via program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

This variable may be used to reverse the meaning of spindle parameters POLARM3 and POLARM4 that set the spindle turning direction for M3 and M4. The variable does not modify the values of the machine parameters.

If the spindle is turning in open loop (M3/M4), the CNC does not apply the changes of this variable immediately; the CNC applies the changes the next time a speed or an M3/M4 is programmed.

Syntax.

·ch· Channel number.

·sn· Name, logic number or index of the spindle.

V.A.POLARITY.S	Spindle S.
V.SP.POLARITY.S	Spindle S.
V.SP.POLARITY	Master spindle.
V.A.POLARITY.4	Spindle with logic number ·4·.
V.SP.POLARITY.2	Spindle with index ·2· in the system.
V.[2].SP.POLARITY.1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

Value.	Meaning.
0	For the M3/M4 functions, the CNC applies parameters POLARM3 and POLARM4.
1	For the M3/M4 functions, the CNC reverses the meaning of parameters POLARM3 and POLARM4.

Remarks.

The variable keeps its value between programs and after a reset; it is initialized to ·0· when turning the CNC on.

Every time the part-program writes or reads this variable, the CNC synchronizes the block preparation and the execution. If it is the PLC who modifies the variable, it must be borne in mind that block preparation takes its value into account every time it detects a change in the spindle movement (Function G63, change of speed or of the turning direction, etc.); therefore, before preparing those blocks, the PLC must have written this variable.

NUMBER OF PULSES SENT BY THE HANDWHEEL.

(V.)G.HANDP[hw]

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Number of pulses sent by the handwheel since the system was started up. While the handwheel is disabled from the PLC (marks INHIBITMPG1 through INHIBITMPG12), the variable does not save the pulses sent by the handwheel.

Syntax.

·hw· Handwheel number.

(V.)G.HANDP[1]	Handwheel ·1·.

Remarks.

The PLC can use these variables to control the machining feedrate depending on how fast the handwheel is turned, for example when machining a part for the first time or to move back in the machining operation when the retrace function is active. Depending on the handwheel pulses read by the variable, it is possible to calculate from the PLC the right feedrate override percentage and set it for the machining operation. This handwheel mode is known as "feed handwheel".

CHANGE THE SIMULATION SPEED FROM THE PLC.

(V.)PLC.SIMUSPEED

Variable that can be read and written from the PLC and read from the program and from the interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

Percentage of simulation speed (between 0% and 100%).

V.PLC.SIMUSPEED

Currently, the simulation displays a horizontal bar on the screen to change the simulation speed. This variable may be used to change its value via PLC. In some cases, the OEM can transfer via PLC the feedrate override value to this variable to control the simulation speed.



The execution and simulation channels can work simultaneously in which case, the feedrate override switch will affect both. It is up to the OEM to use this variable in the right contexts so it only affects one of the channels.

10.27 Variables related to volumetric compensation.

(V.)[ch].A.VOLCOMP.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Value that the volumetric compensation is adding to the axis.

Reading this variable from the oscilloscope allows watching the evolution of the volumetric compensation.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.A.VOLCOMP.Z	Z axis.
V.A.VOLCOMP.3	Axis with logic number ·3·.
V.[2].A.VOLCOMP.3	Axis with index ·3· in the channel ·2·.

(V.)[ch].A.PIVOT.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
Variable valid for analog drive, position Sercos and velocity Sercos.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Distance from the pivot center to the tool tip in machine coordinates. This variable returns the sum of the spindle offsets (parameters TDATA) and the tool offset.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.A.PIVOT.Z	Z axis.
V.A.PIVOT.3	Axis with logic number ·3·.
V.[2].A.PIVOT.3	Axis with index ·3· in the channel ·2·.

10.28 Variables associated with the Mechatrolink bus.

STATUS OF COMMUNICATION AND OF THE MECHATROLINK DEVICES.

(V.)[ch].A.MSTATUS.xn
(V.)[ch].A.MSTATUS.sn
(V.)[ch].SP.MSTATUS.sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified from the oscilloscope environment.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for servo and Mechatrolink inverter.
The variable returns the execution value; reading it interrupts block preparation.

Information on the status of the execution of the command and of the device according to Mechatrolink specifications. Refer to the documentations of the device.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.MSTATUS.Z	Z axis.
V.MPA.MSTATUS.S	Spindle S.
V.SP.MSTATUS.S	Spindle S.
V.SP.MSTATUS	Master spindle.
V.MPA.MSTATUS.4	Axis or spindle with logic number ·4·.
V.[2].MPA.MSTATUS.1	Axis with index ·1· in the channel ·2·.
V.SP.MSTATUS.2	Spindle with index ·2· in the system.
V.[2].SP.MSTATUS.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.MSUBSTAT.xn
(V.)[ch].A.MSUBSTAT.sn
(V.)[ch].SP.MSUBSTAT.sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified from the oscilloscope environment.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for servo and Mechatrolink inverter.
The variable returns the execution value; reading it interrupts block preparation.

Information on the status of the execution of the subcommand and of the device according to Mechatrolink specifications. Refer to the documentations of the device.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.MSUBSTAT.Z	Z axis.
V.MPA.MSUBSTAT.S	Spindle S.
V.SP.MSUBSTAT.S	Spindle S.
V.SP.MSUBSTAT	Master spindle.
V.MPA.MSUBSTAT.4	Axis or spindle with logic number ·4·.
V.[2].MPA.MSUBSTAT.1	Axis with index ·1· in the channel ·2·.
V.SP.MSUBSTAT.2	Spindle with index ·2· in the system.
V.[2].SP.MSUBSTAT.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.MALARM.xn
(V.)[ch].A.MALARM.sn
(V.)[ch].SP.MALARM.sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified from the oscilloscope environment.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for servo and Mechatrolink inverter.
The variable returns the execution value; reading it interrupts block preparation.

Alarm or error code of the device. Refer to the documentations of the device.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.MALARM.Z	Z axis.
V.MPA.MALARM.S	Spindle S.
V.SP.MALARM.S	Spindle S.
V.SP.MALARM	Master spindle.
V.MPA.MALARM.4	Axis or spindle with logic number ·4·.
V.[2].MPA.MALARM.1	Axis with index ·1· in the channel ·2·.
V.SP.MALARM.2	Spindle with index ·2· in the system.
V.[2].SP.MALARM.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.MIOMON.xn
(V.)[ch].A.MIOMON.sn
(V.)[ch].SP.MIOMON.sn

Variable that can only be read from the program, PLC and interface.
Variable that may be modified from the oscilloscope environment.
Variable valid for rotary and linear axes as well as for spindles.
Variable valid for Mechatrolink servo.
The variable returns the execution value; reading it interrupts block preparation.

Status of the device inputs and outputs.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.MPA.MIOMON.Z	Z axis.
V.MPA.MIOMON.S	Spindle S.
V.SP.MIOMON.S	Spindle S.
V.SP.MIOMON	Master spindle.
V.MPA.MIOMON.4	Axis or spindle with logic number ·4·.
V.[2].MPA.MIOMON.1	Axis with index ·1· in the channel ·2·.
V.SP.MIOMON.2	Spindle with index ·2· in the system.
V.[2].SP.MIOMON.1	Spindle with index ·1· in the channel ·2·.

10.29 Variables related to synchronized switching.

The value of these variables is equivalent to the one programmed in the parameters of the instruction #SWTOUT. When executing the instruction without parameters and the variables have a value assigned to it, the CNC assumes these latter values as active. Likewise, when programming the parameters of the instruction, the variables assume these values as their own.

On CNC power-up, the offset wears are set to zero. After power-up, the values programmed for the offsets (using the instruction #SWTOUT or the variables) are maintained even after an error, a reset or an M30.

These variables interrupt block preparation. To modify the offset values without interrupting block preparation, change these values from the PLC or use the parameters of the instruction #SWTOUT. If these variables are changed from the PLC, the new values are assumed when executing the instruction #SWTOUT. If the instruction is active when changing the values from the PLC, it assumes the new values.

V.G.TON

Variable that can be read and written via program, PLC and interface.
The variable returns the execution value; reading/writing it interrupts block preparation.

Time shift (milliseconds) to anticipate the activation of the digital output.

V.G.TON

V.G.TOF

Variable that can be read and written via program, PLC and interface.
The variable returns the execution value; reading/writing it interrupts block preparation.

Time shift (milliseconds) to anticipate the deactivation of the digital output.

V.G.TOF

V.G.PON

Variable that can be read and written via program, PLC and interface.
The variable returns the execution value; reading/writing it interrupts block preparation.

Distance shift (offset, mm(inches) to anticipate the activation of the digital output.

V.G.PON

V.G.POF

Variable that can be read and written via program, PLC and interface.
The variable returns the execution value; reading/writing it interrupts block preparation.

Distance shift (offset, mm(inches) to anticipate the deactivation of the digital output.

V.G.POF

10.30 PWM related variables.

(V.)G.PWMON

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

PWM status, when it has been activated via CNC.

Syntax.

V.G.PWMON

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Laser OFF.
1	Laser ON.

Remarks.

Activating the Laser via PLC has priority over activating it from the CNC.

Laser status from the CNC. Variable (V.)G.PWMON	Laser status from the PLC. PWMON mark	Laser status.
0	0	Laser OFF.
1	0	Laser ON from the CNC.
0	1	Laser ON from the PLC.
1	1	Laser ON from the PLC.

(V.)G.PWMFREQ

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Frequency of the PWM (between 2 and 5000 Hz; by default, 0), when the PWM has been activated via CNC. This variable only works when the Laser is active via CNC.

Syntax.

V.G.PWMFREQ

(V.)G.PWMDUTY

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Duty cycle of the PWM (between 0.1 and 100%; by default, 50%), when the PWM has been activated via CNC. This variable only works when the Laser is active via CNC.

Syntax.

V.G.PWMDUTY

Remarks.

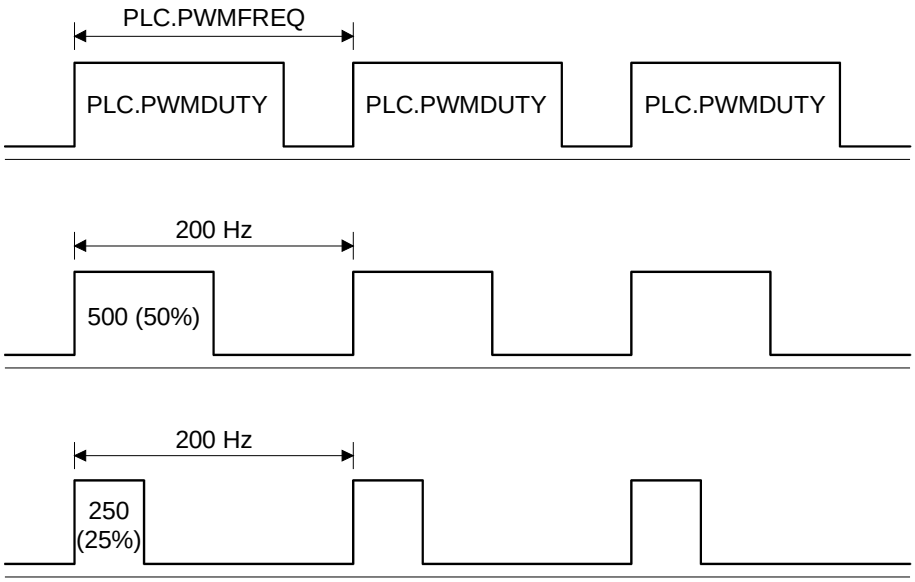
The reading from the PLC will be given in tenths (x10); i.e. for a value ·0.1·, the PLC reading will return a value of ·1·. For a value of ·100·, the readin from the PLC will return a value of ·1000·.

(V.)PLC.PWMFREQ

Variable that can be read and written from the PLC and read from the program and from the interface. The variable returns the execution value; reading it interrupts block preparation.

Frequency of the PWM (between 2 and 5000 Hz; by default, 0), when the PWM has been activated via PLC. Setting variable to ·0· deactivates the output of the PWM. This variable only works when the Laser has been activated via PLC.

PLC.PWMFREQ



Remarks.

The changes from the variable are updated (refreshed) in the next loop; the intermediate times between loops will be ignored.

(V.)PLC.PWMDUTY

Variable that can be read and written from the PLC and read from the program and from the interface. The variable returns the execution value; reading it interrupts block preparation.

Duty cycle of the PWM (between 1 and 100%; by default, 50%), when the PWM has been activated via PLC. This variable only works when the Laser has been activated via PLC.

PLC.PWMDUTY

Remarks.

The changes from the variable are updated (refreshed) in the next loop; the intermediate times between loops will be ignored.

The reading from the CNC will be divided by 10; i.e. for the value of ·1·, the CNC reading will return a value of ·0.1·.

10.31 Variables related to cycle time.

ANALYSIS OF THE LOOP TIME (CYCLE TIME) AT THE CNC.

(V.)G.NCTIMERATE

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Percentage of loop time (cycle time) used by the real time portion of the CNC.

```
V.G.NCTIMERATE
```

This variable serves to evaluate the load of the system and the time that the periodic interruption that controls the movement of the axes leaves the operating time to manage other applications that may be executed in parallel with the CNC. Applications that need time of the operating system are, for example, displaying the screens, user interface applications, refreshing variables on the screen, file management (subroutines or programs that are opened and closed while machining), etc.

If there isn't enough free time for the applications, it is possible to decrease parameter PREPFREQ, increase parameter LOOPTIME, group the subroutines in a the same file or reduce the number of external applications

(V.)G.LOOPTIMERATE

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Percentage of loop time (cycle time) used by the position loop.

```
V.G.LOOPTIMERATE
```

This variable serves as reference to know whether the consumption of interruption time is due to the number of axes or to the path preparation process itself.

If the position loop uses most of the interruption time, it means that the system is overloaded by the axes and, therefore, you will have to evaluate the possibility to increase the LOOPTIME parameter.

ANALYSIS OF THE LOOP TIME (CYCLE TIME) IN THE CHANNEL.

(V.)[ch].G.CHTIMERATE

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Percentage of loop time (cycle time) used by the channel.

This variable helps determine whether the particular execution of a channel is taking up too much time.

Syntax.

·ch· Channel number.

```
V.[2].G.CHTIMERATE                Channel ·2·.
```

(V.)[ch].G.PREPTIMERATE

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Percentage of loop time (cycle time) used by the channel for block preparation.

This variable serves to evaluate the load in path preparation and to know whether it is possible to increase the PREPFREQ parameter or not.

Syntax.

·ch· Channel number.

V.[2].G.PREPTIMERATE	Channel ·2·.

(V.)[ch].G.IPOTIMERATE

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Percentage of loop time (cycle time) used by the interpolator of the channel.

This variable serves to evaluate the overload in the path generating algorithm and the smoothing algorithm.

Syntax.

·ch· Channel number.

V.[2].G.IPOTIMERATE	Channel ·2·.

10.32 Variables associated with the feedback inputs for analog axes.

(V.)[ch].A.COUNTERST.xn
(V.)[ch].A.COUNTERST.sn
(V.)[ch].SP.COUNTERST.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Status of the feedback input.

For a feedback input to be active, it must have an analog axis associated with it.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.COUNTERST.Z	Z axis.
V.A.COUNTERST.S	Spindle S.
V.SP.COUNTERST.S	Spindle S.
V.SP.COUNTERST	Master spindle.
V.A.COUNTERST.4	Axis or spindle with logic number ·4·.
V.[2].A.COUNTERST.1	Axis with index ·1· in the channel ·2·.
V.SP.COUNTERST.2	Spindle with index ·2· in the system.
V.[2].SP.COUNTERST.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.COUNTER.xn
(V.)[ch].A.COUNTER.sn
(V.)[ch].SP.COUNTER.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Feedback input pulses (integer side + fraction side).

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.COUNTER.Z	Z axis.
V.A.COUNTER.S	Spindle S.
V.SP.COUNTER.S	Spindle S.
V.SP.COUNTER	Master spindle.
V.A.COUNTER.4	Axis or spindle with logic number ·4·.
V.[2].A.COUNTER.1	Axis with index ·1· in the channel ·2·.
V.SP.COUNTER.2	Spindle with index ·2· in the system.
V.[2].SP.COUNTER.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.ASINUS.xn
(V.)[ch].A.ASINUS.sn
(V.)[ch].SP.ASINUS.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Fraction side of the A signal.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.ASINUS.Z	Z axis.
V.A.ASINUS.S	Spindle S.
V.SP.ASINUS.S	Spindle S.
V.SP.ASINUS	Master spindle.
V.A.ASINUS.4	Axis or spindle with logic number ·4·.
V.[2].A.ASINUS.1	Axis with index ·1· in the channel ·2·.
V.SP.ASINUS.2	Spindle with index ·2· in the system.
V.[2].SP.ASINUS.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.BSINUS.xn
(V.)[ch].A.BSINUS.sn
(V.)[ch].SP.BSINUS.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Fraction side of the B signal.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.BSINUS.Z	Z axis.
V.A.BSINUS.S	Spindle S.
V.SP.BSINUS.S	Spindle S.
V.SP.BSINUS	Master spindle.
V.A.BSINUS.4	Axis or spindle with logic number ·4·.
V.[2].A.BSINUS.1	Axis with index ·1· in the channel ·2·.
V.SP.BSINUS.2	Spindle with index ·2· in the system.
V.[2].SP.BSINUS.1	Spindle with index ·1· in the channel ·2·.

10.33 Variables associated with the analog inputs and outputs.

(V.)G.ANAI[n]

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

[n] input voltage (in volts).

Syntax.

·n· Number of the analog input.

V.G.ANAI[3]	Voltage of the analog input ·3·.

(V.)G.ANAO[n]

Variable that can be read from the interface and written from the program and from the PLC.
The variable returns the execution value; reading it interrupts block preparation.

[n] output voltage (in volts).

Syntax.

·n· Number of the analog input.

V.G.ANAO[3]	Voltage of the analog output ·3·.

10.34 Variables associated with the velocity command and the feedback of the drive.

VELOCITY COMMAND AND TORQUE FOR SERCOS AXES.

(V.)[ch].A.FTEO.xn
(V.)[ch].A.FTEO.sn
(V.)[ch].SP.FTEO.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Velocity command for Sercos (in rpm).

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.FTEO.Z	Z axis.
V.A.FTEO.S	Spindle S.
V.SP.FTEO.S	Spindle S.
V.SP.FTEO	Master spindle.
V.A.FTEO.4	Axis or spindle with logic number ·4·.
V.[2].A.FTEO.1	Axis with index ·1· in the channel ·2·.
V.SP.FTEO.2	Spindle with index ·2· in the system.
V.[2].SP.FTEO.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.POSCMD.xn
(V.)[ch].A.POSCMD.sn
(V.)[ch].SP.POSCMD.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Position command for Sercos.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.POSCMD.Z	Z axis.
V.A.POSCMD.S	Spindle S.
V.SP.POSCMD.S	Spindle S.
V.SP.POSCMD	Master spindle.
V.A.POSCMD.4	Axis or spindle with logic number ·4·.
V.[2].A.POSCMD.1	Axis with index ·1· in the channel ·2·.
V.SP.POSCMD.2	Spindle with index ·2· in the system.
V.[2].SP.POSCMD.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.TORQUE.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Current torque in Sercos.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.TORQUE.Z	Z axis.
V.A.TORQUE.S	Spindle S.
V.SP.TORQUE.S	Spindle S.
V.SP.TORQUE	Master spindle.
V.A.TORQUE.4	Axis or spindle with logic number ·4·.
V.[2].A.TORQUE.1	Axis with index ·1· in the channel ·2·.
V.SP.TORQUE.2	Spindle with index ·2· in the system.
V.[2].SP.TORQUE.1	Spindle with index ·1· in the channel ·2·.

Remarks.

The reading from the PLC will be given in tenths (x10); i.e. if the parameter value is ·10·, the PLC reading will return a value of ·100·.

FEEDBACK OF THE ANALOG OR SERCOS DRIVE.

(V.)[ch].A.POSNC.xn
(V.)[ch].A.POSNC.sn
(V.)[ch].SP.POSNC.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Position feedback.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.POSNC.Z	Z axis.
V.A.POSNC.S	Spindle S.
V.SP.POSNC.S	Spindle S.
V.SP.POSNC	Master spindle.
V.A.POSNC.4	Axis or spindle with logic number ·4·.
V.[2].A.POSNC.1	Axis with index ·1· in the channel ·2·.
V.SP.POSNC.2	Spindle with index ·2· in the system.
V.[2].SP.POSNC.1	Spindle with index ·1· in the channel ·2·.

10.35 Variables related to the change of gear and set of the Sercos drive.

(V.)[ch].A.SETGE.xn
(V.)[ch].A.SETGE.sn
(V.)[ch].SP.SETGE.sn

Variable that can be read and written via program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Select the set and the gear at the drive. Only one change can be taking place at a time. If other gear or set changes are programmed during this process, even if they are at different drives, the CNC only keeps the one programmed last and ignores the rest of the intermediate changes.

The drive can have 8 work ranges or gears identified from 0 through 7 (drive parameter GP6) and 8 sets of parameters (drive parameter GP4) identified from 0 through 7.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.SETGE.Z	Z axis.
V.A.SETGE.S	Spindle S.
V.SP.SETGE.S	Spindle S.
V.SP.SETGE	Master spindle.
V.A.SETGE.4	Axis or spindle with logic number ·4·.
V.[2].A.SETGE.1	Axis with index ·1· in the channel ·2·.
V.SP.SETGE.2	Spindle with index ·2· in the system.
V.[2].SP.SETGE.1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

The 4 least significant bits indicate the work range and the 4 most significant bits indicate the set of parameters. If the value of any 4-bit set is ·0·, the CNC does not change the active gear or set at the drive. Example of some values of the variable.

Value.	Meaning.
$21	First range or gear (gear ·0·). Second set of parameters (set ·1·).
$40	The drive maintains the active gear. Fourth set of parameters (set ·3·).
$07	Seventh range or gear (gear ·6·). The drive maintains the active parameter set.

Remarks.

The CNC keeps the SERPLCAC mark active during the change process.

10.36 Variables related to loop adjustment.

COORDINATE RELATED VARIABLES.

(V.)[ch].A.IPOPOS.xn
(V.)[ch].A.IPOPOS.sn
(V.)[ch].SP.IPOPOS.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Theoretical position value (coordinate) at the output of the interpolator, before the transformation; i.e. in part coordinates.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.IPOPOS.Z	Z axis.
V.A.IPOPOS.S	Spindle S.
V.SP.IPOPOS.S	Spindle S.
V.SP.IPOPOS	Master spindle.
V.A.IPOPOS.4	Axis or spindle with logic number ·4·.
V.[2].A.IPOPOS.1	Axis with index ·1· in the channel ·2·.
V.SP.IPOPOS.2	Spindle with index ·2· in the system.
V.[2].SP.IPOPOS.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.FILTERIN.xn
(V.)[ch].A.FILTERIN.sn
(V.)[ch].SP.FILTERIN.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Theoretical coordinate of the interpolator before the filter.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.FILTERIN.Z	Z axis.
V.A.FILTERIN.S	Spindle S.
V.SP.FILTERIN.S	Spindle S.
V.SP.FILTERIN	Master spindle.
V.A.FILTERIN.4	Axis or spindle with logic number ·4·.
V.[2].A.FILTERIN.1	Axis with index ·1· in the channel ·2·.
V.SP.FILTERIN.2	Spindle with index ·2· in the system.
V.[2].SP.FILTERIN.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.FILTEROUT.xn
(V.)[ch].A.FILTEROUT.sn
(V.)[ch].SP.FILTEROUT.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Theoretical coordinate of the interpolator after the filter.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.FILTEROUT.Z	Z axis.
V.A.FILTEROUT.S	Spindle S.
V.SP.FILTEROUT.S	Spindle S.
V.SP.FILTEROUT	Master spindle.
V.A.FILTEROUT.4	Axis or spindle with logic number ·4·.
V.[2].A.FILTEROUT.1	Axis with index ·1· in the channel ·2·.
V.SP.FILTEROUT.2	Spindle with index ·2· in the system.
V.[2].SP.FILTEROUT.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.LOOPTPOS.xn
(V.)[ch].A.LOOPTPOS.sn
(V.)[ch].SP.LOOPTPOS.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Theoretical coordinate at the input of the position loop.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.LOOPTPOS.Z	Z axis.
V.A.LOOPTPOS.S	Spindle S.
V.SP.LOOPTPOS.S	Spindle S.
V.SP.LOOPTPOS	Master spindle.
V.A.LOOPTPOS.4	Axis or spindle with logic number ·4·.
V.[2].A.LOOPTPOS.1	Axis with index ·1· in the channel ·2·.
V.SP.LOOPTPOS.2	Spindle with index ·2· in the system.
V.[2].SP.LOOPTPOS.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.LOOPPOS.xn
(V.)[ch].A.LOOPPOS.sn
(V.)[ch].SP.LOOPPOS.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Real coordinate at the input of the position loop.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.LOOPPOS.Z	Z axis.
V.A.LOOPPOS.S	Spindle S.
V.SP.LOOPPOS.S	Spindle S.
V.SP.LOOPPOS	Master spindle.
V.A.LOOPPOS.4	Axis or spindle with logic number ·4·.
V.[2].A.LOOPPOS.1	Axis with index ·1· in the channel ·2·.
V.SP.LOOPPOS.2	Spindle with index ·2· in the system.
V.[2].SP.LOOPPOS.1	Spindle with index ·1· in the channel ·2·.

POSITION INCREMENT AND SAMPLING PERIOD.

(V.)[ch].A.POSINC.xn
(V.)[ch].A.POSINC.sn
(V.)[ch].SP.POSINC.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Real position increment of the current sampling period.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.POSINC.Z	Z axis.
V.A.POSINC.S	Spindle S.
V.SP.POSINC.S	Spindle S.
V.SP.POSINC	Master spindle.
V.A.POSINC.4	Axis or spindle with logic number ·4·.
V.[2].A.POSINC.1	Axis with index ·1· in the channel ·2·.
V.SP.POSINC.2	Spindle with index ·2· in the system.
V.[2].SP.POSINC.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.TPOSINC.xn
(V.)[ch].A.TPOSINC.sn
(V.)[ch].SP.TPOSINC.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Theoretical position increment of the current sampling period.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.TPOSINC.Z	Z axis.
V.A.TPOSINC.S	Spindle S.
V.SP.TPOSINC.S	Spindle S.
V.SP.TPOSINC	Master spindle.
V.A.TPOSINC.4	Axis or spindle with logic number ·4·.

V.[2].A.TPOSINC.1	Axis with index ·1· in the channel ·2·.
V.SP.TPOSINC.2	Spindle with index ·2· in the system.
V.[2].SP.TPOSINC.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.PREVPOSINC.xn
(V.)[ch].A.PREVPOSINC.sn
(V.)[ch].SP.PREVPOSINC.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Real position increment of the previous sampling period.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.PREVPOSINC.Z	Z axis.
V.A.PREVPOSINC.S	Spindle S.
V.SP.PREVPOSINC.S	Spindle S.
V.SP.PREVPOSINC	Master spindle.
V.A.PREVPOSINC.4	Axis or spindle with logic number ·4·.
V.[2].A.PREVPOSINC.1	Axis with index ·1· in the channel ·2·.
V.SP.PREVPOSINC.2	Spindle with index ·2· in the system.
V.[2].SP.PREVPOSINC.1	Spindle with index ·1· in the channel ·2·.

FINE ADJUSTMENT OF FEEDRATE, ACCELERATION AND JERK.

(V.)[ch].A.TFEED.xn
(V.)[ch].A.TFEED.sn
(V.)[ch].SP.TFEED.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Instantaneous theoretical speed value at the input of the position loop.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.TFEED.Z	Z axis.
V.A.TFEED.S	Spindle S.
V.SP.TFEED.S	Spindle S.
V.SP.TFEED	Master spindle.
V.A.TFEED.4	Axis or spindle with logic number ·4·.
V.[2].A.TFEED.1	Axis with index ·1· in the channel ·2·.
V.SP.TFEED.2	Spindle with index ·2· in the system.
V.[2].SP.TFEED.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.FEED.xn
(V.)[ch].A.FEED.sn
(V.)[ch].SP.FEED.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Instantaneous real speed value at the input of the position loop.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.FEED.Z	Z axis.
V.A.FEED.S	Spindle S.
V.SP.FEED.S	Spindle S.
V.SP.FEED	Master spindle.
V.A.FEED.4	Axis or spindle with logic number ·4·.
V.[2].A.FEED.1	Axis with index ·1· in the channel ·2·.
V.SP.FEED.2	Spindle with index ·2· in the system.
V.[2].SP.FEED.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.TACCEL.xn
(V.)[ch].A.TACCEL.sn
(V.)[ch].SP.TACCEL.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Theoretical instantaneous acceleration value.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.TACCEL.Z	Z axis.
V.A.TACCEL.S	Spindle S.
V.SP.TACCEL.S	Spindle S.
V.SP.TACCEL	Master spindle.
V.A.TACCEL.4	Axis or spindle with logic number ·4·.
V.[2].A.TACCEL.1	Axis with index ·1· in the channel ·2·.
V.SP.TACCEL.2	Spindle with index ·2· in the system.
V.[2].SP.TACCEL.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.ACCEL.xn
(V.)[ch].A.ACCEL.sn
(V.)[ch].SP.ACCEL.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Real instantaneous acceleration value.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.ACCEL.Z	Z axis.
V.A.ACCEL.S	Spindle S.
V.SP.ACCEL.S	Spindle S.
V.SP.ACCEL	Master spindle.
V.A.ACCEL.4	Axis or spindle with logic number ·4·.
V.[2].A.ACCEL.1	Axis with index ·1· in the channel ·2·.
V.SP.ACCEL.2	Spindle with index ·2· in the system.
V.[2].SP.ACCEL.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.TJERK.xn
(V.)[ch].A.TJERK.sn
(V.)[ch].SP.TJERK.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Theoretical instantaneous jerk value.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.TJERK.Z	Z axis.
V.A.TJERK.S	Spindle S.
V.SP.TJERK.S	Spindle S.
V.SP.TJERK	Master spindle.
V.A.TJERK.4	Axis or spindle with logic number ·4·.
V.[2].A.TJERK.1	Axis with index ·1· in the channel ·2·.
V.SP.TJERK.2	Spindle with index ·2· in the system.
V.[2].SP.TJERK.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.JERK.xn
(V.)[ch].A.JERK.sn
(V.)[ch].SP.JERK.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Real instantaneous jerk value.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.JERK.Z	Z axis.
V.A.JERK.S	Spindle S.
V.SP.JERK.S	Spindle S.
V.SP.JERK	Master spindle.
V.A.JERK.4	Axis or spindle with logic number ·4·.
V.[2].A.JERK.1	Axis with index ·1· in the channel ·2·.
V.SP.JERK.2	Spindle with index ·2· in the system.
V.[2].SP.JERK.1	Spindle with index ·1· in the channel ·2·.

GAIN SETTING VIA PLC.

(V.)[ch].A.PLCFFGAIN.xn
(V.)[ch].A.PLCFFGAIN.sn
(V.)[ch].SP.PLCFFGAIN.sn

Variable that can be read and written from the PLC and read from the program and from the interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Percentage of feed-forward programmed from the PLC.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.PLCFFGAIN.Z	Z axis.
V.A.PLCFFGAIN.S	Spindle S.
V.SP.PLCFFGAIN.S	Spindle S.
V.SP.PLCFFGAIN	Master spindle.
V.A.PLCFFGAIN.4	Axis or spindle with logic number ·4·.
V.[2].A.PLCFFGAIN.1	Axis with index ·1· in the channel ·2·.
V.SP.PLCFFGAIN.2	Spindle with index ·2· in the system.
V.[2].SP.PLCFFGAIN.1	Spindle with index ·1· in the channel ·2·.

Remarks.

In order for the feed-forward and the AC-forward defined this way to be taken into account, they must be active by machine parameter; i.e. by means of machine parameter FFWTYPE if it is an analog drive or a simulated drive or parameter OPMODEP if it is a Sercos drive.

The values defined by these variables prevail over the ones defined by machine parameters or by program. Setting the variables with a negative value cancels their effect ("0" is a valid value). These variables are initialized neither by a reset nor when validating the parameters.

The reading from the PLC will be given in hundredths (x100); i.e. if the parameter value is ·10·, the PLC reading will return a value of ·1000·.

(V.)[ch].A.PLCACFGAIN.xn
(V.)[ch].A.PLCACFGAIN.sn
(V.)[ch].SP.PLCACFGAIN.sn

Variable that can be read and written from the PLC and read from the program and from the interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Percentage of AC-forward programmed from the PLC.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.PLCACFGAIN.Z	Z axis.
V.A.PLCACFGAIN.S	Spindle S.
V.SP.PLCACFGAIN.S	Spindle S.
V.SP.PLCACFGAIN	Master spindle.
V.A.PLCACFGAIN.4	Axis or spindle with logic number ·4·.
V.[2].A.PLCACFGAIN.1	Axis with index ·1· in the channel ·2·.
V.SP.PLCACFGAIN.2	Spindle with index ·2· in the system.
V.[2].SP.PLCACFGAIN.1	Spindle with index ·1· in the channel ·2·.

Remarks.

In order for the feed-forward and the AC-forward defined this way to be taken into account, they must be active by machine parameter; i.e. by means of machine parameter FFWTYPE if it is an analog drive or a simulated drive or parameter OPMODEP if it is a Sercos drive.

The values defined by these variables prevail over the ones defined by machine parameters or by program. Setting the variables with a negative value cancels their effect ("0" is a valid value). These variables are initialized neither by a reset nor when validating the parameters.

The reading from the PLC will be given in tenths (x10); i.e. if the parameter value is ·10·, the PLC reading will return a value of ·100·.

(V.)[ch].A.PLCPROGAIN.xn
(V.)[ch].A.PLCPROGAIN.sn
(V.)[ch].SP.PLCPROGAIN.sn

Variable that can be read and written from the PLC and read from the program and from the interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Proportional gain programmed from the PLC.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.PLCPROGAIN.Z	Z axis.
V.A.PLCPROGAIN.S	Spindle S.
V.SP.PLCPROGAIN.S	Spindle S.
V.SP.PLCPROGAIN	Master spindle.
V.A.PLCPROGAIN.4	Axis or spindle with logic number ·4·.
V.[2].A.PLCPROGAIN.1	Axis with index ·1· in the channel ·2·.
V.SP.PLCPROGAIN.2	Spindle with index ·2· in the system.
V.[2].SP.PLCPROGAIN.1	Spindle with index ·1· in the channel ·2·.

Remarks.

The values defined by these variables prevail over the ones defined by machine parameters or by program. Setting the variables with a negative value cancels their effect ("0" is a valid value). These variables are initialized neither by a reset nor when validating the parameters.

10.37 Variables related to the loop of the axis or of the tandem spindle.

(V.)[ch].A.TPIIN.xn
(V.)[ch].A.TPIIN.sn
(V.)[ch].SP.TPIIN.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Input of the PI of the master axis of the tandem (in rpm).

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.TPIIN.Z	Z axis.
V.A.TPIIN.S	Spindle S.
V.SP.TPIIN.S	Spindle S.
V.SP.TPIIN	Master spindle.
V.A.TPIIN.4	Axis or spindle with logic number ·4·.
V.[2].A.TPIIN.1	Axis with index ·1· in the channel ·2·.
V.SP.TPIIN.2	Spindle with index ·2· in the system.
V.[2].SP.TPIIN.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.TPIOUT.xn
(V.)[ch].A.TPIOUT.sn
(V.)[ch].SP.TPIOUT.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Output of the PI of the master axis of the tandem (in rpm).

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.TPIOUT.Z	Z axis.
V.A.TPIOUT.S	Spindle S.
V.SP.TPIOUT.S	Spindle S.
V.SP.TPIOUT	Master spindle.
V.A.TPIOUT.4	Axis or spindle with logic number ·4·.
V.[2].A.TPIOUT.1	Axis with index ·1· in the channel ·2·.
V.SP.TPIOUT.2	Spindle with index ·2· in the system.
V.[2].SP.TPIOUT.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.TFILTOUT.xn
(V.)[ch].A.TFILTOUT.sn
(V.)[ch].SP.TFILTOUT.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Output of the pre-load filter of the tandem.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.TFILTOUT.Z	Z axis.
V.A.TFILTOUT.S	Spindle S.
V.SP.TFILTOUT.S	Spindle S.
V.SP.TFILTOUT	Master spindle.
V.A.TFILTOUT.4	Axis or spindle with logic number ·4·.
V.[2].A.TFILTOUT.1	Axis with index ·1· in the channel ·2·.
V.SP.TFILTOUT.2	Spindle with index ·2· in the system.
V.[2].SP.TFILTOUT.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.PRELOAD.xn
(V.)[ch].A.PRELOAD.sn
(V.)[ch].SP.PRELOAD.sn

Variable that can be read and written via program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Preload in the tandem.

Syntax.

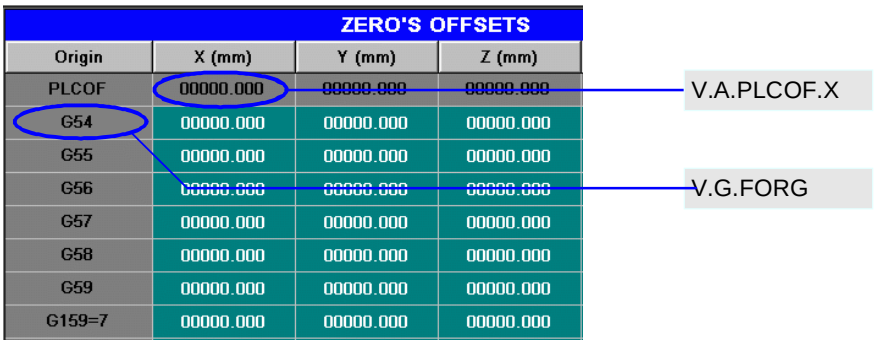
·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.PRELOAD.Z	Z axis.
V.A.PRELOAD.S	Spindle S.
V.SP.PRELOAD.S	Spindle S.
V.SP.PRELOAD	Master spindle.
V.A.PRELOAD.4	Axis or spindle with logic number ·4·.
V.[2].A.PRELOAD.1	Axis with index ·1· in the channel ·2·.
V.SP.PRELOAD.2	Spindle with index ·2· in the system.
V.[2].SP.PRELOAD.1	Spindle with index ·1· in the channel ·2·.

10.38 Variables related to user tables (zero offset table).



(V.)G.FORG

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

First zero offset in the table.

V.G.FORG

(V.)G.NUMORG

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Number of zero offsets in the table.

V.G.FORG

(V.)[ch].A.PLCOF.xn
(V.)[ch].A.PLCOF.sn
(V.)[ch].SP.PLCOF.sn

Variable that can be read and written from the program and from the PLC and read from the interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

xn axis. Offset of the zero offset set by PLC.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.PLCOF.Z	Z axis.
V.A.PLCOF.S	Spindle S.
V.SP.PLCOF.S	Spindle S.
V.SP.PLCOF	Master spindle.
V.A.PLCOF.4	Axis or spindle with logic number ·4·.
V.[2].A.PLCOF.1	Axis with index ·1· in the channel ·2·.
V.SP.PLCOF.2	Spindle with index ·2· in the system.
V.[2].SP.PLCOF.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.ACTPLCOF.xn
(V.)[ch].A.ACTPLCOF.sn
(V.)[ch].SP.ACTPLCOF.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

xn axis. Accumulated PLC offset.

Syntax.

·ch· Channel number.

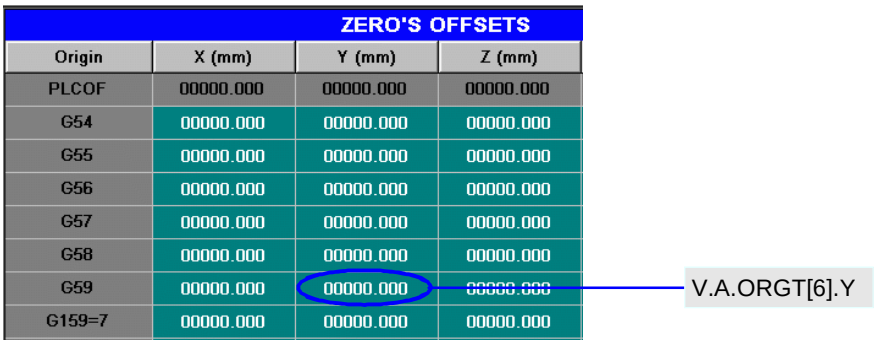
·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.ACTPLCOF.Z	Z axis.
V.A.ACTPLCOF.S	Spindle S.
V.SP.ACTPLCOF.S	Spindle S.
V.SP.ACTPLCOF	Master spindle.
V.A.ACTPLCOF.4	Axis or spindle with logic number ·4·.
V.[2].A.ACTPLCOF.1	Axis with index ·1· in the channel ·2·.
V.SP.ACTPLCOF.2	Spindle with index ·2· in the system.
V.[2].SP.ACTPLCOF.1	Spindle with index ·1· in the channel ·2·.

ZERO OFFSET TABLE (WITHOUT FINE SETTING OF THE ABSOLUTE ZERO OFFSET).

The fine setting of the absolute zero offset is enabled via machine parameters (parameter FINEORG)



(V.)[ch].A.ORG.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the value of block preparation.

xn axis. Value of the active zero offset (absolute G159 + incremental G158).

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.A.ORG.Z	Z axis.
V.A.ORG.3	Axis with logic number ·3·.
V.[2].A.ORG.3	Axis with index ·3· in the channel ·2·.

(V.)[ch].A.ADDORG.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the value of block preparation.

xn axis. Value of the active incremental zero offset (G158).

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.A.ADDORG.Z	Z axis.
V.A.ADDORG.3	Axis with logic number ·3·.
V.[2].A.ADDORG.3	Axis with index ·3· in the channel ·2·.

(V.)[ch].A.ORGT[nb].xn

Variable that can be read and written from the program and from the PLC and read from the interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

xn axis. Offset set in the zero offset [nb].

Syntax.

·ch· Channel number.

·nb· Zero offset number.

·xn· Name, logic number or index of the axis.

V.A.ORGT[1].Z	Zero offset G54 (G159=1). Z axis.
V.A.ORGT[4]0.3	Zero offset G57 (G159=4). Axis with logic number ·3·.
V.[2].A.ORGT[9].3	Zero offset G159=9. Axis with index ·3· in the channel ·2·.

Remarks.

The numbering of zero offsets G54 through G59 is always the same; G54=1, G55=2, G56=3, G57=4, G58=5, G59=6.

ZERO OFFSET TABLE (WITH FINE SETTING OF THE ABSOLUTE ZERO OFFSET).

The fine setting of the absolute zero offset is enabled via machine parameters (parameter FINEORG)

(V.)[ch].A.ORG.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the value of block preparation.

xn axis. Value of the active zero offset (coarse absolute G159 + fine absolute G159 + incremental G158).

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.A.ORG.Z	Z axis.
V.A.ORG.3	Axis with logic number ·3·.
V.[2].A.ORG.3	Axis with index ·3· in the channel ·2·.

(V.)[ch].A.ADDORG.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the value of block preparation.

xn axis. Value of the active incremental zero offset (G158).

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.A.ADDORG.Z	Z axis.
V.A.ADDORG.3	Axis with logic number ·3·.
V.[2].A.ADDORG.3	Axis with index ·3· in the channel ·2·.

(V.)[ch].A.COARSEORG.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the value of block preparation.

xn axis. Value of the active absolute zero offset (G159), coarse part.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.A.COARSEORG.Z	Z axis.
V.A.COARSEORG.3	Axis with logic number ·3·.
V.[2].A.COARSEORG.3	Axis with index ·3· in the channel ·2·.

(V.)[ch].A.FINEORG.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the value of block preparation.

xn axis. Value of the active absolute zero offset (G159), fine part.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.A.FINEORG.Z	Z axis.
V.A.FINEORG.3	Axis with logic number ·3·.
V.[2].A.FINEORG.3	Axis with index ·3· in the channel ·2·.

(V.)[ch].A.ORGT[nb].xn

Variable that can be read and written from the program and from the PLC and read from the interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

xn axis. Offset set in the zero offset [nb]; coarse part plus fine part.

When writing this variable, the value is assigned to the coarse part deleting the fine part.

Syntax.

·ch· Channel number.

·nb· Zero offset number.

·xn· Name, logic number or index of the axis.

V.A.ORGT[1].Z	Zero offset G54 (G159=1). Z axis.
V.A.ORGT[4]0.3	Zero offset G57 (G159=4). Axis with logic number ·3·.
V.[2].A.ORGT[9].3	Zero offset G159=9. Axis with index ·3· in the channel ·2·.

Remarks.

The numbering of zero offsets G54 through G59 is always the same; G54=1, G55=2, G56=3, G57=4, G58=5, G59=6.

(V.)[ch].A.COARSEORGT[nb].xn

Variable that can be read and written from the program and from the PLC and read from the interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

xn axis. Offset set in the zero offset [nb]; coarse part.

Syntax.

·ch· Channel number.

·nb· Zero offset number.

·xn· Name, logic number or index of the axis.

V.A.COARSEORGT[1].Z	Zero offset G54 (G159=1). Z axis.
V.A.COARSEORGT[4].3	Zero offset G57 (G159=4). Axis with logic number ·3·.
V.[2].A.COARSEORGT[9].3	Zero offset G159=9. Axis with index ·3· in the channel ·2·.

(V.)[ch].A.FINEORGT[nb].xn

Variable that can be read and written from the program and from the PLC and read from the interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

xn axis. Offset set in the zero offset [nb]; fine part.

Syntax.

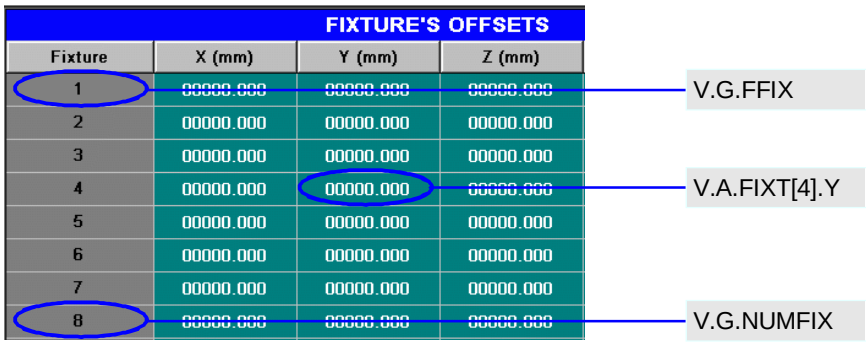
·ch· Channel number.

·nb· Zero offset number.

·xn· Name, logic number or index of the axis.

V.A.FINEORGT[1].Z	Zero offset G54 (G159=1). Z axis.
V.A.FINEORGT[4].3	Zero offset G57 (G159=4). Axis with logic number ·3·.
V.[2].A.FINEORGT[9].3	Zero offset G159=9. Axis with index ·3· in the channel ·2·.

10.39 Variables related to user tables (fixture table).



(V.)G.FFIX

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

First fixture of the table.

V.G.FFIX

(V.)G.NUMFIX

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Number of fixtures in the table.

V.G.NUMFIX

(V.)[ch].G.FIX

Variable that can be read and written from the program and read from the PLC and from the interface.
The variable returns the value of block preparation.
Report variable (to be used from the scripts).

Number of current fixture.

Syntax.

·ch· Channel number.

V.[2].G.FIX	Channel ·2·. Number of current fixture.

(V.)[ch].A.FIX.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the value of block preparation.

xn axis. Offset set in the current fixture.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.A.FIX.Z	Z axis.
V.A.FIX.3	Axis with logic number ·3·.
V.[2].A.FIX.3	Axis with index ·3· in the channel ·2·.

(V.)[ch].A.FIXT[nb].xn

Variable that can be read and written from the program and from the PLC and read from the interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

xn axis. Offset set in the [nb] fixture.

Syntax.

·ch· Channel number.

·nb· Fixture offset number.

·xn· Name, logic number or index of the axis.

V.A.FIXT[1].Z	First zero offset. Z axis.
V.A.FIXT[4].3	Fourth zero offset. Axis with logic number ·3·.
V.[2].A.FIXT[9].3	Ninth zero offset. Axis with index ·3· in the channel ·2·.

10.40 Variables related to user tables (arithmetic parameters table).

LOCAL ARITHMETIC PARAMETERS.

(V.)[ch].G.LUPACT[nb]

Variable that can be read and written via PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

Value of local arithmetic parameter [nb] of the current nesting level.

Syntax.

·ch· Channel number.

·nb· Parameter number.

[2].G.LUPACT[14]	Channel ·2·. Value of parameter ·14·.

(V.)[ch].G.LUP1[nb]
(V.)[ch].G.LUP2[nb]
(V.)[ch].G.LUP3[nb]
(V.)[ch].G.LUP4[nb]
(V.)[ch].G.LUP5[nb]
(V.)[ch].G.LUP6[nb]
(V.)[ch].G.LUP7[nb]

Variable that can be read and written via PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

Value of local arithmetic parameter [nb] of nesting level 1 through 7.

Syntax.

·ch· Channel number.

·nb· Parameter number.

[2].G.LUP1[14]	Channel ·2·. Value of parameter ·14· of nesting level ·1·.
[2].G.LUP7[6]	Channel ·2·. Value of parameter ·6· of nesting level ·7·.

Remarks.

When reading this variable from the PLC, it truncates the decimal side. If the value of the parameter is 54.9876, when read from the PLC, it will return the value 54.

Value of the parameter.	Reading from the PLC.
P14 = 23.1234	G.LUP1[14] = 23 G.LUP1F[14] = 231234
P22 =-12.0987	G.LUP1[22] = -12 G.LUP1F[22] = -120987

(V.)[ch].G.LUP1F[nb]
(V.)[ch].G.LUP2F[nb]
(V.)[ch].G.LUP3F[nb]
(V.)[ch].G.LUP4F[nb]
(V.)[ch].G.LUP5F[nb]
(V.)[ch].G.LUP6F[nb]
(V.)[ch].G.LUP7F[nb]

Variable that can be read and written via PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

Value of local arithmetic parameter [nb] of nesting level 1 through 7 (value x10000).

Syntax.

·ch· Channel number.

·nb· Parameter number.

[2].G.LUP1F[14]	Channel ·2·. Value of parameter ·14· of nesting level ·1·.
[2].G.LUP7F[6]	Channel ·2·. Value of parameter ·6· of nesting level ·7·.

Remarks.

Reading these variables from the PLC returns the parameter value multiplied by 10000. If the value of the parameter is 54.9876, when read from the PLC, it will return the value 549876.

Value of the parameter.	Reading from the PLC.
P14 = 23.1234	G.LUP1[14] = 23 G.LUP1F[14] = 231234
P22 =-12.0987	G.LUP1[22] = -12 G.LUP1F[22] = -120987

GLOBAL ARITHMETIC PARAMETERS.

(V.)[ch].G.GUP[nb]

Variable that can be read and written via PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

Value of the global arithmetic parameter [nb].

Syntax.

·ch· Channel number.

·nb· Parameter number.

[2].G.GUP[114]	Channel ·2·. Value of parameter ·114·.

Remarks.

When reading this variable from the PLC, it truncates the decimal side. If the value of the parameter is 54.9876, when read from the PLC, it will return the value 54.

Value of the parameter.	Reading the variable.
P114 = 124.4567	G.GUP[114] = 124 G.GUPF[114] = 1244567
P200 =-12.0987	G.GUP[200] = -12 G.GUPF[200] = -120987

(V.)[ch].G.GUPF[nb]

Variable that can be read and written via PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

Value of the global arithmetic parameter [nb] (value x10000).

Syntax.

·ch· Channel number.

·nb· Parameter number.

[2].G.GUP[114]	Channel ·2·. Value of parameter ·114·.

Remarks.

Reading these variables from the PLC returns the parameter value multiplied by 10000. If the value of the parameter is 54.9876, when read from the PLC, it will return the value 549876.

Value of the parameter.	Reading the variable.
P114 = 124.4567	G.GUP[114] = 124 G.GUPF[114] = 1244567
P200 =-12.0987	G.GUP[200] = -12 G.GUPF[200] = -120987

COMMON ARITHMETIC PARAMETERS.

(V.)G.CUP[nb]

Variable that can be read and written via PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

Value of the common arithmetic parameter [nb].

Syntax.

·nb· Parameter number.

[2].G.CUP[10014]	Channel ·2·. Value of parameter ·10014·.

Remarks.

When reading this variable from the PLC, it truncates the decimal side. If the value of the parameter is 54.9876, when read from the PLC, it will return the value 54.

Value of the parameter.	Reading the variable.
P10014 = 124.4567	G.CUP[10014] = 124 G.CUPF[10014] = 1244567
P10200 =-12.0987	G.CUP[10200] = -12 G.CUPF[10200] = -120987

(V.)G.CUPF[nb]

Variable that can be read and written via PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

Value of the common arithmetic parameter [nb] (value x10000).

Syntax.

·nb· Parameter number.

[2].G.CUPF[10014]	Channel ·2·. Value of parameter ·10014·.

Remarks.

Reading these variables from the PLC returns the parameter value multiplied by 10000. If the value of the parameter is 54.9876, when read from the PLC, it will return the value 549876.

Value of the parameter.	Reading the variable.
P10014 = 124.4567	G.CUP[10014] = 124 G.CUPF[10014] = 1244567
P10200 =-12.0987	G.CUP[10200] = -12 G.CUPF[10200] = -120987

10.41 Variables related to the position of the axes.

There are real and theoretical coordinates corresponding to the tool base and tool tip. All of them may be referred to Machine Zero or to the current Part Zero.

A theoretical coordinate is the position that the axis must occupy at all times, a real coordinate is the one it actually occupies and the difference between these two is called "following error".

PROGRAMMED COORDINATES.

(V.)[ch].A.PPOS.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the value of block preparation.

Channel [ch]. Programmed part coordinates (tool tip).

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.A.PPOS.Z	Z axis.
V.A.PPOS.3	Axis with logic number ·3·.
V.[2].A.PPOS.3	Axis with index ·3· in the channel ·2·.

Remarks.

This variable returns the target coordinate, in part coordinates and referred to the tool tip, in the current reference system; i.e. taking into consideration the coordinate rotation, scaling factor, active inclined plane, etc.

Programmed movements.	Returned values.
G1 X10	V.A.PPOS.X = 10
#SCALE [2] G1 X10	Scaling factor of ·2·. V.A.PPOS.X = 20
G73 Q90 G1 X10	Coordinate system rotation. V.A.PPOS.Y = 20 The Y axis is the one that moves

The values read from a program or from the PLC and the interface will be different when the coordinate is affected by tool compensation or when machining in round corner mode. The value read by program will be the programmed coordinate whereas the value read from the PLC or interface will be the real (actual) coordinate considering tool radius compensation and corner rounding.

(V.)[ch].G.PLPPOS1

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the value of block preparation.

Channel [ch]. Coordinate programmed in the first axis of the channel (tool tip).

Syntax.

·ch· Channel number.

V.[2].G.PLPPOS1	Channel ·2·.

(V.)[ch].G.PLPPOS2

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the value of block preparation.

Channel [ch]. Coordinate programmed in the second axis of the channel (tool tip).

Syntax.

·ch· Channel number.

V.[2].G.PLPPOS2	Channel ·2·.

(V.)[ch].G.PLPPOS3

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the value of block preparation.

Channel [ch]. Coordinate programmed in the third axis of the channel (tool tip).

Syntax.

·ch· Channel number.

V.[2].G.PLPPOS3	Channel ·2·.

POSITION IN PART COORDINATES.

(V.)[ch].A.APOS.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Real part coordinates of the tool base.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.A.APOS.Z	Z axis.
V.A.APOS.3	Axis with logic number ·3·.
V.[2].A.APOS.3	Axis with index ·3· in the channel ·2·.

(V.)[ch].A.ATPOS.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Theoretical part coordinates of the tool base.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.A.ATPOS.Z	Z axis.
V.A.ATPOS.3	Axis with logic number ·3·.
V.[2].A.ATPOS.3	Axis with index ·3· in the channel ·2·.

(V.)[ch].A.ATIPPOS.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Real part coordinates of the tool tip.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.A.ATIPPOS.Z	Z axis.
V.A.ATIPPOS.3	Axis with logic number ·3·.
V.[2].A.ATIPPOS.3	Axis with index ·3· in the channel ·2·.

(V.)[ch].A.ATIPTPOS.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Theoretical part coordinates of the tool tip.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.A.ATIPTPOS.Z	Z axis.
V.A.ATIPTPOS.3	Axis with logic number ·3·.
V.[2].A.ATIPTPOS.3	Axis with index ·3· in the channel ·2·.

POSITION IN MACHINE COORDINATES.

(V.)[ch].A.POS.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Real machine coordinates of the tool base.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.A.POS.Z	Z axis.
V.A.POS.3	Axis with logic number ·3·.
V.[2].A.POS.3	Axis with index ·3· in the channel ·2·.

(V.)[ch].A.TPOS.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Theoretical machine coordinates of the tool base.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.A.TPOS.Z	Z axis.
V.A.TPOS.3	Axis with logic number ·3·.
V.[2].A.TPOS.3	Axis with index ·3· in the channel ·2·.

(V.)[ch].A.TIPPOS.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Real machine coordinates of the tool tip.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.A.TIPPOS.Z	Z axis.
V.A.TIPPOS.3	Axis with logic number ·3·.
V.[2].A.TIPPOS.3	Axis with index ·3· in the channel ·2·.

(V.)[ch].A.TIPTPOS.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Theoretical machine coordinates of the tool tip.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.A.TIPTPOS.Z	Z axis.
V.A.TIPTPOS.3	Axis with logic number ·3·.
V.[2].A.TIPTPOS.3	Axis with index ·3· in the channel ·2·.

POSITION READ IN THE INTERNAL FEEDBACK (MOTOR FEEDBACK).

(V.)[ch].A.POSMOTOR.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Value of the internal feedback (motor feedback).

When the combined feedback is active (parameter FBMIXTIME), the variable returns the value of the internal feedback (motor feedback). If there is no feedback combination, the value of the variable coincides with (V.)[ch].A.POS.xn.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.A.POSMOTOR.Z	Z axis.
V.A.POSMOTOR.3	Axis with logic number ·3·.
V.[2].A.POSMOTOR.3	Axis with index ·3· in the channel ·2·.

FOLLOWING ERROR OF THE AXIS.

(V.)[ch].A.FLWE.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Following error, lag (difference between the theoretical position value and the actual, real, position value of the axis).

If there is no feedback combination, the CNC reads the actual (real) position from either the internal or the external feedback (parameter FBACKSRC). If feedback combination is active (parameter FBMIXTIME), the CNC calculates the following error (lag) using the combined feedback.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.A.FLWE.Z	Z axis.
V.A.FLWE.3	Axis with logic number ·3·.
V.[2].A.FLWE.3	Axis with index ·3· in the channel ·2·.

(V.)[ch].A.FLWACT.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Following error (lag) with respect to the external feedback (direct feedback).

When the combined feedback is active (parameter FBMIXTIME), the variable returns the following error (axis lag) with respect to the external feedback (direct feedback). If there is no feedback combination, the variable returns the following error (lag) with respect to the active feedback (parameter FBACKSRC), same as the variable "(V.)[ch].A.FLWE.xn".

Active feedback.	V.A.FLWE.xn	V.A.FLWACT.xn
Internal	Following error (lag) with respect to the internal feedback (motor feedback).	
External	Following error (lag) with respect to the external feedback (direct feedback).	
Feedback combination.	Following error (lag) with respect to the combined feedback.	Following error (lag) with respect to the external feedback.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.A.FLWACT.Z	Z axis.
V.A.FLWACT.3	Axis with logic number ·3·.
V.[2].A.FLWACT.3	Axis with index ·3· in the channel ·2·.

(V.)[ch].A.FLWEST.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Linear estimate of the following error (lag).

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.A.FLWE.Z	Z axis.
V.A.FLWE.3	Axis with logic number ·3·.
V.[2].A.FLWE.3	Axis with index ·3· in the channel ·2·.

10.42 Variables related to spindle position.

The following are used when the spindle is working in closed loop, it behaves like an axis. A theoretical coordinate is the position that the spindle must occupy at all times, a real coordinate is the one it actually occupies and the difference between these two is called "following error".

SPINDLE POSITION.

(V.)[ch].A.PPOS.sn
(V.)[ch].SP.PPOS.sn

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Programmed spindle position.

Syntax.

·ch· Channel number.

·sn· Name, logic number or index of the spindle.

V.A.PPOS.S	Spindle S.
V.SP.PPOS.S	Spindle S.
V.SP.PPOS	Master spindle.
V.A.PPOS.5	Spindle with logic number ·5·.
V.SP.PPOS.2	Spindle with index ·2· in the system.
V.[2].SP.PPOS.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.POS.sn
(V.)[ch].SP.POS.sn

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Real spindle position.

Syntax.

·ch· Channel number.

·sn· Name, logic number or index of the spindle.

V.A.POS.S	Spindle S.
V.SP.POS.S	Spindle S.
V.SP.POS	Master spindle.
V.A.POS.5	Spindle with logic number ·5·.
V.SP.POS.2	Spindle with index ·2· in the system.
V.[2].SP.POS.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.TPOS.sn
(V.)[ch].SP.TPOS.sn

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Theoretical spindle position.

Syntax.

·ch· Channel number.

·sn· Name, logic number or index of the spindle.

V.A.TPOS.S	Spindle S.
V.SP.TPOS.S	Spindle S.
V.SP.TPOS	Master spindle.
V.A.TPOS.5	Spindle with logic number ·5·.
V.SP.TPOS.2	Spindle with index ·2· in the system.
V.[2].SP.TPOS.1	Spindle with index ·1· in the channel ·2·.

SPINDLE FOLLOWING ERROR.

(V.)[ch].A.FLWE.sn
(V.)[ch].SP.FLWE.sn

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Spindle following error.

Syntax.

·ch· Channel number.

·sn· Name, logic number or index of the spindle.

V.A.FLWE.S	Spindle S.
V.SP.FLWE.S	Spindle S.
V.SP.FLWE	Master spindle.
V.A.FLWE.5	Spindle with logic number ·5·.
V.SP.FLWE.2	Spindle with index ·2· in the system.
V.[2].SP.FLWE.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.FLWEST.sn
(V.)[ch].SP.FLWEST.sn

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Linear estimate of the spindle following error (lag).

Syntax.

·ch· Channel number.

·sn· Name, logic number or index of the spindle.

V.A.FLWEST.S	Spindle S.
V.SP.FLWEST.S	Spindle S.
V.SP.FLWEST.S	Master spindle.
V.A.FLWEST.5	Spindle with logic number ·5·.
V.SP.FLWEST.2	Spindle with index ·2· in the system.
V.[2].SP.FLWEST.1	Spindle with index ·1· in the channel ·2·.

10.43 Feedrate related variables.

FEEDRATE ACTIVE IN THE CHANNEL.

(V.)[ch].G.FREAL

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Real feedrate on the tool path.

Use the oscilloscope to compare the real and theoretical feedrates along the path and detect adjustment problems when both are different at specific points. Also, using variables V.G.LINEN and V.G.BLKN, it is possible to associate these feedrate changes with the program blocks or lines where they occur.

Syntax.

·ch· Channel number.

V.[2].G.FREAL	Channel ·2·.

Remarks.

This variable takes into account the accelerations and decelerations of the machine. When the axes are stopped, it returns a value of ·0· and when moving it returns the value corresponding to the feedrate type G94/G95. On laser cutting machines, it is recommended to use this variable so the laser power is proportional to the feedrate.

(V.)[ch].G.PATHFEED

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Theoretical feedrate on the tool path.

Use the oscilloscope to compare the real and theoretical feedrates along the path and detect adjustment problems when both are different at specific points. Also, using variables V.G.LINEN and V.G.BLKN, it is possible to associate these feedrate changes with the program blocks or lines where they occur.

Syntax.

·ch· Channel number.

V.[2].G.PATHFEED	Channel ·2·.

(V.)[ch].PLC.G00FEED

Variable that can be read and written from the PLC and read from the program and from the interface.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Maximum feedrate permitted in the channel.

This variable may be used to limit any time, in real time, the maximum feedrate in the channel for any type of movement (G00, G01, etc). The CNC assumes the change immediately and stays active until the variable takes a value of ·0· restoring the limit set by machine parameter.

Syntax.

·ch· Channel number.

V.[2].PLC.G00FEED	Channel ·2·.

Remarks.

Regardless of the value assigned to this variable, the feedrate will never exceed the limits set by parameters G00FEED and MAXFEED.

Variable.	Machine parameters.		Feedrate active in the channel.	
(V.)[ch].PLC.G00FEED	G00FEED	MAXFEED	G00	G01, G02, ...
3000	10000	5000	3000	3000
7000	10000	5000	7000	5000
12000	10000	5000	10000	5000

PROGRAMMING THE FEEDRATE IN G94.

(V.)[ch].G.FEED

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Feedrate active in G94.

Syntax.

·ch· Channel number.

V.[2].G.FEED	Channel ·2·.

Remarks.

The feedrate in G94 may be set by program or by PLC; the one set by PLC has the highest priority.

Feedrates set.	(V.)[ch].PLC.F	(V.)[ch].G.PRGF	(V.)[ch].G.FEED
By program; F2000. By PLC; none.	0	2000	2000
By program; F2000. By PLC; F4000.	4000	2000	4000
By program; F2000. By PLC; F500.	500	2000	500
By program; F2000. By PLC; none. By MDI; F3000.	0	3000	3000
By program; F2000. By PLC; F6000. By MDI; F3000.	6000	3000	6000
By program; F2000. By PLC; F500. By MDI; F3000.	500	3000	500

(V.)[ch].PLC.F

Variable that can be read and written from the PLC and read from the program and from the interface.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Feedrate by PLC in G94.

The feedrate programmed by PLC prevails over the one programmed by program or MDI. To cancel the feedrate by PLC, set the variable to ·0·; the CNC applies the feedrate active by program.

Syntax.

·ch· Channel number.

V.[2].PLC.F	Channel ·2·.

(V.)[ch].G.PRGF

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Feedrate by program in G94.

Being G94 active, programming a new feedrate in MDI mode updates the value of this variable.

Syntax.

·ch· Channel number.

V.[2].G.PRGF	Channel ·2·.

PROGRAMMING THE FEEDRATE IN G95.

(V.)[ch].G.FPREV

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Feedrate active in G95.

Syntax.

·ch· Channel number.

V.[2].G.FPREV	Channel ·2·.

Remarks.

The feedrate in G95 may be set by program or by PLC; the one set by PLC has the highest priority.

Feedrates set.	(V.)[ch].PLC.FPR	(V.)[ch].G.PRGFPR	(V.)[ch].G.FPREV
By program; F0.5. By PLC; none.	0	0.5	0.5
By program; F0.5. By PLC; F0.7.	0.7	0.5	0.7
By program; F0.5. By PLC; F0.12.	0.12	0.5	0.12
By program; F0.5. By PLC; none. By MDI; F1.8.	0	1.8	1.8
By program; F0.5. By PLC; F2.5. By MDI; F1.8.	2.5	1.8	2.5
By program; F0.5. By PLC; F0.7. By MDI; F1.8.	0.7	1.8	0.7

(V.)[ch].PLC.FPR

Variable that can be read and written from the PLC and read from the program and from the interface.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Feedrate by PLC in G95.

The feedrate programmed by PLC prevails over the one programmed by program or MDI. To cancel the feedrate by PLC, set the variable to ·0·; the CNC applies the feedrate active by program.

Syntax.

·ch· Channel number.

V.[2].PLC.FPR	Channel ·2·.

(V.)[ch].G.PRGFPR

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Feedrate by program in G95.

Being G95 active, programming a new feedrate in MDI mode updates the value of this variable.

Syntax.

·ch· Channel number.

V.[2].G.PRGFPR	Channel ·2·.

PROGRAMMING THE MACHINING TIME.

(V.)[ch].G.FTIME

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Machining time in G93 (in seconds).

Syntax.

·ch· Channel number.

V.[2].G.FTIME	Channel ·2·.

PERCENTAGE OF FEEDRATE (FEEDRATE OVERRIDE).

(V.)[ch].G.FRO

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

Channel [ch]. Feedrate percentage active in the channel.

Syntax.

·ch· Channel number.

V.[2].G.FRO	Channel ·2·.

Remarks.

The Feedrate override percentage may be set by program, by PLC or with the selector switch; the one set by program has the highest priority and the one selected with the switch the lowest.

(V.)[ch].G.PRGFRO	(V.)[ch].PLC.FRO	(V.)[ch].G.CNCFRO	(V.)[ch].G.FRO
0	0	70 %	70 %
0	40 %	70 %	40 %
85 %	40 %	70 %	85 %
20 %	90 %	70 %	20 %
20 %	0	70 %	20 %

(V.)[ch].G.PRGFRO

Variable that can be read and written from the program and read from the PLC and from the interface.
The variable returns the value of block preparation.

Channel [ch]. Percentage of feedrate by program.

The percentage set by program has higher priority than the one set by PLC or by the switch. To cancel the value set by program, set the variable to ·0·.

Syntax.

·ch· Channel number.

V.[2].G.PRGFRO	Channel ·2·.

(V.)[ch].PLC.FRO

Variable that can be read and written from the PLC and read from the program and from the interface. The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Percentage of feedrate by PLC.

The percentage set by PLC has higher priority than the one set by the switch, but lower priority than the one set by program.. To cancel the value set by PLC, set the variable to ·0·. To set a 0% value set by PLC, set the variable to -1.

Syntax.

·ch· Channel number.

V.[2].PLC.FRO	Channel ·2·.

(V.)[ch].G.CNCFRO

Variable that can be read and written (asynchronous writing) from the interface and read from the program and from the PLC.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Feedrate percentage at the switch of the operator panel.

The percentage set by the switch of the operator panel has lower priority than the one set by PLC or by program.

Syntax.

·ch· Channel number.

V.[2].G.CNCFRO	Channel ·2·.

10.44 Variables associated with acceleration and jerk on the tool path.

(V.)[ch].G.MAXACCEL

Variable that can be read and written via program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Modify the maximum acceleration allowed on the machining path.

The maximum acceleration allowed on the machining path is set by parameter MAXACCEL and this variable may be used to change this value temporarily at a particular time and in real time. The CNC assumes the change immediately and stays active until M30 or reset is executed, in which case the CNC restores the value set by the machine parameter. If the variable takes the value of ·0·, the CNC does not apply any acceleration limit on the tool path, not even the one set by machine parameter.

The CNC always respects the dynamics of the axes involved in the path; i.e. regardless of the maximum acceleration allowed on the path, the movement will respect the acceleration set for each axis.

Syntax.

·ch· Channel number.

V.[2].G.MAXACCEL	Channel ·2·.

(V.)[ch].G.MAXJERK

Variable that can be read and written via program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Modify the maximum jerk on the machining path.

The maximum jerk allowed on the machining path is set by parameter MAXJERK and this variable may be used to change this value temporarily at a particular time and in real time. The CNC assumes the change immediately and stays active until M30 or reset is executed, in which case the CNC restores the value set by the machine parameter. If the variable takes the value of ·0·, the CNC does not apply any jerk limit on the tool path, not even the one set by machine parameter.

The CNC always respects the dynamics of the axes involved in the path; i.e. regardless of the maximum jerk allowed on the path, the movement will respect the jerk set for each axis.

Syntax.

·ch· Channel number.

V.[2].G.MAXJERK	Channel ·2·.

10.45 Variables related to managing the feedrate in HSC mode.

VARIABLES RELATED TO BLOCK PREPARATION.

(V.)[ch].G.PERFRATE

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Percentage of blocks managed by the CNC in block preparation, with respect to the best possible to reach the maximum feedrate in each section.

Syntax.

·ch· Channel number.

V.[2].G.PERFRATE	Channel ·2·.

Remarks.

This variable can return a value close to 100; if the percentage of blocks is lower than 100%, the HSC might be losing room to increase the feedrate, respecting the available room to brake. To know if this is the case, it will be necessary to analyze the V.G.DROPRATE variable because the reason the feedrate is not increasing might be geometrical and not due to the number of blocks available.

(V.)[ch].G.DROPRATE

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Percentage of reduction of the maximum feedrate possible.

Feedrate reduction may be due to an improper supply of blocks or because a feedrate lower than possible has been programmed.

Syntax.

·ch· Channel number.

V.[2].G.DROPRATE	Channel ·2·.

Remarks.

This variable can return a value close to 100; if the value is lower than 100%, the CNC may increase the feedrate if it has more blocks. To know whether the CNC can supply more blocks or not, it will be necessary to analyze the V.G.PERFRATE variable.

If both variables have a value lower than 100, parameter PREPFREQ may be increased to increase the supply of blocks as long as the system has enough time; in other words, the percentage of cycle time used by the CNC is not too close to the total cycle time (about 50%). This information may be checked in the V.G.NCTIMERATE variable.

ANALYSIS OF THE PROGRAMMED ERROR.

(V.)[ch].G.LIMERROR

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Value of the error that cancels the splines (CONTERROR mode).

This variable may be used to evaluate whether the program is generated with more error than we demand from the HSC mode or not.

Syntax.

·ch· Channel number.

V.[2].G.LIMERROR	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
-1	The generated profile does not exceed the programmed error. The programmed error does not limit the maximum feedrate of the axis.
###	Value of the error that cancels the splines (CONTERROR mode).

(V.)[ch].G.ACTROUND

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Error programmed in HSC mode.

This variable may be used to optimize the OEM subroutines by cancelling the HSC mode and activating G5 with another error to save time and restore the previous execution mode later one.

Syntax.

·ch· Channel number.

V.[2].G.ACTROUND	Channel ·2·.

FEEDRATE LIMITATION AT THE BLOCK BEING EXECUTED.

(V.)[ch].G.AXLIMF

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Logic number of the axis that limits the feedrate at the block being executed.

Together with the V.G.PARLIMF variable, it may be used to evaluate the behavior of the machining operation in a particular section where the feedrate decreases too much or is irregular.

Syntax.

·ch· Channel number.

V.[2].G.AXLIMF	Channel ·2·.

(V.)[ch].G.PARLIMF

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Cause that limits the feedrate at the block being executed.

Syntax.

·ch· Channel number.

V.[2].G.PARLIMF	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
1	Maximum feedrate of the axis.
2	Acceleration due to curvature (parameter CURVACC).
3	Jerk due to curvature (parameter CURVJERK).
6	Error committed by the spline.
7	Insufficient memory due to very small blocks and/or buffer full.
10	Maximum axis feedrate in transformations.
11	Maximum axis acceleration in transformations.

LIMITATION OF THE FEEDRATE AT THE CORNER.

(V.)[ch].G.AXLIMC

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Logic number of the axis that limits the feedrate at the corner, in the block being executed.

Together with the V.G.PARLIMC variable, it may be used to evaluate the behavior of the machining operation at a particular corner where the feedrate decreases too much or is irregular.

Syntax.

·ch· Channel number.

V.[2].G.AXLIMC	Channel ·2·.

(V.)[ch].G.PARLIMC

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Cause that limits the feedrate at the corner in the block being executed.

Syntax.

·ch· Channel number.

V.[2].G.PARLIMC	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
1	Maximum feedrate of the axis.
4	The acceleration at the corner (parameter CORNERACC).
5	The jerk at the corner (parameter CORNERJERK).
8	The chordal error at the corner (CONTERROR).
9	Geometry at the corner (FAST mode).

MODIFY THE DYNAMICS OF ALL THE AXES OF THE CHANNEL.

(V.)[ch].G.DYNOVR

Variable that can be read and written via program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Percentage applicable to all the dynamics variables of all the axes of the channel.

This variable may be used to make changes in real time to the machining dynamics (ignoring look-ahead blocks). The dynamics can also be changed from the interface using the dynamic-override bar of the HSC.



The upper and lower limits (range) for varying the value depends on machine parameters MINDYNOVR and MAXDYNOVR.

Syntax.

·ch· Channel number.

V.[2].G.DYNOVR	Channel ·2·.

10.46 Variables related to spindle speed.

PROGRAMMING THE SPEED.

(V.)[ch].A.SREAL.sn
(V.)[ch].SP.SREAL.sn

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Real spindle speed.

Syntax.

·ch· Channel number.

·sn· Name, logic number or index of the spindle.

V.A.SREAL.S	Spindle S.
V.SP.SREAL.S	Spindle S.
V.SP.SREAL	Master spindle.
V.A.SREAL.5	Spindle with logic number ·5·.
V.SP.SREAL.2	Spindle with index ·2· in the system.
V.[2].SP.SREAL.1	Spindle with index ·1· in the channel ·2·.

Remarks.

This variable takes into account the accelerations and decelerations of the machine. It returns a value of ·0· when the spindle is stopped, it returns the value in rpm when the spindle is in G96/G97 and in degrees/minute when the spindle is in M19.

SPINDLE SPEED IN G97.

(V.)[ch].A.SPEED.sn
(V.)[ch].SP.SPEED.sn

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Active spindle speed in G97.

Syntax.

·ch· Channel number.

·sn· Name, logic number or index of the spindle.

V.A.SPEED.S	Spindle S.
V.SP.SPEED.S	Spindle S.
V.SP.SPEED	Master spindle.
V.A.SPEED.5	Spindle with logic number ·5·.
V.SP.SPEED.2	Spindle with index ·2· in the system.
V.[2].SP.SPEED.1	Spindle with index ·1· in the channel ·2·.

Remarks.

The speed may be set by program or by PLC; the one set by PLC has the highest priority.

Speed set.	V.PLC.S.sn	V.A.PRGS.sn	V.A.SPEED.sn
By program; S5000. By PLC; none.	0	5000	5000
By program; S5000. By PLC; S9000.	9000	5000	9000
By program; S5000. By PLC; S3000.	3000	5000	3000
By program; S5000. By PLC; none. By MDI; S8000.	0	8000	8000
By program; S5000. By PLC; S9000. By MDI; S8000.	9000	8000	9000
By program; S5000. By PLC; S3000. By MDI; S8000.	3000	8000	3000

(V.)[ch].PLC.S.sn

Variable that can be read and written from the PLC and read from the program and from the interface. The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Speed active by PLC for G97.

The speed programmed by PLC prevails over the one programmed by program or MDI. To cancel the speed by PLC, set the variable to ·0·; the CNC applies the speed active by program.

Syntax.

·ch· Channel number.

·sn· Name, logic number or index of the spindle.

V.PLC.S.S2	Spindle S2.
V.PLC.S.5	Spindle with logic number ·5·.
V.[2].PLC.S.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.PRGS.sn
(V.)[ch].SP.PRGS.sn

Variable that can only be read from the program, PLC and interface. The variable returns the value of block preparation.

Channel [ch]. Speed active by program for G97.

Being G97 active, programming a new speed in MDI mode updates the value of this variable.

Syntax.

·ch· Channel number.

·sn· Name, logic number or index of the spindle.

V.A.PRGS.S	Spindle S.
V.SP.PRGS.S	Spindle S.
V.SP.PRGS	Master spindle.
V.A.PRGS.5	Spindle with logic number ·5·.
V.SP.PRGS.2	Spindle with index ·2· in the system.
V.[2].SP.PRGS.1	Spindle with index ·1· in the channel ·2·.

SPINDLE SPEED IN G96 (CSS).

(V.)[ch].A.CSS.sn
(V.)[ch].SP.CSS.sn

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Active spindle speed in G96.

Syntax.

·ch· Channel number.

·sn· Name, logic number or index of the spindle.

V.A.CSS.S	Spindle S.
V.SP.CSS.S	Spindle S.
V.SP.CSS	Master spindle.
V.A.CSS.5	Spindle with logic number ·5·.
V.SP.CSS.2	Spindle with index ·2· in the system.
V.[2].SP.CSS.1	Spindle with index ·1· in the channel ·2·.

Remarks.

The speed may be set by program or by PLC; the one set by PLC has the highest priority.

Speed set.	V.PLC.CSS.sn	V.A.PRGCSS.sn	V.A.CSS.sn
By program; S150. By PLC; none.	0	150	150
By program; S150. By PLC; S250.	250	150	250
By program; S150. By PLC; S100.	100	150	100
By program; S150. By PLC; none. By MDI; S300.	0	300	300
By program; S150. By PLC; S250. By MDI; S200.	250	200	250
By program; S150. By PLC; S100. By MDI; S200.	100	200	100

(V.)[ch].PLC.CSS.sn

Variable that can be read and written from the PLC and read from the program and from the interface.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Speed active by PLC for G96.

The speed programmed by PLC prevails over the one programmed by program or MDI. To cancel the speed by PLC, set the variable to ·0·; the CNC applies the speed active by program.

Syntax.

·ch· Channel number.

·sn· Name, logic number or index of the spindle.

V.PLC.CSS.S2	Spindle S2.
V.PLC.CSS.5	Spindle with logic number ·5·.
V.[2].PLC.CSS.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.PRGCSS.sn
(V.)[ch].SP.PRGCSS.sn

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Speed active by program for G96.

Being G96 active, programming a new speed in MDI mode updates the value of this variable.

Syntax.

·ch· Channel number.

·sn· Name, logic number or index of the spindle.

V.A.PRGCSS.S	Spindle S.
V.SP.PRGCSS.S	Spindle S.
V.SP.PRGCSS	Master spindle.
V.A.PRGCSS.5	Spindle with logic number ·5·.
V.SP.PRGCSS.2	Spindle with index ·2· in the system.
V.[2].SP.PRGCSS.1	Spindle with index ·1· in the channel ·2·.

SPINDLE SPEED IN M19.

(V.)[ch].A.SPOS.sn
(V.)[ch].SP.SPOS.sn

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Active spindle speed in M19 .

Syntax.

·ch· Channel number.

·sn· Name, logic number or index of the spindle.

V.A.SPOS.S	Spindle S.
V.SP.SPOS.S	Spindle S.
V.SP.SPOS	Master spindle.
V.A.SPOS.5	Spindle with logic number ·5·.
V.SP.SPOS.2	Spindle with index ·2· in the system.
V.[2].SP.SPOS.1	Spindle with index ·1· in the channel ·2·.

Remarks.

The speed may be set by program or by PLC; the one set by PLC has the highest priority.

Speed set.	V.PLC.SPOS.sn	V.A.PRGSPOS.sn	V.A.SPOS.sn
By program; S.POS=180. By PLC; none.	0	180	180
By program; S.POS=180. By PLC; S.POS=250.	250	180	250
By program; S.POS=180. By PLC; S.POS=90.	90	180	90
By program; S.POS=180. By PLC; none. By MDI; S.POS=200.	0	200	200
By program; S.POS=180. By PLC; S.POS=250. By MDI; S.POS=200.	250	200	250
By program; S.POS=180. By PLC; S.POS=100. By MDI; S.POS=200.	100	200	100

(V.)[ch].PLC.SPOS.sn

Variable that can be read and written from the PLC and read from the program and from the interface.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Speed active by PLC for M19.

The speed programmed by PLC prevails over the one programmed by program or MDI. To cancel the speed by PLC, set the variable to ·0·; the CNC applies the speed active by program.

Syntax.

·ch· Channel number.

·sn· Name, logic number or index of the spindle.

V.PLC.SPOS.S2	Spindle S2.
V.PLC.SPOS.5	Spindle with logic number ·5·.
V.[2].PLC.SPOS.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.PRGSPOS.sn
(V.)[ch].SP.PRGSPOS.sn

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Speed active by program for M19.

Syntax.

·ch· Channel number.

·sn· Name, logic number or index of the spindle.

V.A.PRGSPOS.S	Spindle S.
V.SP.PRGSPOS.S	Spindle S.
V.SP.PRGSPOS	Master spindle.
V.A.PRGSPOS.5	Spindle with logic number ·5·.
V.SP.PRGSPOS.2	Spindle with index ·2· in the system.
V.[2].SP.PRGSPOS.1	Spindle with index ·1· in the channel ·2·.

SPEED LIMIT.

(V.)[ch].A.SLIMIT.sn
(V.)[ch].SP.SLIMIT.sn

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Active spindle speed limit.

Syntax.

·ch· Channel number.

·sn· Name, logic number or index of the spindle.

V.A.SLIMIT.S	Spindle S.
V.SP.SLIMIT.S	Spindle S.
V.SP.SLIMIT	Master spindle.
V.A.SLIMIT.5	Spindle with logic number ·5·.
V.SP.SLIMIT.2	Spindle with index ·2· in the system.
V.[2].SP.SLIMIT.1	Spindle with index ·1· in the channel ·2·.

Remarks.

The maximum speed may be set by program or by PLC; the one set by PLC has the highest priority.

(V.)[ch].PLC.SL.sn

Variable that can be read and written from the PLC and read from the program and from the interface.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Spindle speed limit active by PLC.

The speed programmed by PLC prevails over the one programmed by program or MDI. To cancel the speed by PLC, set the variable to ·0·; the CNC applies the speed active by program.

Syntax.

·ch· Channel number.

·sn· Name, logic number or index of the spindle.

V.PLC.SL.S2	Spindle S2.
V.PLC.SL.5	Spindle with logic number ·5·.
V.[2].PLC.SL.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.PRGSL.sn
(V.)[ch].SP.PRGSL.sn

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Spindle speed limit active by program.

Syntax.

·ch· Channel number.

·sn· Name, logic number or index of the spindle.

V.A.PRGSL.S	Spindle S.
V.SP.PRGSL.S	Spindle S.
V.SP.PRGSL	Master spindle.
V.A.PRGSL.5	Spindle with logic number ·5·.
V.SP.PRGSL.2	Spindle with index ·2· in the system.
V.[2].SP.PRGSL.1	Spindle with index ·1· in the channel ·2·.

PERCENTAGE OF SPINDLE SPEED (SPINDLE SPEED OVERRIDE).

(V.)[ch].A.SSO.sn
(V.)[ch].SP.SSO.sn

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Active spindle speed override.

Syntax.

·ch· Channel number.

·sn· Name, logic number or index of the spindle.

V.A.SSO.S	Spindle S.
V.SP.SSO.S	Spindle S.
V.SP.SSO	Master spindle.
V.A.SSO.5	Spindle with logic number ·5·.
V.SP.SSO.2	Spindle with index ·2· in the system.
V.[2].SP.SSO.1	Spindle with index ·1· in the channel ·2·.

Remarks.

The spindle speed override percentage may be set by program, by PLC or with the selector switch; the one set by program has the highest priority and the one selected with the switch the lowest.

V.A.PRGSSO.sn V.SP.PRGSSO.sn	V.PLC.SSO.sn	V.A.CNCSSO.sn V.SP.CNCSSO.sn	V.A.SSO.sn V.SP.SSO.sn
0	0	100 %	100 %
0	80 %	100 %	80 %
110 %	80%	100 %	110 %
70 %	80 %	100 %	70 %
70 %	0	100 %	70 %

(V.)[ch].A.PRGSSO.sn
(V.)[ch].SP.PRGSSO.sn

Variable that can be read and written from the program and read from the PLC and from the interface.
The variable returns the value of block preparation.

Channel [ch]. Percentage of spindle speed by program.

The percentage set by program has higher priority than the one set by PLC or by the switch. To cancel the value set by program, set the variable to ·0·.

Syntax.

·ch· Channel number.

·sn· Name, logic number or index of the spindle.

V.A.PRGSSO.S	Spindle S.
V.SP.PRGSSO.S	Spindle S.
V.SP.PRGSSO	Master spindle.
V.A.PRGSSO.5	Spindle with logic number ·5·.
V.SP.PRGSSO.2	Spindle with index ·2· in the system.
V.[2].SP.PRGSSO.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].PLC.SSO.sn

Variable that can be read and written from the PLC and read from the program and from the interface.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Percentage of spindle speed by PLC.

The percentage set by PLC has higher priority than the one set by the switch, but lower priority than the one set by program.. To cancel the value set by PLC, set the variable to ·0·.

Syntax.

·ch· Channel number.

·sn· Name, logic number or index of the spindle.

V.PLC.SSO.S2	Spindle S2.
V.PLC.SSO.5	Spindle with logic number ·5·.
V.[2].PLC.SSO.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.CNCSSO.sn
(V.)[ch].SP.CNCSSO.sn

Variable that can be read and written (asynchronous writing) from the interface and read from the program and from the PLC.
The variable returns the execution value; reading it interrupts block preparation.

Channel [ch]. Spindle speed percentage at the switch of the operator panel.

The percentage set by the switch of the operator panel has lower priority than the one set by PLC or by program.

Syntax.

·ch· Channel number.

·sn· Name, logic number or index of the spindle.

V.A.CNCSSO.S	Spindle S.
V.SP.CNCSSO.S	Spindle S.
V.SP.CNCSSO	Master spindle.
V.A.CNCSSO.5	Spindle with logic number ·5·.
V.SP.CNCSSO.2	Spindle with index ·2· in the system.
V.[2].SP.CNCSSO.1	Spindle with index ·1· in the channel ·2·.

10.47 Variables associated with the tool manager.

(V.)[ch].TM.MZSTATUS

Variable that can be read from the PLC and from the interface.

Tool manager status.

Syntax.

·ch· Channel number.

[2].TM.MZSTATUS	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Normal.
1	Tool manager error.
2	Error at the tool manager; waiting to complete the maneuver in progress.
4	Tool manager emergency.

(V.)[ch].TM.MZRUN

Variable that can be read from the PLC and from the interface.

Tool manager running.

Syntax.

·ch· Channel number.

[2].TM.MZRUN	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	There is no maneuver in progress.
1	There is a maneuver in progress.

(V.)[ch].TM.MZWAIT

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Tool manager processing a maneuver.

Syntax.

·ch· Channel number.

V.[2].TM.MZWAIT	Channel ·2·.

Remarks.

From software version V2.01 on, there is no need to use this variable in the subroutine associated with M06. The subroutine itself waits for the manager's maneuvers to finish and it does not interrupt block preparation.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No need to wait.
1	It has to wait.

(V.)[ch].TM.MZMODE

Variable that can be read and written from the program and interface and read from the PLC. The variable returns the execution value; reading and writing it interrupt block preparation.

Operating mode of the tool manager.

This variable may be used to set the magazine in load or unload mode.

Syntax.

·ch· Channel number.

V.[2].TM.MZMODE	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Regular work mode.
1	Magazine in load mode.
2	Magazine in unload mode.

10.48 Variables related to managing the tool magazine and the tool changer arm.

RELATIONSHIP BETWEEN THE TOOL MAGAZINE AND THE CHANNEL.

(V.)[ch].TM.ACTUALMZ

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Tool Magazine being used by the [ch] channel.

Syntax.

·ch· Channel number.

V.[2].TM.ACTUALMZ	Channel ·2·.

(V.)TM.MZACTUALCH[mz]

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Channel that is using tool magazine [mz].

Syntax.

·mz· Type of tool magazine.

V.TM.MZACTUALCH[2]	Magazine ·2·.

TOOL LOCATION IN THE MAGAZINE.

(V.)TM.T[mz][pos]

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Tool in the [pos] position of the [mz] magazine.

Syntax.

·mz· Type of tool magazine.

·pos· Tool position in the magazine.

V.TM.T[2][15]	Magazine ·2·. Position ·15·.

Remarks.

If the variable is missing the offset number, the variable will then refer to the first one. This variable returns 0 if the tool is not in the magazine.

(V.)TM.P[mz][tl]

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Position of the [t] tool in the [mz] magazine.

Syntax.

·mz· Type of tool magazine.

·tl· Tool number.

V.TM.P[2][15]	Magazine ·2·. Position ·15·.

Remarks.

If the variable is missing the offset number, the variable will then refer to the first one.

TOOL LOCATION IN THE TOOL CHANGER ARM.

(V.)TM.TOOLCH1[mz]

Variable that can be read and written via program, PLC(asynchronous writing) and interface.
The variable returns the execution value; reading it interrupts block preparation.

Tool in the first claw of the magazine arm [mz].

Syntax.

·mz· Type of tool magazine.

V.TM.TOOLCH1[2]	Magazine ·2·.

(V.)TM.TOOLCH2[mz]

Variable that can be read and written via program, PLC(asynchronous writing) and interface.
The variable returns the execution value; reading it interrupts block preparation.

Tool in the second claw of the magazine arm [mz].

Syntax.

·mz· Type of tool magazine.

V.TM.TOOLCH2[2]	Magazine ·2·.

10.49 Variables related to the active tool and to the next one.

Variables related to the active tool are always read synchronously. The writing of these variables is always asynchronous, be it for the active tool or not.

TOOL AND ACTIVE OFFSET.

(V.)[ch].TM.TOOL

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Number of the active tool.

Syntax.

·ch· Channel number.

V.[2].TM.TOOL	Channel ·2·. Active tool.

(V.)[ch].TM.TOD

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Number of the active tool offset.

Syntax.

·ch· Channel number.

V.[2].TM.TOD	Channel ·2·. Active tool offset.

NEXT TOOL AND TOOL OFFSET.

(V.)[ch].TM.NXTOOL

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Number of the next tool.

The "next tool" is the one already selected but waiting to be activated by executing an M06.

Syntax.

·ch· Channel number.

V.[2].TM.NXTOOL	Channel ·2·. Next tool.

Remarks.

Although the variable has writing permission, the value of 0 (zero) cannot be written.

(V.)[ch].TM.NXTOD

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Number of the next tool offset.

Syntax.

·ch· Channel number.

V.[2].TM.NXTOD	Channel ·2·. Next tool offset.

STATUS OF THE ACTIVE TOOL.

(V.)[ch].TM.TSTATUS

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Active tool. Tool status.

Syntax.

·ch· Channel number.

V.[2].TM.TSTATUS	Channel ·2·. Active tool.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Tool available.
1	Tool available.
2	Tool worn out.

FAMILY OF THE ACTIVE TOOL.

(V.)[ch].TM.TLFF

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Active tool. Family code.

Syntax.

·ch· Channel number.

V.[2].TM.TLFF	Channel ·2·.

ACTIVE TOOL MONITORING.

(V.)[ch].TM.TOMON[ofd]

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the active tool. Type of tool life monitoring.

Syntax.

[odf] Tool offset; if missing, the active tool offset.

·ch· Channel number.

V.[2].TM.TOMON	Channel ·2·. Active tool offset.
V.[2].TM.TOMON[3]	Channel ·2·. Tool offset ·3·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Without tool life monitoring.
1	Tool life monitoring; machining time.
2	Tool life monitoring; number of operations.

(V.)[ch].TM.TLFN[ofd]

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the active tool. Nominal life.

Syntax.

[odf] Tool offset; if missing, the active tool offset.

·ch· Channel number.

V.[2].TM.TLFN	Channel ·2·. Active tool offset.
V.[2].TM.TLFN[3]	Channel ·2·. Tool offset ·3·.

(V.)[ch].TM.TLFR[ofd]

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the active tool. Real (actual) life.

Syntax.

[odf] Tool offset; if missing, the active tool offset.

·ch· Channel number.

V.[2].TM.TLFR	Channel ·2·. Active tool offset.
V.[2].TM.TLFR[3]	Channel ·2·. Tool offset ·3·.

(V.)[ch].TM.REMLIFE

Variable that can be read from the PLC and from the interface.

Active tool. Remaining life.

Syntax.

·ch· Channel number.

[2].TM.REMLIFE	Channel ·2·.

"CUSTOM" DATA OF THE ACTIVE TOOL.

(V.)[ch].TM.TOTP1
(V.)[ch].TM.TOTP2
(V.)[ch].TM.TOTP3
(V.)[ch].TM.TOTP4

Variable that can be read and written via program, PLC(asynchronous writing) and interface.
The variable returns the execution value; reading it interrupts block preparation.

Active tool. Custom parameters.

Syntax.

·ch· Channel number.

V.[2].TM.TOTP1	Channel ·2·. Custom parameter ·1·.
V.[2].TM.TOTP2	Channel ·2·. Custom parameter ·2·.
V.[2].TM.TOTP3	Channel ·2·. Custom parameter ·3·.
V.[2].TM.TOTP4	Channel ·2·. Custom parameter ·4·.

TOOL GEOMETRY.

(V.)[ch].TM.NUMOFD

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Active tool. Number of tool offsets.

Syntax.

·ch· Channel number.

V.[2].TM.NUMOFD	Channel ·2·.

(V.)[ch].TM.DTYPE[ofd]

Variable that can be read and written via program, PLC(asynchronous writing) and interface.
The variable returns the execution value; reading it interrupts block preparation.

Active tool. Code of the tool offset type.

Syntax.

·ch· Channel number.

[odf] Tool offset; if missing, the active tool offset.

V.[2].TM.DTYPE	Channel ·2·. Active tool offset.
V.[2].TM.DTYPE[3]	Channel ·2·. Tool offset ·3·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
1	Milling tool..
2	Drilling tool.
3	Surface milling tool.
4	Reaming tool.
5	Boring tool.
6	Threading tool.
7	Grooving or cut-off tool.
8	Turning tool.
9	Other.
10	Measuring probe.

(V.)[ch].TM.DSUBTYPE[ofd]

Variable that can be read and written via program, PLC(asynchronous writing) and interface.
The variable returns the execution value; reading it interrupts block preparation.

Active tool. Code of the tool offset subtype.

Syntax.

·ch· Channel number.

[odf] Tool offset; if missing, the active tool offset.

V.[2].TM.DSUBTYPE	Channel ·2·. Active tool offset.
V.[2].TM.DSUBTYPE[3]	Channel ·2·. Tool offset ·3·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.	Value.	Meaning.
0	Without type.	8	Surface milling endmill. 
1	Flat endmill. 	9	Reamer. 
2	Ball endmill. 	10	Boring tool. 
3	Toric endmill. 	11	Diamond shaped cutter for turning. 
4	Tap. 	12	Square cutter for turning, grooving or cut-off. 
5	Thread cutter. 	13	Round cutter for turning. 
6	Disk endmill. 	14	Measuring probe (milling machine). 
7	Drill bit. 	15	Measuring probe (lathe). 

(V.)[ch].TM.TURNCONFIG[ofd]

Variable that can be read and written via program, PLC(asynchronous writing) and interface.
The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the active tool. Axis orientation.

Only for turning tools. The orientation of the axes is determined by the type of lathe (horizontal or vertical), the position of the turret and the spindle position (on the right or on the left).

Syntax.

·ch· Channel number.

[odf] Tool offset; if missing, the active tool offset.

V.[2].TM.TURNCONFIG	Channel ·2·. Active tool offset.
V.[2].TM.TURNCONFIG[3]	Channel ·2·. Tool offset ·3·.

Values of the variable.

This variable returns one of the following values.

Value.	Axis orientation.	Value.	Axis orientation.
0		4	
1		5	
2		6	
3		7	

(V.)[ch].TM.LOCODE[ofd]

Variable that can be read and written via program, PLC(asynchronous writing) and interface.
The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the active tool. Location code (shape) or tool calibration point.

Only for turning tools. The location code indicates which is the calibrated tool tip and, therefore, the point controlled by the CNC to apply radius compensation. The location code depends on the orientation of the machine axes.

Syntax.

·ch· Channel number.

[odf] Tool offset; if missing, the active tool offset.

V.[2].TM.LOCODE	Channel ·2·. Active tool offset.
V.[2].TM.LOCODE[3]	Channel ·2·. Tool offset ·3·.

Values of the variable.

This variable returns one of the following values.

Value.								
0								
1								
2								
3								
4								
5								
6								

Value.								
7								
8								
9								

(V.)[ch].TM.FIXORI[ofd]

Variable that can be read and written via program, PLC(asynchronous writing) and interface.
The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the active tool. Tool-holder orientation.

Syntax.

·ch· Channel number.

[odf] Tool offset; if missing, the active tool offset.

V.[2].TM.FIXORI	Channel ·2·. Active tool offset.
V.[2].TM.FIXORI[3]	Channel ·2·. Tool offset ·3·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Face axis.
1	Longitudinal axis.

(V.)[ch].TM.SPDLTURDIR[ofd]

Variable that can be read and written via program, PLC(asynchronous writing) and interface.
The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the active tool. Spindle turning direction.

Syntax.

·ch· Channel number.

[odf] Tool offset; if missing, the active tool offset.

V.[2].TM.SPDLTURDIR	Channel ·2·. Active tool offset.
V.[2].TM.SPDLTURDIR[3]	Channel ·2·. Tool offset ·3·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Undefined direction.
1	Right-hand machining direction.
2	Left-hand machining direction.

(V.)[ch].TM.TOR[ofd]

Variable that can be read and written via program, PLC(asynchronous writing) and interface.
The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the active tool. Radius.

This variable is not valid for turning tools.

Syntax.

·ch· Channel number.

[odf] Tool offset; if missing, the active tool offset.

V.[2].TM.TOR	Channel ·2·. Active tool offset.
V.[2].TM.TOR[3]	Channel ·2·. Tool offset ·3·.

(V.)[ch].TM.TOI[ofd]

Variable that can be read and written via program, PLC(asynchronous writing) and interface. The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the active tool. Radius wear.

This variable is not valid for turning tools.

Syntax.

·ch· Channel number.

[odf] Tool offset; if missing, the active tool offset.

V.[2].TM.TOI	Channel ·2·. Active tool offset.
V.[2].TM.TOI[3]	Channel ·2·. Tool offset ·3·.

(V.)[ch].TM.TOL[ofd]

Variable that can be read and written via program, PLC(asynchronous writing) and interface. The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the active tool. Length.

This variable is not valid for turning tools.

Syntax.

·ch· Channel number.

[odf] Tool offset; if missing, the active tool offset.

V.[2].TM.TOL	Channel ·2·. Active tool offset.
V.[2].TM.TOL[3]	Channel ·2·. Tool offset ·3·.

(V.)[ch].TM.TOK[ofd]

Variable that can be read and written via program, PLC(asynchronous writing) and interface. The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the active tool. Length wear.

This variable is not valid for turning tools.

Syntax.

·ch· Channel number.

[odf] Tool offset; if missing, the active tool offset.

V.[2].TM.TOK	Channel ·2·. Active tool offset.
V.[2].TM.TOK[3]	Channel ·2·. Tool offset ·3·.

(V.)[ch].TM.TOAN[ofd]

Variable that can be read and written via program, PLC(asynchronous writing) and interface. The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the active tool. Penetration angle.

Syntax.

·ch· Channel number.

[odf] Tool offset; if missing, the active tool offset.

V.[2].TM.TOAN	Channel ·2·. Active tool offset.
V.[2].TM.TOAN[3]	Channel ·2·. Tool offset ·3·.

(V.)[ch].TM.TOTIPR[ofd]

Variable that can be read and written via program, PLC(asynchronous writing) and interface. The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the active tool. Tip radius.

Syntax.

·ch· Channel number.

[odf] Tool offset; if missing, the active tool offset.

V.[2].TM.TOTIPR	Channel ·2·. Active tool offset.
V.[2].TM.TOTIPR[3]	Channel ·2·. Tool offset ·3·.

(V.)[ch].TM.TOWTIPR[ofd]

Variable that can be read and written via program, PLC(asynchronous writing) and interface. The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the active tool. Tool tip radius wear.

Syntax.

·ch· Channel number.

[odf] Tool offset; if missing, the active tool offset.

V.[2].TM.TOWTIPR	Channel ·2·. Active tool offset.
V.[2].TM.TOWTIPR[3]	Channel ·2·. Tool offset ·3·.

(V.)[ch].TM.TOCUTL[ofd]

Variable that can be read and written via program, PLC(asynchronous writing) and interface. The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the active tool. Cutting length.

Syntax.

·ch· Channel number.

[odf] Tool offset; if missing, the active tool offset.

V.[2].TM.TOCUTL	Channel ·2·. Active tool offset.
V.[2].TM.TOCUTL[3]	Channel ·2·. Tool offset ·3·.

(V.)[ch].TM.NOSEA[ofd]

Variable that can be read and written via program, PLC(asynchronous writing) and interface. The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the active tool. Cutter angle.

Syntax.

·ch· Channel number.

[odf] Tool offset; if missing, the active tool offset.

V.[2].TM.NOSEA	Channel ·2·. Active tool offset.
V.[2].TM.NOSEA[3]	Channel ·2·. Tool offset ·3·.

(V.)[ch].TM.NOSEW[ofd]

Variable that can be read and written via program, PLC(asynchronous writing) and interface.
The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the active tool. Cutter width.

Syntax.

·ch· Channel number.

[odf] Tool offset; if missing, the active tool offset.

V.[2].TM.NOSEW	Channel ·2·. Active tool offset.
V.[2].TM.NOSEW[3]	Channel ·2·. Tool offset ·3·.

(V.)[ch].TM.CUTA[ofd]

Variable that can be read and written via program, PLC(asynchronous writing) and interface.
The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the active tool. Cutting angle of the cutter.

Syntax.

·ch· Channel number.

[odf] Tool offset; if missing, the active tool offset.

V.[2].TM.CUTA	Channel ·2·. Active tool offset.
V.[2].TM.CUTA[3]	Channel ·2·. Tool offset ·3·.

(V.)[ch].TM.TOFL[ofd].xn

Variable that can be read and written via program, PLC(asynchronous writing) and interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the active tool. Tool length offset on the xn axis.

The offsets are used to define the tool dimensions in each axis. The dimensions of the turning tools are defined using these offsets; either these offsets or tool length and radius may be used for the dimensions of the rest of the tools.

On tools that are not just for turning, e.g. endmills and drill bits, the offsets may also be used to define the tool position when using a tool holder or an intermediate tool. In this case, the tool dimensions are defined with the radius and the length.

Syntax.

·ch· Channel number.

·ofd· Tool offset.

·xn· Name, logic number or index of the axis.

V.TM.TOFL[3].Z	Tool offset ·3·. Z axis.
V.TM.TOFL[3].4	Tool offset ·3·. Axis with logic number ·4·.
V.[2].TM.TOFL[3].1	Tool offset ·3·. Axis with index ·1· in the channel ·2·.

(V.)[ch].TM.TOFLW[ofd].xn

Variable that can be read and written via program, PLC(asynchronous writing) and interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the active tool. Tool length offset wear on the xn axis.

Syntax.

·ch· Channel number.

·ofd· Tool offset.

·xn· Name, logic number or index of the axis.

V.TM.TOFLW[3].Z	Tool offset ·3·. Z axis.
V.TM.TOFLW[3].4	Tool offset ·3·. Axis with logic number ·4·.
V.[2].TM.TOFLW[3].1	Tool offset ·3·. Axis with index ·1· in the channel ·2·.

Remarks.

The value of these variables depends on the active units (radius or diameter). In order for these variables to return the value in diameter, these units must be enabled by machine parameter and function G151 must be active.

(V.)[ch].TM.TOFL1
(V.)[ch].TM.TOFL2
(V.)[ch].TM.TOFL3

Variable that can be read and written via program, PLC(asynchronous writing) and interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.

Active offset of the active tool. Tool length offset on the first axes of the channel.

On turning tools, it is used to define the tool length in each axis. On milling tools, it is used to define the tool position when using a tool holder or an intermediate tooling.

Syntax.

·ch· Channel number.

V.[2].TM.TOFL1	Channel ·2·. First axis of the channel.
V.[2].TM.TOFL2	Channel ·2·. Second axis of the channel.
V.[2].TM.TOFL3	Channel ·2·. Third axis of the channel.

(V.)[ch].TM.TOFLW1
(V.)[ch].TM.TOFLW2
(V.)[ch].TM.TOFLW3

Variable that can be read and written via program, PLC(asynchronous writing) and interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.

Active offset of the active tool. Tool length offset wear on the first axes of the channel.

Syntax.

·ch· Channel number.

V.[2].TM.TOFLW1	Channel ·2·. First axis of the channel.
V.[2].TM.TOFLW2	Channel ·2·. Second axis of the channel.
V.[2].TM.TOFLW3	Channel ·2·. Third axis of the channel.

Remarks.

The value of these variables depends on the active units (radius or diameter). In order for these variables to return the value in diameter, these units must be enabled by machine parameter and function G151 must be active.

CANCEL THE PRESET TURNING DIRECTION OF THE TOOL.

(V.)G.SPDLTURDIR

Variable that can be read and written from the program and read from the PLC (asynchronous) and from the interface.
The variable returns the execution value; reading it interrupts block preparation.

Modify the preset turning direction for the active tool.

The preset turning speed of the active tool may be temporarily canceled from the part-program. This is done by setting this variable to ·0·. The variable does not modify the tool table. When changing a tool, this variable will take the corresponding value according to what has been set in the tool table.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Without preset turning direction.
1	Turning direction M03.
2	Turning direction M04.

10.50 Variables associated with any tool.

The variables referred to a tool other than the active one are for synchronous reading if the tool is in the magazine and for asynchronous reading if otherwise. The writing of these variables is always asynchronous, be it for the active tool or not.

TOOL STATUS.

(V.)TM.TSTATUST[tl]

Variable that can be read from the PLC and from the interface.

Tool [tl]. Tool status.

Syntax.

·tl· Tool number.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Tool available.
1	Tool available.
2	Tool worn out.

FAMILY OF THE TOOL.

(V.)TM.TLFFT[tl]

Variable that can be read and written via program, PLC(asynchronous writing) and interface.
The variable returns the execution value; reading it interrupts block preparation.

Tool [tl]. Family code.

Syntax.

·tl· Tool number.

V.TM.TLFFT[23]	Tool ·23·.

TOOL MONITORING.

(V.)TM.TOMONT[tl][ofd]

Variable that can be read and written via program, PLC(asynchronous writing) and interface.
The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the [tl] tool. Type of tool life monitoring.

Syntax.

·tl· Tool number.

·ofd· Tool offset.

V.TM.TOMONT[23][3]	Tool ·23·. Tool offset ·3·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Without tool life monitoring.
1	Tool life monitoring; machining time.
2	Tool life monitoring; number of operations.

(V.)TM.TLFNT[tl][ofd]

Variable that can be read and written via program, PLC(asynchronous writing) and interface.
The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the [tl] tool. Nominal life.

Syntax.

·tl· Tool number.

·ofd· Tool offset.

V.TM.TLFNT[23][3]	Tool ·23·. Tool offset ·3·.

(V.)TM.TLFRT[tl][ofd]

Variable that can be read and written via program, PLC(asynchronous writing) and interface.
The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the [tl] tool. Real (actual) life.

Syntax.

·tl· Tool number.

·ofd· Tool offset.

V.TM.TLFRT[23][3]	Tool ·23·. Tool offset ·3·.

"CUSTOM" DATA OF THE TOOL.

(V.)TM.TOTP1T[tl]
(V.)TM.TOTP2T[tl]
(V.)TM.TOTP3T[tl]
(V.)TM.TOTP4T[tl]

Variable that can be read and written via program, PLC(asynchronous writing) and interface.
The variable returns the execution value; reading it interrupts block preparation.

Tool [tl]. Custom parameters.

Syntax.

·tl· Tool number.

V.TM.TOTP1T[23]	Tool ·23·. Custom parameter ·1·.
V.TM.TOTP2T[23]	Tool ·23·. Custom parameter ·2·.
V.TM.TOTP3T[23]	Tool ·23·. Custom parameter ·3·.
V.TM.TOTP4T[23]	Tool ·23·. Custom parameter ·4·.

TOOL GEOMETRY.

(V.)TM.NUMOFDT[tl]

Variable that can be read and written via program, PLC(asynchronous writing) and interface.
The variable returns the execution value; reading it interrupts block preparation.

Tool [tl]. Number of tool offsets.

Syntax.

·tl· Tool number.

V.TM.NUMOFDT[23]	Tool ·23·.

(V.)TM.DTYPET[tl][ofd]

Variable that can be read and written via program, PLC(asynchronous writing) and interface. The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the [tl] tool. Code of the tool offset type.

Syntax.

·tl· Tool number.

[odf] Tool offset; if missing, the active tool offset.

V.TM.DTYPET[23]	Tool ·23·. Active tool offset.
V.TM.DTYPET[23][3]	Tool ·23·. Tool offset ·3·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
1	Milling tool..
2	Drilling tool.
3	Surface milling tool.
4	Reaming tool.
5	Boring tool.
6	Threading tool.
7	Grooving or cut-off tool.
8	Turning tool.
9	Other.
10	Measuring probe.

(V.)TM.DSUBTYPET[tl][ofd]

Variable that can be read and written via program, PLC(asynchronous writing) and interface. The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the [tl] tool. Code of the tool offset subtype.

Syntax.

·tl· Tool number.

[odf] Tool offset; if missing, the active tool offset.

V.TM.DSUBTYPET[23]	Tool ·23·. Active tool offset.
V.TM.DSUBTYPET[23][3]	Tool ·23·. Tool offset ·3·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.	Value.	Meaning.
0	Without type.	8	Surface milling endmill. 
1	Flat endmill. 	9	Reamer. 
2	Ball endmill. 	10	Boring tool. 
3	Toric endmill. 	11	Diamond shaped cutter for turning. 
4	Tap. 	12	Square cutter for turning, grooving or cut-off. 
5	Thread cutter. 	13	Round cutter for turning. 
6	Disk endmill. 	14	Measuring probe (milling machine). 
7	Drill bit. 	15	Measuring probe (lathe). 

(V.)TM.TURNCONFIGT[tl][ofd]

Variable that can be read and written via program, PLC(asynchronous writing) and interface.
The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the [tl] tool. Axis orientation.

Only for turning tools. The orientation of the axes is determined by the type of lathe (horizontal or vertical), the position of the turret and the spindle position (on the right or on the left).

Syntax.

·tl· Tool number.

[odf] Tool offset; if missing, the active tool offset.

V.TM.TURNCONFIGT[23]	Tool ·23·. Active tool offset.
V.TM.TURNCONFIGT[23][3]	Tool ·23·. Tool offset ·3·.

Values of the variable.

This variable returns one of the following values.

Value.	Axis orientation.	Value.	Axis orientation.
0		4	
1		5	
2		6	
3		7	

(V.)TM.LOCODET[tl][ofd]

Variable that can be read and written via program, PLC(asynchronous writing) and interface.
The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the [tl] tool. Location code (shape) or tool calibration point.

Only for turning tools. The location code indicates which is the calibrated tool tip and, therefore, the point controlled by the CNC to apply radius compensation. The location code depends on the orientation of the machine axes.

Syntax.

·tl· Tool number.

[odf] Tool offset; if missing, the active tool offset.

V.TM.LOCODET[23]	Tool ·23·. Active tool offset.
V.TM.LOCODET[23][3]	Tool ·23·. Tool offset ·3·.

Values of the variable.

This variable returns one of the following values.

Value.								
0								
1								
2								
3								
4								
5								
6								
7								

Value.								
8								
9								

(V.)TM.FIXORIT[tl][ofd]

Variable that can be read and written via program, PLC(asynchronous writing) and interface.
The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the [tl] tool. Tool-holder orientation.

Syntax.

·tl· Tool number.

[odf] Tool offset; if missing, the active tool offset.

V.TM.FIXORIT[23]	Tool ·23·. Active tool offset.
V.TM.FIXORIT[23][3]	Tool ·23·. Tool offset ·3·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Face axis.
1	Longitudinal axis.

(V.)TM.SPDLTURDIRT[tl][ofd]

Variable that can be read and written via program, PLC(asynchronous writing) and interface.
The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the [tl] tool. Spindle turning direction.

Syntax.

·tl· Tool number.

[odf] Tool offset; if missing, the active tool offset.

V.TM.SPDLTURDIRT[23]	Tool ·23·. Active tool offset.
V.TM.SPDLTURDIRT[23][3]	Tool ·23·. Tool offset ·3·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Undefined direction.
1	Right-hand machining direction.
2	Left-hand machining direction.

(V.)TM.TORT[tl][ofd]

Variable that can be read and written via program, PLC(asynchronous writing) and interface.
The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the [tl] tool. Radius.

This variable is not valid for turning tools.

Syntax.

·tl· Tool number.

[odf] Tool offset; if missing, the active tool offset.

V.TM.TORT[23]	Tool ·23·. Active tool offset.
V.TM.TORT[23][3]	Tool ·23·. Tool offset ·3·.

(V.)TM.TOIT[tl][ofd]

Variable that can be read and written via program, PLC(asynchronous writing) and interface.
The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the [tl] tool. Radius wear.

This variable is not valid for turning tools.

Syntax.

·tl· Tool number.

[odf] Tool offset; if missing, the active tool offset.

V.TM.TOIT[23]	Tool ·23·. Active tool offset.
V.TM.TOIT[23][3]	Tool ·23·. Tool offset ·3·.

(V.)TM.TOLT[tl][ofd]

Variable that can be read and written via program, PLC(asynchronous writing) and interface.
The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the [tl] tool. Length.

This variable is not valid for turning tools.

Syntax.

·tl· Tool number.

[odf] Tool offset; if missing, the active tool offset.

V.TM.TOLT[23]	Tool ·23·. Active tool offset.
V.TM.TOLT[23][3]	Tool ·23·. Tool offset ·3·.

(V.)TM.TOKT[tl][ofd]

Variable that can be read and written via program, PLC(asynchronous writing) and interface.
The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the [tl] tool. Length wear.

This variable is not valid for turning tools.

Syntax.

·tl· Tool number.

[odf] Tool offset; if missing, the active tool offset.

V.TM.TOKT[23]	Tool ·23·. Active tool offset.
V.TM.TOKT[23][3]	Tool ·23·. Tool offset ·3·.

(V.)TM.TOANT[tl][ofd]

Variable that can be read and written via program, PLC(asynchronous writing) and interface.
The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the [tl] tool. Penetration angle.

Syntax.

·tl· Tool number.

[odf] Tool offset; if missing, the active tool offset.

V.TM.TOANT[23]	Tool ·23·. Active tool offset.
V.TM.TOANT[23][3]	Tool ·23·. Tool offset ·3·.

(V.)TM.TOTIPRT[tl][ofd]

Variable that can be read and written via program, PLC(asynchronous writing) and interface.
The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the [tl] tool. Tip radius.

Syntax.

·tl· Tool number.

[odf] Tool offset; if missing, the active tool offset.

V.TM.TOTIPRT[23]	Tool ·23·. Active tool offset.
V.TM.TOTIPRT[23][3]	Tool ·23·. Tool offset ·3·.

(V.)TM.TOWTIPRT[tl][ofd]

Variable that can be read and written via program, PLC(asynchronous writing) and interface.
The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the [tl] tool. Tool tip radius wear.

Syntax.

·tl· Tool number.

[odf] Tool offset; if missing, the active tool offset.

V.TM.TOWTIPRT[23]	Tool ·23·. Active tool offset.
V.TM.TOWTIPRT[23][3]	Tool ·23·. Tool offset ·3·.

(V.)TM.TOCUTLT[tl][ofd]

Variable that can be read and written via program, PLC(asynchronous writing) and interface.
The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the [tl] tool. Cutting length.

Syntax.

·tl· Tool number.

[odf] Tool offset; if missing, the active tool offset.

V.TM.TOCUTLT[23]	Tool ·23·. Active tool offset.
V.TM.TOCUTLT[23][3]	Tool ·23·. Tool offset ·3·.

(V.)TM.NOSEAT[tl][ofd]

Variable that can be read and written via program, PLC(asynchronous writing) and interface.
The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the [tl] tool. Cutter angle.

Syntax.

·tl· Tool number.

[odf] Tool offset; if missing, the active tool offset.

V.TM.NOSEAT[23]	Tool ·23·. Active tool offset.
V.TM.NOSEAT[23][3]	Tool ·23·. Tool offset ·3·.

(V.)TM.NOSEWT[tl][ofd]

Variable that can be read and written via program, PLC(asynchronous writing) and interface.
The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the [tl] tool. Cutter width.

Syntax.

·tl· Tool number.

[odf] Tool offset; if missing, the active tool offset.

V.TM.NOSEWT[23]	Tool ·23·. Active tool offset.
V.TM.NOSEWT[23][3]	Tool ·23·. Tool offset ·3·.

(V.)TM.CUTAT[tl][ofd]

Variable that can be read and written via program, PLC(asynchronous writing) and interface.
The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the [tl] tool. Cutting angle of the cutter.

Syntax.

·tl· Tool number.

[odf] Tool offset; if missing, the active tool offset.

V.TM.CUTAT[23]	Tool ·23·. Active tool offset.
V.TM.CUTAT[23][3]	Tool ·23·. Tool offset ·3·.

(V.)TM.TOFLT[tl][ofd].xn

Variable that can be read and written via program, PLC(asynchronous writing) and interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the [tl] tool. Tool length offset on the xn axis.

On turning tools, it is used to define the tool length in each axis. On milling tools, it is used to define the tool position when using a tool holder or an intermediate tooling.

Syntax.

·ofd· Tool offset.

·xn· Name or logic number of the axis.

V.TM.TOFL[23].Z	Tool ·23·. Z axis.
V.TM.TOFL[23][3].4	Tool ·23·. Axis with logic number ·4·.

(V.)TM.TOFLWT[tl][ofd].xn

Variable that can be read and written via program, PLC(asynchronous writing) and interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.

[odf] offset of the [tl] tool. Tool length offset wear on the xn axis.

Syntax.

·ofd· Tool offset.

·xn· Name or logic number of the axis.

V.TM.TOFLWT[23].Z	Tool ·23·. Z axis.
V.TM.TOFLWT[23][3].4	Tool ·23·. Axis with logic number ·4·.

Remarks.

The value of these variables depends on the active units (radius or diameter). In order for these variables to return the value in diameter, these units must be enabled by machine parameter and function G151 must be active.

10.51 Variables associated with the tool being prepared.

The CNC reads several blocks ahead of the one being executed in order to calculate in advance the path to follow.

As can be seen in the following example, the block being prepared is calculated with the tool T6; whereas the tool T1 is the one currently selected.

G1 X100 F200 T1 M6	(Block in execution)
Y200	
G1 X20 F300 T6 M6	
X30 Y60	(Block being prepared)

There are specific variables for consulting and/or modifying the values being used in the preparation. These variables can only be accessed from the program and are evaluated during block preparation. Writing these variables does not modify the tool table; the new value is only used during block preparation.

TOOL AND ACTIVE OFFSET.

(V.)[ch].G.TOOL

Variable that can be read from the program.
The variable returns the value of block preparation.

Number of the tool being prepared.

Syntax.

·ch· Channel number.

V.[2].G.TOOL	Channel ·2·.

(V.)[ch].G.TOD

Variable that can be read from the program.
The variable returns the value of block preparation.

Number of tool offset being prepared.

Syntax.

·ch· Channel number.

V.[2].G.TOD	Channel ·2·.

NEXT TOOL AND TOOL OFFSET.

(V.)[ch].G.NXTOOL

Variable that can be read from the program.
The variable returns the value of block preparation.

Number of the next tool being prepared.

Syntax.

·ch· Channel number.

V.[2].G.NXTOOL	Channel ·2·.

(V.)[ch].G.NXTOD

Variable that can be read from the program.
The variable returns the value of block preparation.

Number of the next tool offset being prepared.

Syntax.

·ch· Channel number.

V.[2].G.NXTOD	Channel ·2·.

TOOL STATUS.

(V.)[ch].G.TSTATUS

Variable that can be read from the program.
The variable returns the value of block preparation.

Tool being prepared. Tool status.

Syntax.

·ch· Channel number.

V.[2].G.TSTATUS	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Tool available.
1	Tool available.
2	Tool worn out.

FAMILY OF THE TOOL.

(V.)[ch].G.TLFF

Variable that can be read from the program.
The variable returns the execution value; reading it interrupts block preparation.

Tool being prepared. Family code.

Syntax.

·ch· Channel number.

V.[2].G.TLFF	Channel ·2·.

TOOL MONITORING.

(V.)[ch].G.TOMON

Variable that can be read and written from the program.
The variable returns the value of block preparation.

Tool offset being prepared. Type of tool life monitoring.

Syntax.

·ch· Channel number.

V.[2].G.TOMON	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Without tool life monitoring.
1	Tool life monitoring; machining time.
2	Tool life monitoring; number of operations.

(V.)[ch].G.TLFN

Variable that can be read from the program.
The variable returns the value of block preparation.

Tool offset being prepared. Nominal life.

Syntax.

·ch· Channel number.

V.[2].G.TLFN	Channel ·2·.

(V.)[ch].G.TLFR

Variable that can be read from the program.
The variable returns the value of block preparation.

Tool offset being prepared. Real (actual) life.

Syntax.

·ch· Channel number.

V.[2].G.TLFR	Channel ·2·.

(V.)[ch].G.REMLIFE

Variable that can be read and written from the program.
The variable returns the value of block preparation.

Tool offset being prepared. Remaining life.

Syntax.

·ch· Channel number.

V.[2].G.REMLIFE	Channel ·2·.

"CUSTOM" DATA OF THE TOOL.

(V.)[ch].G.TOTP1
(V.)[ch].G.TOTP2
(V.)[ch].G.TOTP3
(V.)[ch].G.TOTP4

Variable that can be read and written from the program.
The variable returns the value of block preparation.

Tool being prepared. Custom parameters.

Syntax.

·ch· Channel number.

V.[2].G.TOTP1	Channel ·2·. Custom parameter ·1·.
V.[2].G.TOTP2	Channel ·2·. Custom parameter ·2·.
V.[2].G.TOTP3	Channel ·2·. Custom parameter ·3·.
V.[2].G.TOTP4	Channel ·2·. Custom parameter ·4·.

TOOL GEOMETRY.

(V.)[ch].G.DSUBTYPE

Variable that can be read from the program.
The variable returns the value of block preparation.

Tool being prepared. Code of the tool offset subtype.

Syntax.

·ch· Channel number.

V.[2].G.DSUBTYPE	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.	Value.	Meaning.
0	Without type.	8	Surface milling endmill. 
1	Flat endmill. 	9	Reamer. 
2	Ball endmill. 	10	Boring tool. 
3	Toric endmill. 	11	Diamond shaped cutter for turning. 
4	Tap. 	12	Square cutter for turning, grooving or cut-off. 
5	Thread cutter. 	13	Round cutter for turning. 
6	Disk endmill. 	14	Measuring probe (milling machine). 
7	Drill bit. 	15	Measuring probe (lathe). 

(V.)[ch].G.LOCODE

Variable that can be read from the program.
The variable returns the value of block preparation.

[odf] offset of the tool being prepared. Location code (shape) or tool calibration point.

Only for turning tools. The location code indicates which is the calibrated tool tip and, therefore, the point controlled by the CNC to apply radius compensation. The location code depends on the orientation of the machine axes.

Syntax.

·ch· Channel number.

V.[2].G.LOCODE	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.								
0								
1								
2								
3								
4								
5								
6								
7								
8								
9								

(V.)[ch].G.TOR

Variable that can be read and written from the program.
The variable returns the value of block preparation.

Tool being prepared. Radius.

This variable is not valid for turning tools.

Syntax.

·ch· Channel number.

V.[2].G.TOR	Channel ·2·.

(V.)[ch].G.TOI

Variable that can be read and written from the program.
The variable returns the value of block preparation.

Tool being prepared. Radius wear.

This variable is not valid for turning tools.

Syntax.

·ch· Channel number.

V.[2].G.TOI	Channel ·2·.

(V.)[ch].G.TOL

Variable that can be read and written from the program.
The variable returns the value of block preparation.

Tool being prepared. Length.

This variable is not valid for turning tools.

Syntax.

·ch· Channel number.

V.[2].G.TOL	Channel ·2·.

(V.)[ch].G.TOK

Variable that can be read and written from the program.
The variable returns the value of block preparation.

Tool being prepared. Length wear.

This variable is not valid for turning tools.

Syntax.

·ch· Channel number.

V.[2].G.TOK	Channel ·2·.

(V.)[ch].G.TOAN

Variable that can be read and written from the program.
The variable returns the value of block preparation.

Tool being prepared. Penetration angle.

Syntax.

·ch· Channel number.

V.[2].G.TOAN	Channel ·2·.

(V.)[ch].G.TOTIPR

Variable that can be read and written from the program.
The variable returns the value of block preparation.

Tool being prepared. Tip radius.

Syntax.

·ch· Channel number.

V.[2].G.TOTIPR	Channel ·2·.

(V.)[ch].G.TOWTIPR

Variable that can be read and written from the program.
The variable returns the value of block preparation.

Tool being prepared. Tool tip radius wear.

Syntax.

·ch· Channel number.

V.[2].G.TOWTIPR	Channel ·2·.

(V.)[ch].G.TOCUTL

Variable that can be read and written from the program.
The variable returns the value of block preparation.

Tool being prepared. Cutting length.

Syntax.

·ch· Channel number.

V.[2].G.TOCUTL	Channel ·2·.

(V.)[ch].G.NOSEA

Variable that can be read and written from the program.
The variable returns the value of block preparation.

Tool being prepared. Cutter angle.

Syntax.

·ch· Channel number.

V.[2].G.NOSEA	Channel ·2·.

(V.)[ch].G.NOSEW

Variable that can be read and written from the program.
The variable returns the value of block preparation.

Tool being prepared. Cutter width.

Syntax.

·ch· Channel number.

V.[2].G.NOSEW	Channel ·2·.

(V.)[ch].G.CUTA

Variable that can be read and written from the program.
The variable returns the value of block preparation.

Tool being prepared. Cutting angle of the cutter.

Syntax.

·ch· Channel number.

V.[2].G.CUTA	Channel ·2·.

(V.)[ch].A.TOFL.xn

Variable that can be read and written from the program.
Variable valid for rotary and linear axes.
The variable returns the value of block preparation.

Tool being prepared. Tool length offset on the xn axis.

On turning tools, it is used to define the tool length in each axis. On milling tools, it is used to define the tool position when using a tool holder or an intermediate tooling.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.A.TOFL.Z	Z axis.
V.A.TOFL.4	Axis with logic number ·4·.
V.[2].A.TOFL.1	Axis with index ·1· in the channel ·2·.

(V.)[ch].A.TOFLW.xn

Variable that can be read and written from the program.
Variable valid for rotary and linear axes.
The variable returns the value of block preparation.

Tool being prepared. Tool length offset wear on the xn axis.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.A.TOFLW.Z	Z axis.
V.A.TOFLW.4	Axis with logic number ·4·.
V.[2].A.TOFLW.1	Axis with index ·1· in the channel ·2·.

(V.)[ch].G.TOFL1
(V.)[ch].G.TOFL2
(V.)[ch].G.TOFL3

Variable that can be read and written from the program.
Variable valid for rotary and linear axes.
The variable returns the value of block preparation.

Tool being prepared. Tool length offset on the first axes of the channel.

On turning tools, it is used to define the tool length in each axis. On milling tools, it is used to define the tool position when using a tool holder or an intermediate tooling.

Syntax.

·ch· Channel number.

V.[2].G.TOFL1	Channel ·2·. First axis of the channel.
V.[2].G.TOFL2	Channel ·2·. Second axis of the channel.
V.[2].G.TOFL3	Channel ·2·. Third axis of the channel.

(V.)[ch].G.TOFLW1
(V.)[ch].G.TOFLW2
(V.)[ch].G.TOFLW3

Variable that can be read and written from the program.
Variable valid for rotary and linear axes.
The variable returns the value of block preparation.

Tool being prepared. Tool length offset wear on the first axes of the channel.

Syntax.

·ch· Channel number.

V.[2].G.TOFLW1	Channel ·2·. First axis of the channel.
V.[2].G.TOFLW2	Channel ·2·. Second axis of the channel.
V.[2].G.TOFLW3	Channel ·2·. Third axis of the channel.

10.52 Variables related to jog mode.

MOVEMENT ALLOWED IN JOG MODE.

(V.)[ch].G.INTMAN

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Movements in jog mode are allowed.

Syntax.

·ch· Channel number.

V.[2].G.INTMAN	Channel ·2·.

Remarks.

Jog movements are allowed when the jog mode or the TEACH-IN mode is active, during tool inspection and when functions G200 and G201 are active.

TYPE OF MOVEMENT ACTIVE IN THE CHANNEL.

(V.)G.MANMODE

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

Type of movement active for all the axes.

V.G.MANMODE

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
1	Handwheel mode.
2	Continuous jog mode.
3	Incremental jog mode.

Remarks.

The type of movement may be set by the switch of the operator panel or by PLC; the one set by PLC has the highest priority.

(V.)G.CNCMANMODE

Variable that can be read and written from the interface and read from the program and from the PLC.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

Type of movement selected at the switch for all the axes.

The value set by the switch of the operator panel has lower priority than the one set by PLC.

V.G.CNCMANMODE

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
1	Handwheel mode.
2	Continuous jog mode.
3	Incremental jog mode.

(V.)PLC.MANMODE

Variable that can be read and written from the PLC and read from the program and from the interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

Type of movement selected by PLC for all the axes.

The value selected by PLC prevails over the one selected on the operator panel. To cancel the feedrate set by PLC, set the variable to ·0·.

```
V.PLC.MANMODE
```

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Nothing has been selected by PLC.
1	Handwheel mode.
2	Continuous jog mode.
3	Incremental jog mode.

TYPE OF MOVEMENT ACTIVE ON AN AXIS.

(V.)[ch].A.MANMODE.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

Type of movement active for all the ·xn· axis.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

```
V.A.MANMODE.Z                Z axis.
V.A.MANMODE.4                Axis with logic number ·4·.
V.[2].A.MANMODE.1            Axis with index ·1· in the channel ·2·.
```

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
1	Handwheel mode.
2	Continuous jog mode.
3	Incremental jog mode.
4	Handwheel mode without selected axis. The handwheel mode has been selected but the axis to be moved has not been selected.

Remarks.

The type of movement may be set by the switch of the operator panel or by PLC; the one set by PLC has the highest priority.

(V.)[ch].A.CNCMMODE.xn

Variable that can be read and written from the interface and read from the program and from the PLC.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

Type of movement selected at the switch for the ·xn· axis.

The value set by the switch of the operator panel has lower priority than the one set by PLC.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.A.CNCMMODE.Z	Z axis.
V.A.CNCMMODE.4	Axis with logic number ·4·.
V.[2].A.CNCMMODE.1	Axis with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
1	Handwheel mode.
2	Continuous jog mode.
3	Incremental jog mode.

(V.)[ch].A.PLCMMODE.xn

Variable that can be read and written from the PLC and read from the program and from the interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

Type of movement selected by PLC for the ·xn· axis.

The value selected by PLC prevails over the one selected on the operator panel. To cancel the feedrate set by PLC, set the variable to ·0·.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.A.PLCMMODE.Z	Z axis.
V.A.PLCMMODE.4	Axis with logic number ·4·.
V.[2].A.PLCMMODE.1	Axis with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Nothing has been selected by PLC.
1	Handwheel mode.
2	Continuous jog mode.
3	Incremental jog mode.

Remarks.

If an axis has been set in handwheel mode from the PLC, it can only be deactivated from the PLC; a reset does not deactivate it.

SWITCH POSITION IN HANDWHEEL MODE.

(V.)G.MPGIDX

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

Active position for all the handwheels.

V.G.MPGIDX

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
1	Position 1.
2	Position 10.
3	Position 100.

Remarks.

The value may be set by the switch of the operator panel or by PLC; the one set by PLC has the highest priority.

(V.)G.CNCMPGIDX

Variable that can be read and written from the interface and read from the program and from the PLC.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

Position selected at the switch.

The value set by the switch of the operator panel has lower priority than the one set by PLC.

V.PLC.CNCMPGIDX

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
1	Position 1.
2	Position 10.
3	Position 100.

(V.)PLC.MPGIDX

Variable that can be read and written from the PLC and read from the program and from the interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

Position selected by PLC.

The value selected by PLC prevails over the one selected on the operator panel. To cancel the feedrate set by PLC, set the variable to ·0·.

V.PLC.MPGIDX

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
1	Position 1.
2	Position 10.
3	Position 100.

SWITCH POSITION IN INCREMENTAL JOG MODE.

(V.)G.INCJOGIDX

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

Active position for all the axes.

V.G.INCJOGIDX

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
1	Position 1.
2	Position 10.
3	Position 100.
4	Position 1000.
5	Position 10000.

Remarks.

The value may be set by the switch of the operator panel or by PLC; the one set by PLC has the highest priority.

(V.)G.CNCINCJOGIDX

Variable that can be read and written from the interface and read from the program and from the PLC.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

Position selected by the switch.

The value set by the switch of the operator panel has lower priority than the one set by PLC.

V.G.CNCINCJOGIDX

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
1	Position 1.
2	Position 10.
3	Position 100.
4	Position 1000.
5	Position 10000.

(V.)PLC.INCJOGIDX

Variable that can be read and written from the PLC and read from the program and from the interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

Position selected by PLC.

The value selected by PLC prevails over the one selected on the operator panel. To cancel the feedrate set by PLC, set the variable to ·0·.

```
V.PLC.INCJOGIDX
```

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
1	Position 1.
2	Position 10.
3	Position 100.
4	Position 1000.
5	Position 10000.

JOG FEEDRATES.

(V.)[ch].G.FMAN

Variable that can be read and written from the program and interface and read from the PLC.
The variable returns the execution value; reading it interrupts block preparation.

Jogging feedrate in G94.

Syntax.

·ch· Channel number.

```
V.[2].G.FMAN                    Channel ·2·.
```

Remarks.

This variable also updates its value when setting a new feedrate ("F" field on the screen for jog mode) or MDI/MDA mode and function G94 is active.

(V.)[ch].G.MANFPR

Variable that can be read and written from the program and interface and read from the PLC.
The variable returns the execution value; reading it interrupts block preparation.

Jogging feedrate in G95.

Syntax.

·ch· Channel number.

```
V.[2].G.MANFPR                  Channel ·2·.
```

Remarks.

This variable also updates its value when setting a new feedrate ("F" field on the screen for jog mode) or MDI/MDA mode and function G95 is active.

10.53 Variables related to the programmed functions.

MOVEMENT OF AXES AND SPINDLES.

(V.)[ch].A.INPOS.xn
(V.)[ch].A.INPOS.sn
(V.)[ch].SP.INPOS.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

Axis or spindle in position.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.INPOS.Z	Z axis.
V.A.INPOS.S	Spindle S.
V.SP.INPOS.S	Spindle S.
V.SP.INPOS	Master spindle.
V.A.INPOS.4	Axis or spindle with logic number ·4·.
V.[2].A.INPOS.1	Axis with index ·1· in the channel ·2·.
V.SP.INPOS.2	Spindle with index ·2· in the system.
V.[2].SP.INPOS.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.DIST.xn
(V.)[ch].A.DIST.sn
(V.)[ch].SP.DIST.sn

Variable that can be read and written via program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Distance traveled by the axis or spindle.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.DIST.Z	Z axis.
V.A.DIST.S	Spindle S.
V.SP.DIST.S	Spindle S.
V.SP.DIST	Master spindle.
V.A.DIST.4	Axis or spindle with logic number ·4·.
V.[2].A.DIST.1	Axis with index ·1· in the channel ·2·.
V.SP.DIST.2	Spindle with index ·2· in the system.
V.[2].SP.DIST.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.ACCUDIST.xn
(V.)[ch].A.ACCUDIST.sn
(V.)[ch].SP.ACCUDIST.sn

Variable that can be read and written via program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Distance traveled by the axis or spindle since the last coordinate latching. This variable is initialized to ·0· when a latching event takes place. To add a position offset to this variable at the latch point, just add it from the PLC in a later cycle.

Axis synchronization pemits treating a rotary axis as an infinite axis making it possible to increase the feedback count of the axis indefinitely (wihout limits) regardless of the value of the module. The CNC uses this variable follow up on the axis. This variable may be initialized in a sampling period and keep counting from the initialized value on.

This feature is useful, for example, for a rotary axis or encoder that moves an endless coveyor belt on where the part is. The infinite axis treatment allows synchronizing the coordinate of the conveyor belt with an external event and, this way, count the moving distance of the part beyond the value of the module of the rotary axis that moves the belt.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.ACCUDIST.Z	Z axis.
V.A.ACCUDIST.S	Spindle S.
V.SP.ACCUDIST.S	Spindle S.
V.SP.ACCUDIST	Master spindle.
V.A.ACCUDIST.4	Axis or spindle with logic number ·4·.
V.[2].A.ACCUDIST.1	Axis with index ·1· in the channel ·2·.
V.SP.ACCUDIST.2	Spindle with index ·2· in the system.
V.[2].SP.ACCUDIST.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.PREVACCUDIST.xn
(V.)[ch].A.PREVACCUDIST.sn
(V.)[ch].SP.PREVACCUDIST.sn

Variable that can be read and written via program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Distance traveled by the axis or spindle between the last two coordinate latchings. This variable updates (refreshes) its value at every latching event; that's why the variable will have a ·0· value until the first one takes place.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.PREVACCUDIST.Z	Z axis.
V.A.PREVACCUDIST.S	Spindle S.
V.SP.PREVACCUDIST.S	Spindle S.
V.SP.PREVACCUDIST	Master spindle.
V.A.PREVACCUDIST.4	Axis or spindle with logic number ·4·.
V.[2].A.PREVACCUDIST.1	Axis with index ·1· in the channel ·2·.
V.SP.PREVACCUDIST.2	Spindle with index ·2· in the system.
V.[2].SP.PREVACCUDIST.1	Spindle with index ·1· in the channel ·2·.

WORK PLANE AND AXES.

(V.)[ch].G.PLANE

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Axes making up the work plane.

Syntax.

·ch· Channel number.

V.[2].G.PLANE	Channel ·2·.

Values of the variable.

The values returned by this variable are encoded as follows.

X=10	X1=11	X2=12	X3=13	X4=14	... X9=19
Y=20	Y1=21	Y2=22	Y3=23	Y4=24	... Y9=29
Z=30	Z1=31	Z2=32	Z3=33	Z4=34	... Z9=39
U=40	U1=41	U2=42	U3=43	U4=44	... U9=49
V=50	V1=51	V2=52	V3=53	V4=54	... V9=59
W=60	W1=61	W2=62	W3=63	W4=64	... W9=69
A=70	A1=71	A2=72	A3=73	A4=74	... A9=79
B=80	B1=81	B2=82	B3=83	B4=84	... B9=89
C=90	C1=91	C2=92	C3=93	C4=94	... C9=99

Main axes.	Main plane.	Reading the variable.
X-Y-Z	G17 (XY)	V.[1].G.PLANE = 1020 V.[1].G.LONGAX = 30 V.[1].G.TOOLDIR = 2
X-Y-Z	G18 (ZX)	V.[1].G.PLANE = 3010 V.[1].G.LONGAX = 20 V.[1].G.TOOLDIR = 2
X-V1-Z3	G17 (X-V1) #TOOL AX [V1-]	V.[1].G.PLANE = 1051 V.[1].G.LONGAX = 33 V.[1].G.TOOLDIR = 1

(V.)[ch].G.LONGAX

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Longitudinal axis.

Syntax.

·ch· Channel number.

V.[2].G.LONGAX	Channel ·2·.

Values of the variable.

The values returned by this variable are encoded as follows.

X=10	X1=11	X2=12	X3=13	X4=14	... X9=19
Y=20	Y1=21	Y2=22	Y3=23	Y4=24	... Y9=29
Z=30	Z1=31	Z2=32	Z3=33	Z4=34	... Z9=39
U=40	U1=41	U2=42	U3=43	U4=44	... U9=49
V=50	V1=51	V2=52	V3=53	V4=54	... V9=59
W=60	W1=61	W2=62	W3=63	W4=64	... W9=69

A=70	A1=71	A2=72	A3=73	A4=74	... A9=79
B=80	B1=81	B2=82	B3=83	B4=84	... B9=89
C=90	C1=91	C2=92	C3=93	C4=94	... C9=99

(V.)[ch].G.TOOLDIR

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Tool orientation.

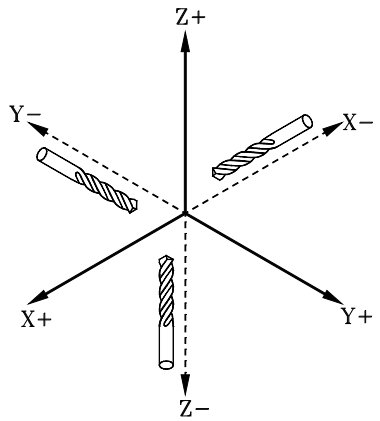
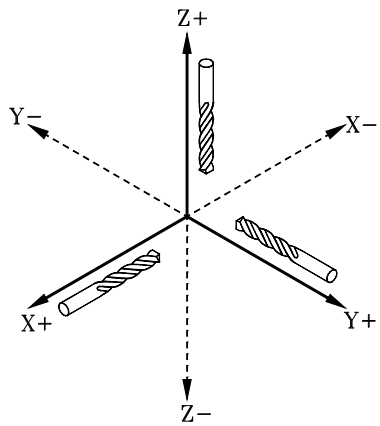
Syntax.

·ch· Channel number.

V.[2].G.TOOLDIR Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
1	The tool is positioned in the positive direction of the axis. 
2	The tool is positioned in the negative direction of the axis. 

(V.)[ch].G.PLAXNAME1
(V.)[ch].G.PLAXNAME2
(V.)[ch].G.PLAXNAME3

Variable to be read via interface.

Name the main axes of the channel.

Syntax.

·ch· Channel number.

[2].G.PLAXNAME1	Channel ·2·. Abscissa axis.
[2].G.PLAXNAME2	Channel ·2·. Ordinate axis.
[2].G.PLAXNAME3	Channel ·2·. Third main axis.

(V.)[ch].G.PLANELONG

Variable to be read via interface.

Index of the tool in the axis channel.

Syntax.

·ch· Channel number.

[2].G.PLANELONG	Channel ·2·.

Remarks.

In this variable, the index in the channel of the first axis will be ·0·, that of the second axis will be ·1· and so on.

"G" AND "M" FUNCTIONS.

(V.)[ch].G.GS[nb]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.
Report variable (to be used from the scripts).

Status of the requested "G" function.

Each function has a bit that indicates whether the relevant function is active (=1) or not (=0).

Syntax.

·ch· Channel number.

·nb· Function number.

V.[2].G.GS[3]	Channel ·2·. G3 function.

(V.)[ch].G.MS[nb]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.
Report variable (to be used from the scripts).

Status of the requested "M" function.

Each function has a bit that indicates whether the relevant function is active (=1) or not (=0).

Syntax.

·ch· Channel number.

·nb· Function number.

V.[2].G.MS[5]	Channel ·2·. M5 function.

(V.)[ch].G.HGS1
(V.)[ch].G.HGS2
(V.)[ch].G.HGS3
(V.)[ch].G.HGS4
(V.)[ch].G.HGS5
(V.)[ch].G.HGS6
(V.)[ch].G.HGS7

(V.)[ch].G.HGS8
(V.)[ch].G.HGS9
(V.)[ch].G.HGS10

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Status of the requested "G" (32 bit) functions.

Syntax.

·ch· Channel number.

V.[2].G.HGS1	Channel ·2·.

Remarks.

Every variable corresponds to a range of 32 G functions and returns a 32-bit value; 1 bit per function. Each bit indicates whether the function is active (bit = 1) or not (bit = 0). The least significant bit corresponds to the lowest function of the range.

Variable.	G function range.	
(V.)[ch].G.HGS1	G0 - G31.	Bit 0 corresponds to G0.
(V.)[ch].G.HGS2	G32-G63	Bit 0 corresponds to G32.
(V.)[ch].G.HGS3	G64-G95	Bit 0 corresponds to G64.
(V.)[ch].G.HGS4	G96-G127	Bit 0 corresponds to G96.
(V.)[ch].G.HGS5	G128-G159	Bit 0 corresponds to G128.
(V.)[ch].G.HGS6	G160-G191	Bit 0 corresponds to G160.
(V.)[ch].G.HGS7	G192-G223	Bit 0 corresponds to G192.
(V.)[ch].G.HGS8	G224-G255	Bit 0 corresponds to G224.
(V.)[ch].G.HGS9	G256-G287	Bit 0 corresponds to G256.
(V.)[ch].G.HGS10	G288-G319	Bit 0 corresponds to G288.

To check the status of function G08 from the part-program.

```
$IF [V.[1].G.HGS1 & [2**8]] == 2**8
```

To check the status of function G101 from the part-program.

```
$IF [V.[1].G.HGS4 & [2**5]] == 2**5
```

To check the status of function G08 from the PLC.

```
DFU B0KEYBD1 = CNCRD(G.HGS1, R100, M100)
B8R100 = ···
```

To check the status of function G101 from the PLC.

```
DFU B0KEYBD1 = CNCRD(G.HGS4, R101, M100)
B5R101 = ···
```

(V.)[ch].G.HGS

Variable to be read via interface.
Report variable (to be used from the scripts).

"G" functions to be displayed in the history.

[2].G.HGS	Channel ·2·.

Remarks.

This variable returns a binary value. Each function has a bit that indicates whether the relevant variable will be displayed (=1) or not (=0). Bit 0, the least significant bit, corresponds to G0, bit 1 to G1 and so on.

(V.)[ch].G.HMS

Variable to be read via interface.
Report variable (to be used from the scripts).

"M" functions of the master spindle to be displayed in the history.

[2].G.HMS	Channel ·2·. Master spindle.

Remarks.

This variable returns a binary value. Each function has a bit that indicates whether the relevant variable will be displayed (=1) or not (=0). Bit 0, the least significant bit, corresponds to M0, bit 1 to M1 and so on.

(V.)[ch].G.HMS1
(V.)[ch].G.HMS2
(V.)[ch].G.HMS3
(V.)[ch].G.HMS4

Variable to be read via interface.
Report variable (to be used from the scripts).

"M" functions of spindle 1 through 4 to be displayed in the history.

[2].G.HMS2	Channel ·2·. Spindle ·2·.

Remarks.

This variable returns a binary value. Each function has a bit that indicates whether the relevant variable will be displayed (=1) or not (=0). Bit 0, the least significant bit, corresponds to M0, bit 1 to M1 and so on.

CANNED CYCLES.

(V.)[ch].G.CYCLETYPEON

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Type of canned cycle active

Syntax.

·ch· Channel number.

V.[2].G.CYCLETYPEON	Channel ·2·.

CANNED CYCLE CALLING PARAMETERS.

(V.)C.A··Z

Variable that can be read and written from the program.
The variable returns the value of block preparation.

Value of the ISO canned cycle calling parameter.

Syntax.

·A··Z· Call parameter.

V.C.F	"F" parameter.

Remarks.

Calling a canned cycle.	Reading the variable.
G90 G81 Z0 I-15	V.C.Z = 0 V.C.Z = -15

(V.)C.name

Variable that can be read and written from the program.
The variable returns the value of block preparation.

Value of the parameters that call upon the canned cycles of the editor.

Syntax.

·name· Call parameter.

V.C.MROUGHIN	Parameter MROUGHIN.

(V.)C.CALLP_A··Z

Variable that can be read from the program.
The variable returns the value of block preparation.

Parameter programmed in the call to a canned cycle.

Syntax.

·A··Z· Call parameter.

V.C.CALLP_F	"F" parameter.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	It has not been programmed.
1	It has been programmed.

Remarks.

Calling a canned cycle.	Reading the variable.
G90 G81 Z0 I-15	V.C.CALLP_Z = 1 V.C.CALLP_I = 1 V.C.CALLP_K = 0

(V.)C.P_A··Z

Variable that can be read and written from the program.
The variable returns the value of block preparation.

Value of the positioning cycle calling parameter.

Syntax.

·A··Z· Call parameter.

V.C.P_F	"F" parameter.

Remarks.

Calling a canned cycle.	Reading the variable.
G160 A30 X100 K10 P6	V.C.P_A = 30 V.C.P_X = 100

(V.)C.P_CALLP_A··Z

Variable that can be read from the program.
The variable returns the value of block preparation.

Parameter programmed in the call to a positioning cycle.

Syntax.

·A··Z· Call parameter.

V.C.P_CALLP_F	"F" parameter.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	It has not been programmed.
1	It has been programmed.

Remarks.

Calling a canned cycle.	Reading the variable.
G160 A30 X100 K10 P6	V.C.P_CALLP_A = 1 V.C.P_CALLP_K = 1 V.C.P_CALLP_R = 0

SUBROUTINE CALLING PARAMETERS.

(V.)C.PCALLP_A··Z

Variable that can be read from the program.
The variable returns the value of block preparation.

Parameter programmed in the call to a subroutine

This variable is valid for OEM subroutines (G18x) and for subroutines called upon using #PCALL or #MCALL..

Syntax.

·A··Z· Call parameter.

V.C.PCALLP_F	"F" parameter.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	It has not been programmed.
1	It has been programmed.

Remarks.

Call to a subroutine.	Reading the variable.
#PCALL sub.nc A12.56 D3	V.C.PCALLP_A = 1 V.C.PCALLP_D = 1

ARC RELATED.

(V.)[ch].G.R

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Radius of the arc.

Syntax.

·ch· Channel number.

V.[2].G.R	Channel ·2·.

(V.)[ch].G.I
(V.)[ch].G.J
(V.)[ch].G.K

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Coordinates referred to the arc center.

Syntax.

·ch· Channel number.

V.[2].G.I	Channel ·2·. First axis of the channel.
V.[2].G.J	Channel ·2·. Second axis of the channel.
V.[2].G.K	Channel ·2·. Third axis of the channel.

Remarks.

Being G20 active, parameters "I", "J" and "K" are associated with the abscissa axis, ordinate axis and the axis perpendicular to the work plane respectively.

(V.)[ch].G.CIRERR[1]
(V.)[ch].G.CIRERR[2]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Arc center correction in the first axes of the channel.

Syntax.

·ch· Channel number.

V.[2].G.CIRERR[1]	Channel ·2·. First axis of the channel.
V.[2].G.CIRERR[2]	Channel ·2·. Second axis of the channel.

Remarks.

Being function G265 active, the CNC recalculates the center if the arc is not exact but it is within tolerance.

Part-program.	Reading the variable.
G00 X0 Y0 G2 X120 Y120.001 I100 J20	V.G.R = 101.980881 V.G.I = 100.0004 V.G.J = 20.0004 V.G.CIRERR[1] = -0.000417 V.G.CIRERR[2] = -0.000417

Being function G264 active, if the arc is not exact, but it is within tolerances, the CNC executes an arc with the radius calculated from the starting point. The CNC maintains the position of the center.

Part-program.	Reading the variable.
G00 X0 Y0 G2 X120 Y120.001 I100 J20	V.G.R = 101.981371 V.G.I = 100 V.G.J = 20 V.G.CIRERR[1] = 0 V.G.CIRERR[2] = 0

POLAR ORIGIN.

(V.)[ch].G.PORGF

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Position of the polar origin referred to part zero (abscissa).

Syntax.

·ch· Channel number.

V.[2].G.PORGF	Channel ·2·.

(V.)[ch].G.PORGS

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Position of the polar origin referred to part zero (ordinate).

Syntax.

·ch· Channel number.

V.[2].G.PORGS	Channel ·2·.

GEOMETRIC ASSISTANCE. MIRROR IMAGE.

(V.)[ch].G.MIRROR

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Active mirror images.

Syntax.

·ch· Channel number.

V.[2].G.MIRROR	Channel ·2·.

Remarks.

This variable returns the result in the least significant bits, one per axis (1= active and 0=not active). The least significant bit is for the first axis, the next one for the second axis and so on.

(V.)[ch].G.MIRROR1
(V.)[ch].G.MIRROR2
(V.)[ch].G.MIRROR3

Variable that can be read from the program.
The variable returns the value of block preparation.

Mirror image active on the first axes of the channel.

Syntax.

·ch· Channel number.

V.[2].G.MIRROR1	Channel ·2·. First axis of the channel.
V.[2].G.MIRROR2	Channel ·2·. Second axis of the channel.
V.[2].G.MIRROR3	Channel ·2·. Third axis of the channel.

GEOMETRIC ASSISTANCE. SCALING FACTOR.

(V.)[ch].G.SCALE

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

It indicates the active general scaling factor.

Syntax.

·ch· Channel number.

V.[2].G.SCALE	Channel ·2·.

GEOMETRIC ASSISTANCE. COORDINATE SYSTEM ROTATION.

(V.)[ch].G.ROTPF

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Position of the rotation center referred to part zero (abscissa).

Syntax.

·ch· Channel number.

V.[2].G.ROTPF	Channel ·2·.

(V.)[ch].G.ROTPS

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Position of the rotation center referred to part zero (ordinate).

Syntax.

·ch· Channel number.

V.[2].G.ROTPS	Channel ·2·.

(V.)[ch].G.ORGROT

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Rotation angle of the coordinate system.

Syntax.

·ch· Channel number.

V.[2].G.ORGROT	Channel ·2·.

BLOCK REPETITION.

(V.)[ch].G.PENDRPT

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Number of pending repetitions with #RPT.

Syntax.

·ch· Channel number.

V.[2].G.PENDRPT	Channel ·2·.

Remarks.

This variable indicates the number of repetitions left to execute. In the first execution, its value is the programmed number of repetitions minus one and in the last one, its value is zero.

(V.)[ch].G.PENDNR

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Number of pending repetitions with NR.

Syntax.

·ch· Channel number.

V.[2].G.PENDNR	Channel ·2·.

Remarks.

This variable indicates the number of repetitions left to execute. In the first execution, its value is the programmed number of repetitions minus one and in the last one, its value is zero.

AXIS SLAVING.

(V.)[ch].G.LINKACTIVE

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Axis slaving (coupling) status.

Syntax.

·ch· Channel number.

V.[2].G.LINKACTIVE	Channel ·2·.

HSC FUNCTION.

(V.)[ch].G.HSC

Variable that can be read from the program.
The variable returns the value of block preparation.

HSC function active.

Syntax.

·ch· Channel number.

V.[2].G.HSC	Channel ·2·.

ACTIVE PROBE.

(V.)[ch].G.ACTIVPROBE

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Number of the active probe.

Syntax.

·ch· Channel number.

V.[2].G.ACTIVPROBE	Channel ·2·.

STATUS OF THE LOCAL PROBES.

(V.)G.PRBST1

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Status of probe ·1·.

V.G.PRBST1

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	The probe is not making contact or has not been initialized.
1	The probe is making contact.

(V.)G.PRBST2

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Status of probe ·2·.

V.G.PRBST2

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	The probe is not making contact or has not been initialized.
1	The probe is making contact.

PROBING (G100/G101/G102).

(V.)[ch].G.MEASOK

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Probing completed.

Syntax.

·ch· Channel number.

V.[2].G.MEASOK	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

(V.)[ch].A.MEASOK.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Probing done on the axis.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.MEASOK.Z	Z axis.
V.A.MEASOK.S	Spindle S.
V.SP.MEASOK.S	Spindle S.
V.SP.MEASOK	Master spindle.
V.A.MEASOK.4	Axis or spindle with logic number ·4·.
V.[2].A.MEASOK.1	Axis with index ·1· in the channel ·2·.
V.SP.MEASOK.2	Spindle with index ·2· in the system.
V.[2].SP.MEASOK.1	Spindle with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

(V.)[ch].G.PLMEASOK1
(V.)[ch].G.PLMEASOK2
(V.)[ch].G.PLMEASOK3

Variable that can be read from the program.
The variable returns the execution value; reading it interrupts block preparation.

Probing on the plane axes completed.

Syntax.

·ch· Channel number.

V.[2].G.PLMEASOK1	Channel ·2·. First axis of the plane.
V.[2].G.PLMEASOK2	Channel ·2·. Second axis of the plane.
V.[2].G.PLMEASOK3	Channel ·2·. Third axis of the plane.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

(V.)[ch].A.MEAS.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Measured value. Machine coordinates of the tool base.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.MEAS.Z	Z axis.
V.A.MEAS.S	Spindle S.
V.SP.MEAS.S	Spindle S.
V.SP.MEAS	Master spindle.
V.A.MEAS.4	Axis or spindle with logic number ·4·.
V.[2].A.MEAS.1	Axis with index ·1· in the channel ·2·.
V.SP.MEAS.2	Spindle with index ·2· in the system.
V.[2].SP.MEAS.1	Spindle with index ·1· in the channel ·2·.

Remarks.

Part-program.	Reading the variable.
G00 X0 Y0 G100 X100 F100	V.A.MEAS.X = 95 V.A.MEASOF.X = -5 V.A.MEASOK.X = 1

(V.)[ch].A.ATIPMEAS.xn

Variable that can be read from the program.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Measured value. Part coordinates of the tool tip.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.ATIPMEAS.Z	Z axis.
V.A.ATIPMEAS.S	Spindle S.
V.SP.ATIPMEAS.S	Spindle S.
V.SP.ATIPMEAS	Master spindle.

V.A.ATIPMEAS.4	Axis or spindle with logic number ·4·.
V.[2].A.ATIPMEAS.1	Axis with index ·1· in the channel ·2·.
V.SP.ATIPMEAS.2	Spindle with index ·2· in the system.
V.[2].SP.ATIPMEAS.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].G.PLMEAS1
(V.)[ch].G.PLMEAS2
(V.)[ch].G.PLMEAS3

Variable that can be read from the program.
The variable returns the value of block preparation.

Value measured on the first axes of the channel. Part coordinates of the tool tip.

Syntax.

·ch· Channel number.

V.[2].G.PLMEAS1	Channel ·2·. First axis of the channel.
V.[2].G.PLMEAS2	Channel ·2·. Second axis of the channel.
V.[2].G.PLMEAS3	Channel ·2·. Third axis of the channel.

(V.)[ch].A.MEASOF.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Difference with respect to programmed point.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.MEASOF.Z	Z axis.
V.A.MEASOF.S	Spindle S.
V.SP.MEASOF.S	Spindle S.
V.SP.MEASOF	Master spindle.
V.A.MEASOF.4	Axis or spindle with logic number ·4·.
V.[2].A.MEASOF.1	Axis with index ·1· in the channel ·2·.
V.SP.MEASOF.2	Spindle with index ·2· in the system.
V.[2].SP.MEASOF.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.MEASIN.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Coordinate that includes a measurement offset.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.MEASIN.Z	Z axis.
V.A.MEASIN.S	Spindle S.
V.SP.MEASIN.S	Spindle S.
V.SP.MEASIN	Master spindle.
V.A.MEASIN.4	Axis or spindle with logic number ·4·.

V.[2].A.MEASIN.1	Axis with index ·1· in the channel ·2·.
V.SP.MEASIN.2	Spindle with index ·2· in the system.
V.[2].SP.MEASIN.1	Spindle with index ·1· in the channel ·2·.

Remarks.

The CNC only refreshes this variable when probing with G101.

MANUAL INTERVENTION.

(V.)[ch].A.MANOF.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.

Distance moved in jog mode or tool inspection.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.A.MANOF.Z	Z axis.
V.A.MANOF.4	Axis with logic number ·4·.
V.[2].A.MANOF.1	Axis with index ·1· in the channel ·2·.

(V.)[ch].A.ADDMANOF.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.

Distance moved with G200 or G201.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.A.ADDMANOF.Z	Z axis.
V.A.ADDMANOF.4	Axis with logic number ·4·.
V.[2].A.ADDMANOF.1	Axis with index ·1· in the channel ·2·.

Remarks.

The value of this variable is maintained during the execution of the program even when canceling manual intervention.

STATUS OF THE ANGULAR TRANSFORMATION.

(V.)[ch].G.ANGAXST

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Overall status of the angular transformation of the channel.

Syntax.

·ch· Channel number.

V.[2].G.ANGAXST	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Off.
1	On.
2	Frozen (suspended).

(V.)G.ANGIDST[nb]

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Status of the angular transformation [nb] of the machine parameters.

Syntax.

·nb· Number of the angular transformation in the order defined in the machine parameters.

V.G.ANGIDST[1]	Angular transformation ·1·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Off.
1	On.
2	Frozen (suspended).

TANGENTIAL CONTROL STATUS.

(V.)[ch].G.TGCTRLST

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Status of tangential control in the channel.

Syntax.

·ch· Channel number.

V.[2].G.TGCTRLST	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Off.
1	On.
2	Frozen (suspended).

(V.)[ch].A.TGCTRLST.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the value of block preparation.

Status of tangential control in the axis.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.A.TGCTRLST.Z	Z axis.
V.A.TGCTRLST.4	Axis with logic number ·4·.
V.[2].A.TGCTRLST.1	Axis with index ·1· in the channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Off.
1	On.
2	Frozen (suspended).

(V.)[ch].A.TANGAN.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.

Angle programmed on the axis.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.A.TANGAN.Z	Z axis.
V.A.TANGAN.4	Axis with logic number ·4·.
V.[2].A.TANGAN.1	Axis with index ·1· in the channel ·2·.

(V.)[ch].G.TANGFEED

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Positioning feedrate for tangential control.

Syntax.

·ch· Channel number.

V.[2].G.TANGFEED	Channel ·2·.

SYNCHRONIZATION OF CHANNELS.

(V.)[ch].G.MEETST[mk]

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Status of the MEET type mark [mk] in the [ch] channel.

Syntax.

·ch· Channel number.

·mk· Number of the synchronization mark.

V.[2].G.MEETST[4]	Channel ·2·. Mark ·4·.

(V.)[ch].G.WAITST[mk]

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Status of the WAIT type mark [mk] in the [ch] channel.

Syntax.

·ch· Channel number.

·mk· Number of the synchronization mark.

V.[2].G.WAITST[4]	Channel ·2·. Mark ·4·.

(V.)[ch].G.MEETCH[nch]

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

MEET type mark originated in channel [nch] and expected by channel [ch].

Syntax.

·ch· Channel number.

·nch· Channel that originates the synchronization mark.

V.[2].G.MEETCH[4]	Channel ·2·. Channel ·4·.

(V.)[ch].G.WAITCH[nch]

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

WAIT type mark originated in channel [nch] and expected by channel [ch].

Syntax.

·ch· Channel number.

·nch· Channel that originates the synchronization mark.

V.[2].G.WAITCH[4]	Channel ·2·. Channel ·4·.

SELECTION OF THE KINEMATICS.

(V.)[ch].G.KINID

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Number of the active kinematics.

If no kinematics is active, the variable will return a ·0· value.

Syntax.

·ch· Channel number.

V.[2].G.KINID	Channel ·2·.

POSITION OF THE AXES OF THE KINEMATICS.

(V.)[ch].G.POSROTF
(V.)[ch].G.POSROTS
(V.)[ch].G.POSROTT

Variable that can be read and written via program, PLC and interface.
The variable returns the value of block preparation.

Current position of the main rotary axes.

Syntax.

·ch· Channel number.

V.[2].G.POSROTF	Current position of the main rotary axis.
V.[2].G.POSROTS	Current position of the secondary rotary axis.
V.[2].G.POSROTT	Current position of the third rotary axis.

(V.)[ch].G.TOOLORIF1
(V.)[ch].G.TOOLORIS1
(V.)[ch].G.TOOLORIT1

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Target position for the main rotary axes (solution 1).

Syntax.

·ch· Channel number.

V.[2].G.TOOLORIF1	Target position for the main rotary axis.
V.[2].G.TOOLORIS1	Target position for the secondary rotary axis.
V.[2].G.TOOLORIT1	Target position for the third rotary axis.

Remarks.

These variables that indicate the position the main axes must occupy in order for the tool to be perpendicular to the define work plane. They are very useful when the spindle is not fully motorized (mono-rotary or manual spindles).

On angular (swivel) spindle heads, there are 2 possible solutions when calculating this target position. These variables indicate the solution that involves the shortest movement of the main rotary axis with respect to the zero position.

(V.)[ch].G.TOOLORIF2
(V.)[ch].G.TOOLORIS2
(V.)[ch].G.TOOLORIT2

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Target position for the main rotary axes (solution 2).

Syntax.

·ch· Channel number.

V.[2].G.TOOLORIF2	Target position for the main rotary axis.
V.[2].G.TOOLORIS2	Target position for the secondary rotary axis.
V.[2].G.TOOLORIT2	Target position for the third rotary axis.

Remarks.

These variables that indicate the position the main axes must occupy in order for the tool to be perpendicular to the define work plane. They are very useful when the spindle is not fully motorized (mono-rotary or manual spindles).

On angular (swivel) spindle heads, there are 2 possible solutions when calculating this target position. These variables indicate the solution that involves the longest movement of the main rotary axis with respect to the zero position.

INCLINED PLANES.

(V.)[ch].G.CS

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.
Report variable (to be used from the scripts).

Number of the active CS function.

Syntax.

·ch· Channel number.

V.[2].G.CS	Channel ·2·.

(V.)[ch].G.ACS

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.
Report variable (to be used from the scripts).

Number of the active ACS function.

Syntax.

·ch· Channel number.

V.[2].G.ACS	Channel ·2·.

(V.)[ch].G.TOOLCOMP

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Longitudinal compensation function active.

Syntax.

·ch· Channel number.

V.[2].G.TOOLCOMP	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
1	RTCP.
2	TLC.
3	None.

DIE RESULTING FROM THE INCLINED PLANE.

(V.)[ch].G.CSMAT1
(V.)[ch].G.CSMAT2
(V.)[ch].G.CSMAT3
(V.)[ch].G.CSMAT4
(V.)[ch].G.CSMAT5
(V.)[ch].G.CSMAT6
(V.)[ch].G.CSMAT7

(V.)[ch].G.CSMAT8
(V.)[ch].G.CSMAT9

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Die resulting from the inclined plane.

These variables correspond to the transformation matrix from theoretical reference system to the real reference system.

Syntax.

·ch· Channel number.

(V.)[ch].G.CSMAT1	Channel ·2·. Element row 1 column 1.
(V.)[ch].G.CSMAT2	Channel ·2·. Element row 1 column 2.
(V.)[ch].G.CSMAT3	Channel ·2·. Element row 1 column 3.
(V.)[ch].G.CSMAT4	Channel ·2·. Element row 2 column 1.
(V.)[ch].G.CSMAT5	Channel ·2·. Element row 2 column 2.
(V.)[ch].G.CSMAT6	Channel ·2·. Element row 2 column 3.
(V.)[ch].G.CSMAT7	Channel ·2·. Element row 3 column 1.
(V.)[ch].G.CSMAT8	Channel ·2·. Element row 3 column 2.
(V.)[ch].G.CSMAT9	Channel ·2·. Element row 3 column 3.

(V.)[ch].G.CSMAT10
(V.)[ch].G.CSMAT11
(V.)[ch].G.CSMAT12

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Offset of the current coordinate system referred to machine zero on the first axes.

These variables correspond to the transformation matrix from theoretical reference system to the real reference system.

Syntax.

·ch· Channel number.

(V.)[ch].G.CSMAT10	Channel ·2·. Offset on the first axis.
(V.)[ch].G.CSMAT11	Channel ·2·. Offset on the second axis.
(V.)[ch].G.CSMAT12	Channel ·2·. Offset on the third axis.

FEED-FORWARD AND AC-FORWARD.

(V.)[ch].A.FFGAIN.xn
(V.)[ch].A.FFGAIN.sn
(V.)[ch].SP.FFGAIN.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Active percentage of feed-forward.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.FFGAIN.Z	Z axis.
V.A.FFGAIN.S	Spindle S.
V.SP.FFGAIN.S	Spindle S.
V.SP.FFGAIN	Master spindle.

V.A.FFGAIN.4	Axis or spindle with logic number ·4·.
V.[2].A.FFGAIN.1	Axis with index ·1· in the channel ·2·.
V.SP.FFGAIN.2	Spindle with index ·2· in the system.
V.[2].SP.FFGAIN.1	Spindle with index ·1· in the channel ·2·.

Remarks.

The reading from the PLC will be given in tenths (x10); i.e. if the parameter value is ·10·, the PLC reading will return a value of ·100·.

(V.)[ch].A.ACFGAIN.xn
(V.)[ch].A.ACFGAIN.sn
(V.)[ch].SP.ACFGAIN.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Active percentage of AC-forward.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.ACFGAIN.Z	Z axis.
V.A.ACFGAIN.S	Spindle S.
V.SP.ACFGAIN.S	Spindle S.
V.SP.ACFGAIN	Master spindle.
V.A.ACFGAIN.4	Axis or spindle with logic number ·4·.
V.[2].A.ACFGAIN.1	Axis with index ·1· in the channel ·2·.
V.SP.ACFGAIN.2	Spindle with index ·2· in the system.
V.[2].SP.ACFGAIN.1	Spindle with index ·1· in the channel ·2·.

Remarks.

The reading from the PLC will be given in tenths (x10); i.e. if the parameter value is ·10·, the PLC reading will return a value of ·100·.

(V.)[ch].A.ACTFFW.xn
(V.)[ch].A.ACTFFW.sn
(V.)[ch].SP.ACTFFW.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Instant feed-forward.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.ACTFFW.Z	Z axis.
V.A.ACTFFW.S	Spindle S.
V.SP.ACTFFW.S	Spindle S.
V.SP.ACTFFW	Master spindle.
V.A.ACTFFW.4	Axis or spindle with logic number ·4·.

V.[2].A.ACTFFW.1	Axis with index ·1· in the channel ·2·.
V.SP.ACTFFW.2	Spindle with index ·2· in the system.
V.[2].SP.ACTFFW.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.ACTACF.xn
(V.)[ch].A.ACTACF.sn
(V.)[ch].SP.ACTACF.sn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Instant AC-forward.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.ACTACF.Z	Z axis.
V.A.ACTACF.S	Spindle S.
V.SP.ACTACF.S	Spindle S.
V.SP.ACTACF	Master spindle.
V.A.ACTACF.4	Axis or spindle with logic number ·4·.
V.[2].A.ACTACF.1	Axis with index ·1· in the channel ·2·.
V.SP.ACTACF.2	Spindle with index ·2· in the system.
V.[2].SP.ACTACF.1	Spindle with index ·1· in the channel ·2·.

ERRORS AND WARNINGS.

(V.)[ch].G.CNCERR

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Number of the error with highest priority in the indicated channel.

Syntax.

·ch· Channel number.

V.[2].G.CNCERR	Channel ·2·.

Remarks.

This variable is initialized with a reset. If several channels are in the same group, the reset of a channel means resetting all of them, thus initializing the variables of all the channels of the group.

If several channels are in the same group, an error in a channel causes the same error in all of them; in this case, this variable will have the same value for all the channels of the group.

(V.)[ch].G.CNCWARNING

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Number of warning displayed in the indicated channel.

Syntax.

·ch· Channel number.

V.[2].G.CNCWARNING	Channel ·2·.

Remarks.

This variable is initialized with a reset. If several channels are in the same group, the reset of a channel means resetting all of them, thus initializing the variables of all the channels of the group.

If there are several warnings, the value of the variable is updated as they are eliminated. When the last warning is eliminated, this variable is initialized to zero.

REPOSITIONING OF AXES AND SPINDLES.

(V.)[ch]G.ENDREP

Variable that can be read and written via program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

All the axes are repositioned.

Syntax.

·ch· Channel number.

V.[2].G.ENDREP	Channel ·2·.

(V.)[ch].G.SPDLREP

Variable that can be read and written via program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

M function to be used to reposition the spindle after a tool inspection.

Syntax.

·ch· Channel number.

V.[2].G.SPDLREP	Channel ·2·.

ACTIVE ZERO OFFSET.

(V.)[ch]G.EXTORG

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Function that defines the zero offset (zero offset or coordinate preset).

Syntax.

·ch· Channel number.

V.[2].G.EXTORG	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	G53. There is no zero offset.
1 - 99	G159. Zero offset (99 available). If 1, G159=1; if 2, G159=2 and so on.
1000	G92. Coordinate preset.

Remarks.

The numbering of zero offsets G54 through G59 is always the same; G54=1, G55=2, G56=3, G57=4, G58=5, G59=6.

10.54 Variables related to the electronic cam.

(V.)G.CAMST[cam]

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Cam status.

Use this variable in the PLC program to condition the cam operations to the right status and consequently prevent execution errors. The operations of the cam from the part-program stay in execution, without issuing an error message, waiting for the right cam status.

Syntax.

·cam· Cam number.

V.G.CAMST[2]	Cam ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	**CAM_NULL** The cam does not exist. The cam is not defined in the machine parameters and there is no file cam available, the CNC or the PLC have not selected it (SELECT) or they have deselected it (DESELECT). • This state only allows loading one file cam from the CNC or PLC. • This status does not allow activating the cam, either from the CNC or from the PLC.
1	**CAM_LOADING** The CNC or the PLC is loading the file that defines the cam. • This state only allows activating the cam from the CNC, where the CNC will remain in execution until the cam reaches the CAM_READY state. The cam cannot be activated from the PLC.
2	**CAM_READY** Cam ready. The CNC has validated the cam defined in the machine parameters or is done loading the cam from a file. • This state allows any cam operation, either from the CNC or from the PLC.
3	**CAM_START** The cam is activated, but it has not entered the range of the master axis yet. In this situation, if the CNC or the PLC cancels the cam, the cam will not reach the CAM_READY state as long as it does not go into and out of the range of the master axis at least once. • This state allows canceling the cam, either from the CNC or from the PLC. • This state allows deselecting the cam, but only from the CNC.
4	**CAM_RUNNING_OUTSIDE** The cam is active and in execution out of the range of the master axis. In this state, the cam has been inside the range of the master axis at least once. If the CNC or the PLC cancel the cam, the cam will go into CAM_READY state. • This state allows canceling the cam, either from the CNC or from the PLC. • This state allows deselecting the cam, but only from the CNC.
5	**CAM_RUNNING_INSIDE** The cam is active and in execution inside the range of the master axis. If the CNC or the PLC cancel the cam, the cam will go into CAM_FINISH state. • This state allows canceling the cam, either from the CNC or from the PLC. • This state allows deselecting the cam, but only from the CNC.
6	**CAM_FINISH** The CNC or the PLC has canceled the cam, but it keeps executing until the cam goes out of the range of the master axis. This situation occurs when the CNC or the PLC cancel a cam that was in CAM_RUNNING_INSIDE state; if they cancel a cam that was in CAM_RUNNING_OUTSIDE state, the cam reaches the CAM_READY state. • This state only allows deselecting the cam, but only from the CNC.

Remarks.

To switch from CAM_READY to CAM_NULL, just deselect the file cam and make sure that the cam is not in the machine parameters. If the dam is defined in the machine parameters, it maintains the CAM_READY state, but the cam data will be defined in the machine parameters.

- To select a file cam, the cam status (state) must be either CAM_NULL or CAM_READY.
- To select a file cam from the PLC, preventing execution errors, the cam status (state) must be CAM_READY. If the cam is activated from the CNC, the channel will remain in execution until the cam reaches the CAM_READY state.
- To deselect a file cam from the PLC, preventing execution errors, the cam status (state) must be CAM_READY. If the cam is deselected from the CNC, the channel will remain in execution until the cam reaches the CAM_READY state.

(V.)G.CAM[cam][index]

Variable that can be read and written via program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Factor to be applied to the range of the slave axis when activating the cam.

Syntax.

·cam· Cam number (between 1 and 16).

·index· Point of the cam. For the cams defined in the machine parameters, a value between 1 and 1024. For the file cams, the line number.

V.G.CAM[2][123]	Cam ·2·. Point or line of file ·123·

10.55 Variables related to the independent axes.

INDEPENDENT INTERPOLATOR.

(V.)[ch].A.INDPOS.xn
(V.)[ch].A.INDPOS.sn
(V.)[ch].SP.INDPOS.sn
Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Theoretical coordinate of the independent interpolator.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.INDPOS.Z	Z axis.
V.A.INDPOS.S	Spindle S.
V.SP.INDPOS.S	Spindle S.
V.SP.INDPOS	Master spindle.
V.A.INDPOS.4	Axis or spindle with logic number ·4·.
V.[2].A.INDPOS.1	Axis with index ·1· in the channel ·2·.
V.SP.INDPOS.2	Spindle with index ·2· in the system.
V.[2].SP.INDPOS.1	Spindle with index ·1· in the channel ·2·.

INDEPENDENT AXIS IN EXECUTION.

(V.)[ch].G.IBUSY
Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

An independent axis is in execution.

Syntax.

·ch· Channel number.

V.[2].G.IBUSY	Channel ·2·.

PERCENTAGE OF FEEDRATE (FEEDRATE OVERRIDE).

(V.)[ch].A.FRO.xn
Variable that can be read and written via program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

Channel [ch]. Feedrate percentage active on the axis.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index in the channel of the axis.

V.A.FRO.Z	Z axis.
V.A.FRO.4	Axis with logic number ·4·.
V.[2].A.FRO.1	Axis with index ·1· in the channel ·2·.

POSITIONING MOVE.

(V.)[ch].A.IPPOS.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.

Coordinate programmed for the independent axis.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index in the channel of the axis.

V.A.IPPOS.Z	Z axis.
V.A.IPPOS.4	Axis with logic number ·4·.
V.[2].A.IPPOS.1	Axis with index ·1· in the channel ·2·.

(V.)[ch].A.ITPOS.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.

Theoretical coordinate of the independent axis.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index in the channel of the axis.

V.A.ITPOS.Z	Z axis.
V.A.ITPOS.4	Axis with logic number ·4·.
V.[2].A.ITPOS.1	Axis with index ·1· in the channel ·2·.

(V.)[ch].A.IPRGF.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.

Feedrate programmed in the independent axis.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index in the channel of the axis.

V.A.IPRGF.Z	Z axis.
V.A.IPRGF.4	Axis with logic number ·4·.
V.[2].A.IPRGF.1	Axis with index ·1· in the channel ·2·.

(V.)[ch].A.IORG.xn

Variable that can be read and written via program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.

Offset for the independent axis.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index in the channel of the axis.

V.A.IORG.Z	Z axis.
V.A.IORG.4	Axis with logic number ·4·.
V.[2].A.IORG.1	Axis with index ·1· in the channel ·2·.

SYNCHRONIZATION MOVEMENT.

(V.)[ch].A.SYNCTOUT.xn
(V.)[ch].A.SYNCTOUT.sn
(V.)[ch].SP.SYNCTOUT.sn

Variable that can be read and written via program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Maximum time to establish synchronism.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.SYNCTOUT.Z	Z axis.
V.A.SYNCTOUT.S	Spindle S.
V.SP.SYNCTOUT.S	Spindle S.
V.SP.SYNCTOUT	Master spindle.
V.A.SYNCTOUT.4	Axis or spindle with logic number ·4·.
V.[2].A.SYNCTOUT.1	Axis with index ·1· in the channel ·2·.
V.SP.SYNCTOUT.2	Spindle with index ·2· in the system.
V.[2].SP.SYNCTOUT.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.SYNCVELxn
(V.)[ch].A.SYNCVEL.sn
(V.)[ch].A.SYNCVEL.sn

Variable that can be read and written via program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Synchronization speed.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.SYNCVEL.Z	Z axis.
V.A.SYNCVEL.S	Spindle S.
V.SP.SYNCVEL.S	Spindle S.
V.SP.SYNCVEL	Master spindle.
V.A.SYNCVEL.4	Axis or spindle with logic number ·4·.
V.[2].A.SYNCVEL.1	Axis with index ·1· in the channel ·2·.
V.SP.SYNCVEL.2	Spindle with index ·2· in the system.
V.[2].SP.SYNCVEL.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.SYNCPOSW.xn
(V.)[ch].A.SYNCPOSW.sn
(V.)[ch].SP.SYNCPOSW.sn

Variable that can be read and written via program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Maximum position difference to start correcting it.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.SYNCPOSW.Z	Z axis.
V.A.SYNCPOSW.S	Spindle S.
V.SP.SYNCPOSW.S	Spindle S.
V.SP.SYNCPOSW	Master spindle.
V.A.SYNCPOSW.4	Axis or spindle with logic number ·4·.
V.[2].A.SYNCPOSW.1	Axis with index ·1· in the channel ·2·.
V.SP.SYNCPOSW.2	Spindle with index ·2· in the system.
V.[2].SP.SYNCPOSW.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.SYNCVELW.xn
(V.)[ch].A.SYNCVELW.sn
(V.)[ch].A.SYNCVELW.sn

Variable that can be read and written via program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Maximum velocity difference to start correcting it.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.SYNCVELW.Z	Z axis.
V.A.SYNCVELW.S	Spindle S.
V.SP.SYNCVELW.S	Spindle S.
V.SP.SYNCVELW	Master spindle.
V.A.SYNCVELW.4	Axis or spindle with logic number ·4·.
V.[2].A.SYNCVELW.1	Axis with index ·1· in the channel ·2·.
V.SP.SYNCVELW.2	Spindle with index ·2· in the system.
V.[2].SP.SYNCVELW.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.SYNCPOSOFF.xn
(V.)[ch].A.SYNCPOSOFF.sn
(V.)[ch].SP.SYNCPOSOFF.sn

Variable that can be read and written via program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Position offset for synchronization.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.SYNCPOSOFF.Z	Z axis.
V.A.SYNCPOSOFF.S	Spindle S.
V.SP.SYNCPOSOFF.S	Spindle S.
V.SP.SYNCPOSOFF	Master spindle.
V.A.SYNCPOSOFF.4	Axis or spindle with logic number ·4·.
V.[2].A.SYNCPOSOFF.1	Axis with index ·1· in the channel ·2·.
V.SP.SYNCPOSOFF.2	Spindle with index ·2· in the system.
V.[2].SP.SYNCPOSOFF.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.SYNCVELOFF.xn
(V.)[ch].A.SYNCVELOFF.sn
(V.)[ch].SP.SYNCVELOFF.sn

Variable that can be read and written via program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Velocity offset for synchronization.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.SYNCVELOFF.Z	Z axis.
V.A.SYNCVELOFF.S	Spindle S.
V.SP.SYNCVELOFF.S	Spindle S.
V.SP.SYNCVELOFF	Master spindle.
V.A.SYNCVELOFF.4	Axis or spindle with logic number ·4·.
V.[2].A.SYNCVELOFF.1	Axis with index ·1· in the channel ·2·.
V.SP.SYNCVELOFF.2	Spindle with index ·2· in the system.
V.[2].SP.SYNCVELOFF.1	Spindle with index ·1· in the channel ·2·.

(V.)[ch].A.GEARADJ.xn
(V.)[ch].A.GEARADJ.sn
(V.)[ch].SP.GEARADJ.sn

Variable that can be read and written via program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.

Fine adjustment of the gear ratio during synchronization.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.GEARADJ.Z	Z axis.
V.A.GEARADJ.S	Spindle S.
V.SP.GEARADJ.S	Spindle S.
V.SP.GEARADJ	Master spindle.
V.A.GEARADJ.4	Axis or spindle with logic number ·4·.
V.[2].A.GEARADJ.1	Axis with index ·1· in the channel ·2·.
V.SP.GEARADJ.2	Spindle with index ·2· in the system.
V.[2].SP.GEARADJ.1	Spindle with index ·1· in the channel ·2·.

Remarks.

The reading from the PLC will be given in hundredths (x100); i.e. if the parameter value is ·10·, the PLC reading will return a value of ·1000·.

(V.)[ch].A.SYNCERR.xn
(V.)[ch].A.SYNCERR.sn
(V.)[ch].SP.SYNCERR.sn

Variable that can be read and written via program, PLC and interface.
Variable valid for rotary and linear axes as well as for spindles.
The variable returns the execution value; reading it interrupts block preparation.

Synchronization error.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

V.A.SYNCERR.Z	Z axis.
V.A.SYNCERR.S	Spindle S.
V.SP.SYNCERR.S	Spindle S.
V.SP.SYNCERR	Master spindle.
V.A.SYNCERR.4	Axis or spindle with logic number ·4·.
V.[2].A.SYNCERR.1	Axis with index ·1· in the channel ·2·.
V.SP.SYNCERR.2	Spindle with index ·2· in the system.
V.[2].SP.SYNCERR.1	Spindle with index ·1· in the channel ·2·.

COORDINATE LATCHING WITH THE HELP OF A PROBE OR A DIGITAL INPUT.

(V.)[ch].A.LATCH1.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.

Machine coordinate obtained from latching probe 1 on the ·xn· axis.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.A.LATCH1.Z	Z axis.
V.A.LATCH1.4	Axis with logic number ·4·.
V.[2].A.LATCH1.1	Axis with index ·1· in the channel ·2·.

(V.)[ch].A.LATCH2.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for rotary and linear axes.
The variable returns the execution value; reading it interrupts block preparation.

Machine coordinate obtained from latching probe 2 on the ·xn· axis.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.A.LATCH2.Z	Z axis.
V.A.LATCH2.4	Axis with logic number ·4·.
V.[2].A.LATCH2.1	Axis with index ·1· in the channel ·2·.

10.56 Variables associated with the virtual tool axis.

(V.)[ch].G.VIRTAXIS

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Logic number of the virtual tool axis.

Syntax.

·ch· Channel number.

V.[2].G.VIRTAXIS	Channel ·2·.

(V.)[ch].G.VIRTAXST

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel [ch]. Status of the virtual tool axis: activated/deactivated.

Syntax.

·ch· Channel number.

V.[2].G.VIRTAXST	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Virtual tool axis deactivated.
1	Virtual tool axis activated.

(V.)[ch].A.VIRTAXOF.xn

Variable that can only be read from the program, PLC and interface.
Variable valid for linear axes.
The variable returns the execution value; reading it interrupts block preparation.

Distance traveled by the axis, due to the movement of the virtual tool axis.

Syntax.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

V.A.VIRTAXOF.Z	Z axis.
V.A.VIRTAXOF.4	Axis or spindle with logic number ·4·.
V.[2].A.VIRTAXOF.1	Axis with index ·1· in the channel ·2·.

10.57 Variables defined by the user.

(V.)P.name

Variable that can be read and written from the program.
The variable returns the value of block preparation.

User local variables.

Syntax.

·name· Name of the variable.

V.P.myvar	Variable with the name of "myvar".

Remarks.

These variables maintain their value in local and global subroutines called upon from the program.

(V.)S.name

Variable that can be read and written from the program.
The variable returns the value of block preparation.

User global variables.

Syntax.

·name· Name of the variable.

V.S.myvar	Variable with the name of "myvar".

Remarks.

These variables maintain their value between programs and after a reset. Variables are deleted when the CNC is turned off and they can also be deleted from the part-program using the #DELETE instruction.

```
#DELETE V.S.myvar1 V.S.myvar2
```

10.58 General variables of the CNC.

HARDWARE TYPE.

(V.)G.HARDTYPE

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Hardware type.

V.G.HARDTYPE

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Simulator
2	ICU central unit.

EMERGENCY RELAY.

(V.)G.ERELAYST

Variable that can only be read from the program, PLC (asynchronous) and interface.
The variable returns the execution value; reading it interrupts block preparation.

Status of the emergency relay.

V.G.ERELAYST

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	Open relay.
1	Closed relay.

SOFTWARE VERSION.

(V.)G.VERSION

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

CNC version and release number.

V.G.VERSION

Values of the variable.

This variable returns a decimal value; to understand the meaning of the variables, resource to their hexadecimal format. The high portion of this variable contains the software version and its low portion the release.

V.G.VERSION	Hexadecimal value.	Meaning.
778	$30A	Version 3.10 Software version: $300 Software release: $0A

DATE, TIME AND AMOUNT OF TIME OF TURNED ON.

(V.)G.DATE

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Date in year-month-day format.

V.G.DATE

Values of the variable.

If the date is "April 25th 1999", the returned value will be 990425.

Remarks.

This variable updates its value every hour.

(V.)G.TIME

Variable that can only be read from the program, PLC (asynchronous) and interface.
The variable returns the execution value; reading it interrupts block preparation.

Time in hours-minutes-seconds format.

V.G.TIME

Values of the variable.

If the time is "18hr 22min 34s", the returned value will be 182234.

(V.)G.CLOCK

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Seconds since the CNC was turned on.

V.G.TIME

INFORMATION ABOUT THE CHANNELS.

(V.)[ch].G.CNCHANNEL

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Channel number.

Syntax.

·ch· Channel number.

V.[2].G.CNCHANNEL	Channel ·2·.

(V.)G.FOCUSCHANNEL

Variable that can be read and written from the PLC and from the interface and read from the program.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

Channel with active focus.

V.G.FOCUSCHANNEL

KEY SIMULATION.

(V.)G.KEY

Variable that can be read and written from the PLC and read from the program and from the interface. The variable returns the execution value; reading it interrupts block preparation.

Code of the last key accepted by the CNC.

V.G.KEY

Remarks.

To read the last key that has been accepted by the CNC or simulate the keyboard from the PLC by writing in it the code of the desired key.

10.59 Variables related to CNC status.

CNC STATUS.

(V.)[ch].G.STATUS

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

CNC status (brief).

Syntax.

·ch· Channel number.

V.[2].G.STATUS	Channel ·2·.

Values of the variable.

This variable returns a decimal value; to understand the meaning of the variables, resource to their hexadecimal format. The list of codes is the following.

Decimal.	Hexadecimal.	Meaning.
0	$0	Not Ready
1	$1	Ready.
2	$2	In execution.
4	$4	Interrupted.
8	$8	In error.

(V.)[ch].G.FULLSTATUS

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

CNC status (detailed).

Syntax.

·ch· Channel number.

V.[2].G.FULLSTATUS	Channel ·2·.

Values of the variable.

This variable returns a decimal value; to understand the meaning of the variables, resource to their hexadecimal format. The high portion of this variable contains the information of the STATUS variable and its low portion provides the substate of the CNC; FULLSATUS = (STATUS)(substate). Thus, if the variable returns a value of 514, it means the following.

Decimal.	Hexadecimal.	Status.	Substate.
514	$0202	$0200 In execution.	$02 In MDI.

The list of codes for the low portion of FULLSTATUS is: In reset state, the low portion of FULLSTATUS is $0, in jog mode is $1, in simulation mode is $D and so on.

Decimal.	Hexadecimal.	Meaning.
0	$0	In Reset
1	$1	In Jog mode
2	$2	In MDI mode
3	$3	In program
4	$4	Stopped by an M0
5	$5	Stopped with CYCLE STOP
6	$6	Stopped in Single block mode
9	$9	Syntax checks.
10	$A	Block search (without moving the axes)
11	$B	Block search finished. In stand by
12	$C	Execution time estimate.
13	$D	In simulation.

Example of some values of the FULLSTATUS variable.

Decimal.	Hexadecimal.	Status.	Substate.
521	$209	$200 In execution.	$09 Syntax checks.
522	$20A	$200 In execution.	$0A Block search.
524	$20C	$200 In execution.	$0C Execution time estimate.
525	$20D	$200 In execution.	$0D In simulation.
515	$203	$200 In execution.	$03 In program.

SELECTED AXES.

(V.)[ch].G.SELECTEDAXIS

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Axes selected for home search, repositioning, coordinate preset or movement to a coordinate.

Syntax.

·ch· Channel number.

```
V.[2].G.SELECTEDAXIS                Channel ·2·.
```

Values of the variable.

The variable returns a 32-bit value where each bit represents one axis; the least significant bit corresponds to the axis with the lowest logic number. Each bit indicates whether the axis is selected (bit = 1) or not (bit = 0).

DETAILED CNC STATUS IN JOG MODE.

(V.)[ch].G.CNCMANSTATUS

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

Detailed CNC status in jog mode.

Syntax.

·ch· Channel number.

V.[2].G.CNCMANSTATUS	Channel ·2·.

Values of the variable.

This variable returns a hexadecimal value with the following meaning.

Value.	Meaning.
$1	Jog mode.
$2	Jog or handwheel movements.
$4	Manual tool calibration.
$8	Part centering (mill model).
$10	Home search with subroutine. The variable takes this value when pressing the [ZERO] key and maintains it until the CNC returns to the READY state after the home search.
$20	Home search on an axis. The variable takes this value when pressing the [ZERO] key with an axis selected and maintains it until the CNC returns to the READY state after the home search.
$40	"T" field selected. The variable removes the bit when the CNC returns to the READY state after pressing [START].
$80	"F" field selected. The variable removes the bit when the CNC returns to the READY state after pressing [START].
$100	"S" field selected. The variable removes the bit when the CNC returns to the READY state after pressing [START].
$200	Axis ready for coordinate preset or for movement to a coordinate. Axis selected and coordinate set, waiting for [ENTER] or [START].
$10000000	MDI/MDA mode.
$20000000	CNCEX.

Example of some values of the variable.

CNC status.	Variables.
Tool calibration screen is active.	V.G.CNCMANSTATUS = $5
An axis has been selected for home search, but [START] has not been pressed.	V.G.CNCMANSTATUS = $21 V.G.STATUS = $1 (READY)
A home search is in progress on an axis.	V.G.CNCMANSTATUS = $21 V.G.STATUS = $2 (IN EXECUTION)
Home search on an axis has been interrupted.	V.G.CNCMANSTATUS = $21 V.G.STATUS = $4 (INTERRUPTED)
The CNC is executing a block in MDI/MDA from jog mode.	V.G.CNCMANSTATUS=$10000001 V.G.STATUS = $2 (IN EXECUTION)
The execution of a block in MDI/MDA from jog mode has been interrupted.	V.G.CNCMANSTATUS=$10000001 V.G.STATUS = $4 (INTERRUPTED)

DETAILED CNC STATUS IN AUTOMATIC MODE.

(V.)[ch].G.CNCAUTSTATUS

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

Detailed CNC status in automatic mode.

Syntax.

·ch· Channel number.

V.[2].G.CNCAUTSTATUS	Channel ·2·.

Values of the variable.

This variable returns a hexadecimal value with the following meaning.

Value.	Meaning.
$1	Automatic mode or MDI/MDA mode over the automatic mode.
$2	First block selected.
$4	Stop condition selected.
$8	The option to execute the blocks of the program independently (EXBLK softkey) is active.
$10	Block search is selected.
$20	The M H F S function display is selected (during block search or tool inspection).
$40	The M H F S function selection is active (during tool inspection).
$80	Tool inspection is selected.
$100	The "cancel and resume" option is active.
$200	Axis repositioning is selected (during block search or tool inspection).
$400	An axis has been selected for repositioning.
$800	The CNC is executing an interruption subroutine.
$1000	Simulated execution is active. Theoretical travel.
$2000	Simulated execution is active. G functions.
$4000	Simulated execution is active. GMST functions.
$8000	Simulated execution is active. Main plane.
$10000	Simulated execution is active. Rapid.
$20000	Simulated execution is active. Rapid S=0.
$40000	The cycle editor is executing a cycle.
$80000	The cycle editor is simulating a cycle.
$10000000	MDI/MDA mode active over the automatic mode.
$20000000	The PLC is executing a CNCEX command.

Example of some values of the variable.

CNC status.	Variables.
There is a program in execution.	V.G.CNCAUTSTATUS = $1 V.G.STATUS = $2 (IN EXECUTION)
The execution has been interrupted and tool inspection is active.	V.G.CNCAUTSTATUS = $81 V.G.STATUS = $4 (INTERRUPTED)
Tool inspection is active and there is a jog movement.	V.G.CNCAUTSTATUS = $81 V.G.CNCMANSTATUS = $2 V.G.STATUS = $2 (IN EXECUTION)

CNC status.	Variables.
Tool inspection is active and an axis has been selected for repositioning.	V.G.CNCAUTSTATUS = $681 V.G.STATUS = $4 (INTERRUPTED)
The CNC is executing a block in MDI/MDA from automatic mode.	V.G.CNCAUTSTATUS = $10000001 V.G.STATUS = $2 (IN EXECUTION)
The execution of a block in MDI/MDA from automatic mode has been interrupted.	V.G.CNCAUTSTATUS=$10000001 V.G.STATUS = $4 (INTERRUPTED)

10.60 Variables associated with the part-program being executed.

PART-PROGRAM INFORMATION.

(V.)[ch].G.FILENAME

Variable to be read via interface.
Report variable (to be used from the scripts).

Name of the program in execution.

Syntax.

·ch· Channel number.

[2].G.FILENAME	Channel ·2·.

(V.)[ch].G.PRGPATH

Variable to be read via interface.
Report variable (to be used from the scripts).

Path of the program in execution.

Syntax.

·ch· Channel number.

[2].G.PRGPATH	Channel ·2·.

(V.)[ch].G.FILEOFFSET

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Number of characters up to the line being executed or prepared.

Syntax.

·ch· Channel number.

V.[2].G.FILEOFFSET	Channel ·2·.

Remarks.

The variable returns the value of execution or preparation as follows. Reading the variable from the PLC or from the interface returns the value of the program being executed; reading it from the part-program or MDI returns the program being prepared.

(V.)[ch].G.BLKN

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Last block executed (number).

If no label has been executed, the variable will return a -1 value.

Syntax.

·ch· Channel number.

V.[2].G.BLKN	Channel ·2·.

(V.)[ch].G.LINEN

Variable that can only be read from the program, PLC and interface.
The variable returns the value of block preparation.

Number of program blocks executed or prepared.

Syntax.

·ch· Channel number.

V.[2].G.LINEN	Channel ·2·.

Remarks.

The variable returns the value of execution or preparation as follows. Reading the variable from the PLC or from the interface returns the number of blocks executed; reading it from the part-program or MDI returns the number of blocks prepared.

INFORMATION ON PROGRAM EXECUTION.

(V.)[ch].G.FIRST

Variable that can be read and written from the interface and read from the program and from the PLC.
The variable returns the execution value; reading it interrupts block preparation.

First time a program is executed.

Syntax.

·ch· Channel number.

V.[2].G.FIRST	Channel ·2·.

Values of the variable.

This variable returns one of the following values.

Value.	Meaning.
0	No.
1	Yes.

Remarks.

First time execution means every time a new program is selected. This variable is initialized when changing the program being executed in the channel, even with the #EXEC instruction.

When the instruction #EXEC is involved in a program, the variable behaves as follows:

1. The variable is initialized when selecting and executing the program.
2. When executing the #EXEC instruction, the variable is re-initialized because the program in execution has changed.
3. If then, this program is executed again, the program in execution changes again and the variable is updated.

(V.)[ch].G.PARTC

Variable that can be read and written via program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

Parts counter.

Syntax.

·ch· Channel number.

V.[2].G.PARTC	Channel ·2·.

Remarks.

This variable is initialized when changing the program being executed and it increments its value every time an M02 or M30 is executed. This variable is initialized when changing the program being executed in the channel, even with the #EXEC instruction.

When the instruction #EXEC is involved in a program, the variable behaves as follows:

1 The variable is initialized when selecting and executing the program.

2 When executing the #EXEC instruction, the variable is re-initialized because the program in execution has changed.

3 If then, this program is executed again, the program in execution changes again and the variable is updated.

In this case, to keep track of how many times the program has been executed, it is recommended to use an arithmetic parameter at the end of the program like a counter.

(V.)[ch].G.CYTIME

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.

Part-program execution time (in hundredths of a second).

Syntax.

·ch· Channel number.

V.[2].G.CYTIME	Channel ·2·.

Remarks.

This variable is set to 0 at every new execution even of the same program. It does not measure the time that execution has been stopped.

EXECUTING OPTIONS; SINGLE BLOCK, RAPID, ETC..

(V.)[ch].G.SBOUT

Variable that can only be read from the program, PLC and interface.
The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

"Single block" function activated.

Syntax.

·ch· Channel number.

V.[2].G.SBOUT	Channel ·2·.

Remarks.

The single block function may be activated or canceled from the keyboard or from the PLC (SBLOCK mark). To activate it, just set one of them high (=1), but to cancel it both must be low (=0).

(V.)[ch].G.SBLOCK

Variable that can be read and written from the interface and read from the program and from the PLC.
The variable returns the execution value; reading it interrupts block preparation.

"Single block" function requested via keyboard.

Syntax.

·ch· Channel number.

V.[2].G.SBLOCK	Channel ·2·.

(V.)[ch].G.BLKSKIP

Variable that can be read and written from the interface and read from the program and from the PLC. The variable returns the execution value; reading it interrupts block preparation.

Block skip function (\ ) activated.

Syntax.

·ch· Channel number.

V.[2].G.BLKSKIP	Channel ·2·.

Remarks.

The "block skip" function is selected from the PLC using the BLKSKIP1 mark.

(V.)[ch].G.M01STOP

Variable that can be read and written from the interface and read from the program and from the PLC. The variable returns the execution value; reading it interrupts block preparation.

Conditional stop function (M01) activated.

Syntax.

·ch· Channel number.

V.[2].G.M01STOP	Channel ·2·.

Remarks.

The "conditional stop" function is activated from the PLC using the M01STOP mark.

(V.)[ch].G.RAPID

Variable that can be read and written from the interface and read from the program and from the PLC. The variable returns the execution value; reading it interrupts block preparation.
Report variable (to be used from the scripts).

"Rapid traverse" function activated.

Syntax.

·ch· Channel number.

V.[2].G.RAPID	Channel ·2·.

Remarks.

The "rapid traverse" function is selected from the PLC using the MANRAPID mark.

10.61 Interface related variables.

(V.)[ch].E.PROGSELECT

Variable that can be read and written via program, PLC and interface.
Report variable (to be used from the scripts).

Name of the program selected in automatic.

This variable can only be written with the value of ·0· to eliminate the program selected in automatic mode. When simulating the program, writing this variable has no effect.

Syntax.

·ch· Channel number.

[2].E.PROGSELECT	Channel ·2·.

(V.)E.COMPONENTNAME

Variable to be read via interface.
Report variable (to be used from the scripts).

Component name.

G.COMPONENTNAME

(V.)[ch].E.MMCMODE

Variable to be read via interface.
Report variable (to be used from the scripts).

Syntax.

·ch· Channel number.

[2].E.MMCMODE	Channel ·2·.

(V.)E.NERRORS

Variable to be read via interface.
Report variable (to be used from the scripts).

Number of errors at the CNC.

G.NERRORS

(V.)E.ERRORPRI

Variable to be read via interface.
Report variable (to be used from the scripts).

Error priority.

G.ERRORPRI

(V.)[ch].E.MANUALMODE

Variable to be read via interface.
Report variable (to be used from the scripts).

Exclusive jog mode active.

Syntax.

·ch· Channel number.

[2].E.MANUALMODE	Channel ·2·.

(V.)E.CALCRESUL

Variable to be read via interface.
Report variable (to be used from the scripts).

Result of the calculator.

G.CALCRESULT

(V.)E.NUMJOGPANELS

Variable to be read via interface.

Number of jog panels.

G.NUMJOGPANELS

(V.)E.HELPDATA

Variable that can be read and written from the interface.
Report variable (to be used from the scripts).

Mark of the help file corresponding to the help page.

G.HELPDATA

(V.)E.CNCMSG

Variable to be read via interface.
Report variable (to be used from the scripts).

Error text.

G.CNCMSG

(V.)[ch].E.PATHSELECT

Variable to be read via interface.
Report variable (to be used from the scripts).

Path of the program selected in automatic.

Syntax.

·ch· Channel number.

[2].E.PATHSELECT	Channel ·2·.

(V.)[ch].E.CMPNTNUMBER

Variable to be read via interface.
Report variable (to be used from the scripts).

Component where the channel is.

Syntax.

·ch· Channel number.

[2].E.CMPNTNUMBER	Channel ·2·.

(V.)[ch].E.PAGENUMBER

Variable to be read via interface.
Report variable (to be used from the scripts).

Page where the channel is.

Syntax.

·ch· Channel number.

[2].E.PAGENUMBER	Channel ·2·.

(V.)[ch].E.GRAPHTYPE

Variable to be read via interface.
Report variable (to be used from the scripts).

Graphics configuration selected in parameter GRAPHTYPE.

Syntax.

·ch· Channel number.

[2].E.GRAPHTYPE	Channel ·2·.

10.

CNC VARIABLES.

Interface related variables.

FAGOR

CNC 8060
CNC 8065

(REF: 1402)





FAGOR

CNC 8060
CNC 8065

(REF: 1402)