CNCelite

8058/8060
8065/8070

Installation manual.

Ref: 2210

FAGOR
AUTOMATION

TRANSLATION OF THE ORIGINAL MANUAL

This manual is a translation of the original manual. This manual, as well as the documents derived from it, have been drafted in Spanish. In the event of any contradictions between the document in Spanish and its translations, the wording in the Spanish version shall prevail. The original manual will be labeled with the text "ORIGINAL MANUAL".

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 the following 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.

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.

DUAL-USE PRODUCTS

Products manufactured by FAGOR AUTOMATION since April 1st 2014 will include "-MDU" in their identification if they are included on the list of dual-use products according to regulation UE 428/2009 and require an export license depending on destination.

FAGOR AUTOMATION

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, hence no absolute match is guaranteed. However, the contents of this document are regularly checked and updated implementing the necessary corrections in a later 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

[4] CONCEPTS.

CHAPTER 7 GENERAL SETUP.

CHAPTER 8 FEEDBACKS.

CHAPTER 9 HANDWHEELS.

CHAPTER 10 CONFIGURING AN AXIS.

CHAPTER 11 TANDEM AXIS (SPINDLE).

CHAPTER 12 VOLUMETRIC COMPENSATION.

CHAPTER 13 CONFIGURING THE HSC MODE.

CHAPTER 14 TOOL AND MAGAZINE MANAGEMENT.

ABOUT THIS MANUAL.

Title.

Installation manual.

Models.

CNCelite
8058 8060
8065 8070

Type of documentation.

OEM manual. This manual describes the CNC setup; machine parameters, PLC, etc.

Remarks.

Always use the manual reference associated with the software version or a later manual reference. You can download the latest manual reference from the download section on our website.

Limitations.

The availability of some of the features described in this manual are dependent on the acquired software options. Moreover, the machine manufacturer (OEM) customizes the CNC performance of each machine using the machine parameters and the PLC. Because of this, the manual may describe features that are not available for the CNC or the machine. Consult the machine manufacturer for the available features.

Electronic document.

man_elite_58_60_65_70_inst.pdf. Manual available from the download section of our website.

Language.

English [EN]. Refer to our website, download area, the languages available for each manual.

Date of publication.

October, 2022

Manual reference

Ref: 2210

Associated version.

v2.21.02

Responsibility exemption.

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

Trademarks.

This manual may contain third-party trademarks or trade names, however, they do not have them associated ® or ™ symbols. 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.

Website / Email.

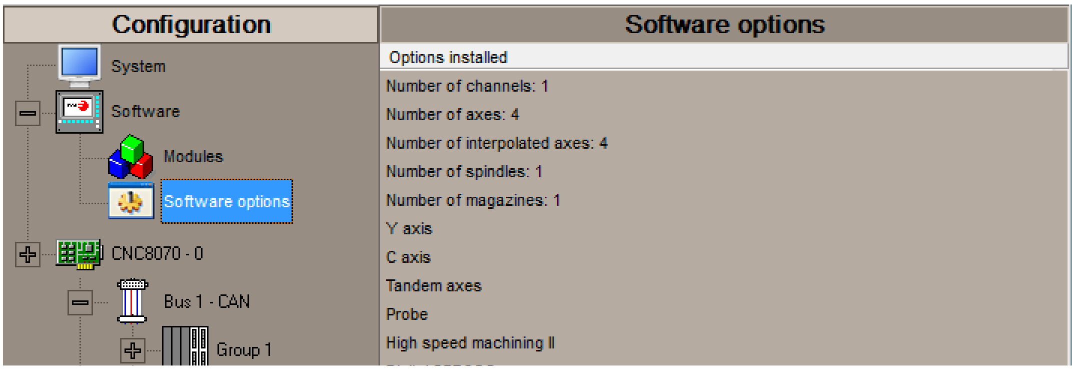
http://www.fagorautomation.com

Email: contact@fagorautomation.es

About the product.

SOFTWARE OPTIONS.

Some of the features described in this manual are dependent on the acquired software options. The active software options for the CNC can be consulted in the diagnostics mode (accessible from the task window by pressing [CTRL] [A]), under software options. Consult Fagor Automation regarding the software options available for your model.



Software option	Description.
SOFT ADDIT AXES	Option to add axes to the default configuration.
SOFT ADDIT SPINDLES	Option to add spindles to the default configuration.
SOFT ADDIT TOOL MAGAZ	Option to add magazines to the default configuration.
SOFT ADDIT CHANNELS	Option to add channels to the default configuration.
SOFT 4 AXES INTERPOLATION LIMIT	Limited to 4 interpolated axes.
SOFT DIGITAL SERCOS	Option for a Sercos digital bus.
SOFT THIRD PARTY DRIVES	Option to use EtherCAT third party drives.
SOFT THIRD PARTY I/Os	Option to use third party I/O modules.
SOFT OPEN SYSTEM	Option for open systems. The CNC is a closed system that offers all the features needed to machine parts. Nevertheless, at times there are some customers who use third-party applications to take measurements, perform statistics or other tasks apart from machining a part. This feature must be active when installing this type of application, even if they are Office files. Once the application has been installed, it is recommended to close the CNC in order to prevent the operators from installing other kinds of applications that could slow the system down and affect the machining operations.

Software option	Description.
SOFT i4.0 CONNECTIVITY PACK	Options for Industry 4.0 connectivity. This option provides various data exchange standards (for example, OPC UA), which allows the CNC (and therefore the machine tool) to be integrated into a data acquisition network or into a MES or SCADA system.
SOFT EDIT/SIMUL	Option to enable edisimu mode (edition and simulation) on the CNC, which can edit, modify and simulate part programs.
SOFT DUAL-PURPOSE (M-T)	Option to enable the dual-purpose machine, which allows milling and turning cycles. On Y-axis lathes, this option allows for pockets, bosses and even irregular pockets with islands to be made during milling cycles. On a C-axis mill, this option allows turning cycles to be used.
SOFT TOOL RADIUS COMP	Option to enable radius compensation. This compensation programs the contour to be machined based on part dimensions without taking into account the dimensions of the tool that will be used later on. This avoids having to calculate and define the tool paths based on the tool radius.
SOFT PROFILE EDITOR	Option to enable the profile editor in edisimu mode and in the cycle editor. This editor can graphically, and in a guided way, define rectangular, circular profiles or any profile made up of straight and circular sections an it can also import dxf files. After defining the profile, the CNC generates the required blocks and add them to the program.
SOFT HD GRAPHICS In a multi-channel system, this feature requires the MP-PLUS (83700201) processor.	High definition solid 3D graphics for the execution and simulation of part-programs and canned cycles of the editor. During machining, the HD graphics display, in real time, the tool removing the material from the part, allowing the condition of the part to be seen at all times. These graphics are required for the collision control (FCAS).
SOFT IIP CONVERSATIONAL	The IIP (Interactive Icon-based Pages) mode, or conversational mode, works with the CNC in a graphical and guided way based on predefined cycles. There is no need to work with part programs, have any previous programming knowledge or be familiar with Fagor CNCs. Working in conversational mode is easier than in ISO mode, as it ensures proper data entry and minimizes the number of operations to be defined.
SOFT RTCP This feature requires the MP-PLUS (83700201) processor.	Option to enable dynamic RTCP (Rotating Tool Center Point) required to machine with 4, 5 and 6 axis kinematics; for example, angular and orthogonal spindles, tilting tables, etc. The RTCP orientation of the tool may be changed without modifying the position occupied by the tool tip on the part.
SOFT C AXIS	Option to enable C-axis kinematics and associated canned cycles. The machine parameters of each axis or spindle indicate whether it can operate as a C axis or not. For this reason, it is not necessary to add specific axes to the configuration.
SOFT Y AXIS	Option to enable lathe Y-axis kinematics and associated canned cycles.
SOFT TANDEM AXES	Option to enable tandem axle control. A tandem axis consists of two motors mechanically coupled to each other forming a single transmission system (axis or spindle). 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. When activating this feature, it should be kept in mind that for each tandem axis of the machine, another axis must be added to the entire configuration. For example, on a large 3-axis lathe (X Z and tailstock), if the tailstock is a tandem axis, the final purchase order for the machine must indicate 4 axes.

Software option	Description.
SOFT SYNCHRONISM	Option to enable the synchronization of paired axes and spindles, in speed or position, and through a given ratio.
SOFT KINEMATIC CALIBRATION	Option to enable tool calibration. For the first time, this kinematics calibration allows for the kinematics offsets to be calculated using various approximate data and, also, from time to time to correct any possible deviations caused by day-to-day machining operations.
SOFT 60 HSSA I MACHINING SYSTEM	Option to enable the HSSA-I (High Speed Surface Accuracy) algorithm for high speed machining (HSC). This new HSSA algorithm allows for high speed machining optimization, where higher cutting speeds, smoother contours, a better surface finishing and greater precision are achieved.
SOFT HSSA II MACHINING SYSTEM	Option to enable the HSSA-II (High Speed Surface Accuracy) algorithm for high speed machining (HSC). This new HSSA algorithm allows for high speed machining optimization, where higher cutting speeds, smoother contours, a better surface finishing and greater precision are achieved. The HSSA-II algorithm has the following advantages compared to the HSSA-I algorithm. • Advanced algorithm for point preprocessing in real time. • Extended curvature algorithm with dynamic limitations. Improved acceleration and jerk control. • Greater number of pre-processed points. • Filters to smooth out the dynamic machine behavior.
SOFT TANGENTIAL CONTROL	Option to enable tangential control. "Tangential Control" maintains a rotary axis always in the same orientation with respect to the programmed tool path. The machining path is defined on the axes of the active plane and the CNC maintains the orientation of the rotary axis along the entire tool path.
SOFT PROBE	Option to enable functions G100, G103 and G104 (for probe movements) and probe canned cycles (which help to measure part surfaces and to calibrate tools). For the laser model, it only activates the non-cycle function G100. The CNC may have two probes; usually a tabletop probe to calibrate tools and a measuring probe to measure the part.
SOFT FVC STANDARD **SOFT FVC UP TO 10m3** **SOFT FVC MORE TO 10m3**	Options to enable volumetric compensation. The precision of the parts is limited by the machine manufacturing tolerances, wear, the effect of temperature, etc., especially on 5-axis machines. Volumetric compensation corrects these geometric errors to a larger extent, thus improving the precision of the positioning. The volume to be compensated is defined by a point cloud and for each point the error to be corrected is measured. When mapping the total work volume of the machine, the CNC knows the exact position of the tool at all times. There are 3 options, which depend on the size of the machine. • FVC STANDARD: Compensation for 15625 points (maximum 1000 points per axis). Quick calibration (time), but less precise than the other two, but sufficient for the desired tolerances. • FVC UP TO 10m3: Volume compensation up to 10 m^3. More accurate than FVC STANDARD, but requires a more accurate calibration using a Tracer or Tracker laser). • FVC MORE TO 10m3: Volume compensation greater than 10 m^3. More accurate than FVC STANDARD, but requires a more accurate calibration using a Tracer or Tracker laser.

Software option	Description.
SOFT CONV USER CYCLES	Option to enable user conversational cycles. The user and the OEM can add their own canned cycles (user cycles) using the FGUIM application that comes installed on the CNC. The application offers a guided way to define a new component and its softkey menu without having to be familiar with script languages. User cycles work in a similar way as Fagor canned cycles.
SOFT PROGTL3	Option to enable the ProGTL3 programming language (ISO language extension), allowing profiles to be programmed using a geometric language and without the need to use an external CAD system. This language can program lines and circles where the end point is defined as the intersection of 2 other sections, pockets, ruled surfaces, etc.
SOFT PPTRANS	Option to enable the program translator, which can convert programs written in other languages to Fagor ISO code.
SOFT DMC	Option to enable the DMC (Dynamic Machining Control). DMC adapts the feedrate during machining to maintain the cutting power as close as possible to ideal machining conditions.
SOFT FMC	Option to enable the FMC (Fagor Machining Calculator). The FMC application consists of a database of materials to be machined and machining operations, with an interface to choose suitable cutting conditions for these operations.
SOFT FFC	Option to enable the FFC (Fagor Feed Control). During the execution of a canned cycle of the editor, the FFC function makes it possible to replace the feedrate and speed programmed in the cycle with the active values of the execution, which are acted upon by the feed override and speed override.
SOFT 60/65/70 OPERATING TERMS	Option to enable a temporary user license for the CNC, which is valid until the date set by the OEM. While the license is valid, the CNC will be fully operational (according to the purchased software options).
SOFT FCAS	Option to enable the FCAS (Fagor Collision Avoidance System). The FCAS option, within the system limitations, monitors the automatic, MDI/MDA, manual and tool inspection movements in real time, so as to avoid collisions between the tool and the machine. The FCAS option requires that the HD graphics to be active and that there is a defined a model configuration of the machine adjusted to reality (.xca file), which includes all its moving parts.
SOFT GENERATE ISO CODE	ISO generation converts canned cycles, calls to subroutines, loops, etc. into their equivalent ISO code (G, F, S, etc functions), so the user can modify it and adapt it to his needs (eliminate unwanted movements, etc.). The CNC generates the new ISO code while simulating the program, either from the DISIMU mode or from the conversational mode.
SOFT PWM CONTROL	Option to enable PWM (Pulse - Width Modulation) control on laser machines. This feature is essential for cutting very thick sheets, where the CNC must create a series of PWM pulses to control laser power when drilling the initial point. This function is only available for Sercos bus control systems and must also use one of the two fast digital outputs available from the central unit.
SOFT GAP CONTROL	Option to enable gap control, which makes it possible to set a fixed distance between the laser nozzle and the sheet surface with the use of a sensor. The CNC compensates the difference between the distance measured by the sensor and the programmed distance with additional movements on the axis programmed for the gap.

Software option	Description.
SOFT MANUAL NESTING	Option to enable nesting in the automatic option. Nesting consists of creating a pattern on the sheet material using previously defined figures (in dxf, dwg or parametric files), so as to use most of the sheet as possible. Once the pattern has been defined, the CNC creates a program. During manual nesting, the operator distributes the parts on top of the sheet material.
SOFT AUTO NESTING	Option to enable nesting in the automatic option. Nesting consists of creating a pattern on the sheet material using previously defined figures (in dxf, dwg or parametric files), so as to use most of the sheet as possible. Once the pattern has been defined, the CNC creates a program. During automatic nesting, the application distributes the figures on the sheet material and optimizes the spaces.
SOFT DRILL CYCL OL	Option to enable ISO drilling cycles (G80, G81, G82, G83).

EC DECLARATION OF CONFORMITY, WARRANTY CONDITIONS AND QUALITY CERTIFICATES.

EC-DECLARATION OF CONFORMITY

The declaration of conformity is available from the downloads section of the Fagor Automation corporate website.

https://www.fagorautomation.com/en/downloads/

Type of file: Declaration of conformity.

NRTL CERTIFICATE FOR USA AND CANADA

The quality certificates are available from ·the company· label on the Fagor Automation corporate website.

https://www.fagorautomation.com/en/sections/quality/

WARRANTY TERMS

The sales and warranty conditions are available from the downloads section of the Fagor Automation corporate website.

https://www.fagorautomation.com/en/downloads/

Type of file: General sales - warranty conditions.

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 or material damage originated from not complying with these basic safety rules.

Before start-up, verify that the machine that integrates this CNC meets the 2006/42/EC Directive.

PRECAUTIONS BEFORE CLEANING THE UNIT

Do not get into the inside of the unit.	Only personnel authorized by Fagor Automation may access the interior of this unit.
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 powered.

PRECAUTIONS DURING REPAIRS

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 access the interior of this unit.
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 powered.

PRECAUTIONS AGAINST PERSONAL HARM

Interconnection of modules.	Use the connection cables provided with the unit.
Use proper cables.	To prevent risks, only use cables and Sercos fiber recommended for this unit. To prevent a risk of electrical shock at the central unit, use the proper connector (supplied by Fagor); use a three-prong power cable (one of them being ground).
Avoid electric shocks.	To prevent electrical shock and fire risk, do not apply electrical voltage out of the indicated range.
Ground connection.	In order to avoid electrical discharges, connect the ground terminals of all the modules to the main ground terminal. Also, before connecting the inputs and outputs of this product, make sure that the ground connection has been done. 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.	In order to avoid electrical discharges, always work with a relative humidity (non-condensing).
Do not work in explosive environments.	In order to avoid risks, harm or damages, do not work in explosive environments.

PRECAUTIONS AGAINST DAMAGE TO THE PRODUCT

Work 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, homes, etc.).
Install this unit in the proper place.	It is recommended, whenever possible, to install the CNC away from coolants, chemical product, blows, etc. that could damage it. This unit meets 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 mains as the unit. Nearby portable transmitters (radio-telephones, Ham radio transmitters). Nearby radio / TC transmitters. Nearby arc welding machines. Nearby high voltage lines.
Enclosures.	It is up to the manufacturer to guarantee that the enclosure where the unit has been installed meets all the relevant directives of the European Union.
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, operator panel and the remote modules.
Connecting the power supply to ground.	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.	Maintain the CNC within the recommended temperature range, both when running and not running. See the corresponding chapter in the hardware manual.
Central unit enclosure.	To maintain the right ambient conditions in the enclosure of the central unit, it must meet the requirements indicated by Fagor. See the corresponding chapter in the hardware manual.
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.

SAFETY SYMBOLS

Symbols that may appear in the manual.

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

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

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

Information symbol.
This symbol indicates notes, warnings and advises.

Symbol for additional documentation.
This symbol indicates that there is another document with more detailed and specific information.

Symbols that the product may carry.

	Ground symbol. *This symbol indicates that that point must be under voltage.*
	ESD components. *This symbol identifies the cards as ESD components (sensitive to electrostatic discharges).*

RETURNING CONDITIONS.

Pack it in its original package along with its original packaging material. If you do not have the original packaging material, pack it as follows:

1. Get a cardboard box whose 3 inside dimensions are at least 15 cm (6 inches) larger than those of the unit itself. The cardboard being used to make the box must have a resistance of 170 Kg (375 lb.).
2. Attach a label to the device indicating the owner of the device along with contact information (address, telephone number, email, name of the person to contact, type of device, serial number, etc.). In case of malfunction also indicate symptom and a brief description of the problem.
3. Protect the unit wrapping it up with a roll of polyethylene or with similar material. When sending a central unit with monitor, protect especially the screen.
4. Pad the unit inside the cardboard box with polyurethane foam on all sides.
5. Seal the cardboard box with packaging tape or with 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. Never use air compressed at high pressure to clean the unit because it could cause the accumulation of electrostatic charges that could result in electrostatic shocks.

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 supplied with power. Before handling these connectors (I/O, feedback, etc.), make sure that the unit is not powered.
- Do not get into the inside of the unit. Only personnel authorized by Fagor Automation may access the interior of this unit.

NEW FEATURES.

Manual reference:	Ref: 2210
Date of publication:	October, 2022
Associated software:	v2.21.02

Below is a list of the features added in this software version and the manuals that describe them.

List of features.	Manual
Selection of the keyboard language. - Machine parameter: KEYBOARDLANG	[INST]
The number of available PLC registers increases to 4096.	[INST]
The number of available PLC brands increases to 10,240.	[INST]

[CHN]................. Execution channels.
[INST]................ Installation manual.
[RIOS]................ Módulos remotos (RIO5, RIOW, RIOR, RIOR-E).
[VAR].................. CNC variables.

PART 1.
SOFTWARE INSTALLATION.

THE CNC SOFTWARE.

1

1.1 Introduction.

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.

Fagor supplies the CNC with properly installed software. It is the responsibility of the OEM (machine manufacturer) to configure the CNC for your machine using the machine, PLC and other parameters. The OEM can also customize the appearance of the CNC using the Fguim customization tool (hereinafter, the Fguim), which can modify the pages, components (work modes), add canned cycles, etc. Before using Fguim, ask Fagor Automation for the corresponding documentation.

Folder tree.

The necessary files for the CNC are located in the folder c:\FagorCnc and its relevant subfolders. See *"1.6 Folder tree."* on page 43. The subfolders ..\MTB and ..\Users are of particular importance.

MTB	This folder is for the OEM. 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.
USERS	This folder is for the user. The purpose of this folder is to provide the user with a memory space for storing part-programs, profiles, etc. as they are generated.

1.1.1 Disk access modes and protection.

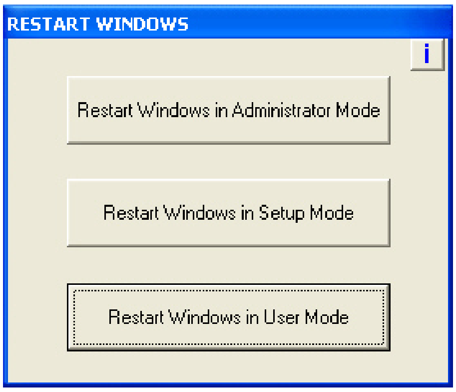
Fagor delivers the unit with the disk that is write-protected except for the folders or files that must be unprotected for the normal operation of the CNC. Any changes made to protected folders or files will be effective until the CNC is restarted, at which time the initial configuration will be restored. The changes made to unprotected areas of the disk will remain.

The device has three preset protection modes. The unit indicates the active protection mode by displaying an icon on the task bar of the operating system, next to the clock. When the CNC is turned on, the status bar displays the active protection mode using icons.

Icon.	Access and protection mode.
	Administrator mode.
	Setup mode.
	User mode.

DiskMonitor application.

The DiskMonitor application toggles between the CNC work and protection modes (administrator, setup and user modes), and it also protects/unprotects folders.



DiskMonitor application.

Accessing the DiskMonitor window.

- While the CNC application is turned off, press the [ALT][D] hotkey or the active protection mode icon on the task bar of the operative system (alongside the clock).
- With the CNC application open, press the [ALT] [D] hotkey or in diagnosis mode ("Apps" softkey).

Changing the protection mode.

This window displays a button for each protection mode. Each of these buttons closes the CNC application and restarts it in the selected mode. To start the CNC in administrator and setup modes, first enter the corresponding password (if the manufacturer has defined it in this way).

Protecting or unprotecting folders.

Administrator mode allows folders to be protected or unprotected. The [i] button at the top right of the DiskMonitor displays the list of unprotected files and folders. With this list visible, pressing [CTRL]+[ALT]+[SHIFT]+[TAB] protects or unprotect any system folder or file that is not essential for the proper control operation.

1.1.2 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 protection mode, it will request the access password.

1.1.3 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 protection mode, it will request the access password.

1.1.4 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. Management of files, pendrives, Ethernet, etc. can only be done 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.

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

The purpose of the following restrictions is for the OEM to deliver its machines with the CNC in user mode (protected 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.

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 Changing the language of the help files.

Fagor supplies the CNC with the help files installed in English. You can replace these files with the help of the SetupHelp_Vxx.xx.exe programme (where Vxx_xx is the version name), available in the download area of the Fagor Automation corporate website. The available languages are Spanish, English, Italian, German, French and Brazilian Portuguese.

- http://www.fagorautomation.com/en/downloads/.
- Type of file: Programs.
- File: CNC Help Language Changer.

Execute the SetupHelp_Vxx.xx.exe file (where Vxx_xx is the version name) and follow the steps indicated by the program. 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. The help files installed at the CNC are located in the following folder.

```
C:\FagorCNC\Fagor\MMC\Help
```

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.

Insert the Fagor pendrive in the USB port and execute the setup_qc_Vxx_xx.exe program (where Vxx_xx is the version number). 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.

Updating remote nodes of the Sercos III bus.

Each time the CNC is powered up, it verifies the versions of the remote nodes detected in the Sercos III 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 Sercos III bus cannot be guaranteed, the CNC will display the corresponding error message.

1.4 Requirements before and after CNC setup.

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

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 SETUP (not safe) 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 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.

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 Installing third-party software.

The CNC you have purchased is an industrial PC equipped with Windows 7 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.

To install software, there must be sufficient space on the CNC's hard drive. 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 that are generated (for example, files) should be saved in an unprotected directory, for example ..\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 Folder tree.

The necessary files for the CNC are located in the folder c:\FagorCnc and its relevant 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 different MTB system folders; MTB_T for lathes, MTB_M for milling machines and MTB_TT for dual turret lathes. The CNC manages these folders as follows. • When the CNC starts up, and depending on the validation code, it will move the appropriate folder from to C:\FagorCnc and rename it as MTB. • When the validation code is changed, the system will return the MTB folder to the ..\Configuration folder and will rename it with its previous name (MTB_M, MTB_T, etc); then, the CNC will move the corresponding folder C:\FagorCnc and rename it as 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.
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 do not affect the contents of the MTB and USERS directories. 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.

i

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.
DRIVERS	This folder contains some necessary drives for proper operation of the CNC.
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)

PART 2.
MACHINE PARAMETERS.

CONNECTIONS AND MACHINE PARAMETERS.

2

This chapter describes the meaning of all the machine parameters of the CNC. The CNC shows the right parameters depending on the model and the active software options.

Machine parameters tables.

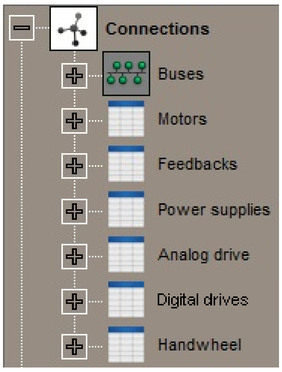
(A) Connection tree and parameters.
(B) Table status icons (validated and saved).
(C) Mnemonic or parameter name.
(D) Value edited by the user (the table shows non-validated values in green).
(E) Saved or validated value of the parameter, selected using the horizontal softkeys.
(F) Parameter units.
(G) Limit values (maximum and minimum) and default value of the parameter.
(H) Parameter description.

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 Connection tree.

The connection tree shows the buses and connections (motors, handwheels, etc.) of the CNC. Within the "Buses" branch, only the simulated bus nodes can be modified. The nodes for the other buses are defined by the CNC depending on the configuration detected at startup. All other connections (motors, feedbacks, etc.) are defined manually. Move the cursor onto one of the branches and press the [INS] key to add a connection. Move the cursor onto one of the connections and press the [DEL] key to delete it.



Buses.

Buses and nodes detected by the CNC at startup; Bus cannot be added manually.

Motors.

List of system motors. Each motor has the following machine parameters. Only motors are configured in this branch. The axis/spindle parameters associates each motor with its axis or spindle. The connection table associates each motor with a physical drive input.

All motors that are active at the same time must be defined (for example, different axes), even if they are the same. The CNC allows the same motor to be associated with several axes or sets, but only if they are not active at the same time (for example, for all sets on the same axis).

Feedbacks.

List of system feedbacks. Each feedback has the following machine parameters. Feedback is only configured in this branch. The axis/spindle parameters associates each feedback with its axis or spindle. The connection table associates each feedback with a physical drive input.

All feedbacks that are active at the same time must be defined (for example, different axes), even if they are the same. The CNC allows the same feedback to be associated with several axes or sets, but only if they are not active at the same time (for example, for all sets on the same axis).

Power supplies.

List of system RPS power supplies. Each RPS power supply has the following machine parameters. Only power supplies are configured in this branch. At startup, the CNC associates each RPS power supply with its node.

Digital drive.

List of BCSD drives in the system. Each BCSD drive has its own parameters. See related documentation.

Analog drive.

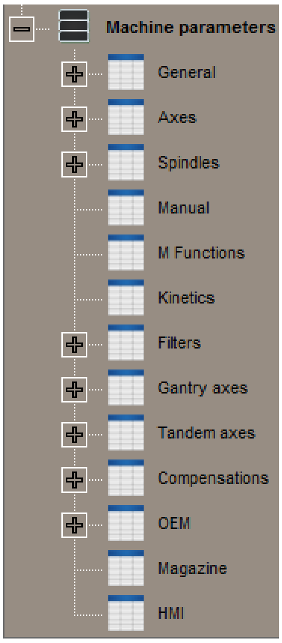
List of system analog drives. Each analog drive has the following machine parameters.

Handwheel.

List of system handwheels. Each handwheel has the following machine parameters. The connection table associates each handwheel with a physical input. Handwheels may be connected to the local feedback inputs of the central unit, a operator panel or a Sercos node feedback input.

2.2 Machine parameter tree.

In order for the machine 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 can include the number of axes, feedrate, accelerations, feedback, magazine type, tool changer, etc. This data is determined by the OEM (machine manufacturer), which is indicated to the CNC through the machine parameters. The machine parameters of the CNC are shown grouped as follows.



General parameters.

These parameters set the general CNC properties, conditions after powering-up, subroutines associated with certain functions, general probe settings, etc.

The CNC can have several execution channels. Move the cursor onto the “General” branch and press the [INS] key to add a channel. Move the cursor onto a channel and press the [DEL] key to delete it.

Axis parameters.

These parameters indicate the properties of the axes, adjustment of the feedrate and gains, travel limits, automatic and manual movement dynamics, probe properties, etc.

Move the cursor onto the “Axis” branch and press the [INS] key to add an axis. Move the cursor onto an axis and press the [DEL] key to delete it.

Each axis can have several parameter sets. Move the cursor onto an axis and press the [INS] key to add a set. Move the cursor onto a set and press the [DEL] key to delete it.

Spindle parameters.

These parameters indicate the spindle properties, adjustment of the speed and gains, movement dynamics, etc.

Move the cursor onto the “Spindle” branch and press the [INS] key to add a spindle. Move the cursor onto a spindle and press the [DEL] key to delete it.

Each spindle can have several parameter sets. Move the cursor onto a spindle and press the [INS] key to add a set. Move the cursor onto a set and press the [DEL] key to delete it.

Jog parameters.

These parameters allow you to customize the jog keys and set the user keys as jog keys.

Parameters of the M function table.

These parameters can define new M functions, set the synchronization type and define the subroutine associated with each function.

Parameters for the Kinematics table.

These parameters can define the type and characteristics of up to six kinematics, orthogonal spindles, tilting tables, C axis, etc.

Filters.

These parameters can configure the axis frequency filters, spindles, HSC mode, etc. Only filters can be configured in this branch; to enable them, they must be assigned to an axis or spindle. One filter can be enabled on several axes.

Move the cursor onto the “Filters” branch and press the [INS] key to add a filter. Move the cursor onto a filter and press the [DEL] key to delete it.

Gantry axes.

Parameters to define and enable the system gantry axes.

Move the cursor onto the “Gantry axes” branch and press the [INS] key to add a gantry axis. Move the cursor onto one of the gantry axes and press the [DEL] key to delete it.

Tandem axes.

Parameters to define and enable the system tandem axes.

Move the cursor onto the “Tandem Axes” branch and press the [INS] key to add a tandem axis. Move the cursor onto one of the tandem axes and press the [DEL] key to delete it.

Compensations.

Parameters to define ballscrew and cross compensation. Compensations are only configured in this branch. To enable them, they must be assigned to an axis or spindle.

- Ballscrew compensation corrects the measurement error caused by the inaccuracy of the spindles being used on each axis.
- Cross compensation is used when an axis suffers position variations due to the movement of another axis.

Move the cursor onto the “Compensation” branch and press the [INS] key to add a compensation. Move the cursor onto a compensation and press the [DEL] key to delete it.

OEM parameters.

These parameters can edit cams and create OEM parameters. OEM parameters are generic parameters for the manufacturer to use as machine parameters.

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.

2.3 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.

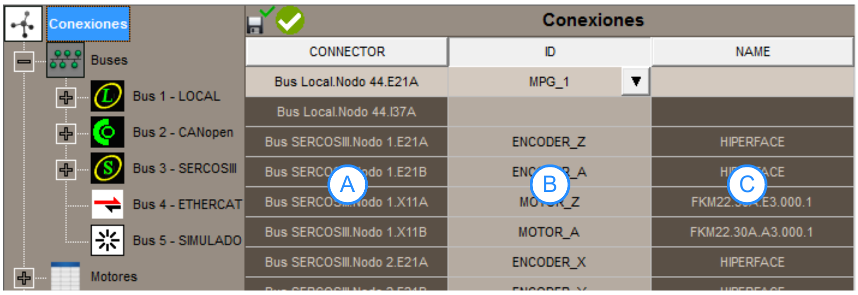
Parameter.	Meaning.
FBACKDIFF	Maximum difference between feedbacks. The CNC uses this parameter for dual-feedback configured axes (parameters SPEEDFBID and POSITIONFBID not equal). This parameter should be set to a value other than ·0·.
DSYNCVELW	Synchronization of axes and spindles. This parameter is 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.
POSLIMIT NEGLIMIT	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	G00 feedrate.

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.

2.4 Connections table.

When the cursor is moved onto the “Connections” or “Buses” branch, the CNC shows the connection table. In this table the CNC must be indicated which device (motor, feedback, etc.) is connected to each connector.

Connections table.



(A) Bus name, node number and connector name.
(B) Device ID (motor, feedback, etc.).
(C) Device Name.

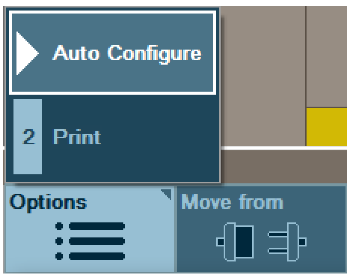
The first column shows the connectors detected by the CNC at startup on each node of each bus. Nodes or connectors can be added manually, except on the simulated bus. For each connector, the table shows the bus and the node to which it belongs.

In the second column, the connected device must be selected. The list of devices is made up of the motors, feedbacks, etc. that are defined in the connection tree. The list only displays unassigned devices.

The third column shows the name or the ID of the connected device.

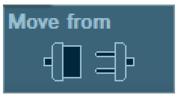
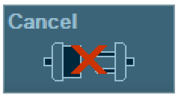
Topology changes during power-up.

If the CNC detects a configuration change at power-up, it will offer a softkey to automatically configure the buses.

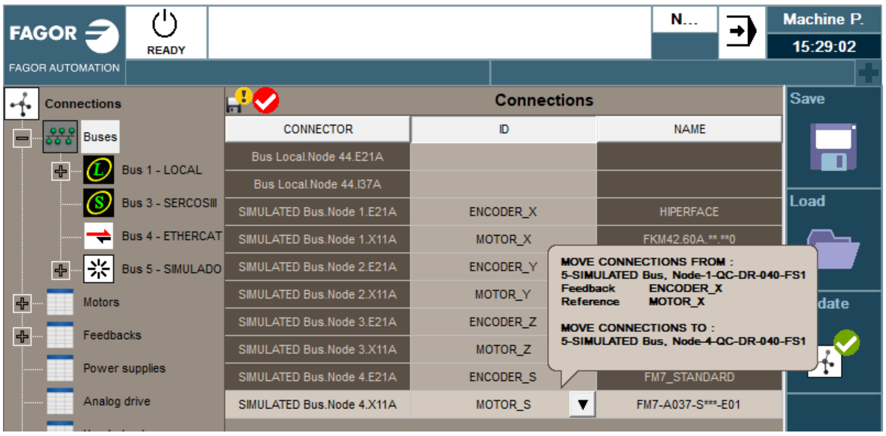


Move connections between nodes.

The softkey menu allows you to move a connection from one node to another. These softkeys allow you to easily move connections between nodes, e.g. from simulated to real nodes. This makes it possible to parameterise a machine in a simulator (simulated nodes for motor and encoder) and copy the configuration to a machine after detecting the topology.

Softkey.	Meaning.
	Select the connection to move it.
	Move a connection to the selected position.
	Cancel the movement of the connection.

The CNC displays information on the selected connection and the destination node, the one where the focus is located.

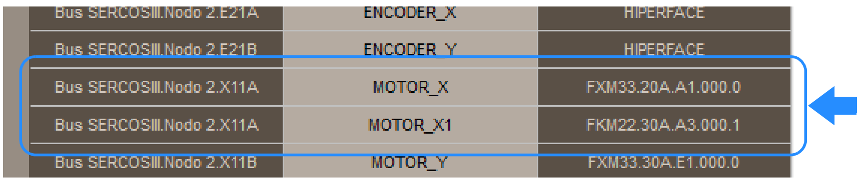


Assigning multiple devices to one connector.

A connector can have several devices connected to it (for example, for multi-axis group management). To do this, the horizontal "Options" softkey can be used to duplicate an existing connector in the table.

Softkey.	Meaning.
Adding a connection to the connector.	This softkey can be used to duplicate the selected connector.
Deleting a connection from the connector.	This softkey can be used to delete a duplicated connector.

Connections table with the duplicated X11A connector on node 2 of the Sercos III bus. Each connector has a different motor connected.

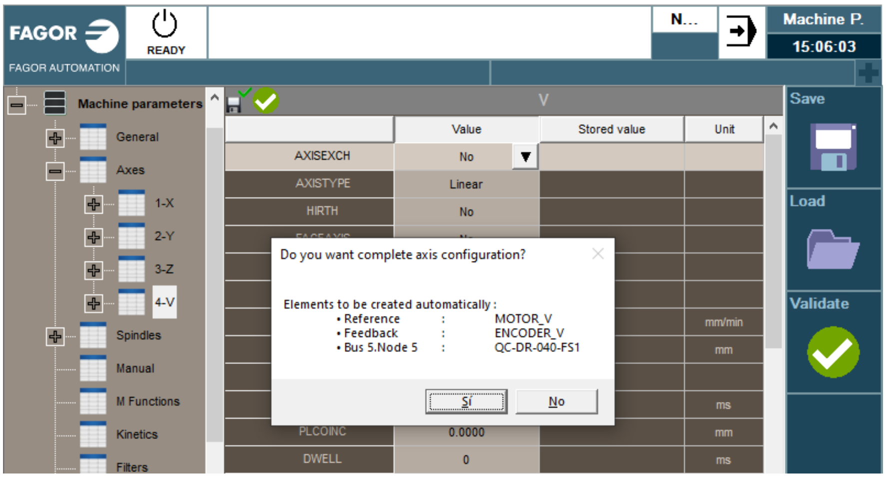


Bus SERCOSIII.Nodo 2.E21A	ENCODER_X	HIPERFACE
Bus SERCOSIII.Nodo 2.E21B	ENCODER_Y	HIPERFACE
Bus SERCOSIII.Nodo 2.X11A	MOTOR_X	FXM33.20A.A1.000.0
Bus SERCOSIII.Nodo 2.X11A	MOTOR_X1	FKM22.30A.A3.000.1
Bus SERCOSIII.Nodo 2.X11B	MOTOR_Y	FXM33.30A.E1.000.0

Add a new axis to the configuration.

By adding an axis to the configuration, the CNC offers the possibility to create all the necessary elements to have a complete basic axis configuration.

- The axis.
- The motor and the encoder.
- A simulated node and the necessary motor and encoder connections.

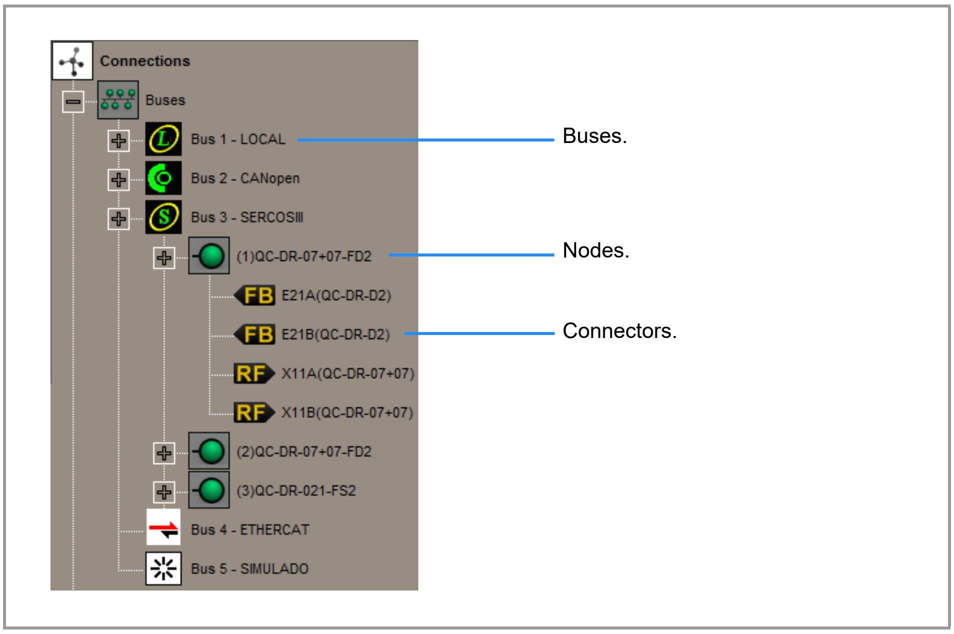


2.5 Connections; buses.

List of buses detected by the CNC at startup, including the simulated bus. Buses cannot be added manually. When the cursor is moved onto the “Connections” or “Buses” branch, the CNC shows the connection table. See *"2.4 Connections table."* on page 55.

- Local Bus. — This bus shows the elements connected to the local inputs and outputs.
- Sercos III bus. — This bus shows the elements connected to the Sercos III bus, drives, RPS power supplies, etc.
- CANopen bus. — This bus shows the elements connected to the CAN bus, operator panels, remote modules, etc.
- EtherCAT bus — This bus shows the elements connected to the EtherCAT bus, operator panels, remote modules, etc.
- Simulated bus. — This bus can be configured by the user and can simulate the system configuration.

Each bus shows the nodes and connectors detected at startup. Nodes or connectors cannot be added manually, except on the simulated bus. For each connector, the name and the device it is connected to are displayed.

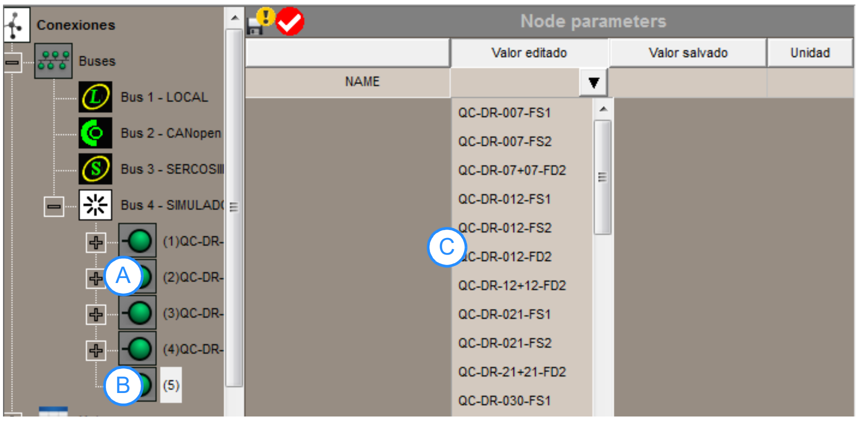


Connector icons and names.

Connection and connector.	Meaning.
FB E21A(QCD-C-S1)	Feedback input.
RF X11A(QCD-B-21)	Motor power output.
mpg MPG_IB(MPG3)	Handwheel input.
AO I37A(IOB-AO1)	Analog output.
SR X11A(QC-RPS-120)	Power input to the main power supply.

2.5.1 Simulated bus configuration.

The simulated bus can be configured, by adding or removing nodes, using the horizontal "Options" softkey or the keyboard. By default, nodes are added empty without any associated device (drive, RPS power supply, etc.). To assign a device to the node, move the cursor onto it and from window on the right select a device from the list. When adding a device, the node will display the connectors associated to it.



(A) Configured nodes.
(B) Unconfigured Node.
(C) Available devices (drives, RPS power supplies, etc.).

Below are the softkeys and keys used to configure the simulated bus.

Softkey.	Key.	Meaning.
Add new.	[INS]	Add a node to the simulated bus.
Clone.	- - -	Clone a simulated bus node.
Delete.	[DEL]	Delete a simulated bus node.

The simulated bus nodes also appear in the connection table. See "2.4 Connections table." on page 55.

2.6 Connections; motors.

2.6.1 Configure a motor.

MOT_NAME
Possible values: Select a device from the list.
Default value: -.
Associated variable: (V.)MPMOT.MOT_NAME.nb

Motor identification and initialization of the parameters linked to the motor. For Fagor motors, the "AUTODETECT" option automatically identifies the connected motor. To add motors from third parties to the list, these motors must be defined in the file motors_OEM.Xml in folder C:\FagorCNC\MTB\Data.

MOT_ENCODER
Possible values: Select a device from the list.
Default value: -.
Associated variable: (V.)MPMOT.MOT_ENCODER.nb

Type of motor encoder; it must match that of the encoder associated to the set, as defined in CURRFBID. If the parameter is set to "Selectable", the encoder connected to the axis can be of any type. To add encoders from third parties to the list, these encoders must be defined in the file encoders_OEM.Xml in folder C:\FagorCNC\MTB\Data.

MOT_KT
Possible values: From 0 to 1000.0000 Nm/A.
Default value: 0 Nm/A.
Associated variable: (V.)MPMOT.MOT_KT.nb

Torque constant of the synchronous motor, (motor torque according to the rms current).

MOT_INOM
Possible values: From 0 to 250.0000 A
Default value: 0 A.
Associated variable: (V.)MPMOT.MOT_INOM.nb

Rated motor current.

MOT_I0
Possible values: From 0 to 200.0000 A
Default value: 0 A.
Associated variable: (V.)MPMOT.MOT_I0.nb

Stall current. On asynchronous motors, this current is the same as the rated current of the motor. They are different on synchronous motors.

MOT_IMAX
Possible values: From 0 to 500.0000 A
Default value: 0 A.
Associated variable: (V.)MPMOT.MOT_IMAX.nb

Maximum motor current.

MOT_POLESPAIR
Possible values: From 0 to 60.0000.
Default value: 0.
Associated variable: (V.)MPMOT.MOT_POLESPAIR.nb

Number of pairs of poles.

MOT_VOLTAGE

Possible values: From 0 to 480 V
Default value: 0 V.
Associated variable: (V.)MPMOT.MOT_VOLTAGE.nb

Rated motor voltage. Note that for a synchronous motor, this parameter is just for information purposes, it is not used in the control.

MOT_SLIP

Possible values: From 0 to 6000.0000 rpm.
Default value: 0 rpm.
Associated variable: (V.)MPMOT.MOT_SLIP.nb

Slip of the asynchronous motor.

MOT_RS

Possible values: From 0 to 1000.0000 Ohm.
Default value: 0 Ohms.
Associated variable: (V.)MPMOT.MOT_RS.nb

Phase-neuter resistance of the stator at 20 °C.

MOT_LSTATORLEAKAGE

Possible values: From 0 to 10 000.0000 H.
Default value: 0 H.
Associated variable: (V.)MPMOT.MOT_LSTATORLEAKAGE.nb

Phase-neuter leak inductance of the stator of an asynchronous motor. Note that for a synchronous motor, this parameter represents the inductance of the motor stator and not the leak inductance because leak inductance is not applicable to these motors.

MOT_PNOM

Possible values: From 0 to 200 000.0000 W.
Default value: 0 W.
Associated variable: (V.)MPMOT.MOT_PNOM.nb

Motor rated power.

MOT_PMAX

Possible values: From 0 to 200 000.0000 W.
Default value: 0 W.
Associated variable: (V.)MPMOT.MOT_PMAX.nb

Maximum motor power.

MOT_TERM_CONST

Possible values: From 1 to 12 000.0000 s.
Default value: 120 s.
Associated variable: (V.)MPMOT.MOT_TERM_CONST.nb

Thermal time constant of the motor.

MOT_TEMP_SENSOR

Possible values: SENSOR_0 — SENSOR_7.
Default value: SENSOR_1
Associated variable: (V.)MPMOT.MOT_TEMP_SENSOR.nb

Type of motor temperature sensor.

Sensor.	Description.
SENSOR_0	PTC over-temperature sensor (yes/no) in SPM and FXM motors.
SENSOR_1	Single sensor for AXM motors.
SENSOR_2	PTC KTY84-130 sensor for FKM (except FKM1 series) and FM9 motors.

Sensor.	Description.
SENSOR_3	NTC sensor for FM7 motors.
SENSOR_4	Without sensor (SENSORLESS).
SENSOR_5	Linear sensor.
SENSOR_6	PTC sensor in FKM1 motors.
SENSOR_7	RTD Pt1000 sensor for FKM motors.

MOT_PHAS_ORDER

Possible values: ORDER_0 — ORDER_3
Default value: ORDER_0.
Associated variable: (V.)MPMOT.MOT_PHAS_ORDER.nb

Power phase inversion. This parameter may be used to change the sequence of the motor power supply phases without having to physically swap the wires.

Sensor.	Description.
ORDER_0	Without changes.
ORDER_1	Swapping of U and V phases.
ORDER_2	Swapping of U and W phases.
ORDER_3	Swapping of V and W phases.

MOT_DELTA_STAR

Possible values: Star (Y) / Triangle (Δ).
Default value: Star (Y).
Associated variable: (V.)MPMOT.MOT_DELTA_STAR.nb

Winding connection type for asynchronous motors.

MOT_INERTIA

Possible values: From 0.01 to 1 000 000.0000 kg·cm².
Default value: 140 kg·cm².
Associated variable: (V.)MPMOT.MOT_INERTIA.nb

Motor inertia.

MOT_SNOM

Possible values: From 0 to 60 000.0000 rpm.
Default value: 0 rpm.
Associated variable: (V.)MPMOT.MOT_SNOM.nb

Rated speed of a synchronous motor or base speed (below the constant power area) of a synchronous motor.

MOT_SMAX

Possible values: From 0 to 60 000.0000 rpm.
Default value: 0 rpm.
Associated variable: (V.)MPMOT.MOT_SMAX.nb

Maximum motor speed.

MOT_ROTORRES

Possible values: From 0 to 1000.0000 Ohm.
Default value: 0 Ohms.
Associated variable: (V.)MPMOT.MOT_ROTORRES.nb

Phase-neuter resistance of the rotor at 20 °C.

MOT_LROTORLEAKAGE

Possible values: From 0 to 10 000.0000 H.
Default value: 0 H.
Associated variable: (V.)MPMOT.MOT_LROTORLEAKAGE.nb

Phase-neuter leak inductance of the rotor.

MOT_LMUTUAL

Possible values: From 0 to 10 000.0000 H.
Default value: 0 H.
Associated variable: (V.)MPMOT.MOT_LMUTUAL.nb

Magnetizing inductance.

MOT_LMUTUALGAIN

Possible values: From 0 to 10 000.0000.
Default value: 1.
Associated variable: (V.)MPMOT.MOT_LMUTUALGAIN.nb

Gain of the magnetizing inductance of the motor.

MOT_MAGN_CURR

Possible values: From 0 to 200.0000 A
Default value: 0 A.
Associated variable: (V.)MPMOT.MOT_MAGN_CURR.nb

Motor rms current without load.

MOT_MAGN_VOLT

Possible values: From 0 to 460 V
Default value: 0 V.
Associated variable: (V.)MPMOT.MOT_MAGN_VOLT.nb

Motor phase-phase rms voltage without load.

MOT_TEMP_EMAX

Possible values: From 0 ºC to 200 ºC.
Default value: 145 ºC.
Associated variable: (V.)MPMOT.MOT_TEMP_EMAX.nb

Maximum temperature limit that can be set for the motor. A zero value in this parameter means that the temperature limit will not be checked.

MOT_FWEAK_SPEED

Possible values: From 0 to 60 000 rpm.
Default value: 0 rpm.
Associated variable: (V.)MPMOT.MOT_FWEAK_SPEED.nb

Speed at which a synchronous motor working as spindle motor will begin «increasing its magnetic field or magnetic flux».

MOT_KE

Possible values: From 0 to 500.000 V/1000 rpm.
Default value: 0 V/1000 rpm.
Associated variable: (V.)MPMOT.MOT_KE.nb

BEMF (in volts) generated between each pair of phases of the stator winding per every 1000 rpm of the motor. This parameter is used to weaken the magnetic flux in synchronous motors working as a spindle. When set to 0, the decrease in magnetic flux will be zero.

MOT_MNOM

Possible values: From 0 to 100 000 Nm.
Default value: 0 Nm.
Associated variable: (V.)MPMOT.MOT_MNOM.nb

Rated motor torque.

MOT_M0

Possible values: From 0 to 100 000 Nm.
Default value: 0 Nm.
Associated variable: (V.)MPMOT.MOT_M0.nb

Motor stall torque.

MOT_TYPE

Possible values: Synchronous / Asynchronous.
Default value: Synchronous.
Associated variable: (V.)MPMOT.MOT_TYPE.nb

Type of motor, either synchronous or asynchronous.

MOT_CTRL_TYPE

Possible values: Standard / Sensorless / V/f.
Default value: Standard.
Associated variable: (V.)MPMOT.MOT_CTRL_TYPE.nb

Type of control of the asynchronous motor. For synchronous motors, always set this parameter to "Standard".

Sensor.	Description.
Standard.	Vector control by rotor flux orientation. This control is used with speed feedback.
Sensorless	Sensorless vector control based on the voltage model.
V/f	V/f control.

PWMFREQ

Possible values: 4 kHz / 8 kHz.
Default value: 4 kHz.
Associated variable: (V.)MPMOT.PWMFREQ.nb

IGBT switching frequency.

CURRENT_PROGAIN

Possible values: From 0 to 1000,0000 A/rpm.
Default value: 0 A/rpm.
Associated variable: (V.)MPMOT.CURRENT_PROGAIN.nb

Value of the proportional gain of the current loop.

CURRENT_INT_TIME

Possible values: From 0 to 1000.0000 Hz
Default value: 0 Hz.
Associated variable: (V.)MPMOT.CURRENT_INT_TIME.nb

Value of the integral gain of the current loop.

CURRENT_FB_DER_TIME

Possible values: From 0 to 10.0000 ms
Default value: 0 ms.
Associated variable: (V.)MPMOT.CURRENT_FB_DER_TIME.nb

Value of the derivative gain of the current loop.

CLOOP_LR_GAIN

Possible values: Yes / No.
Default value: Yes.
Associated variable: (V.)MPMOT.CLOOP_LR_GAIN.nb

This parameter can automatically calculate the value of the proportional (Kp) (CURRENT_PROGAIN) and integral (Ti) (CURRENT_INT_TIME) gain for the current loop adjustment in non-Fagor synchronous motors as well as the adjustment of the current, flux and counter electromotive force loops in non-Fagor asynchronous motors. The parameter uses the values defined in MOT_RS (resistance) and MOT_LSTATORLEAKGE (inductance) for this calculation.

CLOOP_BANDWIDTH

Possible values: From 0 to 2000.0000 Hz
Default value: 1000 Hz.
Associated variable: (V.)MPMOT.CLOOP_BANDWIDTH.nb

Bandwidth of the current loop of an asynchronous motor at zero speed. The drive uses this in its internal calculations to maintain this bandwidth at high rpms.

MOT_L_SERIES

Possible values: From 0 to 10 000.0000 H.
Default value: 0 H.
Associated variable: (V.)MPMOT.MOT_L_SERIES.nb

This parameter is only applicable to high speed spindles due to its very low leak inductance. In order to properly control of the current loop, a three-phase inductance in series with that of the motor trying to increase the leak inductance must be inserted (between the drive and the motor) . The minimum value of the motor leak inductance will be 100 µH. The MOT_L_SERIES value will be that of the inductance in series.

CURRENT_LIMIT

Possible values: From 0 to 300.0000 A
Default value: 0 A.
Associated variable: (V.)MPMOT.CURRENT_LIMIT.nb

Limit of the current command that reaches the system's current loop.

FLUX_PROGAIN

Possible values: From 0 to 1000.0000 A/Wb.
Default value: 0 A/Wb.
Associated variable: (V.)MPMOT.FLUX_PROGAIN.nb

Value of the proportional gain of the flux loop.

FLUX_INT_TIME

Possible values: From 0 to 1000.0000 milliseconds.
Default value: 0 milliseconds.
Associated variable: (V.)MPMOT.FLUX_INT_TIME.nb

Value of the integral time of the flux loop.

BEMF_PROGAIN

Possible values: From 0 to 100,0000 Wb/V.
Default value: 0 Wb/V.
Associated variable: (V.)MPMOT.BEMF_PROGAIN.nb

Value of the proportional gain of the BEMF (back electromotive force).

BEMF_IN

Possible values: From 0 to 1000.0000 milliseconds.
Default value: 0 milliseconds.
Associated variable: (V.)MPMOT.BEMF_IN.nb

Value of the integral time of the BEMF (back electromotive force).

RR_EST_ACTIVE

Possible values: Yes / No.
Default value: Yes.
Associated variable: (V.)MPMOT.RR_EST_ACTIVE.nb

Activate rotor resistance estimate. Rotor resistance varies by the effect of temperature and rotor turning speed variations. To activate the estimation of the new resistance value, the power must be 20 % greater than the rated value and this parameter must be set to "Yes".

ROTOR_FIXED_TEMP

Possible values: From 0 ºC to 150 ºC.
Default value: 0 ºC.
Associated variable: (V.)MPMOT.ROTOR_FIXED_TEMP.nb

Fixed temperature of the rotor. The value of the rotor resistance will be the one corresponding to the temperature indicated in this parameter. It will only be taken into account if the estimation is not enabled (parameter RR_EST_ACTIVE) or if there is no temperature sensor (parameter MOT_TEMP_SENSOR different from SENSOR_2, SENSOR_3, SENSOR_6 and SENSOR_7).

FLUX_REDUCTION

Possible values: from 0 to 100 %.
Default value: 100 %.
Associated variable: (V.)MPMOT.FLUX_REDUCTION.nb

Percentage reduction of flux level. This parameter indicates the percentage of magnetizing current to be applied to the unloaded motor. Momentarily decreasing the magnetizing current reduces the noise generated by the motor as well as the heat generated by the motor turning in a vacuum. To cancel the effect of the parameter, set it to 100%.

Since the setting of the flux and maximum motor torque has a delay, it is not recommended to use this flux reduction on motors used to feed the axes.

MRAS_PROGAIN

Possible values: From 0 to 100 000 (rad/s)/Wb².
Default value: 500 (rad/s)/Wb².
Associated variable: (V.)MPMOT.MRAS_PROGAIN.nb

Value of the proportional gain of the PI used to estimate the speed. Only applicable with SENSORLESS asynchronous motor control.

MRAS_INT_TIME

Possible values: From 0 to 1000 milliseconds.
Default value: 20 milliseconds.
Associated variable: (V.)MPMOT.MRAS_INT_TIME.nb

Value of the integral gain of the PI used to estimate the speed. Only applicable with SENSORLESS asynchronous motor control.

FLUX_VOLT_FILTER

Possible values: From 1 to 8000.0000 Hz.
Default value: 2 Hz.
Associated variable: (V.)MPMOT.FLUX_VOLT_FILTER.nb

Cutoff frequency of the voltage model filter.

FLUX_CPROGAIN

Possible values: From 0 to 1000.0000.
Default value: 0.
Associated variable: (V.)MPMOT.FLUX_CPROGAIN.nb

Proportional gain of the flux current.

FLUX_C_INT_TIME

Possible values: From 0 to 1000.0000 ms.
Default value: 0 milliseconds.
Associated variable: (V.)MPMOT.FLUX_C_INT_TIME.nb

Integral gain of the flux current.

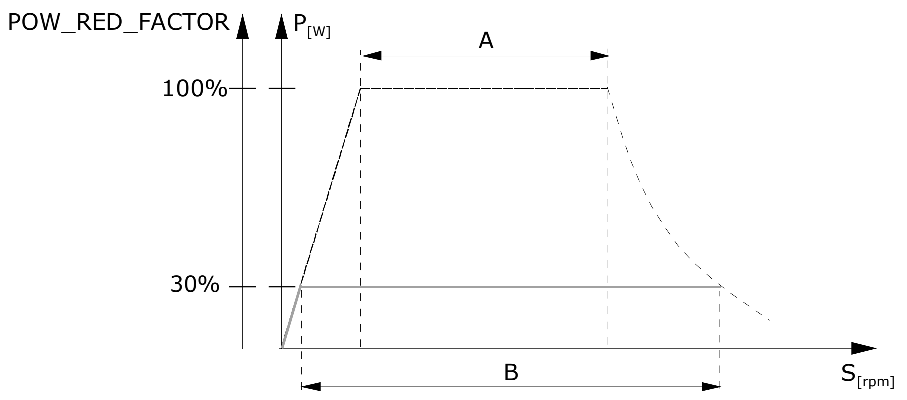
POW_RED_FACTOR

Possible values: from 0 to 100 %.
Default value: 100 %.
Associated variable: (V.)MPMOT.POW_RED_FACTOR.nb

Percentage reduction of motor power level. This parameter is used to reduce power when a wide range of speeds at constant power is required. The rated power of the selected motor must be greater than what the application requires in order to be able to provide the required power even at very low speed.

The same for high speed. The range where the motor outputs the application's minimum power is wider than when using a motor whose rated power is what the application requires.

Limiting the power at the motor does not mean that it can be controlled with a smaller drive. However, the power demanded from the power supply will be lower.



(A) Speed range with constant power equal to 100% of the motor rated power.
(B) Speed range with constant power equal to 30 % of the motor rated power.

COEFI2T_A

Possible values: Within ±10.
Default value: 0.
Associated variable: (V.)MPMOT.COEFI2T_A.nb

Coefficient A of the stall torque curve.

COEFI2T_B

Possible values: Within ±10.
Default value: 0.
Associated variable: (V.)MPMOT.COEFI2T_B.nb

Coefficient B of the stall torque curve.

COEFI2T_C

Possible values: Within ±10.
Default value: 0.
Associated variable: (V.)MPMOT.COEFI2T_C.nb

Coefficient C of the stall torque curve.

MAXPEAKCURRENTTIME

Possible values: From 0 to 100 s.
Default value: 30 s.
Associated variable: (V.)MPMOT.MAXPEAKCURRENTTIME.nb

Maximum peak current time.

POLAR_PITCH

Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.007 87 inches.
Default value: 0.
Associated variable: (V.)MPMOT.POLAR_PITCH.nb

Linear motor polar pitch.

AUTOPHASING_METHOD

Possible values: GC3 / GC11.
Default value: GC11.
Associated variable: (V.)MPMOT.AUTOPHASING_METHOD.nb

Method for motor autophasing. Autophasing is an estimate of the RHO electrical position (position of the rotor magnets relative to the stator) when the motor is moving.

AUTOPHASING_STATUS

Possible values: NOCALC / DONE / EXEC.
Default value: NOCALC.
Associated variable: (V.)MPMOT.AUTOPHASING_STATUS.nb

Motor autophasing status.

MOTOR_RHO

Possible values: From 0 to 65 535.
Default value: 0.
Associated variable: (V.)MPMOT.MOTOR_RHO.nb

Rho value.

AUTOPHASING_CURRENT

Possible values: from 0 to 400 %.
Default value: 200 %.
Associated variable: (V.)MPMOT.AUTOPHASING_CURRENT.nb

Current (percentage of nominal current) during the execution of the autophasing.

MOT_LIN

Possible values: Yes / No.
Default value: No.
Associated variable: (V.)MPMOT.MOT_LIN.nb

Linear motor.

MOT_I0_RHOSYNC

Possible values: Yes / No.
Default value: No.
Associated variable: (V.)MPMOT.MOT_I0_RHOSYNC.nb

Synchronize the vector control with the motor I0.

CURRFB_INVERT

Possible values: Yes / No.
Default value: No.
Associated variable: (V.)MPMOT.CURRFB_INVERT.nb

Feedback inversion of the motor current loop.

COGGING_AMPLITUDE1
..
COGGING_AMPLITUDE4

Possible values: From 0 to 100.0000 %.
Default value: 0 %.
Associated variable: (V.)MPMOT.COGGING_AMPLITUDE1.nb
(V.)MPMOT.COGGING_AMPLITUDE4.nb

Parameter used for the cogging adjustment process of a synchronous motor. Cogging compensation amplitude, defined as a percentage of the rated motor current (parameter MOT_INOM).

COGGING_PHASE1
..
COGGING_PHASE4

Possible values: Within ±180.0000 º.
Default value: 0 º.
Associated variable: (V.)MPMOT.COGGING_PHASE1.nb
(V.)MPMOT.COGGING_PHASE4.nb

Parameter used for the cogging adjustment process of a synchronous motor. Cogging mismatch with respect to RHO.

COGGING_NSINES1
..
COGGING_NSINES4

Possible values: From 0 to 32767.0000.
Default value: 0.
Associated variable: (V.)MPMOT.COGGING_NSINES1.nb
Associated variable: (V.)MPMOT.COGGING_NSINES4.nb

Parameter used for the cogging adjustment process of a synchronous motor. Number of stator slots, i.e. number of sinewings per mechanical revolution of the motor (parameter COGGING_NTURN4).

COGGING_NTURN1
..
COGGING_NTURN4

Possible values: From 1 to 100.0000.
Default value: 1.
Associated variable: (V.)MPMOT.COGGING_NTURN1.nb
Associated variable: (V.)MPMOT.COGGING_NTURN4.nb

Parameter used for the cogging adjustment process of a synchronous motor. Number of mechanical turns of the motor to compensate for the defined number of sinewings (parameter COGGING_NSINES4).

COGGING_BAND_RPM_L1
..
COGGING_BAND_RPM_L4

Possible values: From 0 to 100000.0000 rpm.
Default value: 1000 rpm.
Associated variable: (V.)MPMOT.COGGING_BAND_RPM_L1.nb
(V.)MPMOT.COGGING_BAND_RPM_L4.nb

Parameter used for the cogging adjustment process of a synchronous motor. Speed below which the cogging compensation is total.

COGGING_BAND_RPM_H1
..
COGGING_BAND_RPM_H4

Possible values: From 0 to 100000.0000 rpm.
Default value: 2000 rpm.
Associated variable: (V.)MPMOT.COGGING_BAND_RPM_H1.nb
(V.)MPMOT.COGGING_BAND_RPM_H4.nb

Parameter used for the cogging adjustment process of a synchronous motor. Speed above which the cogging compensation is null.

MOT_MASS

Possible values: From 0.0100 to 1000000.0000 kg.
Default value: 140 kg.
Associated variable: (V.)MPMOT.MOT_MASS.nb

Motor mass.

VF_MAXVOLTAGE

Possible values: From 0 to 480 V.
Default value: 0
Associated variable: (V.)MPMOT.VF_MAXVOLTAGE.nb

Maximum voltage to be applied to the V/f control.

VF_VOLT_CURVE1

Possible values: From 0 to 100 %.
Default value: 100 %.
Associated variable: (V.)MPMOT.VF_VOLT_CURVE1.nb

Percentage of the rated motor voltage (MOT_VOLTAGE) defining the ordinate of point 1 of the characteristic curve.

VF_FREQ_CURVE1

Possible values: From 0 to 100 %.
Default value: 100 %.
Associated variable: (V.)MPMOT.VF_FREQ_CURVE1.nb

Percentage of rated engine speed (MOT_SMON) defining the abscissa of point 1 of the characteristic curve.

VF_VOLT_CURVE2

Possible values: From 0 to 100 %.
Default value: 100 %.
Associated variable: (V.)MPMOT.VF_VOLT_CURVE2.nb

Percentage of the rated motor voltage (MOT_VOLTAGE) defining the ordinate of point 2 of the characteristic curve.

VF_FREQ_CURVE2

Possible values: From 0 to 100 %.
Default value: 100 %.
Associated variable: (V.)MPMOT.VF_FREQ_CURVE2.nb

Percentage of rated engine speed (MOT_SMON) defining the abscissa of point 2 of the characteristic curve.

VF_BOOSTVOLTAGE

Possible values: From 0 to 100 %.
Default value: 100 %.
Associated variable: (V.)MPMOT.VF_BOOSTVOLTAGE.nb

Percentage of motor rated voltage (MOT_VOLTAGE) to be applied at zero speed.

EXT_BALLAST_RESISTANCE

Possible values: From 18.0000 to 1000.0000 Ohm.
Default value: 18 Ohm.
Associated variable: (V.)MPMOT.EXT_BALLAST_RESISTANCE.nb

Ohmic value of the external Ballast resistor. It is useful for the I²t protection of that resistor.

EXT_BALLAST_POWER

Possible values: From 0 to 15000.0000 w.
Default value: 950 w.
Associated variable: (V.)MPMOT.EXT_BALLAST_POWER.nb

Power value of the external Ballast resistor. It is useful for the I²t protection of that resistor.

EXT_BALLAST_ENERGY

Possible values: From 0 to 400000.0000 J.
Default value: 60000 J.
Associated variable: (V.)MPMOT.EXT_BALLAST_ENERGY.nb

Value of the energy pulse that can be dissipated by the external ballast resistor. It is useful for the I²t protection of that resistor.

EXT_BALLAST_I2T

Possible values: Yes / No.
Default value: Yes.
Associated variable: (V.)MPMOT.EXT_BALLAST_I2T.nb

Activate the I²t protection of the external ballast resistor.

2.7 Connections; feedbacks.

2.7.1 Select the feedback type.

FBMODEL

Possible values: Select a device from the list.
Default value: -.
Associated variable: (V.)MPFB.FB_MODEL.nb

List of available protocols and encoders. Depending on the type of selected encoder, some of the parameters associated with the feedback are preset and cannot be changed (a lock icon will appear alongside the parameter).

i

To define an encoder that cannot be found on the list, please contact Fagor Automation.

PROTOCOL

Possible values: None / SSI / EnDat / FeeDat / Hiperface / BiSS.
Default value: None.
Associated variable: (V.)MPFB.PROTOCOL.nb

Absolute signal feedback protocol.

ABSMODE

Possible values: None / Absolute / Absolute + incremental.
Default value: Absolute.
Associated variable: (V.)MPFB.ABSMODE.nb

Absolute feedback work mode. The work mode depends on the ABSFEEDBACK and INCRSIGNAL parameters. See *"ABSFEEDBACK"* on page 85.

Value.	Meaning.
None.	The feedback is not absolute. (parameter ABSFEEDBACK = No)
Absolute.	The CNC uses the absolute signal to find the initial position of the axis after startup and while it is moving. (parameter ABSFEEDBACK = Total) (parameter INCRSIGNAL = None)
Absolute + incremental.	The CNC uses the absolute signal to find the initial position of the axis after startup and the incremental signal while it is moving. (parameter ABSFEEDBACK = One turn) (parameter INCRSIGNAL = TTL / differential TTL / 1 Vpp)

INCRSIGNAL

Possible values: None / TTL / differential TTL / 1 Vpp.
Default value: None.
Associated variable: (V.)MPFB.INCRSIGNAL.nb

Incremental signal type.

ENCTYPE

Possible values: FBTYPE ROT / FBTYPE LIN.
Default value: FBTYPE LIN.
Associated variable: (V.)MPFB.ENCTYPE.nb

Type of encoder; rotary or linear. Informative parameter; non-modifiable.

FBTYPE

Possible values: FBTYPE ROT / FBTYPE LIN.
Default value: FBTYPE LIN.
Associated variable: (V.)MPFB.FBTYPE.nb

This parameter indicates how the feedback will work depending on the axis type; as a rotary or linear feedback.

2.7.2 Counting resolution of a shaft.

NPULSES

Possible values: From 0 to 65 535 pulses.
Default value: 1250 pulses.
Associated variable: (V.)MPFB.NPULSES.nb

This meaning of this parameter depends on the type of encoder.

Type of encoder.	Meaning.
Linear encoder.	The parameter must have a value of 0.
Rotary encoder.	The parameter indicates the number of pulses per turn of the encoder.

If the parameter NPULSES is set with a value other than "0", parameter REFINI determines whether the CNC homes the spindle on its first movement or not.

Gear ratio between the motor and the encoder.

The gear ratio between the motor and the encoder can be taken into account when defining the NPULSES and PITCH parameters, although it is recommended to use the parameters INPUTREV and OUTPUTREV. If the reduction is included in the parameters NPULSES and PITCH, the parameters INPUTREV and OUTPUTREV will be set to 1. See *"INPUTREV"* on page 83. See *"OUTPUTREV"* on page 83.

LINEAR_PITCH

Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.007 87 inches.
Default value: 5.000 mm / 0.196 85 inches.
Associated variable: (V.)MPFB.LINEAR_PITCH.nb

This meaning of this parameter depends on the type of axis and encoder.

Type of axis and encoder.	Meaning.
Linear axis. Linear encoder.	Encoder resolution (pitch).
Linear axis. Rotary encoder.	Not being used.
Rotary axis or spindle.	Not being used.

PITCH

Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.007 87 inches.
Possible values (3): From 0 to 99 999.9999 degrees.
Default value: 5 mm / 0.196 85 inches / 360º.
Associated variable: (V.)MPFB.PITCH.nb

This meaning of this parameter depends on the type of axis and encoder.

Type of axis and encoder.	Meaning.
Linear axis. Linear encoder.	This parameter indicates the ballscrew pitch.
Linear axis. Rotary encoder.	This parameter indicates the ballscrew pitch.
Rotary axis or spindle.	This parameter indicates the number of degrees per turn of the encoder.

2.7.3 Configuring the SSI protocol.

SSICLKFREQ

Possible values: From 150 to 16 000 kHz.
Default value: 150 kHz.
Associated variable: (V.)MPFB.SSICLKFREQ.nb

SSI communication frequency. The frequency depends on the length of the cable.

Length.	Maximum frequency.
Up to 20 meters (65 feet).	600 kHz.
Up to 50 meters (164 feet).	400 kHz.
Up to 75 meters (246 feet).	300 kHz.
Up to 100 meters (328 feet).	250 kHz.

SSIDATALEN

Possible values: From 5 to 32 bits.
Default value: 25 bits.
Associated variable: (V.)MPFB.SSIDATALEN.nb

Number of bits of the SSI transmission that make up the position value (coordinate). For example, 32 bit for a Fagor absolute linear encoder using default parameters.

SSIDATAFORMAT

Possible values: Binary / Gray.
Default value: Binary.
Associated variable: (V.)MPFB.SSIDATAFORMAT.nb

SSI data format; binary code or Gray code.

SSIRESOL

Possible values: From 0 to 4.295E+9.
Default value: 4096.
Associated variable: (V.)MPFB.SSIRESOL.nb

Digital counting resolution or number of digital counting units contained in a pitch.

Example: Linear encoder.
Fagor absolute linear encoder with a pitch of 20 µm and a digital resolution of 0.1 µm.
SSIRESOL = 20 µm / 0,1 µm = 200.

Example: Rotary encoder.
Rotary encoder with 8192 pulses per turn and a ballscrew with a 10 mm pitch.
PITCH = 10 mm.
SSIRESOL = 8192.
The axis resolution will be; 10 / 8192 = 0.0012 mm.

Example: Inductosyn module.
For SSITYPE = ABSIND (Inductosyn ROT+ABS), for a 2 or 4-degree turn and depending on whether it is a high or low resolution encoder, the counting increment is 10 000 units.
PITCH = 2º or 4º.
SSIRESOL = 10 000
The resolution of the axis will be; 2º or 4º / 10 000 = 0.0002 or 0.0004 degrees.

SSIDATACHECKTYPE

Possible values: Do not calculate / Fagor / Inductosyn / Even parity / Odd parity.
Default value: Do not calculate.
Associated variable: (V.)MPFB.SSIDATACHECKTYPE.nb

CRC type of the SSI feedback. 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.	Even if the CNC receives the CRC bits, they are not processed at all by the CNC and, therefore, no errors are reported when the transmission is erroneous. This is not a recommendable option and should only be used during setup.
Fagor.	Data checking algorithm used in Fagor Automation feedback systems.
Inductosyn.	Data checking algorithm used when connecting to an Inductosyn module.
Even parity.	Even parity bit method. This method detects errors, however, it does not correct them.
Odd parity.	Odd parity bit method. This method detects errors, however, it does not correct them.

SSICRCBITS

Possible values: From 0 to 16.
Default value: 0.
Associated variable: (V.)MPFB.SSICRCBITS.nb

Number of bits that making up the valid transmission check (CRC, checksum, etc.) of the SSI transmission.

Example.
Fagor Absolute encoders can be configured to transmit 5 CRC bits together with the position value.
SSIDATALEN=27
SSICRCBITS =5

SSIDATAMODE

Possible values: LSB / MSB.
Default value: LSB.
Associated variable: (V.)MPFB.SSIDATAMODE.nb

Transmission order of the most significant bit.

Value.	Meaning.
LSB	The LSB (Least Significant Bit) is the first one.
MSB	The MSB (Most Significant Bit) is the first one.

SSIPACKFORMAT

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.)MPFB.SSIPACKFORMAT.nb

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 =
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 CRC bits.

SSIALARMBITS

Possible values: From 0 to 8.
Default value: 0.
Associated variable: (V.)MPFB.SSIALARMBITS.nb

Number of alarm bits. The feedback device can send one or more bits indicating an alarm condition or one or more acknowledgment bits if the transmission is successful.

SSIALARMLEVEL

Possible values: From 0 to 255.
Default value: 0.
Associated variable: (V.)MPFB.SSIALARMLEVEL.nb

Alarm signal levels. Value that the alarm/acknowledge bits for an error condition to occur. This parameter only makes sense when SSIALARMBITS is other than 0.

Example 1.
The encoder indicates an error condition using 2 bits, where the first is a 1 and the second a 0.
SSIALARMBITS = 2
SSIALARMLEVEL = 2.

Example 2.
The encoder has a correct acknowledgement transmission bit and indicates it with a logical 0.
SSIALARMBITS = 1
SSIALARMLEVEL = 1. If it is 0 it can be assume that everything is correct.

SSITYPE

Possible values: Select a type from the list.
Default value: Undefined.
Associated variable: (V.)MPFB.SSITYPE.nb

SSI transmission format. 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.
Undefined.	
Fagor.	
G user.	User configuration.
User.	User configuration.
INDUCTOSYN_LINEAR_ABS.	Absolute linear Inductosyn.
INDUCTOSYN_ROTATIVE_ABS.	Absolute rotary Inductosyn in one turn.
INDUCTOSYN_RESOLVER_ABS.	Absolute rotary Inductosyn in one turn.
INDUCTOSYN_LINEAR.	Absolute linear Inductosyn in the pitch. The linear encoder must be homed.
INDUCTOSYN_ROTATIVE.	Absolute rotary Inductosyn in the pitch. The linear encoder must be homed.
FAGOR_LA.	Fagor linear encoder
FAGOR_GA_SA_SVA.	Fagor linear encoder

Value.	Meaning.
FAGOR_HAD200_SAD_170.	Fagor rotary encoder.
FAGOR_HAD90_SAD90.	Fagor rotary encoder.
SSI	SSI configuration for BISS protocol.

STARDELAY

Possible values: From 0 to 255.
Default value: 0.
Associated variable: (V.)MPFB.STARDELAY.nb

Number of clocks to wait between the first down flank and the first up-flank. This value is needed to implement the wait time to convert the signal in some encoders.

Example.
SSICLOCKFREQ = 400 kHz
STARTDELAY = 3
Wait time = (1/400×1000)×3 = 7,5 clocks.

2.7.4 Home search.

I0TYPE

Possible values: Normal / Increasing distance-coded / Decreasing coding / Not evaluated.
Default value: Normal.
Associated variable: (V.)MPFB.I0TYPE.nb

This parameter indicates that it is the I0 type of encoder.

Value.	Meaning.
Normal.	The encoder can have of one or more I0 marks, but there is only one selected mark (for example, with a micro (home-switch)). The encoder references the position using this I0 mark. Depending on the initial position, long movements may be necessary until the position is homed.
Increasing distance-coded.	The encoder has several I0 marks, separated at different distances, following a mathematical formula. The encoder refers to the position after exceeding two contiguous I0 marks, that is, after a few millimeters. • The I0 codification will be increased when the distance between the I0 marks increases according to the direction of movement during the home search. • The I0 codification will be decreased when the distance between the I0 marks decreases according to the direction of movement during the home search.
Decreasing distance-coded.	
Not evaluated.	The CNC does not use the I0 of the encoder. If the parameter DECINPUT=Yes, the CNC uses the home switch.

REFPULSE

Possible values: None / Positive / Negative.
Default value: None.
Associated variable: (V.)MPFB.REFPULSE.nb

Flank of the marker pulse (I0) used by the CNC to complete the home search. For 1 Vpp signals, always select “Positive”.

Value.	Meaning.
None.	The CNC does not evaluate the marker pulse (I0); it considers that the position is homed when the axis releases the home switch. This option is valid when accuracy is not an important factor; for example, the position of a magazine, feeders, etc.
Positive.	The CNC considers that the position is homed when it receives the up-flank of the marker pulse (I0).
Negative.	The CNC considers that the position is homed when it receives the down flank of the marker pulse (I0).

I0CODDI1

Possible values: From 0 to 32 000.
Default value: 1000.
Associated variable: (V.)MPFB.I0CODDI1.nb

Pitch between 2 fixed I0 signals. For encoders with coded I0 marks (I0TYPE parameter), this parameter indicates the pitch between two fixed I0 marks, set in number of waves. The pitch is calculated as the distance between two consecutive fixed I0 marks divided by the signal period. See machine parameter I0CODDI2.

I0CODDI2

Possible values: From 0 to 32 000.
Default value: 100.
Associated variable: (V.)MPFB.I0CODDI2.nb

Pitch between 2 variable I0 signals. For encoders with coded I0 marks (I0TYPE parameter), this parameter indicates the pitch between two variable I0 marks, set in number of waves.

The pitch is calculated as the distance between two consecutive variable I0 marks divided by the signal period.

Example.
Distance between fixed I0s; 20 000 mm.
Distance between variable I0s; 20 020 mm.
Signal period; 20 µm.
Number of waves between fixed I0s; I0CODDI1 = 20 000 / (20 x EXTMULT) = 1000.
Number of waves between variable I0s; I0CODDI2 = 20 020 / (20 x EXTMULT) = 1001.

Values to be assigned for Fagor encoders with distance-coded I0.

Linear encoder.	I0CODDI1	I0CODDI2
SOP SVOP SOX SVOX MOY MOVY GOP GOX MOP MOC MOT MOVP MOX MOVX COP COC COT COVP COX COVX COY FOP FOT FOX	1000	1001
LOP LOX	2000	2001
LOX	2000	2001

Rotary encoder.	I0CODDI1	I0CODDI2
HO SO (90 000 pulses)	1000	1001
HO SO (180 000 pulses)	1000	1001
HOP SOP (180 000 pulses)	1000	1001

EXTMULT

Possible values: From 0 to 65 535.
Default value: 1.
Associated variable: (V.)MPFB.EXTMULT.nb

External multiplying factor. For encoders with coded I0 marks (I0TYPE parameter), this parameter indicates the relationship between the mechanical period (of the graduation on the glass) and the electrical period (of the counting signal being applied to the CNC).

Example.
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.

Linear encoder.	EXTMULT
SOP SVOP GOP MOP MOC MOT MOVP COP COC COT COVP FOP LOP	1
SOX SVOX GOX MOX MOVX COX COVX FOT	5
MOY MOVY COY LOX	10
FOX	25

Rotary encoder.	EXTMULT
HO SO (90 000 pulses)	5
HO SO (180 000 pulses)	10
HOP SOP (180 000 pulses)	1

FREQ_LIMIT

Possible values: ±21 475E+9.
Default value: 1 848 576.0000
Associated variable: (V.)MPFB.FREQ_LIMIT.nb

Counting frequency limit.

REFSYNC

Possible values: A - B / not A - B / A - not B / not A - not B / Not synchronized.
Default value: A - B.
Associated variable: (V.)MPFB.REFSYNC.nb

Position of the I0 signal compared to the encoder A and B signals. For encoders with initial digital positioning without an I0 signal, this parameter indicates the starting quadrant for the digital counting of the sine/cosine cycle. This data must be supplied by the encoder manufacturer.

Value.	Meaning.
A B.	The reference signal of the feedback device is at level A and B.
No-A B.	The reference signal of the feedback device is at level No-A and B.
A No-B.	The reference signal of the feedback device is at level A and No-B.
No-a No-b	The reference signal of the feedback device is at level No-A and No-B.
Not synchronized.	I0 not synchronized.

2.7.5 Feedback alarm.

FBACKAL

Possible values: Yes / No.
Default value: Yes.
Associated variable: (V.)MPFB.FBACKAL.nb

This parameter allows all the feedback associated alarms to be activated/deactivated. If the alarms are deactivated, the CNC displays a power-up message indicating that this safety feature is disabled. This situation is only allowed during setup; once the setup is completed, these alarms must be enabled. When a feedback alarm occurs, the PLC turns the REFPOIN mark off.

TTL analogic feedback devices.

- Verify any disconnected or broken feedback cables (only differential TTL).
- Verify the number of pulses between I0s.

Analog feedback devices using sinusoidal signals.

- Verify any disconnected or broken feedback cables.
- Verify that the sin/cosine values are within the optimum working range.
- Verify the number of pulses between I0s.

Digital feedback devices.

- Verify communications with the feedback device.
- CRC, checksum and/or alarm bit protections associated with each type of feedback device.

2.7.6 Gear ratio.

INPUTREV

Possible values: From 1 to 32 767.
Default value: 1.
Associated variable: (V.)MPFB.INPUTREV.nb

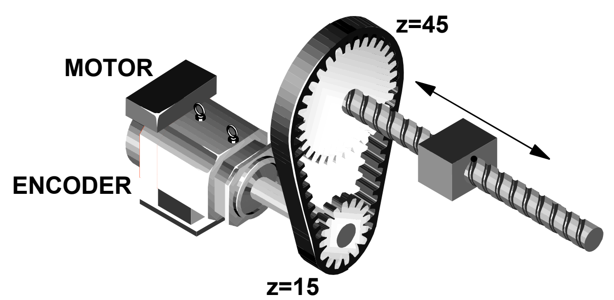
Input shaft turns. See machine parameter OUTPUTREV.

OUTPUTREV

Possible values: From 1 to 32 767.
Default value: 1.
Associated variable: (V.)MPFB.OUTPUTREV.nb

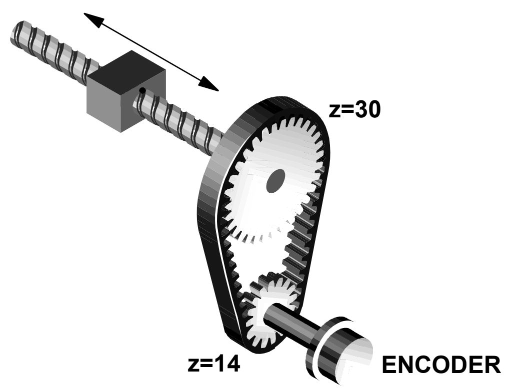
Output shaft turns. Both the parameters INPUTREV and OUTPUTREV set the gear ratio between the encoder and the shaft moving the load. For an axis or spindle with various gear ratios (for example, a spindle with four ranges), a feedback must be defined for each one.

- For a linear encoder, the gear ratio is 1: 1.
- For a rotary encoder and motor feedback, these parameters set the gear ratio between the motor shaft (parameter INPUTREV) and the final shaft moving the load (parameter OUTPUTREV).



INPUTREV=15
OUTPUTREV=45

- For a rotary encoder and direct feedback, these parameters set the gear ratio between the encoder (parameter INPUTREV) and the shaft that moves the load (parameter OUTPUTREV).



INPUTREV=14
OUTPUTREV=30

Gear ratio between the motor and the encoder (parameters NPULSES and PITCH).

The possible gear ratio between the motor and the encoder may also be entered directly using parameters NPULSES and PITCH; in this case, parameters INPUTREV and OUTPUTREV must be set to 1. If the gear ratio is not a whole number, it is recommended to set the real values of the gear ratio for INPUTREV and OUTPUTREV instead of leaving a inaccurate number in PITCH and NPULSES.

2.7.7 Reversing the counting sign.

INVERT

Possible values: Yes / No.
Default value: No.
Associated variable: (V.)MPFB.INVERT.nb

Changing the counting sign. If the shaft is packed, the CNC displays the following error; change the value of the parameter LOOPCH. If the shaft is not packed, but the counting direction is not correct, change the value of the parameters INVERT and LOOPCH.

2.7.8 Backlash compensation.

BACKLASH

Possible values (1): Within ±3.2768 mm.
Possible values (2): Between ±0.129 01 inches.
Possible values (3): Within ±3.2768 degrees.
Default value: 0.
Associated variable: (V.)MPFB.BACKLASH.nb

Shaft backlash. 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. It usually occurs on axes with an encoder and on old machines where there is wear between the ballscrew and the support.

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.

On axes with a linear encoder, set BACKLASH = 0.

2.7.9 Machine Reference point.

ABSFEEDBACK

Possible values: No / one turn / total.
Default value: No.
Associated variable: (V.)MPFB.ABSFEEDBACK.nb

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.

ABSTREATMENT

Possible values: Default / No / One turn / Total.
Default value: Default.
Associated variable: (V.)MPFB.ABSTREATMENT.nb

This parameter indicates how the CNC handles the single-turn absolute encoder. With Default value, the CNC treats the encoder according to the ABSFEEDBACK parameter. With a value other than Default, the CNC will treat the feedback according to the chosen value.

REFVALUE

Possible values (1): Between ±99 999.9999 mm.
Possible values (2): Between ±3937.007 87 inches.
Possible values (3): Between ±99 999.9999 degrees.
Default value: 0.
Associated variable: (V.)MPFB.REFVALUE.nb

This parameter sets the coordinate of the reference point with respect to machine zero. 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.

REFSHIFT

Possible values (1): Between ±99 999.9999 mm.
Possible values (2): Between ±3937.007 87 inches.
Possible values (3): Between ±99 999.9999 degrees.
Default value: 0.
Associated variable: (V.)MPFB.REFSHIFT.nb

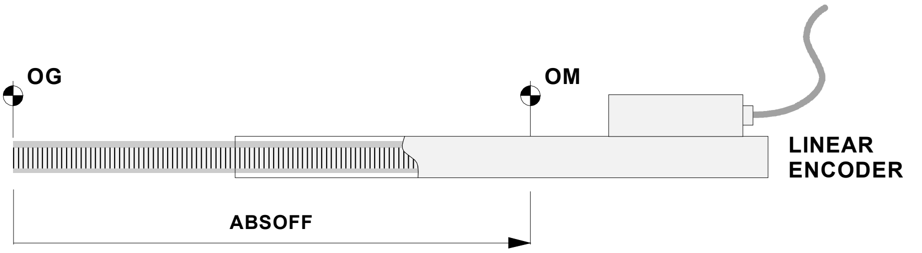
Offset of the reference point. Sometimes, to readjust the machine, it is necessary to take down the feedback device, thus when putting back up, the new home point might no coincide with the previous one. Since the home point must still be the same, the difference between the new point and the old point 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.

ABSOFF

Possible values (1): Between ±99 999.9999 mm.
Possible values (2): Between ±3937.007 87 inches.
Possible values (3): Between ±99 999.9999 degrees.
Default value: 0.
Associated variable: (V.)MPFB.ABSOFF.nb

Feedback device offset with respect to I0. For absolute encoders, the CNC always registers the position of the axis relative to the glass zero point (OC). In encoders with distance-coded “I0”, the CNC registers the position of the axis relative to the glass zero point (OC) after a slight movement of the axis. In order for the CNC to show the position with respect to Machine Zero (OM), this parameter must be assigned the position of the machine zero (OM) with respect to the glass zero point (OC). Depending on its design, the glass zero point (start of distance coding) can be inside or outside the encoder.



(OG) Zero position of the glass.
(OM) Machine reference zero position.

2.7.10 Compensating for the feedback signal.

SC_SIN_GAIN

Possible values: From 0.7000 to 1,3000.
Default value: 1.
Associated variable: (V.)MPFB.SC_SIN_GAIN.nb

Compensation (proportional gain mode) of the amplitude of the feedback sine signal. See parameter SC_COS_GAIN.

SC_COS_GAIN

Possible values: From 0.7000 to 1,3000.
Default value: 1.
Associated variable: (V.)MPFB.SC_COS_GAIN.nb

Compensation (proportional gain mode) of the amplitude of the feedback cosine signal. Both the parameters SC_SIN_GAIN and SC_COS_GAIN set the compensation (proportional gain mode) of the amplitude of the sine/cosine signal sent by the feedback (signals A and B) with respect to the functions $\sin\theta$ and $\cos\theta$. In an ideal case, the gain value should be 1.

This parameter is only valid when the circle of the sine and cosine signals are set manually (parameter SC_ADJ_MODE).

SC_SIN_OFF

Possible values: Between ±200.0000 mV.
Default value: 0 mV.
Associated variable: (V.)MPFB.SC_SIN_OFF.nb

Compensation (offset mode) of the feedback sine signal. See parameter SC_COS_OFF.

SC_COS_OFF

Possible values: Between ±200.0000 mV.
Default value: 0 mV.
Associated variable: (V.)MPFB.SC_COS_OFF.nb

Compensation (offset mode) of the feedback cosine signal. Both the parameters SC_SIN_OFF and SC_COS_OFF set the compensation (offset mode) of the sine/cosine signal sent by the feedback (signals A and B) with respect to the functions $\sin\theta$ and $\cos\theta$. In an ideal case, the offset value should be 0.

This parameter is only valid when the circle of the sine and cosine signals are set manually (parameter SC_ADJ_MODE).

SC_ADJ_MODE

Possible values: None / Manual / Automatic.
Default value: Automatic.
Associated variable: (V.)MPFB.SC_ADJ_MODE.nb

Circle adjustment of the encoder sine and cosine signals. The circle adjustment is a process used to adjust the offset and gains with those of the feedback signals that are treated by the drive software, so that its A and B signals (sine and cosine) come closer to the $\sin\theta$ and $\cos\theta$ functions (and therefore become mathematically correct; meaning that they create a perfect circumference).

The gain and offset adjustments compensate the amplitude and offset of the A and B signals with respect to the $\sin\theta$ and $\cos\theta$ functions. The ideal values for the gain and the offset are 1 and 0 respectively.

This procedure can only be applied to encoders, not to resolvers.

Value.	Meaning.
None.	There is no circle adjustment. The CNC does not compensate the offset or gains; it ignores the parameters SC_SIN_GAIN, SC_COS_GAIN, SC_SIN_OFF and SC_COS_OFF.
Automatic.	The CNC adjusts the circle automatically and continuously. The result of the adjustment is not reflected in the parameters SC_SIN_GAIN, SC_COS_GAIN, SC_SIN_OFF or SC_COS_OFF.
Manual.	Manual circle adjustment using the parameters SC_SIN_GAIN, SC_COS_GAIN, SC_SIN_OFF and SC_COS_OFF.

2.8 Connections; power supplies.

2.8.1 Set up a power supply.

PWSCTLTYPE

Possible values: Constant / Predictive.
Default value: Constant.
Associated variable: (V.)MPPWS.PWSCTLTYPE.nb

Type of DC bus voltage control.

AUTODETECT

Possible values: Yes / No.
Default value: Yes.
Associated variable: (V.)MPPWS.AUTODETECT.nb

When AUTODETECT=Yes, the parameter setting suggested by the CNC is only valid if there are no other additional units connected to the DC Bus; for example, other drives that do not share the Sercos Bus, capacitor modules, etc. In this case, the calculated capacity does not match and it will be necessary to manually adjust the control loop (AUTODETECT=No).

Autodetection to adjust the RPS control loop. As for shaft drives, the RPS has control loops whose adjustments depend on the following:

- The resistance and inductance values of the choke inside the installation and the total capacity in the DC bus (capacitors).
- Fagor recommends that each RPS model must be installed with a specific type of choke. Each RPS-choke combination must switch to a set frequency. Refer to the RPS manual for the proper switching frequency.

Value.	Meaning.
Yes	The CNC auto-configures the RPS control loops, while taking into account the following. • The CNC assumes that the detected RPS module matches the choke recommended by Fagor. • The CNC does a search for the nodes connected to the Sercos bus and assumes the total capacity for all of them. The CNC auto-configures the parameters at startup and saves them. It is likely that the auto-configured parameters will not match the machine parameters the first time. In this case, the new values are written and saved in the machine parameters and no warning is prompted.
No	The capacitor, resistance and inductance values must be set manually to adjust the control loops of the RPS.

SSTIME

Possible values: From 500 to 30 000.0000 milliseconds.
Default value: 5000 milliseconds.
Associated variable: (V.)MPPWS.SSTIME.nb

Maximum time for the DC bus to reach a stable voltage during the bus charging process (SoftStart).

VOLTAGE_REFERENCE

Possible values: From 500 to 750.0000 V.
Default value: 625 V.
Associated variable: (V.)MPPWS.VOLTAGE_REFERENCE.nb

Voltage velocity command. This parameter sets the voltage level achieved by the DC bus when RPS is enabled and increasing.

CURRENT_LIMIT

Possible values: From 0 to 500.0000 A.
Default value: 0 A.
Associated variable: (V.)MPPWS.CURRENT_LIMIT.nb

Current limit. This parameter establishes the maximum operating current for the drive. If it is set to 0, the current is not limited, meaning that the machine will operate at the maximum current it was designed for. For regenerative power supplies (RPS), the maximum value for this parameter is determined by the maximum current of the drive.

RESISTANCE

Possible values: From 0 to 1000.0000 Ohm.
Default value: 0 Ohms.
Associated variable: (V.)MPPWS.RESISTANCE.nb

Choke winding resistance per phase.

INDUCTANCE

Possible values: From 0 to 10 000.0000 H.
Default value: 0.4 H.
Associated variable: (V.)MPPWS.INDUCTANCE.nb

Choke winding inductance per phase.

CAPACITANCE

Possible values: From 0 to 1 000 000.0000 H.
Default value: 1000 H.
Associated variable: (V.)MPPWS.CAPACITANCE.nb

Total capacity of the DC bus capacitors.

PWMFREQ

Possible values: 4 kHz / 8 kHz.
Default value: 4 kHz.
Associated variable: (V.)MPPWS.PWMFREQ.nb

IGBT switching frequency. Refer to the RPS manual for the proper switching frequency.

MAINSFREQ

Possible values: 50 Hz / 60 kHz.
Default value: 50 kHz.
Associated variable: (V.)MPPWS.MAINSFREQ.nb

Mains frequency.

VOLTAGE_PROGAIN

Possible values: From 0 to 1000.0000 A/V.
Default value: 0 A/V.
Associated variable: (V.)MPPWS.VOLTAGE_PROGAIN.nb

Proportional gain of the voltage loop. The parameter CLOOP_LR_GAIN can auto-calculate this parameter value.

VOLTAGE_INT_TIME

Possible values: From 0 to 10 000.0000 ms.
Default value: 0 ms.
Associated variable: (V.)MPPWS.VOLTAGE_INT_TIME.nb

Integral gain of the voltage loop. The parameter CLOOP_LR_GAIN can auto-calculate this parameter value.

CURRENT_PROGAIN

Possible values: From 0 to 1000.0000 A/V.
Default value: 0 A/V.
Associated variable: (V.)MPPWS.CURRENT_PROGAIN.nb

Proportional gain of the current loop. The parameter CLOOP_LR_GAIN can auto-calculate this parameter value.

CURRENT_INT_TIME

Possible values: From 0 to 10 000.0000 ms.
Default value: 0 ms.
Associated variable: (V.)MPPWS.CURRENT_INT_TIME.nb

Integral gain of the current loop. The parameter CLOOP_LR_GAIN can auto-calculate this parameter value.

CLOOP_LR_GAIN

Possible values: Yes / No.
Default value: Yes.
Associated variable: (V.)MPPWS.CLOOP_LR_GAIN.nb

Parameter to auto-calculate the gains of the voltage and current loops, based on the choke resistance and inductance values and the total capacity of the DC bus.

2.9 Connections; analog drives.

2.9.1 Configuring an analog drive.

MINANOUT

Possible values: From 0 to 32 767.
Default value: 0.
Associated variable: (V.)MPAD.MINANOUT.nb

Minimum velocity command. The setpoint is expressed in same units used by the D/A converter, allowing any integer between 0 and 32 767, where the 32 767 value corresponds to a setpoint of 10 V.

MINANOUT	Velocity command.
1 … 3277 … 32 767	0,3 mV. … 1 V. … 10 V.

SERVOOFF

Possible values: Between ±32 767.
Default value: 0.
Associated variable: (V.)MPAD.SERVOOFF.nb

Setpoint that is used as an offset. The setpoint is expressed in same units used by the D/A converter, allowing any integer between ±32 767, where the ±32 767 value corresponds to a setpoint of ±10 V.

SERVOOFF	Velocity command.
-32 767 … -3277 … 1 … 3277 … 32 767	-10 V. … -1 V. … 0.3 mV. … 1 V. … 10 V.

MAXVOLT

Possible values: From 0 to 10 000.0000 mV.
Default value: 9500.0000 mV.
Associated variable: (V.)MPAD.MAXVOLT.nb

Lathe and mill models.

This is the velocity command the CNC must output so the axis can reach its maximum rapid traverse feedrate G00FEED.

Laser model.

This is the command the CNC must provide for the spindle to reach the maximum generator power set in parameter G00FEED.

2.10 Connections; handwheels.

2.10.1 Configuring a handwheel.

MPGTYPE

Possible values: Default / HBH.
Default value: Default.
Associated variable: (V.)MPMPG.MPGTYPE.nb

Handwheel type.

Value.	Meaning.
Default.	General handwheel or individual handwheel.
HBH.	HBH3 or HBH4 terminal.

MPGAXIS

Possible values: No value or axis from the list.
Default value: No value (general handwheel).
Associated variable: (V.)MPMPG.MPGAXIS.nb

Axis associated with the handwheel. The CNC may have general handwheels to move any axis or individual handwheels that will only move their associated axes. This parameter determines whether the handwheel is either individual or general. 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.

Value.	Meaning.
No value.	General handwheel. To move an axis using a general handwheel, select it from the jog panel.
Any axis from the list.	Individual handwheel associated with the selected axis. To move an axis with an individual handwheel, no prior axis selection is needed.

To set the resolution for each handwheel, the distance it can move at each switch position, set the machine axis parameter MPGRESOL.

Compatibility between handwheels.

An axis may be moved indistinctly with its individual handwheel or with a general handwheel.

- If there are several axes selected while 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.

HWFBTYPE

Possible values: TTL / TTL differential.
Default value: TTL.
Associated variable: (V.)MPMPG.HWFBTYPE.nb

Type of handwheel signal.

Value.	Meaning.
TTL	TTL signal.
TTLd	Differential TTL signal.

MPGDIRECT

Possible values: Positive / Negative.
Default value: Positive.
Associated variable: (V.)MPMPG.MPGDIRECT.nb

Handwheel turning direction. If the turning direction is correct, leave the parameter as it is; to reverse it, change the parameter.

HBHIP

Possible values: IP address.
Default value: 0.0.0.0.
Associated variable: (V.)MPMPG.HBHIP.nb

IP address of terminal HBH3 or HBH4.

2.11 General machine parameters.

2.11.1 Channel configuration.

NCHANNEL

Possible values: From 1 to 2.
Default value: 1.
Associated variable: (V.)MPG.NCHANNEL

This parameter must be modified directly in the parameter tree (left panel of the table), by adding or deleting channels in the "General" branch. This parameter cannot be modified directly in the table.

Number of channels. 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. In a multichannel system, the channels can act independently or together; this means they can communicate, synchronize and carry out coordinated actions.

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; etc.

Operation of a channel.

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, arithmetic parameters and, if necessary, they may be synchronized via part-program. Each channel can only control its own axes and spindles, however, when using a part-program or MDI/MDA mode it can command the movements of axes or spindles from other channels.

The axes and spindles of a channel.

In order to be able to move an axis or spindle, it must be assigned to a channel. The axes and spindles of one channel can act independently or in parallel with the other 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.

2.11.2 Configuring the axes of the system.

NAXIS

Possible values: From 1 to 10.
Default value: 3.
Associated variable: (V.)MPG.NAXIS

This parameter must be modified directly in the parameter tree (left panel of the table), by adding or deleting axes in the "Axes" branch. This parameter cannot be modified directly in the table.

This parameter sets the number of axes of the system, whether they are servo-controlled or not. Both the number of axes in the system and the sum of axes and spindles are limited depending on the CNC model.

Example:
A system is available with the following options.
- Number of axes: 3 to 5.
- Number of spindles: 0 to 2.
- Number of axes + spindles: 5.

Since the sum of the axes and spindles is limited to 5, it can be configured from having 5 axes and no spindle (NSPDL= 0 NAXIS= 5) to 2 spindles and 3 axes (NSPDL= 3 NAXIS= 2). It is not possible for it to have 5 axes and two spindles (NSPDL= 2 NAXIS= 5).

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 possible (maximum value of parameter NAXIS).

AXISNAME

This parameter shows the table to define the names of the axes. The table has one parameter for each axis.

Parameter.	Meaning.
AXISNAME n	Axis name.

AXISNAME | AXISNAME n

Possible values: X, X1..X9, Y, Y1..Y9, .. E, E1..E9.
Default value: Starting with AXISNAME1: X, Y, Z, U, V, W, A, B, C.
Associated variable: (V.)MPG.AXISNAMEn

Name of the axes. 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 spindles may be within the range X, X1…X9,...C, C1…C9. For example X, X1, Y3, Z9, W, W7, C...

Name of the axes according to DIN 66217.

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.
X Y Z	Main axes. Two axes make up the work plane and the third axis corresponds to the axis perpendicular to the plane.
U V W	Auxiliary axes, parallel to X-Y-Z respectively.
A B C	Rotary axes, on X-Y-Z respectively.

Logical number of the axes.

The order in which the axes are defined in the AXISNAME table determines their logical number. The first axis will be the logical axis 1, the second axis will be the logic axis 2, and so on. As with the axis name, the logic number permits identifying the axis in PLC variables, marks, etc.

2.11.3 Configuring the spindles of the system.

NSPDL

Possible values: From 0 to 3.
Default value: 1.
Associated variable: (V.)MPG.NSPDL

This parameter must be modified directly in the parameter tree (left panel of the table), by adding or deleting spindles in the "Spindles" branch. This parameter cannot be modified directly in the table.

This parameter sets the number of spindles of the system, whether they are servo-controlled or not. Both the number of spindles in the system and the sum of axes and spindles of the system are limited depending on the CNC model.

Example:

A system is available with the following options.

- Number of axes: 3 to 5.
- Number of spindles: 0 to 2.
- Number of axes + spindles: 5.

Since the sum of the axes and spindles is limited to 5, it can be configured from having 5 axes and no spindle (NSPDL= 0 NAXIS= 5) to 2 spindles and 3 axes (NSPDL= 3 NAXIS= 2). It is not possible for it to have 5 axes and two spindles (NSPDL= 2 NAXIS= 5).

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

SPDLNAME

This parameter shows the table to define the names of the spindles. The table has one parameter for each spindle.

Parameter.	Meaning.
SPDLNAME n	Spindle name.

SPDLNAME | SPDLNAME n

Possible values: S, S1··S9.
Default value: Starting from SPDLNAME1; S, S1, S2··S9.
Associated variable: (V.)MPG.SPDLNAMEn

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

Logical number of the axes.

The order in which the spindles are defined in the SPDLNAME table determines their logical number. The logical numbering of spindles continues from the last logical axis; this means that for a system with 3 axes, the first spindle in the table will be logical spindle 4, the second will be logical spindle 5, and so on. As with the spindle name, the logic number permits identifying the spindle in PLC variables, marks, etc.

AXISNAME n	SPDLNAME n	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·.

2.11.4 Time setting (system).

CNCTIME

Possible values: From 1 to 20 milliseconds.
Default value: 4 milliseconds.
Associated variable: (V.)MPG.CNCTIME

Cycle time (sampling period) of the CNC interpolator.

SYSTEMTIME

Possible values: from 0.1250 to 0.2500 milliseconds.
Default value: 0.2500 milliseconds.
Associated variable: (V.)MPG.SYSTEMTIME

This parameter sets the following.

- Sercos bus time base.
- Frequency of execution of the velocity loop of the axes and spindles.

ETHERCATTIME

Possible values: From 0.25 to 10 ms.
Default value: 1 ms.
Associated variable: (V.)MPG.ETHERCATTIME

EtherCAT bus cycle time.

- This parameter must be greater than or equal to the speed loop time (SYSTEMTIME).
- This parameter may be less than the interpolator cycle time (CNCTIME), e.g., for laser devices connected to the bus.
- The position loop time for EtherCAT axes (POS_LOOPTIME) must be a multiple of this parameter.
- The cycle time for the EtherCAT bus, defined in the configurator, must be equal to this parameter.

PRGFREQ

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 CNCTIME = 4 ms of and a frequency of PRGFREQ = 2, the PLC program will be executed every 4 x 2 = 8 ms.

2.11.5 CAN bus configuration.

CANOPENFREQ

Possible values: Autoscan / 1 Mbps / 800 kbps / 500 kbps / 250 kbps.
Default value: Autoscan.
Associated variable: (V.)MPG.CANOPENFREQ

CANopen bus communication frequency. When using the CANopen protocol, the transmission speed at the bus is defined in each node and they all must run at the same speed. The transmission speed depends on the total length of the bus. Using the following illustrative values; assigning other values may cause communication errors due to signal distortion.

Speed	Length of the CAN bus.
Autoscan	The CNC adjusts the bus frequency at every startup depending on the speed of the rest of the modules. Depending on the configuration of the bus, this option can make the CNC startup slower than if there is a set frequency.
1000 kHz	Up to 20 meters.
800 kHz	From 20 to 40 meters.
500 kHz	From 40 to 100 meters.
250 kHz	From 100 to 500 meters.

The speed of 250 kHz is only available to communicate with the keyboards and RIOW and RIOR series remote modules; this speed is not available for the RIO5 series remote modules.

Peculiarities of the laser model.

For the laser model with RIOR modules, it is recommended to use a communications frequency of 1 MHz.

2.11.6 Serial line configuration.

RSTYPE

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

Type of serial line. 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.

2.11.7 Default conditions (sytem).

INCHES

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. They are also the units for the offsets from machine zero (#MCS). To change the units from the part-program, use function G70 or G71.

PRESSURE

Possible values: Bar / Psi.
Default value: Bar.
Associated variable: (V.)MPG.PRESSURE

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.

2.11.8 Arithmetic parameters.

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

Maximum local arithmetic parameter. See general machine parameter MINLOCP.

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

Minimum local arithmetic parameter. Parameters MINLOCP and MAXLOCP define the group of local arithmetic parameters to be used. Local parameters can only be accessed from the program or subroutine where they have been programmed. Since the main program or a subroutine can call another subroutine and this, in turn, can call a second subroutine and so on (up to 20 levels of nesting), the CNC has seven levels of local parameter nesting.

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

Maximum global arithmetic parameter. Check general machine parameter MINGLBP.

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

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

Maximum read-only global arithmetic parameter. See general machine parameter ROPARMIN.

ROPARMIN
Possible values: From 100 to 9999.
Default value: 0.
Associated variable: (V.)MPG.ROPARMIN

Minimum read-only global arithmetic parameter. 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
Possible values: From 10 000 to 19 999.
Default value: 10 025.
Associated variable: (V.)MPG.MAXCOMP

Maximum common arithmetic parameter. See general machine parameter MINCOMP.

MINCOMP

Possible values: From 10 000 to 19 999.
Default value: 10 000.
Associated variable: (V.)MPG.MINCOMP

Minimum common arithmetic parameter. 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.

2.11.9 Volumetric compensation tables.

VOLCOMP

This parameter shows the volumetric compensation table. Volumetric compensations are configured in the machine parameters and are activated from the PLC (marks VOLCOMP1 to VOLCOMP4).

VOLCOMP | VOLCOMP n

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

Medium and large volumetric compensation.

Parameter.	Meaning.
VCOMPAXIS1 VCOMPAXIS2 VCOMPAXIS3	Name of the first, second and third axis to be compensated.
VCOMPFILE	File containing volumetric compensation data.

Basic volumetric compensation.

Parameter.	Meaning.
VMOVAXIS1 VMOVAXIS2 VMOVAXIS3	In the basic volumetric compensation, the name of the axis that generates variations when moved.
NPOINTSAX1 NPOINTSAX2 NPOINTSAX3	Number of points of basic volumetric compensation on each axis.
INIPOSAX1 INIPOSAX2 INIPOSAX3	Initial position of the basic volumetric compensation on each MOVAXIS axis.
INCREAX1 INCREAX2 INCREAX3	Interval between points for the MOVAXIS axis in basic volumetric compensation.
VCOMPAXIS1 VCOMPAXIS2 VCOMPAXIS3	Name of the first, second and third axis to be compensated.
VCOMPFILE	File containing volumetric compensation data.

VOLCOMP | VOLCOMP n | VMOVAXIS1
VOLCOMP | VOLCOMP n | VMOVAXIS2
VOLCOMP | VOLCOMP n | VMOVAXIS3

Possible values: Any axes defined in AXISNAME.
Default value: None.
Associated variable: (V.)MPG.VMOVAXIS1[tbl]
Associated variable: (V.)MPG.VMOVAXIS2[tbl]
Associated variable: (V.)MPG.VMOVAXIS3[tbl]

Axis whose movement affects another axis. At least one axis must be defined. These linear or rotary axes define the volume to be compensated. If it is a gantry axis, it is only necessary to define the master axis.

These parameters are only available for basic volumetric compensation; they have no function in medium and large volumetric compensations, where the order of the axes is implicit in the file. The CNC displays these parameters when loading a file with the extension csv in the parameter VCOMPFILE.

VOLCOMP | VOLCOMP n | NPOINTSAX1
VOLCOMP | VOLCOMP n | NPOINTSAX2
VOLCOMP | VOLCOMP n | NPOINTSAX3

Possible values: From 2 to 1000 points.
Default value: 10 points.
Associated variable: (V.)MPG.NPOINTSAX1[tbl]
Associated variable: (V.)MPG.NPOINTSAX2[tbl]
Associated variable: (V.)MPG.NPOINTSAX3[tbl]

Number of points that define the volume to be compensated for each axis. The total number of points for the volume is limited to 15,625, and is calculated in the following manner: NPOINTSAX1 × NPOINTSAX2 × NPOINTSAX3. The total number of points for the volume can be distributed among all the axes, up to a maximum of 1000 points per axis.

These parameters are only available for basic volumetric compensation; they have no function in medium and large volumetric compensations. The CNC displays these parameters when loading a file with the extension csv in the parameter VCOMPFILE.

VOLCOMP | VOLCOMP n | INIPOSAX1
VOLCOMP | VOLCOMP n | INIPOSAX2
VOLCOMP | VOLCOMP n | INIPOSAX3

Possible values: Within ±99999.9999 mm / Within ±3937.00787 inch.
Default value: 0.
Associated variable: (V.)MPG.INIPOSAX1[tbl]
Associated variable: (V.)MPG.INIPOSAX2[tbl]
Associated variable: (V.)MPG.INIPOSAX3[tbl]

Initial position of the basic volumetric compensation for the MOVAXIS axis.

These parameters are only available for basic volumetric compensation; they have no function in medium and large volumetric compensations. The CNC displays these parameters when loading a file with the extension csv in the parameter VCOMPFILE.

VOLCOMP | VOLCOMP n | INCREAX1
VOLCOMP | VOLCOMP n | INCREAX2
VOLCOMP | VOLCOMP n | INCREAX3

Possible values: Within ±99999.9999 mm / Within ±3937.00787 inch.
Default value: 10 mm / 0.3937 inch.
Associated variable: (V.)MPG.INCREAX1[tbl]
Associated variable: (V.)MPG.INCREAX2[tbl]
Associated variable: (V.)MPG.INCREAX3[tbl]

Interval between points for the MOVAXIS axis in basic volumetric compensation.

These parameters are only available for basic volumetric compensation; they have no function in medium and large volumetric compensations. The CNC displays these parameters when loading a file with the extension csv in the parameter VCOMPFILE.

VOLCOMP | VOLCOMP n | VCOMPAXIS1
VOLCOMP | VOLCOMP n | VCOMPAXIS2
VOLCOMP | VOLCOMP n | VCOMPAXIS3

Possible values: Any axes defined in AXISNAME.
Default value: None.
Associated variable: (V.)MPG.VCOMPAXIS1[tbl]
Associated variable: (V.)MPG.VCOMPAXIS2[tbl]
Associated variable: (V.)MPG.VCOMPAXIS3[tbl]

These parameters set the axes to be compensated with volumetric compensation. The axes may be linear or rotary, and at least one axis must be defined. 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. An axis can be included in several different compensations, but compensations that share axes cannot be active at the same time. If it is a gantry axis, it is only necessary to define the master axis; the CNC also applies the compensation to the slave axis.

VOLCOMP | VOLCOMP n | VCOMPFILE

Associated variable: (V.)MPG.VCOMPFILE[tbl]

File containing volumetric compensation data.

Medium and large volumetric compensation.

The compensation tables are generated by the calibration application; they are not editable from the CNC. The units in the file data (millimeters or inches) must be those defined by the CNC (parameter INCHES).

Basic volumetric compensation.

The data to be compensated must be a text file (in csv format). The OEM or the company responsible for the calibration must generate this file and define it in the machine parameter VCOMPFILE. The units in the file data (millimeters or inches) must be those defined by the CNC (parameter INCHES).

2.11.10 Execution times.

MINAENDW

Possible values: From 0 to 65 535 milliseconds.
Default value: 10 milliseconds.
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 function 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 (CNCTIME x PRGFREQ).

REFTIME

Possible values: From 0 to 1 000 000 milliseconds.
Default value: 0. milliseconds.
Associated variable: (V.)MPG.REFTIME

Estimated home searching time. See general machine parameter TTIME.

HTIME

Possible values: From 0 to 1 000 000 milliseconds.
Default value: 0. milliseconds.
Associated variable: (V.)MPG.HTIME

Estimated time for an H function. See the general machine parameter TTIME.

DTIME

Possible values: From 0 to 1 000 000 milliseconds.
Default value: 0. milliseconds.
Associated variable: (V.)MPG.DTIME

Estimated time for a D function. See the general machine parameter TTIME.

TTIME

Possible values: From 0 to 1 000 000 milliseconds.
Default value: 0. milliseconds.
Associated variable: (V.)MPG.TTIME

Estimated time for an T function. 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 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. The spindle machine parameter SPDLTIME indicates the estimated time for executing an S function. The machine parameter MTIME of the M function table indicates the estimated time to execute an M function.

2.11.11 Numbering the digital inputs of the CANopen bus. Base index.

Available up to version v2.00.

NDIMOD

Possible values: From 0 to 42.
Default value: 0 (no numbering is defined).
Associated variable: (V.)MPG.NDIMOD

This parameter shows the number of digital input modules connected to the same bus CAN. After defining this parameter, it is possible to set the numbering of the digital inputs of each module by assigning a base index to the first of them. If this parameter is not set, the CNC numbers the digital inputs sequentially according to the order of the modules in the bus.

DIMODADDR

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.

DIMODADDR | DIMOD n

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

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

2.11.12 Numbering the digital outputs of the CANopen bus. Base index.

Available up to version v2.00.

NDOMOD

Possible values: From 0 to 42.
Default value: 0 (no numbering is defined).
Associated variable: (V.)MPG.NDOMOD

This parameter shows the number of digital outputs modules connected to the same bus CAN. After defining this parameter, it is possible to set the numbering of the digital outputs of each module by assigning a base index to the first of them. If this parameter is not set, the CNC numbers the digital outputs sequentially according to the order of the modules in the bus.

DOMODADDR

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.

DOMODADDR | DOMOD n

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

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

2.11.13 Numbering the digital inputs of the CANopen bus. Logic blocks.

NDIMOD

Possible values: From 0 to 42.
Default value: 0 (no numbering is defined).
Associated variable: (V.)MPG.NDIMOD

This parameter shows the number of logic blocks that the digital input modules connected to the CANopen bus are divided into. After defining this parameter, it is possible to set the numbering of the digital inputs of each logic block by assigning a base index to the first of them.

- In RIO5 and RIOR nodes, each module is a logic block, several modules cannot be grouped in a block or divide a module into blocks. The logic blocks must have the same number and size as the modules physically detected in the bus.
- In the RIO5 nodes, each double module (module with two boards) it counts as two, in other words, two logic blocks.
- In RIOW nodes and third-party nodes, it is possible to display as many logic blocks as desired, both in number and in size.

If this parameter is not set, the CNC numbers the digital inputs sequentially according to the order of the modules in the bus.

DIMODADDR

This parameter shows the table to set the digital input logic blocks. The following fields must be defined for each block.

Column.	Meaning.
MNEMONIC	Mnemonic of the logic block.
NODE	Node number of the header to which the I/O logic block belongs.
BLOCK	Number of the logic block.
ADDRESS	Base address of the logic block.
NDI	Number of digital inputs of the module.

The configuration must meet the following requirements. If the parameters are not set according to these requirements, the CNC will ignore it and it will number the digital inputs sequentially according to the order of the modules in the bus.

- The parameters of all I/O nodes of the bus must be set; no node can be left undefined.
- The parameters of all the resources of a node must be set; no I/O can be left without a number.
- The maximum number of logic blocks in the system will be 64, both for digital inputs and digital outputs.
- The number of I/Os in a logic block must be other than zero and a multiple of 8 (8, 16, 24, 32, ...1024).
- The values of the base index must be comply with the formula "8n + 1" (i.e. 1, 9, 17, 25, etc.).
- The blocks may be distributed at will within the range of resources managed by PLC (1...1024). The PLC admits blanks, but the blocks cannot overlap; in other words, a PLC digital input or output can only belong to a single node-module of the bus.

NODE

Possible values: From 1 to 126.
Default value: 0.
Associated variable: (V.)MPG.DIMODNODE[nb]

Node number of the header to which the I/O logic block belongs. The parameters of all I/O nodes of the bus must be set; no node can be left undefined.

BLOCK

Possible values: From 1 to 64.
Default value: 0.
Associated variable: (V.)MPG.DIMODBLOCK[nb]

Number of the logic block. The maximum number of logic blocks in the system will be 64, both for digital inputs and digital outputs.

ADDRESS

Possible values: From 1 to 1017.
Default value: The first valid value.
Associated variable: (V.)MPG.DIMODADDR[nb]

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

In the RIO5 nodes, each double module (module with two boards) it counts as two, in other words, two logic blocks. These modules must be assigned two base indexes for the inputs and two base indexes for the outputs; one for each board.

NDI

Possible values: From 8 to 1024.
Default value: 0.
Associated variable: (V.)MPG.DIMODNDI[nb]

Number of digital inputs of the logic block; it must be other than zero and a multiple of 8 (8, 16, 24, 32, ...1024). The parameters of all the resources of a node must be set; no I/O can be left without a number.

2.11.14 Numbering the digital inputs of the CANopen bus. Logic blocks.

NDOMOD

Possible values: From 0 to 42.
Default value: 0 (no numbering is defined).
Associated variable: (V.)MPG.NDOMOD

This parameter shows the number of logic blocks that the digital output modules connected to the CANopen bus are divided into. After defining this parameter, it is possible to set the numbering of the digital outputs of each logic block by assigning a base index to the first of them.

- In RIO5 and RIOR nodes, each module is a logic block, several modules cannot be grouped in a block or divide a module into blocks. The logic blocks must have the same number and size as the modules physically detected in the bus.
- In the RIO5 nodes, each double module (module with two boards) it counts as two, in other words, two logic blocks.
- In RIOW nodes and third-party nodes, it is possible to display as many logic blocks as desired, both in number and in size.

If this parameter is not set, the CNC numbers the digital outputs sequentially according to the order of the modules in the bus.

DOMODADDR

This parameter shows the table to set the digital output logic blocks. The following fields must be defined for each block.

Column.	Meaning.
MNEMONIC	Mnemonic of the logic block.
NODE	Node number of the header to which the I/O logic block belongs.
BLOCK	Number of the logic block.
ADDRESS	Base address of the logic block.
NDO	Number of digital outputs of the module.

The configuration must meet the following requirements. If the parameters are not set according to these requirements, the CNC will ignore it and it will number the digital outputs sequentially according to the order of the modules in the bus.

- The parameters of all I/O nodes of the bus must be set; no node can be left undefined.
- The parameters of all the resources of a node must be set; no I/O can be left without a number.
- The maximum number of logic blocks in the system will be 64, both for digital inputs and digital outputs.
- The number of I/Os in a logic block must be other than zero and a multiple of 8 (8, 16, 24, 32, ...1024).
- The values of the base index must be comply with the formula "8n + 1" (i.e. 1, 9, 17, 25, etc.).
- The blocks may be distributed at will within the range of resources managed by PLC (1...1024). The PLC admits blanks, but the blocks cannot overlap; in other words, a PLC digital input or output can only belong to a single node-module of the bus.

NODE

Possible values: From 1 to 125.
Default value: 0.
Associated variable: (V.)MPG.DOMODNODE[nb]

Node number of the header to which the I/O logic block belongs. The parameters of all I/O nodes of the bus must be set; no node can be left undefined.

BLOCK

Possible values: From 1 to 64.
Default value: 0.
Associated variable: (V.)MPG.DOMODBLOCK[nb]

Number of the logic block. The maximum number of logic blocks in the system will be 64, both for digital inputs and digital outputs.

ADDRESS

Possible values: From 1 to 1017.
Default value: The first valid value.
Associated variable: (V.)MPG.DOMODADDR[nb]

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

In the RIO5 nodes, each double module (module with two boards) it counts as two, in other words, two logic blocks. These modules must be assigned two base indexes for the inputs and two base indexes for the outputs; one for each board.

NDO

Possible values: From 8 to 1024.
Default value: 0.
Associated variable: (V.)MPG.DOMODNDO[nb]

Number of digital outputs of the logic block; it must be other than zero and a multiple of 8 (8, 16, 24, 32, ...1024). The parameters of all the resources of a node must be set; no I/O can be left without a number.

2.11.15 Numbering of analog inputs for temperature sensors PT100.

NPT100

Possible values: From 0 to 20.
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 must be active if it has one of these temperature sensors connected to it. If the input is active in the machine parameter and there is no sensor connected, the CNC returns the corresponding error.

PT100

This parameter shows the list of PT100 inputs active in the CAN bus. The table has one parameter for each PT100 input.

Parameter.	Meaning.
PT100 n	Analog input associated with the PT100 input.

PT100 n

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

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. When numbering the analog inputs, the CNC considers the PT100 inputs as analog inputs. This way, for numbering, the CNC considers the following:

- The RIO5 modules have 6 analog inputs; four analog inputs (AI1..AI4) plus two PT100 inputs (AI5..AI6).
- The RIOR modules have 4 analog inputs; the two analog inputs (AI1..AI2) and the two PT100 inputs (AI3..AI4).

Example.
In a system with an RIO5 module (node 1) and an RIOR (node 2), the CNC identifies the analog inputs as follows.

- (address = 0) CNC
- (address = 1) RIO5
  - Analog inputs. - 1..4
  - PT100 inputs. - 5..6
- (address = 2) RIOR
  - Analog inputs. - 7..8
  - PT100 inputs. - 9..10

To have 3 PT100 inputs active (the two of the first module and the first one of the second module), the PT100 parameters may be set as follows.

NPT100 = 3
PT100 1 = 5
PT100 2 = 6
PT100 3 = 9

2.11.16 Number of special analog inputs (non-voltage).

NSAI

Possible values: From 0 to 30.
Default value: 0 (all inputs are analog voltage inputs).
Associated variable: (V.)MPG.NSAI

Number of analog inputs that are non-voltage inputs; for example, the analog inputs of the RIOR module can be configured by current.

SAI

This parameter shows the table to define the special analog inputs of the system. The following fields must be defined for each analog input.

SAI	Meaning.
ADDRESS	Analog input
FACTOR	Conversion factor.

ADDRESS

Possible values: From 0 to 60.
Default value: 0.
Associated variable: (V.)MPG.ADDRESS[n]

Analog input number.

FACTOR

Possible values: Within ±999999999.999999999.
Default value: 0.0061035.
Associated variable: (V.)MPG.FACTOR[n]

Conversion factor. The factor indicates the resolution of the A/D converter of the analog input; this means it represents the increment of the analog value for each converter unit.

- For the current analog inputs of the RIOR or RIOW module, use the default value so that the CNC can read the values in deciamperes.
- For third-party analog inputs, calculate the conversion factor based on the measurement range (magnitude) and resolution (number of bits). Consult the module manufacturer for both these values.

Example. Analog inputs for the RIOR module.

- 16 bit analog inputs de (15 bits + sign); meaning values within ±32767.
- Measuring range ±20 milliamperes.

Factor = 200 / 32767 = 0.0061035 (to return the value in deciamperes)

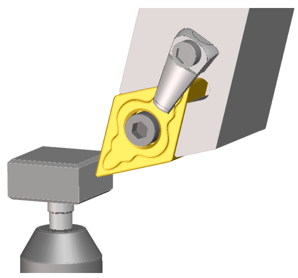
- For analog inputs that have a non-linear conversion or an asymmetric offset, set the factor as "1" and carry out the transformation with the PLC.

2.11.17 Probe setting.

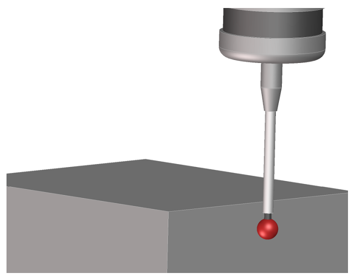
PROBE

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 two probes; usually a tabletop probe to calibrate tools and a measuring probe to measure the part.



Tabletop probe (lathe).



Measuring probe.

PROBEDATA

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.

Parameter.	Meaning.
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.

PROBEDATA | PROBETYPE1
PROBEDATA | PROBETYPE2

Possible values: Remote/Local/Gap (laser model only).
Default value: Remote.
Associated variable: (V.)MPG.PROBETYPE1
Associated variable: (V.)MPG.PROBETYPE2

This parameter indicates the type of probe; “remote” if it is connected to a digital input for remote modules or “local” if it is connected to a probe input for a central unit or "gap" in order to use the proximity sensor as a probe (for example, to measure the inclination of a sheet). The “Gap” option is only available for the laser model. The PROBETYPE1 parameter corresponds to probe 1 and the PROBETYPE2 parameter to probe 2.

PROBEDATA | PRBDI1
PROBEDATA | PRBDI2

Possible values (1): If a remote probe; from 1 to 1024.
Possible values (2): If a local probe; from 1 to 2.
Default value: 0 (no probe is connected).
Associated variable: (V.)MPG.PRBDI1
Associated variable: (V.)MPG.PRBDI2

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; for a proximity sensor it is not necessary to define this parameter. The parameter PRBDI1 corresponds to probe 1 and parameter PRBDI2 to probe 2.

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.

PROBEDATA | PRBPULSE1
PROBEDATA | PRBPULSE2

Possible values: Positive / Negative.
Default value: Positive.
Associated variable: (V.)MPG.PRBPULSE1
Associated variable: (V.)MPG.PRBPULSE2

This parameter indicates whether the probe functions are active high (24V or 5 V) or active low (0 V) of the signal provided by the probe. The probe signal must be at least 20 ms long for the CNC to consider it valid. The parameter PRBPULSE1 corresponds to probe 1 and parameter PRBPULSE2 to probe 2.

To use the proximity sensor as a probe, set PRBPULSE=negative to use G100 or PRBPULSE=positive for G103. The probing will finalize when the INTOL input of the PLC is deactivated.

2.11.18 Shared PLC memory.

PLCDATASIZE

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

Size of the PLC's shared data area. 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.

2.11.19 Configuring local digital outputs (XTX platform).

NLOCOUT

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.

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

EXPSCHK

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

When a local digital output is active (NLOCOUT parameter), 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.

2.11.21 UPS control.

UPS

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

This parameter indicates whether a UPS power supply is connected.

2.11.22 Synchronized switching.

SWTOUTPUT

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 canceled when switching from G1/G2/G3 to G0.

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

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

Possible values: From 0 to 100 milliseconds.
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.

SWTG0FEED

Possible values: Yes / No.
Default value: Yes.
Associated variable: (V.)MPG.SWTG0FEED

The SWTG0FEED parameter determines the feedrate for the sections in G0.

SWTG0FEED	Meaning.
Yes	Movements in G0 are carried out at a rapid feedrate (parameter G00FEED).
No	The cut is made at a constant feedrate (with the feedrate of the previous G1/G2/G3), without accelerating in the sections in G0. This option makes it possible to prevent possible problems with machine frequency excitation, thus producing better quality precision and surface cuts.

The behavior can be modified from the part program with the instruction #SWTOUT. If according to the parameter or the command (#SWTOUT) the G0s are executed as G1 (constant feedrate), G0 is displayed in the history.

2.11.23 PWM (Pulse-Width Modulation).

PWMOUTPUT

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.

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

PWMCANCEL

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.

Parameter.	Meaning.
Yes	The CNC cancels the PWM after a M30 and reset. When there is an error in the channel when the PWM is enabled (or in any channel if it is managed via a PLC), the CNC disables the PWM. 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.
No	The CNC does not disable the PWM after a M30 and reset. When there is an error in the channel when the PWM is enabled (or in any channel if it is managed via a PLC), the CNC keeps the status of the output; meaning that it does not disable the output if it is enabled.

2.11.24 Power control.

PWRCTRLACT

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

Manage power based on the actual feed. The CNC allows control of the power, laser or PWM duty, based on the feed. This parameter defines the feed to be used by the CNC to manage the power.

Value.	Meaning.
Yes.	The CNC managers the power based on the actual feed (variable (V.)A.FEED.xn).
No	The CNC manages the power based on the theoretical feed (variable (V.)A.TFEED.xn).

PWRCTRLCANCEL

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

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

Value.	Meaning.
Yes.	The CNC cancels the power control. A channel error cancels the power control. If an error disables the power control, it will not be re-enabled when the error disappears. It should be programmed again via the program or PLC.
No	The CNC does not cancel the power control. A channel error does not cancel the power control; this will maintain its previous status. At the beginning of the execution, after a reset and upon accessing the tool inspection, the CNC synchronizes the preparation of blocks with the execution.

2.11.25 Gap control.

GAPCONTROL

This parameter shows the table to define the gap control/

GAPCONTROL	
GAPANAINTYPE	Type of analog input connected to the sensor.
GAPANAINID	Number of the analog input connected to the sensor.
GAPDISTLIMIT	Distance corresponding to the limit analog signal of the sensor.
GAPVOLTLIMIT	Voltage corresponding to the limit path of the sensor .
GAPSENSOROFFSET	Offset applied to the sensor.
GAPSENSORCH	Change the sensor signal sign.
GAPGAIN	Proportional gain when applying in GAPCTRL.
GAPAPPROACHDYN	Dynamic response during the final part of the movement approaching the plate.
GAPSENSORFILTER	Table for configuring the sensor filter.
GAPERRORCANCEL	Canceling the gap error outside the range defined by GAPMIN/GAPMAX.
GAPMIN	Minimum sensor limit.
GAPMAX	Maximum sensor limit.
GAPTOLCANCEL	Canceling the gap error outside the range defined by GAPTOL.
GAPTOL	Error tolerance margin.
GAPCOLLISIONMODE	CNC behavior in the event of a sensor collision.
GAPTONEG	Movement block in #GAPCTRL towards descending coordinate.
GAPSENSORWATCH	This parameter displays the table for defining the gap sensor monitoring.
GAPSMINLEVEL	Minimum signal value for emergency action.
GAPSPEEDUP	Incorrect probe signal upload speed.
GAPSPEEDDOWN	Incorrect downstream speed on probe signal.
GAPSTEP	Maximum position setpoint increment below signal level.
GAPMINTIMEOK	Minimum time of correct signal to exit from surveillance.
GAPMAXTIMEKO	Maximum time of incorrect signal to enter surveillance.

GAPCONTROL | GAPANAINTYPE

Possible values: CAN remote / EtherCAT.
Default value: EtherCAT.
Associated variable: (V.)MPG.GAPANAINTYPE

This parameter indicates the location of the analog output connected to the sensor. It is advisable to use an EtherCAT input.

Value.	**Meaning.**
Remote CAN.	The analog input on remote CAN modules.
EtherCAT	The gap sensor is connected to an analog input of an EtherCAT bus.

GAPCONTROL | GAPANAINID

Possible values (1): If GAPANAINTYPE=Remote CAN, from 1 to 60.
Possible values (2): If GAPANAINTYPE=EtherCAT, from 1 to 1024.
Default value: None (no gap control).
Associated variable: (V.)MPG.GAPANAINID

Number of the analog input connected to the sensor. In either case, this parameter indicates the analog output being used.

Analog input of the CAN 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. If there are several analog-output modules in each group, the order is from left to right.

Analog input of remote EtherCAT modules.

In this case, this parameter indicates the PLC register that maps the analog input connected to the sensor.

GAPCONTROL | GAPDISTLIMIT

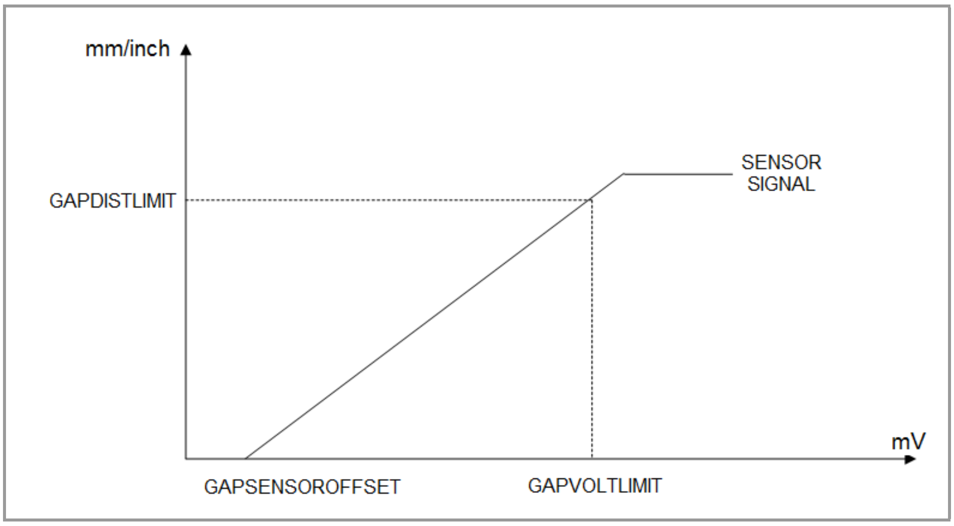
Possible values: From 0 to 99999.9999 mm / From 0 to 3937.00787 inches.
Default value: 10.0000 mm / 0.39370 inch.
Associated variable: (V.)MPG.GAPDISTLIMIT

Distance corresponding to the limit analog signal of the sensor (parameter GAPVOLTLIMIT).

GAPCONTROL | GAPVOLTLIMIT

Possible values: From 0 to 10000 mV
Default value: 10000 mV.
Associated variable: (V.)MPG.GAPVOLTLIMIT

Voltage (in millivolts) corresponding to the limit path of the sensor (parameter GAPDISTLIMIT).



GAPCONTROL | GAPSENSOROFFSET

Possible values: Between ±10000 mV.
Default value: 0 mV.
Associated variable: (V.)MPG.GAPSENSOROFFSET

Offset (in millivolts) to apply to the sensor from the CNC. If a Fagor analog input is used, this parameter is not required, as each has its own offset.

GAPCONTROL | GAPSENSORCH

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

Change the sensor signal sign.

GAPCONTROL | GAPGAIN

Possible values: From 0.0000 to 100.0000.
Default value: 0.2.
Associated variable: (V.)MPG.GAPGAIN

Proportional gain applied to the signal sensor, in position. To use values greater than 0.2 (close to 1), it is advisable to use the parameter GAPSENSORFILTER.

GAPCONTROL | GAPAPPROACHDYN

Possible values: From 10 to 120 %.
Default value: 80 %.
Associated variable: (V.)MPG.GAPAPPROACHDYN

Dynamic response during the final part of the movement approaching the plate. This parameter reduces, by a set percentage, the GAPGAIN value of the sensor axis during the approach to the sheet. The CNC applies this parameter in the approach associated with the gap control (instruction #GAPCTRL) as well as in that associated with leapfrog with gap control (instruction #LEAP). This parameter does not have any effect on the rest of the movements when the gap control is active.

Values below 100% permit a smoother approach, eliminating any possible overshooting of the gap during these approaches. Increasing the value of this parameter can improve the lowering speed, but it can also cause a small overshooting of the position. Reducing this parameter too much can cause a small jolt at the end of the approach when the movement begins on the XY axis with gap control, as there is a sudden change in GAPGAIN.

GAPCONTROL | GAPSENSORFILTERID

Possible values: Any filter defined in the "Filters" branch.
Default value: None.
Associated variable: (V.)[ch].MPG.GAPSENSORFILTERID

Gap sensor filter identifier.

GAPCONTROL | GAPERRORCANCEL

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

The GAPERRORCANCEL parameter establishes the CNC behavior when the gap exceeds the GAPMIN - GAPMAX range. Regardless of the value of this parameter, the CNC keeps monitoring the INPOSGAP mark.

Value.	Meaning.
Yes.	The CNC cancels the out-of-range gap error and does not stop the movement of the axes.
No	The CNC returns an out-of-range gap error and stops the movement of the axes, according to the braking ramp and controlling the gap during the ramp.

GAPCONTROL | GAPMIN

Possible values: From 0 to 99999.9999 mm / From 0 to 3937.00787 inches.
Default value: 0.
Associated variable: (V.)MPG.GAPMIN

Minimum value permitted for the gap. If the gap exceeds the range defined in the parameters GAPMIN and GAPMAX, the PLC deactivates the INPOSGAP mark. The behavior of the CNC depends on the GAPTOLCANCEL parameter.

GAPCONTROL | GAPMAX

Possible values: From 0 to 99999.9999 mm / From 0 to 3937.00787 inches.
Default value: 5.0000 mm / 0.19685 inch.
Associated variable: (V.)MPG.GAPMAX

Maximum value permitted for the gap. This parameter should be lower than GAPDISTLIMIT. If the gap exceeds the range defined in the parameters GAPMIN and GAPMAX, the PLC deactivates the INPOSGAP mark. The behavior of the CNC depends on the GAPTOLCANCEL parameter.

GAPCONTROL | GAPTOLCANCEL

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

The GAPTOLCANCEL parameter establishes the behavior of the CNC when the gap exceeds the margin set in the GAPTOL parameter.

Value.	Meaning.
Yes.	The CNC cancels the out-of-tolerance gap error and does not stop the movement of the axes.
No	The CNC returns an out-of-tolerance gap error and stops the movement of the axes, according to the braking ramp and controlling the gap during the ramp.

GAPCONTROL | GAPTOL

Possible values: From 0 to 99999.9999 mm / From 0 to 3937.00787 inches.
Default value: 0.1000 mm / 0.00394 inch.
Associated variable: (V.)MPG.GAPTOL

This parameter defines a positioning window on the programmed gap, to finish the approach block. If the axis exceeds the tolerance set in the GAPTOL parameter, the PLC deactivates the INTOL mark. The behavior of the CNC depends on the GAPTOLCANCEL parameter.

GAPCONTROL | GAPCOLLISIONMODE

Possible values: Do not display error / Display error.
Default value: Issue an error.
Associated variable: (V.)MPG.GAPCOLLISIONMODE

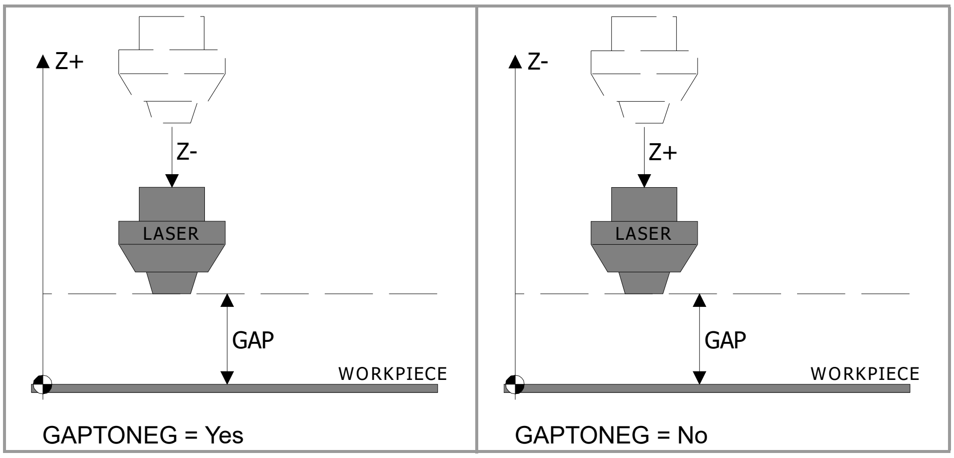
CNC behavior in the event of a sensor collision. The gap control sensor may have a collusion signal connected to the CNC, which should be managed by the PLC via the GAPCOLLISION mark. If the PLC activates the GAPCOLLISION mark, the CNC will display an error or not, depending on how this parameter has been set. The error does not activate the emergency.

GAPCONTROL | GAPTONEG

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

This parameter indicates whether the movement down from the Z axis to reach the gap is in a negative direction (normal conditions) or a positive direction (inverted Z axis).

Value.	Meaning.
Yes.	The positive direction of the Z axis is up and the negative is down (under normal conditions). The axis moves in the negative direction (down) to reach the gap.
No	The positive direction of the Z axis is down and the negative direction is up (inverted Z axis). The axis moves in a positive direction (down) to reach the gap.



GAPCONTROL | GAPSENSORWATCH

This parameter displays the table for defining the gap sensor monitoring.

GAPCONTROL	
GAPSMINLEVEL	Minimum signal value for emergency action.
GAPSPEEDUP	Incorrect probe signal upload speed.
GAPSPEEDDOWN	Incorrect downstream speed on probe signal.
GAPSTEP	Maximum position setpoint increment below signal level.
GAPMINTIMEOK	Minimum time of correct signal to exit from surveillance.
GAPMAXTIMEKO	Maximum time of incorrect signal to enter surveillance.

GAPCONTROL | GAPSENSORWATCH | GAPSMINLEVEL

Possible values: From 0 to 6.0000 mm / From 0 to 0.23622 inch.
Default value: 0.3 mm / 0.01181 inch.
Associated variable: (V.)MPG.GAPSMINLEVEL

Minimum signal value for emergency action.

GAPCONTROL | GAPSENSORWATCH | GAPSPEEDUP

Possible values: From 0 to 327000.0000 mm/min / From 0 to 12874.0200 inch/min.
Default value: 2000 mm/min / 0.00394 inch/min
Associated variable: (V.)MPG.GAPSPEEDUP

Incorrect probe signal upload speed.

GAPCONTROL | GAPSENSORWATCH | GAPSPEEDDOWN

Possible values: From 0 to 327000.0000 mm/min / From 0 to 12874.0200 inch/min.
Default value: 2000 mm/min / 0.00394 inch/min
Associated variable: (V.)MPG.GAPSPEEDDOWN

Incorrect downstream speed on probe signal.

GAPCONTROL | GAPSENSORWATCH | GAPSTEP

Possible values: From 0 to 99999.9999 mm / From 0 to 3937.00787 inches.
Default value: 0.3 mm / 0.01181 inch.
Associated variable: (V.)MPG.GAPSTEP

Maximum position setpoint increment below signal level.

GAPCONTROL | GAPSENSORWATCH | GAPMINTIMEOK

Possible values: From 0 to 655 ms.
Default value: 8 ms.
Associated variable: (V.)MPG.GAPMINTIMEOK

Minimum time of correct signal to exit from surveillance.

GAPCONTROL | GAPSENSORWATCH | GAPMAXTIMEKO

Possible values: From 0 to 655 ms.
Default value: 2 ms.
Associated variable: (V.)MPG.GAPMAXTIMEKO

Maximum time of incorrect signal to enter surveillance.

2.11.26 Leapfrog.

LEAPDYNOVR

Possible values: From 10 to 100000 %.
Default value: 100 %.
Associated variable: (V.)MPG.LEAPDYNOVR

This parameter enables the dynamic response of the Z axis (the longitudinal axis) to be reduced during the leapfrog for a smoother leap, in both the up and down flanks. Making the movement smoother will result in a longer machining time when there are leapfrogs.

Height of the leapfrog.

The height level that is reached in a leapfrog (#LEAP) depends on the ACCJERK, DECJERK, ACEL and DECEL values. The leapfrog requires high ACEL and DECEL values (of around 15,000 mm/s² or greater), as well as ACCJERK and DECJERK values (of around 20 times the ACEL and DECEL).

Parameter.	Meaning.
SLOPETYPE	Type of acceleration by default; linear, trapezoidal and square sine.
ACCEL DECEL	Acceleration and deceleration. Parameters ACCEL and DECEL set the acceleration values when it is trapezoidal or square-sine (bell shaped) (parameter SLOPETYPE).
ACCJERK DECJERK	Acceleration and deceleration jerk. 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.

Plate approach.

The CNC executes the approach to the plate with linear acceleration LACC1 and LACC2, and thus this must not be very high (around a third of the ACEL and DECEL).

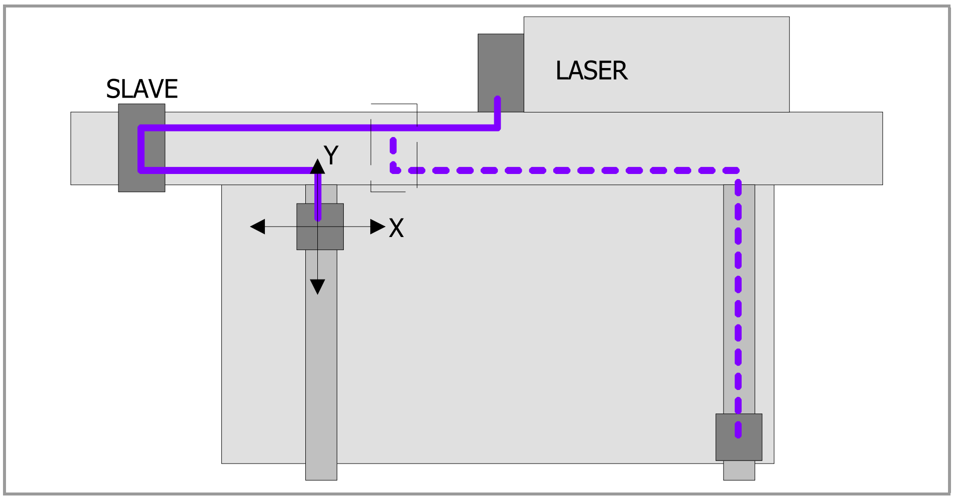
Parameter.	Meaning.
LACC1 LACC2	Acceleration of the first and second section. Parameters LACC1 and LACC2 set the acceleration values when it is linear (SLOPETYPE parameter).

Example.
For an ACEL and DECEL = 18,000 mm/s².
- LACC1 and LACC2 = 6000 mm/s².
- ACCJERK and DECJERK= 360,000 mm/s³.

2.11.27 Compensating the dispersion by the CO2 laser path.

LASERFOLLOW

Compensate the dispersion by the CO2 laser path. This parameter shows the table to define the compensation. The goal of the compensation is to keep the total laser path constant in order to prevent dispersion variations in the cutoff point on the sheet.



For the compensation, a slave axis, which follows the main axes, is used. The dispersion compensation is activated from the program using the instruction #FOLLOW or from the PLC using the instruction FOLLOW.

LASERFOLLLOW	
LASERFOLLOWAXIS	Slave axis applied to which FOLLOW is applied.
LASERFOLLOWOFFSET	Offset of the X Y Z axes with respect to the slave axis.
LASERFOLLOW1	Indicates how the first of the X Y Z axes affect the slave axis.
LASERFOLLOW2	Indicates how the second of the X Y Z axes affect the slave axis.
LASERFOLLOW3	Indicates how the third of the X Y Z axes affect the slave axis.

LASERFOLLOW | LASERFOLLOWAXEIS

Possible values: X, X1··X9, ·· , C, C1··C9.
Default value: 0 (no compensation).
Associated variable: (V.)MPG.LASERFOLLOW

Name of the slave axis for compensation (axis on which the instruction #FOLLOW or the PLC command FOLLOW acts). The variable V.A.FLWMASTER.xn of this axis will be the main element of the synchronization.

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 spindles may be within the range X, X1…X9,...C, C1…C9. For example X, X1, Y3, Z9, W, W7, C...

LASERFOLLOW | LASERFOLLOWOFFSET

Possible values: Within ±99999.9999 mm / Within ±3937.00787 inch.
Default value: 0.
Associated variable: (V.)MPG.LASERFOLLOWOFFSET

Source offset of the X Y Z axes regarding the slave axis. This offset allows the minimization of the total laser path.

LASERFOLLOW | LASERFOLLOW1

Possible values: No effect / Positive / Negative.
Default value: Not affected.
Associated variable: (V.)MPG.LASERFOLLOW1

Effect of the first of the X Y Z axes on the slave axis. This parameter indicates if the first axis (according to the machine parameters) of the channel that manages the compensation affects the slave axis, as well as its direction.

LASERFOLLOW | LASERFOLLOW2

Possible values: No effect / Positive / Negative.
Default value: Not affected.
Associated variable: (V.)MPG.LASERFOLLOW2

Effect of the second of the X Y Z axes on the slave axis. This parameter indicates if the second axis (according to the machine parameters) of the channel that manages the compensation affects the slave axis, as well as its direction.

LASERFOLLOW | LASERFOLLOW3

Possible values: No effect / Positive / Negative.
Default value: Not affected.
Associated variable: (V.)MPG.LASERFOLLOW3

Effect of the second of the X Y Z axes on the slave axis. This parameter indicates if the third axis (according to the machine parameters) of the channel that manages the compensation affects the slave axis, as well as its direction.

- If any of the parameters LASERFOLLOW1, LASERFOLLOW2 or LASERFOLLOW3 is different from 0, the CNC will automatically update the variable V.A.FLWMASTER.xn of the slave axis (LASERFOLLOWAXIS parameter) as follows.

  V.A.FLWMASTER.xn = V.MPG.LASERFOLLOWOFFSET +
  V.MPG.LASERFOLLOW1 * V.A.TPOS.X +
  V.MPG.LASERFOLLOW2 * V.A.TPOS.Y +
  V.MPG.LASERFOLLOW3 * V.A.TPOS.Z

- If parameters LASERFOLLOW1, LASERFOLLOW2 and LASERFOLLOW3 are 0, the OEM should enter the value of the variable V.A.FLWMASTER.xn (e.g. from the PLC).

2.11.28 Backup of non-volatile data.

BKUPREG

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 non-volatile RAM. 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

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 non-volatile RAM. 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.

BKUPCUP

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

This parameter indicates the number of common arithmetic counters that are saved in non-volatile RAM. The CNC saves the first parameters, from the first one to the one set in this parameter.

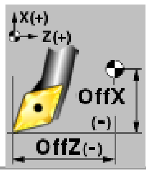
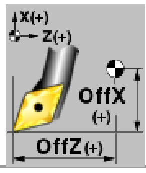
Non-volatile data means that their values must be maintained from one session to another and after turning the CNC off. The CNC saves this data when the CNC is turned off, when power is removed, when a hardware error occurs, etc.

2.11.29 Tool offset and wear.

TOOLOFSG

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

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.

Value.	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.

2.11.30 Spindle synchronization.

SYNCCANCEL

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.

2.11.31 Define the number of jog panels and their relationship with the channels.

NKEYBD

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

Number of jog panels.

KEYBDCH

This parameter shows the table to assign the jog panels to the channels. The table shows a parameter for each operator panel.

Parameter.	Meaning.
KEYBDnCH	Channel to which the operator panel is assigned.

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

KEYBDCH | KEYBDnCH

Possible values: Disabled / CH1 / CH2 / CH3 / CH4.
Default value: CH1.
Associated variable: (V.)MPG.KEYBDCH[jog]

Channel to which the operator panel is assigned. Each operator panel may be disabled or assigned to a specific channel. When an operator panel is assigned to a channel, it is always operative even when 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. The assignment of jog panels can be modified from the PLC using the marks KEYBD1CH to KEYBD8CH.

2.11.32 Rename the axes and the spindles.

RENAMECANCEL

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 cancels the names of the axes and spindles 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.

2.11.33 Zero offsets.

FINEORG

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.

- Absolute zero offset table (without fine setting of the zero offset).

Channel 1 : Zero offsets			
Origin	X (mm)	Y (mm)	Z (mm)
PLCOF	00000.0000	00000.0000	00000.0000
G158	00054.5000	00010.0000	00000.0000
G54 (G159=1)	00000.0000	00000.0000	00000.0000
G55 (G159=2)	00000.0000	00000.0000	00000.0000
G56 (G159=3)	00000.0000	00000.0000	00000.0000

- Absolute zero offset table (with fine setting of the zero offset).

Channel 1 : Zero offsets			
Origin	X (mm)	Y (mm)	Z (mm)
PLCOF	00000.0000	00000.0000	00000.0000
G158	00054.5000	00010.0000	00000.0000
G54 (G159=1)	00050.0000	00000.0000	00000.0000
△	00003.0000	00000.0000	00000.0000
G55 (G159=2)	00000.0000	00000.0000	00000.0000
△	00000.0000	00000.0000	00000.0000
G56 (G159=3)	00010.0000	00000.0000	00000.0000
△	00000.0000	00000.0000	00000.0000

2.12 General machine parameters. Execution channels.

2.12.1 Channel configuration.

GROUPID

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

Channel group identifier. 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

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

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

Hidden channel. 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.

2.12.2 Configuring the axes of the channel.

CHNAXIS

Possible values: From 0 to 10.
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

This parameter shows the table to define the names of the axes. The table has one parameter for each axis.

Parameter.	Meaning.
CHAXISNAME n	Axis name.

CHAXISNAME | CHAXISNAME n

Possible values: Any axis defined in the parameter "AXISNAME".
Default value: CHAXISNAME1=X, CHAXISNAME2=Y, CHAXISNAME3=Z, etc.
Associated variable: (V.)[ch].MPG.CHAXISNAMEn

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

Index of the axis in the channel.

The order in which the axes are defined in the CHAXISNAME table determines their index in the channel. Index 1 will be given to the first axis, Index 2 to the second axis, 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

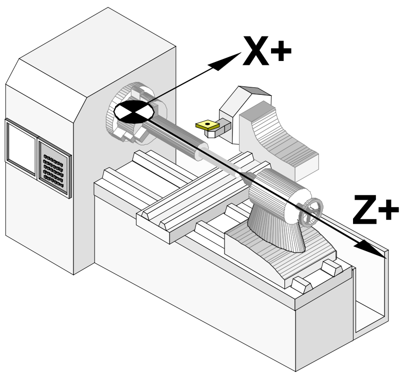
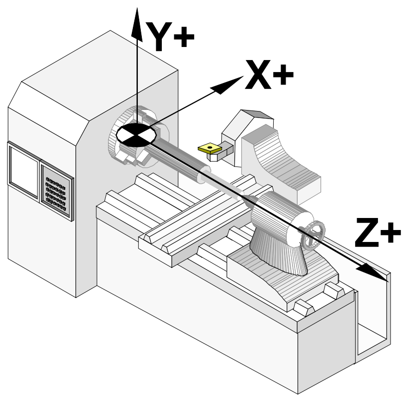
The order of the axes and work planes (lathe model).

The order of the axes of the channel and parameter GEOCONFIG set the main work planes. See *"GEOCONFIG"* on page 135.

GEOCONFIG

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

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

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.

2.12.3 Configuring the spindles of the channel.

CHNSPDL

Possible values: From 0 to 3.
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

This parameter shows the table to define the names of the spindles. The table has one parameter for each axis.

Parameter.	Meaning.
CHSPDLNAME n	Name of the spindles.

CHSPDLNAME | CHSPDLNAME n

Possible values: Any spindle defined in the parameter SPDLNAME.
Default value: CHSPDLNAME1=S, CHSPDLNAME2=S1, CHSPDLNAME3=S2, etc.
Associated variable: (V.)[ch].MPG.SPDLNAMEn

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

Index of the axis in the channel.

The order in which the spindles are set in the CHAXISNAME table determines their index in the channel. Index 1 will be given to the first spindle, Index 2 to the second spindle, and so on. As with the spindle name, the index in the channel permits identifying the axis in PLC variables, marks, etc.

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

2.12.4 Configuration of the C axis.

CAXNAME

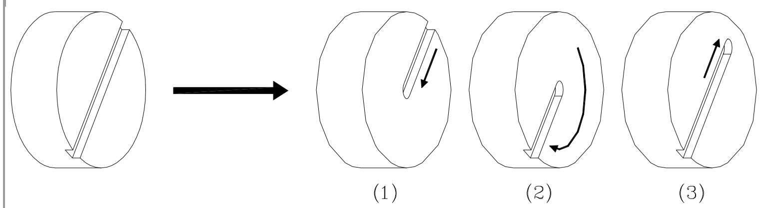
Possible values: Any valid axis name; X, X1··X9, ·· , C, C1··C9.
Default value: C.
Associated variable: (V.)[ch].MPG.CAXNAME

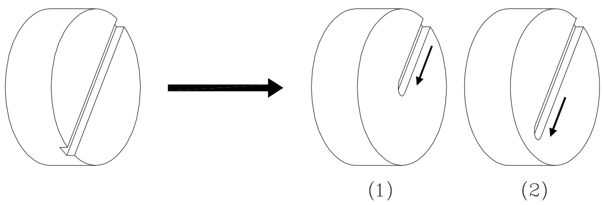
C axis default name This parameter must be defined whenever an axis or spindle is set as the C axis (CAXIS parameter).

ALIGNC

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

"C" axis alignment for diametrical machining.

<table>
<tr><th>Value.</th><th>Meaning.</th></tr>
<tr><td>Yes.</td><td>The tool cannot machine the entire surface diametrically in a single run; the axis C must be aligned.


(1) Machining up to the center.
(2) Rotate the "C" axis 180°.
(3) Resume machining by moving the tool back.</td></tr>
<tr><td>No.</td><td>The tool can machine the entire surface diametrically in a single run; the axis C does not have to be aligned.

</td></tr>
</table>

2.12.5 Time setting (channel).

PREPFREQ

Possible values: From 1 to 12.
Default value: 1.
Associated variable: (V.)[ch].MPG.PREPFREQ

While executing a program, the CNC reads in advance the blocks to execute in order to calculate the tool path, which is known as block preparation. The number of blocks to be prepared depends on the type of machining operation; processing blocks generated with a CAD-CAM and blocks derived from a program edited manually or with canned cycles are not the same thing.

This parameter sets the maximum number of blocks the CNC must process in each cycle (parameter CNCTIME). 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

Possible values: From 0 to 10 000 000 milliseconds.
Default value: 0.
Associated variable: (V.)[ch].MPG.ANTIME

Anticipation time. 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.

2.12.6 Configuration of the HSC mode (channel).

HSC

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

Parameter.	Meaning.
HSCDEFAULTMODE	Default mode to activate HSC machining.
FEEDAVRG	Calculating feedrate average.
MINCORFEED	Minimum feedrate at the corners.
CORNER	Maximum angle between two paths, under which the machining operation is carried out in square corner mode
SMOOTHFREQ	Smoothing frequency for the tool path interpolation.
HSCFILTFREQ	Axis filter frequency (CONTERROR mode).
FASTFACTOR	Desired acceleration percentage of the maximum that the CNC can reach in FAST and SURFACE modes.
FTIMELIM	Admissible time loss in order to optimize quality (FAST and SURFACE mode).
FSMOOTHFREQ	Smoothing frequency of the speed profile (FAST and SURFACE modes).
FASTFILTFREQ	Frequency of profile filter (FAST mode).
SURFFILTFREQ	Axis filter frequency (SURFACE mode).
FREQRES	First resonance frequency of the machine, which the CNC must remove when generating the setpoint.
SOFTFREQ	Path filter frequency for profiles of linear acceleration.
HSCROUND	Default value of the maximum path error in HSC.
ORISMOOTH	Orientation smoothing of the rotary axes working with RTCP.
HSCFILTERID	HSC filter identifier.

HSC | HSCDEFAULTMODE

Possible values: SURFACE / CONTERROR / FAST.
Default value: SURFACE.
Associated variable: (V.)[ch].MPG.HSCDEFAULTMODE

Default mode to activate HSC machining (high speed machining).

Value.	Meaning.
SURFACE	Optimize the surface finish.
CONTERROR	Optimize the contouring error.
FAST	Optimize the machining feedrate.

HSC | FEEDAVRG

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

HSC | MINCORFEED

Possible values (1): From 0 to 200 000.0000 mm/min.
Possible values (2): From 0 to 7874.01575 inches/min.
Default value: 0.
Associated variable: (V.)[ch].MPG.MINCORFEED

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

HSC | CORNER

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.

HSC | SMOOTHFREQ

Possible values: From 0 to 500.0000 Hz.
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.

HSC | HSCFILTFREQ

Possible values: From 0 to 500.0000 Hz.
Default value: 40.
Associated variable: (V.)[ch].MPG.HSCFILTFREQ

Axis 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.

HSC | FASTFACTOR

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

This parameter determines the feedrate at the corners and indicates the desired acceleration percentage 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.

HSC | FTIMELIM

Possible values: From 0 to 100 000.0000 %.
Default value: 200 %.
Associated variable: (V.)[ch].MPG.FTIMELIM

Admissible time loss in order to optimize quality (FAST and SURFACE mode). The feedrate interpolation in HSC FAST and SURFACE modes 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.

HSC | FSMOOTHFREQ

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

This parameter sets the smoothing frequency of the speed profile in HSC FAST and SURFACE modes. This parameter avoids accelerating and decelerating throughout a path, beyond a particular frequency when generating an averaged speed.

HSC | FASTFILTFREQ

Possible values: From 0 to 500.0000 Hz.
Default value: 17 Hz.
Associated variable: (V.)[ch].MPG.FASTFILTFREQ

Frequency of profile filter (FAST mode). 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.

HSC | SURFFILTFREQ

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

Axis filter frequency (SURFACE mode). 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.

HSC | FREQRES

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

First resonance frequency of the machine, which the CNC must remove when generating the setpoint. 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.

HSC | SOFTFREQ

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

Path filter frequency for profiles of linear acceleration. This parameter may be used to smooth the velocity profile in HSC SURFACE mode, which improves machining time and surface quality. This parameter must be set using the FineTune application.

HSC | HSCROUND

Possible values (1): From 0.0001 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Default value: 0,06 mm / 0,00236 inches.
Associated variable: (V.)[ch].MPG.HSCROUND

Default value of the maximum path error in HSC.

HSC | ORISMOOTH

Possible values: From 0 to 200 milliseconds.
Default value: 60 milliseconds.
Associated variable: (V.)[ch].MPG.ORISMOOTH

Orientation smoothing of the rotary axes working with RTCP. This parameter smooths the orientation of the rotary axes, without tool tip error, when working with RTCP in HSC SURFACE mode.

HSC | HSCFILTERID

Possible values: Any filter defined in the "Filters" branch.
Default value: None.
Associated variable: (V.)[ch].MPG.HSCFILTERID

HSC filter identifier.

2.12.7 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

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

Name of the virtual tool axis. 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

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.

2.12.8 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

Possible values: From 0 to 6 / None.
Default value: None (the CNC does not assume any kinematics on power-up).
Associated variable: (V.)[ch].MPG.KINID

This parameter indicates the number of kinematics (no type) assumed by the CNC at startup, after executing an M02, M30 or after a reset; machine parameter validation does not change the active kinematics. If set to ·0·, on power-up the CNC restores the kinematics that was active when it was turned off. If set to "None", 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

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. When the value is “No”, the CNC saves the final result of the CS + ACS programmed in a single CS; it does not maintain the programmed nesting.

LINKCANCEL

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

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

This parameter indicates whether the mirror image is canceled after executing an M02/M30 and after a reset.

SLOPETYPE

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.

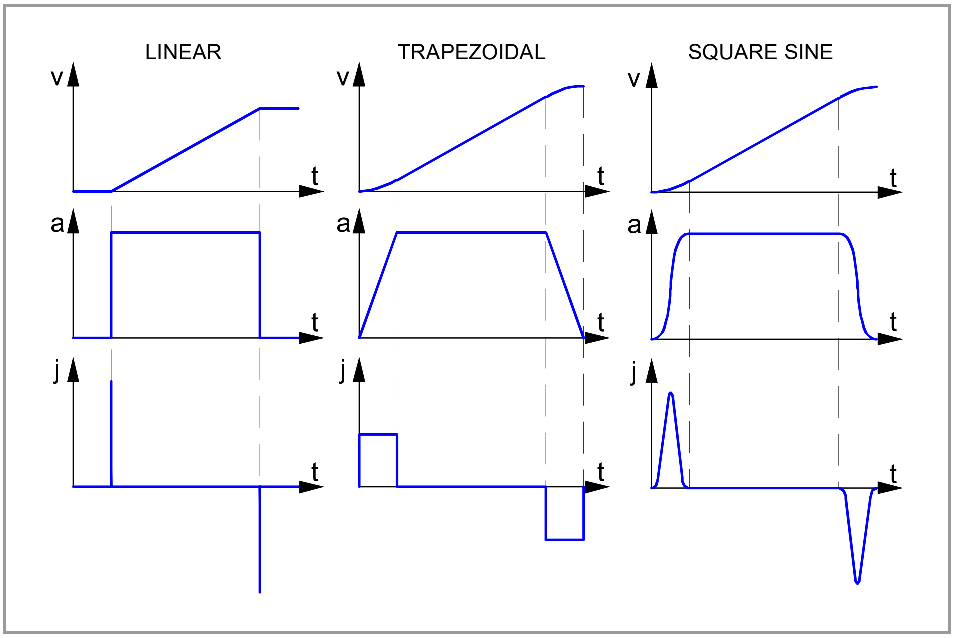
Depending on the type of acceleration selected, the machine parameters will show the ones needed to configure the acceleration. To select a different acceleration via part-program, for the automatic mode, use the instruction #SLOPE.

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 next figure shows the feedrate (v), acceleration (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.



Parameter SLOPETYPE for the spindle.

The parameter SLOPETYPE does not affect the spindle (M3/M4), which always uses jerk acceleration (parameters ACCEL/DECEL and ACCJERK/DECJERK). When the spindle works as a C axis, it will use linear acceleration or jerk acceleration depending on the type of machining.

Acceleration type.	Spindle as C axis. Type of machining.
Linear.	El CNC uses the parameters LACC1/LACC2/LFEED. • Non-HSC Mode (G7/G50/G5) when SLOPETYPE=Linear. • HSC CONTERROR mode when SLOPETYPE=Linear. • HSC SURFACE/FAST mode.
Trapezoidal.	El CNC uses the parameters ACCEL/DECEL/ACCJERK/DECJERK. • Non-HSC Mode (G7/G50/G5) when SLOPETYPE=Trapezoidal. • HSC CONTERROR mode when SLOPETYPE=Trapezoidal.
Square sine (bell shaped).	El CNC uses the parameters ACCEL/DECEL/ACCJERK/DECJERK. • Non-HSC Mode (G7/G50/G5) when SLOPETYPE=Square sine. • HSC CONTERROR mode when SLOPETYPE=Square sine.

IPLANE

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

This parameter indicates the work plane assumed by the CNC at startup. The parameter PLANECANCEL determines whether the CNC also assumes this plane after executing an M02, M30 or after a reset. A machine parameter validation does not change the active plane.

The axes that form the work plane depend on machine parameter CHAXISNAME.

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

M models (mill), L models (laser), OL (Open Line) and GL (General Line).

It is recommended to define the IPLANE parameter as G17. To change the work plane via part-program, use function G17, G18, G19 or G20.

T model (lathe). Configuration of "Trihedron" type axes.

It is recommended to define the IPLANE parameter as G17. To change the work plane via part-program, use function G17, G18, G19 or G20.

T model (lathe). Configuration of "plane" type axes.

The work plane is always G18; machine parameter IPLANE is not applied and it is not possible to change planes via the part-program, only the longitudinal axis (G20).

ISYSTEM

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

This parameter indicates the type of coordinates assumed by the CNC at startup, after executing an M02, M30 or after a reset; machine parameter validation does not change the type of active coordinates. 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

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

This parameter indicates the type of movement assumed by the CNC at startup, after executing an M02, M30 or after a reset; machine parameter validation does not change the type of active coordinates. Movements in G0 are carried out at the feedrate set in 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.

Parameter G0MODAL; maintaining function G0 as modal or non-modal.

The value IMOVE=G0 has priority over G0MODAL=No. The G0 function is defined as non-modal, however, with IMOVE=G0, the CNC assumes the G0 function at start-up after a reset or M30. After programming G1, G2 or G3, the CNC recovers G1 as a modal function until the following reset or M30.

IFEED

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

This parameter indicates the type of feedrate assumed by the CNC at startup, after executing an M02, M30 or after a reset; machine parameter validation does not change the type of active feedrate.

- 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.

G0MODAL

Possible values: Yes / No.
Default value: Yes.
Associated variable: (V.)[ch].MPG.G0MODAL.

This parameter defines whether the G00 function is modal or not.

G0MODAL	Meaning.
Yes.	The G00 function is modal. After it has been programmed, it remains active until G01, G02, G03, G33 or G63 are programmed.
No.	The G00 function is non-modal, it must be programmed for each rapid traverse block. For the following program block, if there is no programmed movement function (G0, G1, G2, G3, G33 or G63), the CNC uses G1, meaning that G1 remains as modal. Within the range of influence of a canned cycle, the last programmed G will be maintain active, while G0 remains being modal.

IMOVE parameter; type of movement (G0/G1) at startup, after a reset or M30.

The value IMOVE=G0 has priority over G0MODAL=No. The G0 function is defined as non-modal, however, with IMOVE=G0, the CNC assumes the G0 function at start-up after a reset or M30. After programming G1, G2 or G3, the CNC recovers G1 as a modal function until the following reset or M30.

G2G3MODAL

Possible values: Yes / No.
Default value: Yes.
Associated variable: (V.)[ch].MPG.G2G3MODAL.

This parameter defines whether the G2/G3 functions are modal or not.

G2G3MODAL	Meaning.
Yes.	The G02 and G03 functions are modal. Once they are programmed, they remain active until G00, G01, G33 or G63 are programmed.
No.	The G02 and G03 functions are non-modal. These must be programmed for each circular interpolation block. For the following program block, if there is no programmed movement function (G0, G1, G2, G3, G33 or G63), the CNC uses G1.

FPRMAN

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

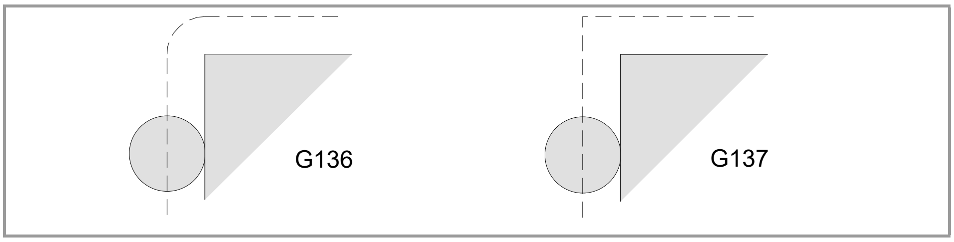
This parameter indicates whether or not the G95 function (feed per turn) is allowed for jog movements from manual mode; it does not affect movement with handwheels or in MDI/MDA mode.

Value.	Meaning.
Yes	The G95 function (feed per turn) is allowed for jog movements from manual mode. The CNC applies the G95 function to the main axes of the channel; the other axes are not affected. The real feedrate (Freal) displayed in jog and conversational modes always is in mm/turn or inch/turn, whether the moving axis is a main axis or not.
No	The G95 function (feed per turn) is not allowed for jog movements from manual mode. When switching from automatic mode to manual mode with G95 active, the history still shows G95 but the jog movements will be in G94 (forward per minute). The same applies if G95 is activated from MDI over manual mode. The F programmed in manual mode, with G95 in the history, will be (V.)G.FMAN (feed per minute) for jog movements and (V.)G.PRGFPR (feed per revolution) for automatic mode. For example, when programming F10 in manual mode with G95 in history: • The axis will move at 10mm/min in jog. • The axis shall move at 10mm/rev*SReal[rpm] in MDI or when moving an axis to a [X][START] coordinate.

IRCOMP

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

This parameter indicates the compensation mode assumed by the CNC at startup, after executing an M02, M30 or after a reset; machine parameter validation does not change the type of active compensation mode. Being the radius compensation active, the compensated paths may be blended together using circular paths (G136) or linear paths (G137).



COMPCANCEL

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.

G00COMP

Possible values: Yes / No.
Default value: Yes (milling and lathe) / No (laser).
Associated variable: (V.)[ch].MPG.G00COMP

This parameter indicates whether the CNC applies tool radius compensation or not in G0 movements.

LCOMPTYP

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.

PLANECANCEL

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

This parameter determines whether the CNC must assume the default plane (parameter IPLANE) after an M30/RESET or to keep the active plane.

Value.	Meaning.
Yes	On power-up, after executing an M02 or M30, and after a reset, the CNC assumes the plane defined in parameter IPLANE.
No	On power-up, the CNC assumes the plane defined in parameter IPLANE; after executing an M02 or M30, and after a reset, the CNC maintains the main active plane.

ICORNER

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

This parameter indicates the type of edge assumed by the CNC at startup, after executing an M02, M30 or after a reset; machine parameter validation does not change the type of active edge. 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

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 (G05). 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

Possible values (1): From 0.02 to 99 999.9999 mm or degrees.
Possible values (2): From 0.00079 to 3937.00787 inches.
Default value: 0.1000 mm or degrees / 0.00394 inches.
Associated variable: (V.)[ch].MPG.MAXROUND

This parameter sets the maximum rounding error in G05 (maximum permitted deviation between the programmed point and the profile resulting from edge rounding). 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

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 in G5. The CNC takes it into account if ROUNDTYPE = % Feedrate.

2.12.9 Arc center correction.

CIRINERR

Possible values (1): From 0 to 99 999.9999 mm or degrees.
Possible values (2): From 0 to 3937.00787 inches.
Default value: 0.0100 mm or degrees / 0.00039 inches.
Associated variable: (V.)[ch].MPG.CIRINERR

Maximum absolute error for circle center correction. See general machine parameter CIRINFACT.

CIRINFACT

Possible values: from 0 to 100.0 %.
Default value: 0.1 %.
Associated variable: (V.)[ch].MPG.CIRINFACT

Maximum relative error for circle center correction. Both parameters CIRINERR and CIRINFACT 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.

2.12.10 Behavior of the feedrate and the feedrate override.

MAXOVR

Possible values: From 0 to 255 Hz.
Default value: 200 Hz.
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

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

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

Breaks down the feedrate for all the axes of the channel.

Value.	Meaning.
Yes.	The CNC breaks down the programmed feedrate for all the axes of the channel.
No.	The CNC breaks down the programmed feedrate for in the main axes of the channel. The rest of the axes move at their corresponding feedrate to end the movement of them all at the same time. The CNC only limits the programmed feedrate if any one of the axes 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.

2.12.11 Override of the dynamics.

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 a machining operation. This parameter sets the minimum percentage to be applied using the slider. If both parameters MINDYNOVR and MAXDYNOVR are set at 100%, the CNC will not show the slider and it will not be possible to change in real time the machining dynamics.

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 a machining operation. This parameter sets the maximum percentage to be applied using the slider. If both parameters MINDYNOVR and MAXDYNOVR are set at 100%, the CNC will not show the slider and it will not be possible to change in real time the machining dynamics.

2.12.12 DMC configuration.

MINDMCOVR

Possible values: From 10 to 100 %.
Default value: 80 %.
Associated variable: (V.)[ch].MPG.MINDMCOVR

Minimum DMC override of all channel axes. See machine parameter MAXDMCOVR.

MAXDMCOVR

Possible values: From 100 to 255 %.
Default value: 120 %.
Associated variable: (V.)[ch].MPG.MAXDMCOVR

Maximum DMC override of all channel axes. Parameters MINDMCOVR and MAXDMCOVR indicate the minimum and maximum percentage of feedrate (feedrate override) that DMC (dynamic machining control) can apply on the channel. Both percentages can be changed in the instruction #DMC ON. The feedrate percentage applied by DMC is added to that selected using the operator panel switch.

$$\mathrm{Override}(\%) = \frac{\mathrm{OverrideDMC}(\%) \times \mathrm{OverrideJOG}(\%)}{100}$$

The resulting override (overrideDMC + overrideJOG) may exceed the set limits for both parameters. The CNC always abides by the maximum limit set in the machine parameter MAXOVR.

If the user selects an override with the switch on the operator panel that is lower than MINDMCOVR, the CNC inhibits DMC (but does not deactivate it); when the override once again exceeds MINDMCOVR, DMC will once again operate as normal.

The feedrate percentage set by the program (variable V.G.PRGFRO) or by the PLC (variable V.PLC.FRO) inhibit DMC, but they do not deactivate it; if canceled, DMC resumes control over the feedrate percentage.

DMCPEAKSIZE

Possible values: From 0 to 100 %.
Default value: 20 %.
Associated variable: (V.)[ch].MPG.DMCPEAKSIZE

Excess power percentage, above the rated power, for monitoring the power peaks in the DMC. If it is set to 0, the peaks are not monitored. If the power read exceeds the power to achieve (target power+no-load power) in the defined percentage, the CNC issues a warning. It stops the axes while moving the spindle and activates the DMCPWRPEAK mark.

This parameter enables the detection of a power peak, which may be due to the breakage of the tool, depth of cut too large, etc.

DMCPWRFACTOR

Possible values: From 0.0 to 100.0.
Default value: 2.0.
Associated variable: (V.)[ch].MPG.DMCPWRFACTOR

Multiplication factor of the target power to be deemed excessive consumed power. If defined with a value smaller than or equal to "1", power consumption will not be monitored. If all the values of the power read over 2 seconds exceeds the power to achieve multiplied by this factor, the CNC will issue a warning and activate the DMCPWRHOLD mark.

This parameter enables the detection of excessive power consumption which may be due to tool wear.

2.12.13 Movement of the independent axes.

IMOVEMACH

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

Independent movement without coordinate transformation.

Value.	Meaning.
Yes	The independent movement of the axes is carried out using the machine coordinates, before RTCP, CS and ACS coordinate transformations.
No	The independent movement of the axes is carried out using the part coordinates, before RTCP, CS and ACS coordinate transformations.

XFITOIND

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.

Value.	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.

2.12.14 Definition of the subroutines.

SUBTABLE

The manufacturer's subroutines are those stored in folder ..\MTB\Sub. These subroutines are associated with functions of the CNC such as tool change, G74, G180 to G189, G380 to G399, etc.

If the same subroutines are executed repeatedly during execution, it is more efficient to load them into the RAM memory of the CNC because this way, they may be accessed faster and execution time may be consequently optimized. To load a subroutine into RAM, it must have the extension .fst.

Being the CNC in USER mode, the OEM routines having a .fst extension are loaded into RAM memory when starting up the CNC application. Being the CNC in SETUP mode, the OEM subroutines whose extension is .fst will be loaded into RAM memory the first time they are executed in the program.

SUBTABLE | TOOLSUB

Possible values: Any subroutine in the folder ..\Mtb\ Sub.
Associated variable: (V.)[ch].MPG.TOOLSUB

Subroutine associated with the T function. This subroutine is executed automatically every time a T function (tool selection) is executed.

SUBTABLE | CAXSUB

Possible values: Any subroutine in the folder ..\Mtb\ Sub.
Associated variable: (V.)[ch].MPG.CAXSUB

Subroutine associated with the #CAX instruction. This subroutine is executed automatically every time the CNC executes the #CAX function (spindle activation as C axis).

SUBTABLE | REFPSUB (G74)

Possible values: Any subroutine in the folder ..\Mtb\ Sub.
Associated variable: (V.)[ch].MPG.REFPSUB

Subroutine associated with function G74. 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.

SUBTABLE | INT1SUB
..
SUBTABLE | INT4SUB

Possible values: Any subroutine in the folder ..\Mtb\ Sub.
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.

SUBTABLE | INITIALSUB

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

This parameter indicates the name of the subroutine associated with the instruction #INITIALSUB. Every time the CNC executes this instruction, its associated subroutine is called upon.

SUBTABLE | PIERCING

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

This parameter indicates the name of the subroutine associated with the instruction #PIERCING. Every time the CNC executes this instruction, its associated subroutine is invoked (by default, Piercing.fst).

The OEM is responsible for ensuring that the subroutine has all the safety aspects related to handling the laser.

At the start of the subroutine (or when the OEM deems it necessary), the OEM must activate the PIERCING mark (assigning it the value 1) to indicate to the PLC that it must start the piercing operation. At the end of the subroutine (or when the OEM deems it necessary), the OEM must deactivate this mark (assigning it the value 0) to indicate to the CNC that it has finished the piercing operation.

SUBTABLE | CUTTINGON

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

This parameter indicates the name of the subroutine associated with the instruction #CUTTING ON. Every time the CNC executes this instruction, its associated subroutine is invoked (by default, Cuttingon.fst).

The OEM is responsible for ensuring that the subroutine has all the safety aspects related to handling the laser.

At the start of the subroutine Cuttingon.fst (or when the OEM deems it necessary), the OEM must activate the CUTTING mark (assigning it a value of 1) to indicate to the PLC that it should start the cutting operation.

SUBTABLE | CUTTINGOFF

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

This parameter indicates the name of the subroutine associated with the instruction #CUTTING OFF. Every time the CNC executes this instruction, its associated subroutine is invoked (by default, Cuttingoff.fst).

The OEM is responsible for ensuring that the subroutine has all the safety aspects related to handling the laser.

At the end of the Cuttingoff.fst subroutine (or when the OEM deems it necessary), the OEM must deactivate the CUTTING mark (assigning it a value of 0) to indicate to the CNC that it has finished the cutting operation.

SUBTABLE | FINALSUB

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

This parameter indicates the name of the subroutine associated with the instruction #FINALSUB. Every time the CNC executes this instruction, its associated subroutine is called upon.

SUBTABLE | OEMSUB (G180)

..

SUBTABLE | OEMSUB (G189)

Possible values: Any subroutine in the folder ..\Mtb\ Sub.
Associated variable: (V.)[ch].MPG.OEMSUB1 / (V.)[ch].MPG.OEMSUB10

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.

SUBTABLE | OEMSUB (G380)

..

SUBTABLE | OEMSUB (G399)

Possible values: Any subroutine in the folder ..\Mtb\ Sub.
Associated variable: (V.)[ch].MPG.OEMSUB11 / (V.)[ch].MPG.OEMSUB30

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.

SUBPATH

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.

2.12.15 Tabletop probe position.

PROBEDATA

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.

Value.	Meaning.
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).

PROBEDATA | PRB1MAX

Possible values (1): Between ±99 999.9999 mm.
Possible values (2): Within ±3937.00787 inches.
Default value: 0.
Associated variable: (V.)[ch].MPG.PRB1MAX

Maximum probe coordinate (first axis of the channel). See general machine parameter PRB3MIN.

PROBEDATA | PRB1MIN

Possible values (1): Between ±99 999.9999 mm.
Possible values (2): Within ±3937.00787 inches.
Default value: 0.
Associated variable: (V.)[ch].MPG.PRB1MIN

Minimum probe coordinate (first axis of the channel). See general machine parameter PRB3MIN.

PROBEDATA | PRB2MAX

Possible values (1): Between ±99 999.9999 mm.
Possible values (2): Within ±3937.00787 inches.
Default value: 0.
Associated variable: (V.)[ch].MPG.PRB2MAX

Maximum probe coordinate (second axis of the channel). See general machine parameter PRB3MIN.

PROBEDATA | PRB2MIN

Possible values (1): Between ±99 999.9999 mm.
Possible values (2): Within ±3937.00787 inches.
Default value: 0.
Associated variable: (V.)[ch].MPG.PRB2MIN

Minimum probe coordinate (second axis of the channel). See general machine parameter PRB3MIN.

PROBEDATA | PRB3MAX

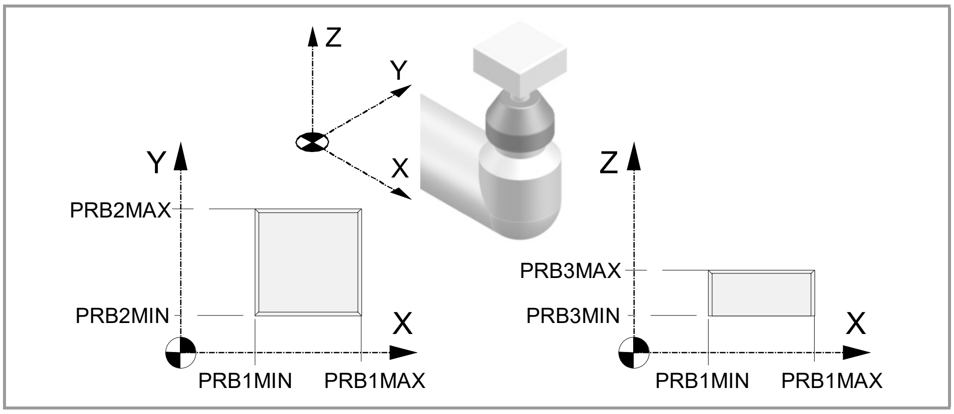
Possible values (1): Between ±99 999.9999 mm.
Possible values (2): Within ±3937.00787 inches.
Default value: 0.
Associated variable: (V.)[ch].MPG.PRB3MAX

Maximum probe coordinate (third axis of the channel). See general machine parameter PRB3MIN.

PROBEDATA | PRB3MIN

Possible values (1): Between ±99 999.9999 mm.
Possible values (2): Within ±3937.00787 inches.
Default value: 0.
Associated variable: (V.)[ch].MPG.PRB3MIN

Minimum probe coordinate (third axis of the channel). 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.



2.12.16 Block search.

FUNPLC

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.

Value.	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.

2.12.17 Interruption subroutines.

SUBINTSTOP

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.

2.12.18 Machining feedrate.

MAXFEED

Possible values (1): From 0 to 500 000.0000 mm/min.
Possible values (2): From 0 to 19 685.03937 inches/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 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.

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.

This parameter cannot be set with a value higher than that of parameter 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.

DEFAULTFEED

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

Assuming MAXFEED for movements in G1/G2/G3 without programming the programmed feedrate. The CNC takes this parameter into account when MAXFEED has been set with a value other than ·0·.

Value.	Meaning.
Yes	When a movement is programmed in G01/G02/G03 without programming the feedrate, the axes move at the maximum machining feedrate (parameter MAXFEED)
No	When a movement is programmed in G01/G02/G03 without programming the feedrate, the CNC will prompt an error.

2.12.19 Rapid traverse for the automatic mode.

RAPIDEN

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.

Value.	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 operator panel.
EXRAPID and rapid key.	To activate the rapid feed, just activate the PLC mark EXRAPID and press the "rapid" key of the operator 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

Possible values (1): From 0 to 500 000.0000 mm/min.
Possible values (2): From 0 to 19 685.03937 inches/min.
Default value: 0.
Associated variable: (V.)[ch].MPG.FRAPIDEN

Maximum rapid feedrate 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.

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).

2.12.20 Maximum acceleration and jerk on the tool path.

MAXACCEL

Possible values (1): From 0.0010 to 600 000 000.0000 mm/min.
Possible values (2): From 0.00004 to 23 622 047.24409 inches/min.
Default value: 0 (no maximum acceleration).
Associated variable: (V.)[ch].MPG.MAXACCEL

Maximum acceleration on the machining path. 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

Possible values (1): From 0.0010 to 6E11 mm/min.
Possible values (2): From 0.00004 to 2.362E10 inches/min.
Default value: 0 (no maximum jerk).
Associated variable: (V.)[ch].MPG.MAXJERK

Maximum jerk on the machining path. 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.

2.12.21 Maximum frequency on the tool path.

MAXFREQ

Possible values: From 0 to 500 Hz.
Default value: 0.
Associated variable: (V.)[ch].MPG.MAXFREQ

Maximum frequency generated on the machining path. 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.

2.12.22 "Retrace" function.

RETRACAC

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

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

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.

2.12.23 Enable tool withdrawal (retraction) in threading.

RETRACTTHREAD

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

Enable tool withdrawal (retraction) in threading. This parameter defines the CNC behavior when interrupting a threading ([STOP] key or PLC mark _FEEDHOL).

Value.	Meaning.
No	The CNC stops the axes at the end of the pass.
Yes	• In electronic threading (G33/G34), function G233 defines the distance that the axes withdraw from the part. If G233 is not active, the axes stop at the end of the pass. • In threading canned cycles, (-T- model), both ISO and conversational, the CNC withdraws the axes from the part. The distance that the tool withdraws depends on how the cycle is programmed.

2.12.24 Master spindle.

MASTERSPDL

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

Treatment of the master spindle after executing an M02, M30, after an emergency or reset and after restarting the CNC.

Value.	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.

2.13 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.

In the laser model, the parameters related to spindle speed (defined in rpm) define the power (defined in watts).

2.13.1 Belonging to the channel.

AXISEXCH

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

Channel change permission. 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.

Value.	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.

2.13.2 Axis type.

AXISTYPE

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

Type of axis.

2.13.3 Hirth axis.

HIRTH

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

Hirth toothed axis. A Hirth axis is the one that can only be positioned at positions multiple of a given value (parameter HPITCH). 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

Parameter valid for linear and rotary axes.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Possible values (3): From 0 to 99 999.9999 degrees.
Default value: 1 mm / 0,03937 inches / 1 degree.
Associated variable: (V.)[ch].MPA.HPITCH.xn

Hirth axis 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 on-screen display of the coordinate with decimals of a Hirth axis is configured from the Fguim application.

2.13.4 Axis configuration for lathe type machines.

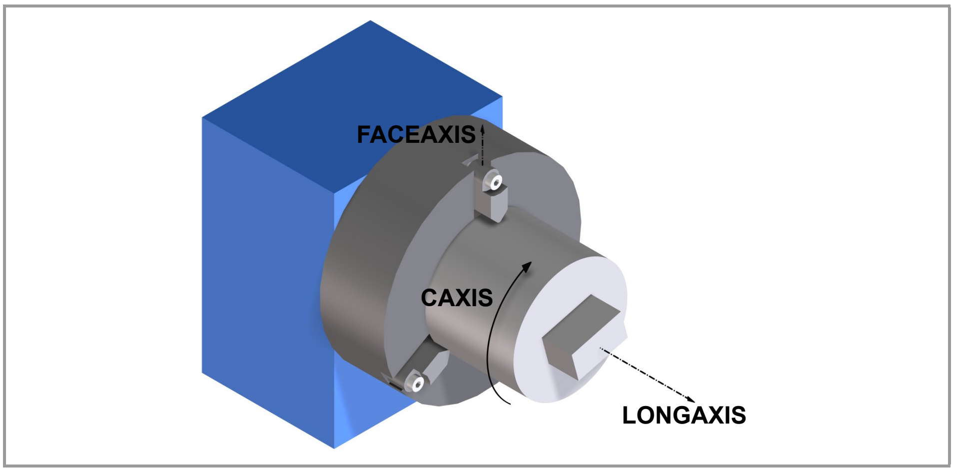
FACEAXIS

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

Working as a cross axis. On turning machines, you must indicate which one is the longitudinal axis and which one is the cross axis. See axis machine parameters LONGAXIS.

LONGAXIS

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



Typical lathe setting:

For lathes, the longitudinal axis and the cross axis must be indicated.

Typical lathe setting:		
X axis.	FACEAXIS = Yes	LONGAXIS = No
Z axis.	FACEAXIS = No	LONGAXIS = Yes
Rest of the axes.	FACEAXIS = No	LONGAXIS = No

Typical customization for mill without rotary axes:

For this type of machine, the longitudinal axis and the cross axis do not have to be indicated.

Typical Mill setting:		
All the axes.	FACEAXIS = No	LONGAXIS = No

Typical customization for mill with rotary axes:

For this type of machine, in order to use the rotary axis cycles, the longitudinal axis and the cross axis must be indicated. The rotary axis must have its machine parameter CAXIS = Yes; the linear axis associated with the rotary axis must have its machine parameter LONGAXIS= Yes; the cross axis must have its machine parameter FACEAXIS= Yes.

Typical customization for lathes (when then rotary axis is A):			
A axis.	- - -	- - -	CAXIS=Yes
X axis (linear axis associated with the rotary axis).	FACEAXIS = No	LONGAXIS = Yes	CAXIS=No
Z axis.	FACEAXIS = Yes	LONGAXIS = No	CAXIS=No

Typical customization for lathes (when then rotary axis is B):			
B axis.	- - -	- - -	CAXIS=Yes
Y axis (linear axis associated with the rotary axis).	FACEAXIS = No	LONGAXIS = Yes	CAXIS=No
Z axis.	FACEAXIS = Yes	LONGAXIS = No	CAXIS=No

Typical customization for lathes (when then rotary axis is C):			
C axis.	- - -	- - -	CAXIS=Yes
Z axis (linear axis associated with the rotary axis).	FACEAXIS = No	LONGAXIS = Yes	CAXIS=No
Either X or Y axis (only one of the two).	FACEAXIS = Yes	LONGAXIS = No	CAXIS=No

Customization of the dual-purpose machine (combined machine).

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.

Xca files and parameters FACEAXIS and LONGAXIS.

The *.def file accompanying the machine configuration (xca file) defines whether the graphics work with the tool base or tool tip coordinates. Also, for each tool, the FIXORI parameter defines whether the tool is oriented according to the axes defined by FACEAXIS and LONGAXIS.

- If the HD graphics work with the tool tip coordinates, then they respond to the parameters FACEAXIS and LONGAXIS.
- If the HD graphics work with the tool base coordinates, then they respond to the parameters FACEAXIS and LONGAXIS.

2.13.5 Synchronization of axes and spindles.

SYNCSET

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

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0 to 200 000.0000 mm/min.
Possible values (2): From 0 to 7874.01575 inches/min.
Possible values (3): From 0 to 36 000 000.0000 degrees/min.
Possible values (4): From 0 to 100 000 rpm.
Default value: 100 mm/min / 3.937 inch/min / 3600 degrees/min / 10 rpm.
Associated variable: (V.)[ch].MPA.DSYNCVELW.xn

Velocity synchronization window. 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

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0 to 99 999.9999 mm/min.
Possible values (2): From 0 to 3937.00787 inches/min.
Possible values (3): From 0 to 99 999,9999 degrees/min.
Default value: 0.0100 mm/min / 0.00039 inch/min / 0.0100 degrees/min.
Associated variable: (V.)[ch].MPA.DSYNCPOSW.xn

Position synchronization window. 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.

2.13.6 Configuration of the rotary axes.

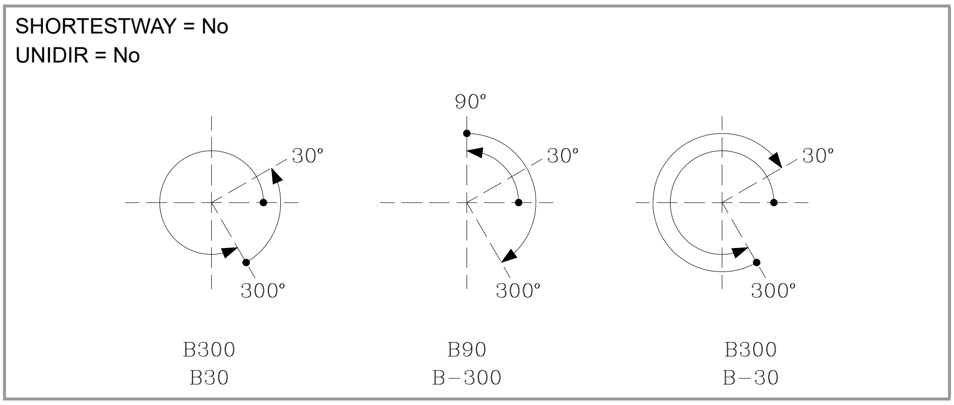
AXISMODE

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

Type of rotary axis. 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 can be programmed in G0/G1 and G90/G91.
- 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.



- 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 determined by set parameters MODUPLIM and MODLOWLIM. The coordinates are always displayed within the module limits, by default 0 and 360º.

  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.
- Parameters SHORTESTWAY and UNIDIR must be set.
- Parameters POSLIMIT and NEGLIMIT have not meaning.

Behavior when AXISMODE = Linearlike.

- The axis behaves like a linear axis; movements can be programmed in G0/G1 and G90/G91.
- The reading (pulse counting) is free and in degrees (it does not affect the millimeters / inches change)
- There are travel limits set by POSLIMIT and NEGLIMIT.
- Parameters SHORTESTWAY and UNIDIR have not meaning.
- The parameters MODUPLIM and MODLOWLIM do not apply.

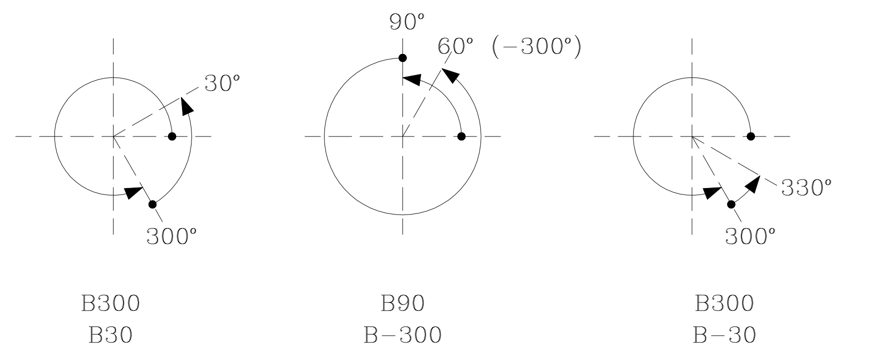
UNIDIR

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

Unidirectional rotation. 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 if 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.

UNIDIR = Positive.



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.

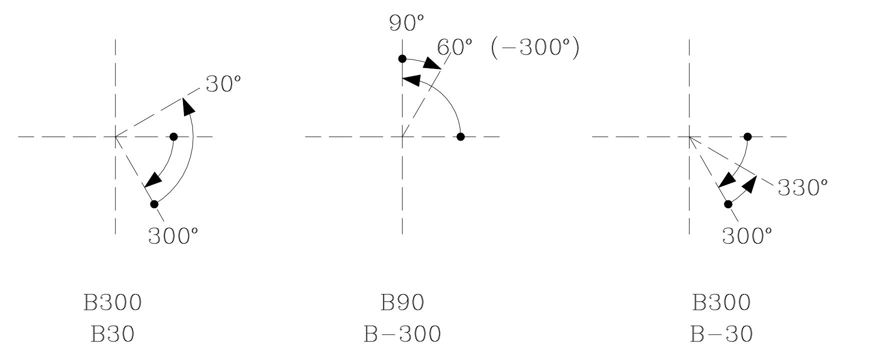
SHORTESTWAY

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.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.

SHORTESTWAY = Yes



The movements in G91 are carried out in the programmed direction.

2.13.7 Module configuration (rotary axes and spindle).

MODCOMP

Parameter valid for rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
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.

Enabling the module compensation. 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.

2.13.8 Activating the spindle for DMC.

DMCSPDL

Possible values: Yes / No.
Default value: No.
Associated variable: (V.)[ch].MPA.DMCSPDL.sn

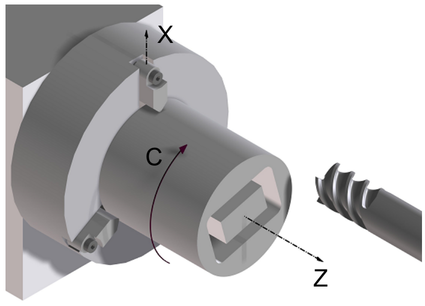
Spindle with enabled power control. This parameter indicates whether or not the DMC can act upon the spindle.

2.13.9 Configuration of the C axis.

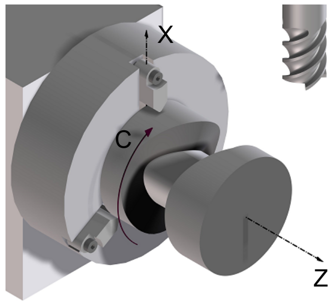
CAXIS

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

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

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.

2.13.10 Configuration of the spindle.

AUTOGEAR

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

Automatic gear change. 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

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

Upper rpm limit OK. See axis machine parameters UPSPDLIM.

UPSPDLIM

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

Lower rpm limit OK. When working with M3 and M4, the REVOK signal is set to high logic when the actual spindle revolutions are between the percentages set by LOSPDLIM y UPSPDLIM. The REVOK signal may be used to manage the Feedhold signal and avoid machining at lower or higher rpm than the ones programmed.

SPDLPOS

This parameter shows the table to set the special spindle positions.

Parameter.	Meaning.
SPDLPOS n	Special spindle position.

SPDLPOS | SPDLPOS 1
..
SPDLPOS | SPDLPOS 5

Parameter valid for spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 99 999.9999º.
Default value: 0.
Associated variable: (V.)[ch].MPA.SPDLPOS.xn

Special spindle position.

SPDLTIME

Parameter valid for spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 1 000 000 milliseconds.
Default value: 0 milliseconds.
Associated variable: (V.)[ch].MPA.SPDLTIME.xn

Estimated time for an S function. 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 this calculation, this parameter can be set to indicate the estimated time to process the S function. When assigning a value other than 0, the CNC assumes that the S value must be passed on to the PLC using the signals SSTROBE + SFUN1.

SPDLSTOP

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, function M5 will have to be programmed. The spindle errors and an emergency always stop the spindle.

- If it is set to ·No·, function G96 remains active with an M2, M30 or a reset.
- If it is set to ·Yes·, function G96 will not remain active with an M2, M30 or a reset.

SREVM05

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.

STEPOVR

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

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

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).

2.13.11 Spindle override change while threading.

THREADOVR

Parameter valid for spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: from 0 to 100 %.
Default value: 0 (The override cannot be changed).
Associated variable: (V.)[ch].MPA.THREADOVR.xn

Maximum variation of spindle override during threading. 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 does not allow the spindle override to be modified while threading if it detects that the feed forward (parameter FFWTYPE) is not active in a parameters set of the involved axes 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

Parameter valid for spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 1 000 000 milliseconds.
Default value: 0.
Associated variable: (V.)[ch].MPA.OVRFILTER.xn

Time filter to make the override change effective. The override change is applied progressively during the indicated period of time. The CNC takes this parameter into account only if THREADOVR other than 0 (zero).

2.13.12 Runaway protection and tendency test.

TENDENCY

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drives and Sercos drives.
Possible values: Yes / No.
Default value: No.
Associated variable: (V.)[ch].MPA.TENDENCY.xn

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 TENDTIME.

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 the setup is completed, this alarm must be enabled.

TENDTIME

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drives and Sercos drives.
Possible values: From 0 to 65 535 milliseconds.
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.

2.13.13 PLC Offset.

PLCOINC

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Possible values (3): From 0 to 99 999.9999 degrees.
Default value: 0 (they are assumed instantaneously).
Associated variable: (V.)[ch].MPA.PLCOINC.xn

PLC offset increment per cycle. 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.)A.PLCOF.xn. The PLC offset applied so far may be checked in the variable (V.)A.ACTPLCOF.xn.

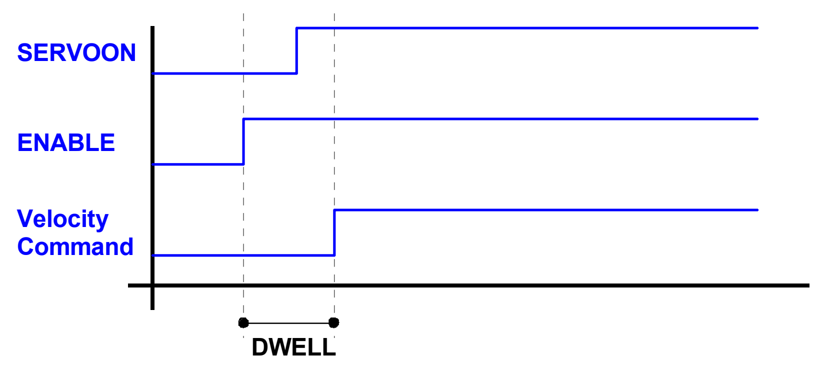
2.13.14 Dwell for dead axes.

DWELL

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 1 000 000 milliseconds.
Default value: 0 (there is 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 routine 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.

2.13.15 Radius / diameter.

DIAMPROG

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

Programming in diameters. To change the type of coordinates via program, use function G151 or G152. Offsets with respect to machine zero (#MCS) are always performed according to this parameter. For the cross axis of the lathes (parameter FACEAXIS = Yes), it is recommended that this parameter be set to “Yes".

2.13.16 Home search.

REFDIREC

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.

REFINI

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. This parameter is only taken into account when the parameter NPULSES has been set to a value other than 0.

2.13.17 Configuration of the probing movement.

PROBEAXIS

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.PROBEAXIS.xn

This parameter indicates if the axis can be involved in the probing movements.

PROBERANGE

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Possible values (3): From 0 to 99 999.9999 degrees.
Default value: 1.0000 mm / 0,03937 inches / 1.0000 degree.
Associated variable: (V.)[ch].MPA.PROBERANGE.xn

This parameter sets the maximum braking distance for the probe after probing, so as to avoid it from being damaged (ceramic probes, etc). The CNC issues an error messages when this distance is exceeded.

PROBEFEED

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0 to 36 000 000.0000 mm/min.
Possible values (2): From 0 to 1 417 322.83465 inches/min.
Possible values (3): From 0 to 36 000 000.0000 degrees/min.
Default value: 0.
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. If PROBEFEED is set to 0, the CNC calculates and uses the maximum possible value of this parameter.

PROBEDELAY
PROBEDELAY2

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Between ±100 000.0000 milliseconds.
Default value: 0 (no delay).
Associated variable: (V.)[ch].MPA.PROBEDELAY2.xn

Delay for the probe signal. 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. For the laser model, if the proximity probe for the gap control has a delay, it is defined in these parameters.

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.

2.13.18 Repositioning of the axes in tool inspection.

REPOSFEED

Parameter valid for linear and rotary axes.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0 to 200 000.0000 mm/min.
Possible values (2): From 0 to 7874.01575 inches/min.
Possible values (3): From 0 to 36 000 000.0000 degrees/min.
Default value: 0.
Associated variable: (V.)[ch].MPA.REPOSFEED.xn

Maximum repositioning feedrate after a tool inspection or interruption subroutine. If not set, the CNC assumes the repositioning feedrate as the one set for the continuous jog mode (parameter JOGFEED). The value of parameter REPOSFEED must always be lower than G00FEED, MAXMANFEED and JOGRAPFEED.

2.13.19 Configuration of the independent axis.

POSFEED

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0 to 200 000.0000 mm/min.
Possible values (2): From 0 to 7874.01575 inches/min.
Possible values (3): From 0 to 36 000 000.0000 degrees/min.
Possible values (4): From 0 to 100 000.0000 rpm.
Default value: 1000.0000 mm/min / 39.37008 inch/min / 1000.0000 degrees/min / 1000 rpm.
Associated variable: (V.)[ch].MPA.POSFEED.xn

Default positioning feedrate for an axis performing an independent movement.

2.13.20 Configure the maximum safety limit for the feedrate and for the speed.

FLIMIT

Parameter valid for linear and rotary axes.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0 to 500 000.0000 mm/min.
Possible values (2): From 0 to 19 685.03937 inches/min.
Possible values (3): From 0 to 72 000 000.0000 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

Parameter valid for spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 200 000.0000 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

2.13.21 Sampling period of the position loop.

POS_LOOPTIME

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0.2500 to 20.0000 milliseconds.
Default value: 1.0000 milliseconds.
Associated variable: (V.)[ch].MPA.POS_LOOPTIME.xn

Execution frequency of the position loop. The value of the POS_LOOPTIME must be the same or a multiple of that set for the parameter SYSTEMTIME.

2.13.22 JOG mode.

MANUAL

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.

Parameter.	Meaning.
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.

MANUAL | JOGFEED

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0 to 200 000.0000 mm/min.
Possible values (2): From 0 to 7874.01575 inches/min.
Possible values (3): From 0 to 36 000 000.0000 degrees/min.
Default value: 1,000.0000 mm/min / 39.37008 inches/min / 1,000.0000 degrees/min.
Associated variable: (V.)[ch].MPA.JOGFEED.xn

This parameter sets the feedrate for movements in continuous jog mode.

MANUAL | JOGRAPFEED

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0 to 200 000.0000 mm/min.
Possible values (2): From 0 to 7874.01575 inches/min.
Possible values (3): From 0 to 36 000 000.0000 degrees/min.
Default value: 10 000.0000 mm/min / 393.70079 inch/min / 10 000.0000 degrees/min.
Associated variable: (V.)[ch].MPA.JOGRAPFEED.xn

This parameter sets the rapid feedrate for movements in continuous jog mode.

MANUAL | MAXMANFEED

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0 to 200 000.0000 mm/min.
Possible values (2): From 0 to 7874.01575 inches/min.
Possible values (3): From 0 to 36 000 000.0000 degrees/min.
Default value: 10 000.0000 mm/min / 393.70079 inch/min / 10 000.0000 degrees/min.
Associated variable: (V.)[ch].MPA.MAXMANFEED.xn

This parameter sets the maximum feedrate for movements in continuous jog mode.

MANUAL | MAXMANACC

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0.0010 to 600 000 000.0000 mm/s².
Possible values (2): From 0.00004 to 23 622 047.2440 inches/s².
Possible values (3): From 0.0010 to 600 000 000.0000 degrees/s².
Default value: 10 000.0000 mm/s² / 393.70079 inch/s² / 10 000.0000 degrees/s².
Associated variable: (V.)[ch].MPA.MAXMANACC.xn

This parameter sets the maximum acceleration for movements in continuous jog mode.

MANUAL | INCJOGDIST

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 incremental jog selector switch. The table shows five parameters, one per each position of the operator panel.

Parameter.	Meaning.
INCJOGDIST1	Switch position ·1·.
INCJOGDIST2	Switch position ·10·.
INCJOGDIST3	Switch position ·100·.
INCJOGDIST4	Switch position ·1000·.
INCJOGDIST5	Switch position ·10000·.

MANUAL | INCJOGDIST | INCJOGDIST 1
··
MANUAL | INCJOGDIST | INCJOGDIST 5

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0.0001 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Possible values (3): From 0.0001 to 99 999.9999 degrees.
Default value: See the table below.
Associated variable: (V.)[ch].MPA.INCJOGDIST[pos].xn

Each parameter sets the distance that the axis moves in incremental jog each time the jog key is pressed. The most typical values are those set by default.

Parameter.	Default value.
INCJOGDIST1	0.0010 mm or degrees / 0.00003937 inches.
INCJOGDIST2	0.0100 mm or degrees / 0.00039370 inches.
INCJOGDIST3	0.1000 mm or degrees / 0.00393700 inches.
INCJOGDIST4	1.0000 mm or degrees / 0.03937007 inches.
INCJOGDIST5	10.000 mm or degrees / 0.39370078 inches.

INCJOGFEED

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.

This parameter shows the table to set the axis feedrate for each incremental jog selector switch position. The table shows five parameters, one per each position of the operator panel.

Parameter.	Meaning.
INCJOGFEED1	Switch position ·1·.
INCJOGFEED2	Switch position ·10·.
INCJOGFEED3	Switch position ·100·.
INCJOGFEED4	Switch position ·1000·.
INCJOGFEED5	Switch position ·10000·.

MANUAL | INCJOGFEED | INCJOGFEED 1
..
MANUAL | INCJOGFEED | INCJOGFEED 5

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 1 to 200 000.0000 mm/min.
Possible values (2): 0.03937 to 7874.01575 inches/min.
Possible values (3): From 1 to 200 000.0000 degrees/min.
Default value: 1,000.0000 mm/min / 39.37008 inches/min / 1,000.0000 degrees/min.
Associated variable: (V.)[ch].MPA.INCJOGFEED[pos].xn

Each parameter sets the feedrate of the incremental jog axis for each incremental jog switch position.

MANUAL | MPGRESOL

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.

Parameter.	Meaning.
MPGRESOL1	Switch position ·1·.
MPGRESOL2	Switch position ·10·.
MPGRESOL3	Switch position ·100·.

MANUAL | MPGRESOL | MPGRESOL 1
..
MANUAL | MPGRESOL | MPGRESOL 3

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): Between ± 99 999.9999 mm (0 not allowed).
Possible values (2): Between ± 3937.00787 inches (0 not allowed).
Possible values (3): Between ± 99 999.9999 degrees (0 not allowed).
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.	Default value.
MPGRESOL1	0.0010 mm or degrees / 0.00003937 inches.
MPGRESOL2	0.0100 mm or degrees / 0.00039370 inches.
MPGRESOL3	0.1000 mm or degrees / 0.00393700 inches.

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.

MANUAL | MPGFILTER

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: from 1 to 1000 cycles.
Default value: 10 cycles.
Associated variable: (V.)[ch].MPA.MPGFILTER.xn

Filter time for the handwheel. This filter smoothes the handwheel movements avoiding sudden variations. This parameter indicates the number of CNC cycles used to are the handwheel pulses read.

MANPOSSW

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): Between ±99 999.9999 mm.
Possible values (2): Within ±3937.00787 inches.
Possible values (3): Between ±99 999.9999 degrees.
Default value: 99 999.9999 mm / 3937.00787 inches / 99 999.9999 degrees.
Associated variable: (V.)[ch].MPA.MANPOSSW.xn

Maximum travel in a positive direction, in additive manual mode. See axis machine parameters MANNEGSW.

MANNEGSW

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): Between ±99 999.9999 mm.
Possible values (2): Within ±3937.00787 inches.
Possible values (3): Between ±99 999.9999 degrees.
Default value: -99 999.9999 mm / -3937.00787 inches / -99 999,9999 degrees.
Associated variable: (V.)[ch].MPA.MANNEGSW.xn

Maximum travel in a negative direction, in additive manual mode. 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

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

Maximum % of jog feedrate in G201. When using function G201, manual mode laid over automatic mode, this parameter indicates the maximum feedrate percentage for manual mode. It is recommended that the sum of the parameters MANFEEDP and IPOFEEDP do not exceed 100%, so as to avoid exceeding the dynamic limits of the machine under certain conditions.

IPOFEEDP

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

Maximum % of execution feedrate in G201. When using function G201, manual mode laid over automatic mode, this parameter indicates the maximum feedrate percentage for automatic mode. It is recommended that the sum of the parameters MANFEEDP and IPOFEEDP do not exceed 100%, so as to avoid exceeding the dynamic limits of the machine under certain conditions. When activating function G201, the feedrate assumes instantaneously the value set by IPOFEEDP.

MANACCP

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

Maximum % of jog acceleration in G201. When using function G201, manual mode laid over automatic mode, this parameter indicates the maximum acceleration for manual mode. It is recommended that the sum of the parameters MAXACCP and IPOACCP do not exceed 100%, so as to avoid exceeding the dynamic limits of the machine under certain conditions.

IPOACCP

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

Maximum % of execution acceleration in G201. When using function G201, manual mode laid over automatic mode, this parameter indicates the maximum acceleration for automatic mode. It is recommended that the sum of the parameters MAXACCP and IPOACCP do not exceed 100%, so as to avoid exceeding 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 operator 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.

2.13.23 Filters.

FILTERID 1
FILTERID 2
FILTERID 3

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Any value defined in the filter table.
Default value: None.
Associated variable: (V.)[ch].MPA.FILTERID[nb].xn

Filter identifier. Each of these parameters displays the list with the available system filters. The available identifiers are defined in the "Filters" branch of the parameter tree (left panel of the table). See *"2.18 Machine parameters; frequency filters."* on page 319.

Frequency filters may be applied to axes and spindles. Up to 3 different filters may be defined for each axis or spindle, hence being possible to eliminate several resonance frequencies. The filters defined for the spindle will only be applied when it works as a C axis or it is doing a rigid tapping. To obtain a good machining quality, all the axes interpolating with each other should be defined with the same type of filter and with the same frequency.

The CNC has different filters; a band-rejection and a low-pass filters. Only one of these 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-pass filter.

2.13.24 Work sets.

NPARSETS

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 1 to 12.
Default value: 1.
Associated variable: (V.)[ch].MPA.NPARSETS.xn

Number of parameter sets.

This parameter must be modified directly in the parameter tree (left panel of the table), by adding or deleting the necessary sets in the branch of each axis. This parameter cannot be modified directly in the table.

DEFAULTSET

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to NPARSETS.
Default value: 1.
Associated variable: (V.)[ch].MPA.DEFAULTSET.xn

Set of default parameters on power-up.

Axis behavior:

On the axes, the DEFAULTSET parameter indicates the parameter set that the CNC assumes at power-up, after a reset and at the start of a new program. When defined with a "0" value, the parameter set is always maintained. To select a parameter set on the axis or via the part-program, use function G112.

Spindle behavior.

The parameter DEFAULTSET is only used when moving the spindle for the first time without a set and without S programmed. In this case, the CNC uses the set indicated in the parameter DEFAULTSET. If DEFAULTSET is set at the value of ·0·, the CNC uses the set of 1.

- The CNC starts-up without a set, and just after starting-up the CNC runs an M19 or G74.
- The CNC starts-up without a set, and just after starting-up the spindles that do not have the SYNCSET parameter defined are synchronized.
- If the Sercos ring falls and the drive uses the default set.

If the CNC starts-up without a set, when the spindle starts-up for the first time (M3/M4 with S programmed) the CNC uses the appropriate set, which id dependant on S.

The start-up and reset do not use the parameter DEFAULTSET; both circumstances are managed from the PLC maneuver. During power-up, the CNC uses the set indicated by the PLC; if there is no set indicated, then the CNC starts-up without a set. After a reset, the CNC uses the set indicated by the PLC; if there is no set indicated, then the CNC remains as it is.

To select a parameter set on a spindle from the part-program and make a gear change, use functions M41 through M44.

M41SET

Parameter valid for spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Any set.
Default value: 1.
Associated variable: (V.)[ch].MPA.M41SET.xn

Set number associated with the M41 gear.

M421SET

Parameter valid for spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Any set.
Default value: 1.
Associated variable: (V.)[ch].MPA.M42SET.xn

Set number associated with the M42 gear.

M43SET

Parameter valid for spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Any set.
Default value: 1.
Associated variable: (V.)[ch].MPA.M43SET.xn

Set number associated with the M43 gear.

M44SET

Parameter valid for spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Any set.
Default value: 1.
Associated variable: (V.)[ch].MPA.M44SET.xn

Set number associated with the M44 gear.

2.14 Machine parameters; axis set.

2.14.1 Operating mode of the drive.

DRV_OPMODE

Possible values: Default / Positioner / Velocity Loop.
Default value: Velocity loop.
Associated variable: (V.)[ch].MPA.DRV_OPMODE[set].xn

Operating mode of the drive. The value of this parameter must be the same in all parameter sets of all axes and spindles.

Value.	Description.
By default.	Current loop in the drive.
Positioning drive.	Position loop at the drive.
Velocity loop.	Velocity loop in the drive.

2.14.2 Software axis limits.

LIMITENABLE

Parameter valid for linear and rotary axes.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Yes / No.
Default value: Yes.
Associated variable: (V.)[ch].MPA.LIMITENABLE[set].xn

Enable software limits (parameters POSLIMIT and NEGLIMIT).

POSLIMIT

Parameter valid for linear and rotary axes.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): Between ±99 999.9999 mm.
Possible values (2): Within ±3937.00787 inches.
Possible values (3): Between ±99 999.9999 degrees.
Default value: 99 999.9999 mm / 3937.00787 inches / 99 999.9999 degrees.
Associated variable: (V.)[ch].MPA.POSLIMIT[set].xn

Positive software limit. See axis machine parameters NEGLIMIT.

NEGLIMIT

Parameter valid for linear and rotary axes.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): Between ±99 999.9999 mm.
Possible values (2): Within ±3937.00787 inches.
Possible values (3): Between ±99 999.9999 degrees.
Default value: -99 999.9999 mm / -3937.00787 inches / -99 999,9999 degrees.
Associated variable: (V.)[ch].MPA.NEGLIMIT[set].xn

On rotary axes, the CNC takes this parameter into account when AXISMODE = Linearlike.

Negative software limit. On linear and rotary axes, these parameters set the axis travel limits. If both are set to "0", there will be no software limits.

i

The software limits are always set in radius, regardless of the setting of parameter DIAMPROG.

SWLIMITTOL

Parameter valid for linear and rotary axes.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): Between ±99 999.9999 mm.
Possible values (2): Within ±3937.00787 inches.
Possible values (3): Between ±99 999.9999 degrees.
Default value: 0.1000 mm / 0.00394 inches / 0.1000 degree.
Associated variable: (V.)[ch].MPA.SWLIMITTOL[set].xn

Software limit tolerance. 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.

2.14.3 Work zones.

ZONELIMITTOL

Parameter valid for linear and rotary axes.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 99 999.9999 mm or degrees / from 0 to 3937.00787 inch.
Default value: 0.1000 mm or degrees / 0.00394 inches.
Associated variable: (V.)[ch].MPA.ZONELIMITTOL[set].xn

Safety distance of the limits of the work zones. Safety distance applied by the CNC to the axis with respect to the limit of the work zone. When a work zone is active, the CNC stops the axes when one of them reaches the safety distance. The safety distance is defined per axis and each axis will have the same safety distance for all the zones.

The safety distance may be modified with the following variable which will assume the value of this parameter on startup.

(V.)[ch].A.ZONELIMITTOL.xn	Safety distance of the limits of the work zones.

2.14.4 Loop setting.

LOOPCH

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

Change the sign of the setpoint. If the shaft is packed and the following error is displayed; change the value of the parameter LOOPCH.

INPOSW

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0.0001 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Possible values (3): From 0.0001 to 99 999.9999 degrees.
Default value: 0.0100 mm / 0.00039 inches / 0.0100 degree.
Associated variable: (V.)[ch].MPA.INPOSW[set].xn

In-position zone. 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.

If a movement smaller than the INPOSW parameter is programmed on a dead axis, the CNC does not enable or move the axis.

2.14.5 Backlash compensation with additional command pulse.

BAKANOUT

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drives and Sercos drives.
Possible values: From 0 to 72 000 000.0000 mm/min.
Default value: 0 (Not applied).
Associated variable: (V.)[ch].MPA.BAKANOUT[set].xn

Additional analog pulse to recover possible leadscrew backlash in movement reversals. 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.

When setting an additional command pulse, parameters BAKTIME, ACTBAKAN and PEAKDISP must also be set.

BAKTIME

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 100 milliseconds.
Default value: 0.
Associated variable: (V.)[ch].MPA.BAKTIME[set].xn

Parameter BAKTIME indicates the duration of the additional velocity command pulse to make up for backlash in movement reversals. The CNC only takes this parameter into consideration when working with an additional velocity command pulse; BAKANOUT other than zero.

BAKDELAY

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drives and Sercos drives.
Possible values: Within ±-2 147 483 647 ms.
Default value: 0.
Associated variable: (V.)[ch].MPA.BAKDELAY[set].xn

Delay to apply the backlash compensation for movement reversals.

ACTBAKAN

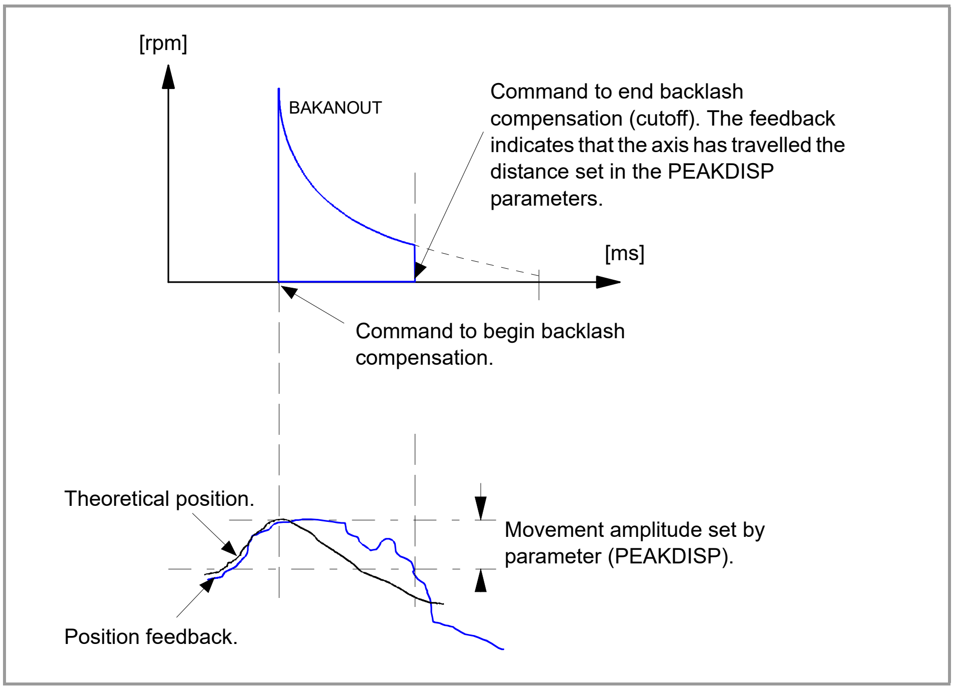
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drives and Sercos drives.
Possible values: Always / G2-G3.
Default value: Always.
Associated variable: (V.)[ch].MPA.ACTBAKAN[set].xn

Parameter ACTBAKAN determines when the additional command pulse is applied to compensate for backlash peaks. The CNC only takes this parameter into consideration when working with an additional velocity command pulse; BAKANOUT other than zero.

PEAKDISP

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drives and Sercos drives.
Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Possible values (3): From 0 to 99 999.9999 degrees.
Default value: 0.0050 mm / 0.00020 inches / 0.0050 degrees.
Associated variable: (V.)[ch].MPA.PEAKDISP[set].xn

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). The CNC only takes this parameter into consideration when working with an additional velocity command pulse; BAKANOUT other than zero.



REVEHYST

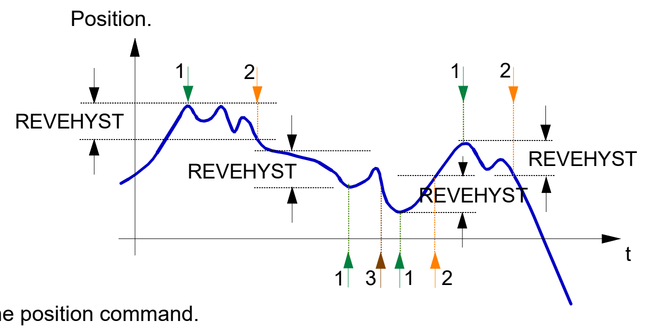
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drives and Sercos drives.
Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Possible values (3): From 0 to 99 999.9999 degrees.
Default value: 0.
Associated variable: (V.)[ch].MPA.REVEHYST[set].xn

Hysteresis for applying the additional command pulse in movement reversals. 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.

2.14.6 Configuring feedbacks.

FBACKDIFF

Parameter valid for linear and rotary axes and spindles.
Parameter valid for a Sercos drive.
Possible values (1): Within ±99 999.9999 mm or degrees.
Possible values (2): Within ±3937.00787 inches.
Default value: 0 (no monitoring).
Associated variable: (V.)[ch].MPA.FBACKDIFF[set].xn

Maximum difference between feedbacks. When the system has dual feedback (parameters SPEEDFBID and POSITIONFBID not equal), 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.

FBMIXTIME

Parameter valid for linear and rotary axes and spindles.
Parameter valid for a Sercos drive.
Possible values: From 0 to 3200.0 milliseconds.
Default value: 0.
Associated variable: (V.)[ch].MPA.FBMIXTIME[set].xn

This parameter may be used to set the time constant to be used for combined feedback; this means that it sets the delay between the coordinates entered in the position loop from both feedbacks. The CNC takes this parameter into account if the SPEEDFBID and POSITIONFBID parameters are not the same.

If the parameter is set to a value other than ·0·, the combined feedback is enabled; if set to ·0·, only the feedback set to POSITIONFBID is enabled.

For machines that have a lot of backlash and use both feedbacks to obtain greater accuracy, these may suffer some instability. These kinds of machines run smoothly with motor feedback, but they may lose some precision; with a direct feedback, however, accuracy improves but the machine movements can be jerky. Combining both feedbacks both precision and smoothness may be achieved.

The CNC uses the combined feedback to calculate the position feedback, but it uses the feedback set to POSITIONFBID to calculate the compensations, circularity test, etc.

2.14.7 Adjustment of rapid traverse G00 and maximum speed.

G00FEED

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0 to 500 000.0000 mm/min.
Possible values (2): From 0 to 19 685.03937 inches/min.
Possible values (3): From 0 to 72 000 000.0000 degrees/min.
Possible values (4): From 0 to 200 000.0000 rpm.
Default value: 20 000.0000 mm/min / 787.40157 inch/min / 1080000 degrees/min / 3000.0000 rpm.
Associated variable: (V.)[ch].MPA.G00FEED[set].xn

Lathe and milling machine models.

Rapid positioning moves (G00) are always carried out at the maximum feedrate possible, the one indicated by G00FEED.

Laser model.

Maximum generator power. In order for the power control through an analog output to be managed correctly (#PWRCTRL ON[OUT], the watts defined in the parameter G00FEED as the generator power must match the volt level defined in the parameter MAXVOLT.

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

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0 to 500 000.0000 mm/min.
Possible values (2): From 0 to 19 685.03937 inches/min.
Possible values (3): From 0 to 500 000,0000 degrees/min.
Possible values (4): 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

2.14.8 Rapid traverse for the automatic mode.

FRAPIDEN

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0 to 500 000.0000 mm/min.
Possible values (2): From 0 to 19 685.03937 inches/min.
Possible values (3): From 0 to 500 000,0000 degrees/min.
Possible values (4): From 0 to 1388.8889 rpm.
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

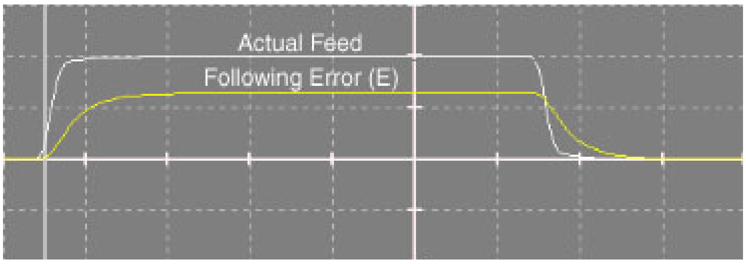
2.14.9 Gain setting.

PROGAIN

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 100.0 (1000/min).
Default value: 1.
Associated variable: (V.)[ch].MPA.PROGAIN[set].xn

Proportional gain. 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

POSREFTIME

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 10 000.0000 ms.
Default value: 0.
Associated variable: (V.)[ch].MPA.POSREFTIME[set].xn

Time constant of the position reference model. First order filter to be applied to the setpoint generated by the position loop.

FFWTYPE

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.

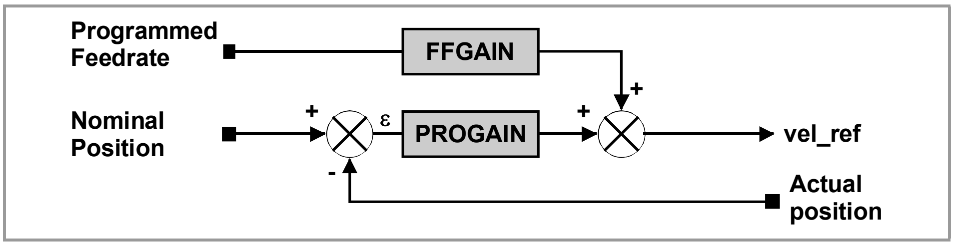
FFGAIN

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: from 0 to 120 %.
Default value: 0.
Associated variable: (V.)[ch].MPA.FFGAIN[set].xn

Percentage of Feed-Forward in automatic. The CNC takes this parameter into account only when working with feed forward (parameter AFFWTYPE). 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 achieved when the following error is minimized as much as possible, but without changing 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

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: from 0 to 120 %.
Default value: 0.
Associated variable: (V.)[ch].MPA.MANFFGAIN[set].xn

Percentage of Feed-Forward in JOG The CNC takes this parameter into account only when working with feed forward (parameter AFFWTYPE).

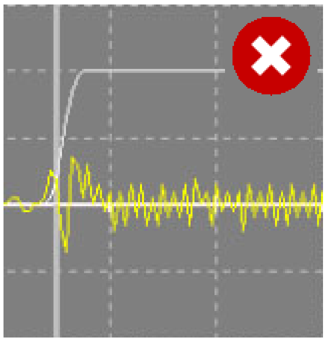
Sometimes, the Feed Forward selected for the automatic mode may be too high for the Jog mode, when there is no need to control the following error that closely. In those cases, parameter MANFFGAIN helps adapt the Feed-Forward applied to the jog mode.

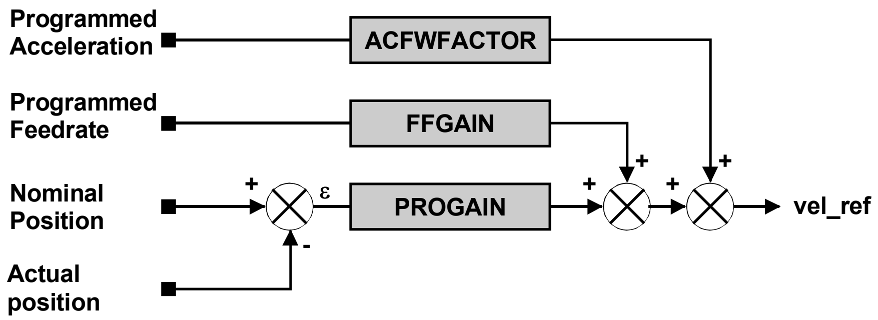
An axis that is going to work as an independent axis should have the same feed forward value in automatic and in jog modes; in other words, parameters FFGAIN and MANFFGAIN should be the same.

ACFWFACTOR

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0.001 to 1 000 000.0000 milliseconds.
Default value: 1000.0000 milliseconds.
Associated variable: (V.)[ch].MPA.ACFWFACTOR[set].xn

Acceleration time constant. 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.

Correct adjustment.	Inverted peaks; poor adjustment.
	
System response without AC-forward gain (10 µm per square).	System response with AC-forward gain (1 µm per square).



ACFGAIN

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: from 0 to 120 %.
Default value: 0.
Associated variable: (V.)[ch].MPA.ACFGAIN[set].xn

Percentage of AC-Forward in automatic. See axis machine parameters MANACFGAIN.

MANACFGAIN

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: from 0 to 120 %.
Default value: 0.
Associated variable: (V.)[ch].MPA.MANACFGAIN[set].xn

Percentage of Feed-Forward in jog mode. The CNC takes this parameter into account only when working with AC-forward (parameter ACFWFACTOR). 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.

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.

2.14.10 Linear acceleration.

LACC1

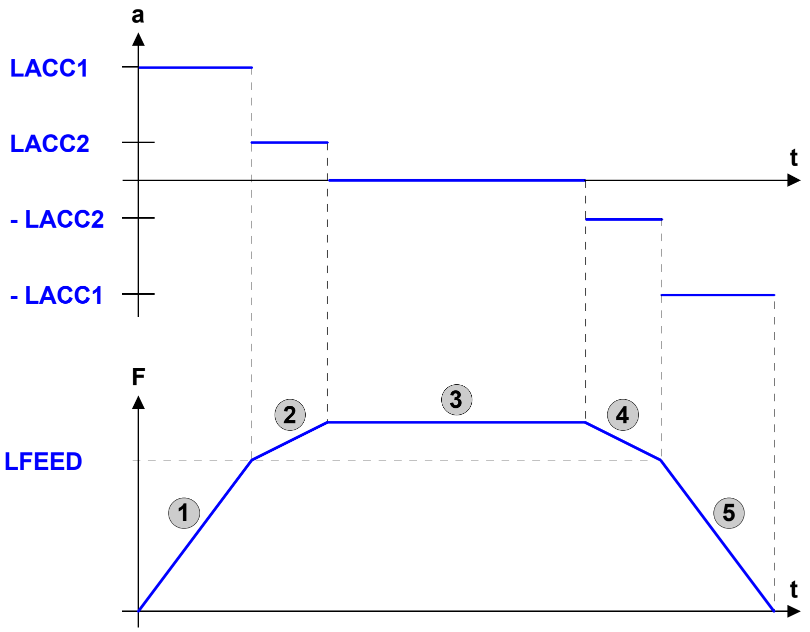
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0 to 600 000 000.0000 mm/s².
Possible values (2): From 0 to 23 622 047.24409 inches/s².
Possible values (3): From 0 to 600 000 000.0000 degrees/s².
Default value: 1000.0000 mm/s² / 39.37008 inch/s² / 1000.0000 degrees/s².
Associated variable: (V.)[ch].MPA.LACC1[set].xn

Linear acceleration; acceleration of the first section. See parameter LACC2.

LACC2

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0 to 600 000 000.0000 mm/s².
Possible values (2): From 0 to 23 622 047.24409 inches/s².
Possible values (3): From 0 to 600 000 000.0000 degrees/s².
Default value: 1000.0000 mm/s² / 39.37008 inch/s² / 1000.0000 degrees/s².
Associated variable: (V.)[ch].MPA.LACC2[set].xn

Linear acceleration; acceleration of the second section. 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 use these parameters when G0ACDCJERK = No.



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

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0 to 200 000.0000 mm/min.
Possible values (2): From 0 to 7874.01575 inches/min.
Possible values (3): From 0 to 36 000 000.0000 degrees/min.
Possible values (4): From 0 to 100 000.0000 rpm.
Default value: 3000 mm/min / 118.11023 inches/min / 1080000 degrees/min / 3000 rpm.
Associated variable: (V.)[ch].MPA.LFEED[set].xn

Linear acceleration; rate of change of acceleration. 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.

2.14.11 Trapezoidal and square sine acceleration.

ACCEL

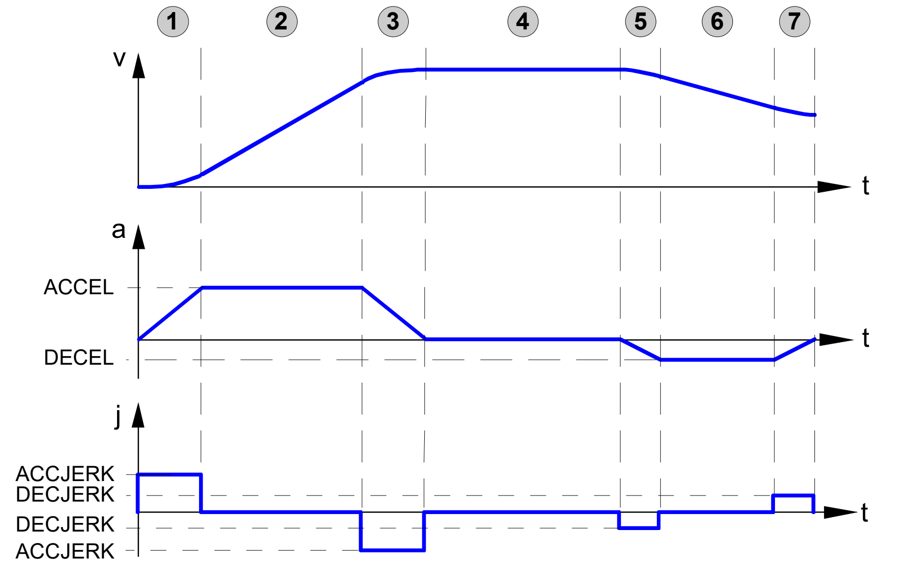
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0.0010 to 600 000 000.0000 mm/s².
Possible values (2): From 0.00004 to 23 622 047.2440 inches/s².
Possible values (3): From 0.0010 to 600 000 000.0000 degrees/s².
Default value: 3000.0000 mm/s² / 118.12023 inch/s² / 3000.0000 degrees/s².
Associated variable: (V.)[ch].MPA.ACCEL[set].xn

Trapezoidal and square sine acceleration. See machine axis parameter DECEL.

DECEL

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0.0010 to 600 000 000.0000 mm/s².
Possible values (2): From 0.00004 to 23 622 047.2440 inches/s².
Possible values (3): From 0.0010 to 600 000 000.0000 degrees/s².
Default value: 3000.0000 mm/s² / 118.12023 inch/s² / 3000.0000 degrees/s².
Associated variable: (V.)[ch].MPA.DECEL[set].xn

Trapezoidal and square sine deceleration. 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 use these parameters when G0ACDCJERK = No.



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

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0.0010 to 6E+011 mm/s³.
Possible values (2): From 0.00004 to 2.362E+010 inches/s².
Possible values (3): From 0.0010 to 6E+011 mm/s³.
Default value: 45 000.000 mm/s² / 1771.65354 inch/s² / 45 000,000 degrees/s².
Associated variable: (V.)[ch].MPA.ACCJERK[set].xn

Acceleration jerk. See machine axis parameter DECJERK.

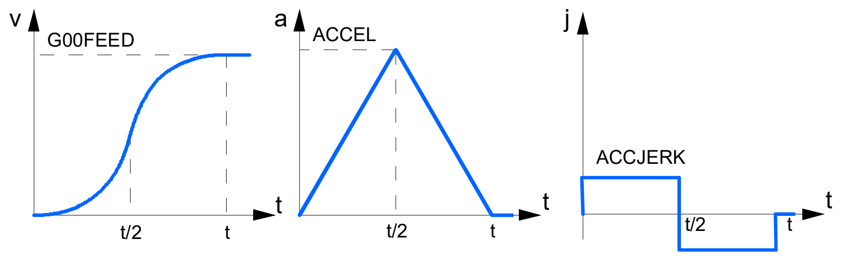
DECJERK

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0.0010 to 6E+011 mm/s³.
Possible values (2): From 0.00004 to 2.362E+010 inches/s².
Possible values (3): From 0.0010 to 6E+011 mm/s³.
Default value: 45 000.000 mm/s² / 1771.65354 inch/s² / 45 000,000 degrees/s².
Associated variable: (V.)[ch].MPA.DECJERK[set].xn

Deceleration jerk. 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.

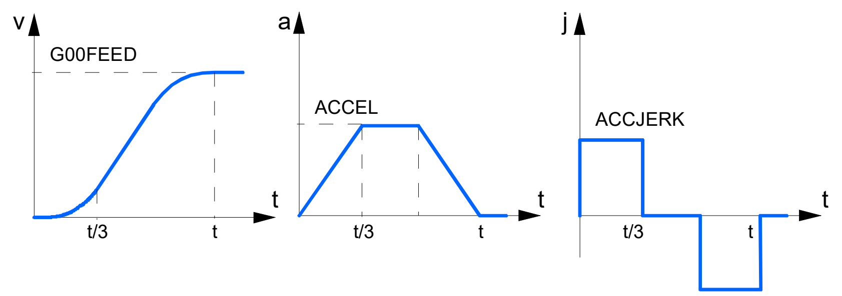
The CNC applies parameters ACCJERK and DECJERK to the movements in G1, G2 and G3; movements in G0 use these parameters when G0ACDCJERK = No. The CNC ignores these parameters in threading movements (G33) and with HSC FAST.

Example. Trapezoidal acceleration.
When an axis is stopped, it must reach G00FEED at maximum acceleration within a certain time (0.5 seconds). The axis parameters may be set with maximum acceleration and minimum jerk.



$$\text{ACCEL} = 2 \times \frac{\text{G00FEED}}{60 \times 0{,}5} \qquad \text{ACCJERK} = 2 \times \frac{\text{ACCEL}}{0{,}5}$$

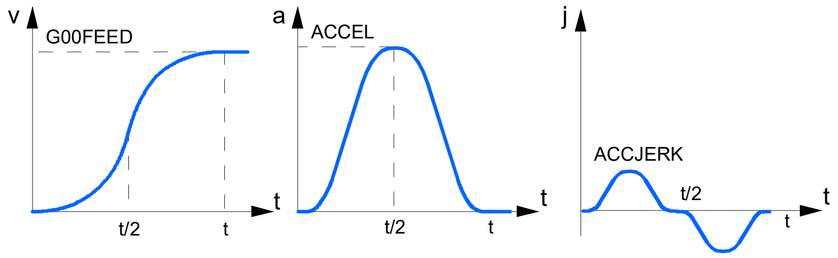
Example. Trapezoidal acceleration.
When an axis is stopped, it must reach G00FEED at maximum acceleration within a certain time (0.5 seconds). The axis parameters may be set with less acceleration and greater jerk so the axis can reach the minimum acceleration faster and it stays longer at maximum acceleration.



$$\text{ACCEL} = \frac{3}{2} \times \frac{\text{G00FEED}}{60 \times 0{,}5} \qquad \text{ACCJERK} = 3 \times \frac{\text{ACCEL}}{0{,}5}$$

Example. Square sine (bell shaped) acceleration.

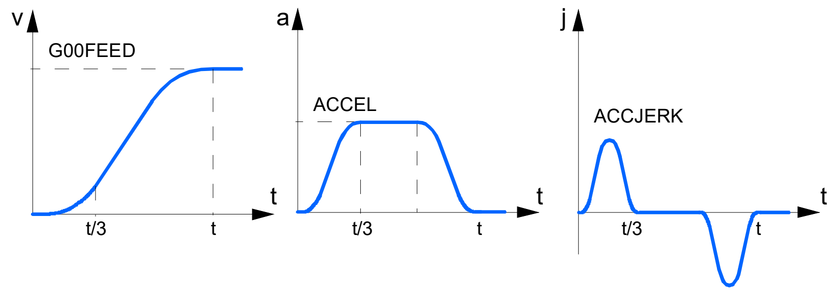
When an axis is stopped, it must reach G00FEED at maximum acceleration within a certain time (0.5 seconds). The axis parameters may be set with maximum acceleration and minimum jerk.



$$ACCEL = 2 \times \frac{G00FEED}{60 \times 0{,}5} \qquad ACCJERK = 2 \times \frac{ACCEL}{0{,}5}$$

Example. Square sine (bell shaped) acceleration.

When an axis is stopped, it must reach G00FEED at maximum acceleration within a certain time (0.5 seconds). The axis parameters may be set with less acceleration and greater jerk so the axis can reach the minimum acceleration faster and it stays longer at maximum acceleration.



$$ACCEL = \frac{3}{2} \times \frac{G00FEED}{60 \times 0{,}5} \qquad ACCJERK = 3 \times \frac{ACCEL}{0{,}5}$$

2.14.12 Enable specific acceleration values for movements in G0.

G0ACDCJERK

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. • Acceleration of the second section. • Acceleration changing speed.	 LACC1 LACC2 LFEED	 LACC1G0 LACC2G0 LFEEDG0
Trapezoidal and square sine acceleration (G0 movements). • Acceleration. • Deceleration. • Acceleration jerk. • Deceleration jerk.	 ACCEL DECEL ACCJERK DECJERK	 ACCELG0 DECELG0 ACCJERKG0 DECJERKG0

2.14.13 Linear acceleration (G0 movements).

LACC1G0

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0 to 600 000 000.0000 mm/s².
Possible values (2): From 0 to 23 622 047.24409 inches/s².
Possible values (3): From 0 to 600 000 000.0000 degrees/s².
Default value: 1000.0000 mm/s² / 39.37008 inch/s² / 1000.0000 degrees/s².
Associated variable: (V.)[ch].MPA.LACC1G0[set].xn

Linear acceleration; acceleration of the first section (G0 movements). See parameter LACC2G0.

LACC2G0

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0 to 600 000 000.0000 mm/s².
Possible values (2): From 0 to 23 622 047.24409 inches/s².
Possible values (3): From 0 to 600 000 000.0000 degrees/s².
Default value: 1000.0000 mm/s² / 39.37008 inch/s² / 1000.0000 degrees/s².
Associated variable: (V.)[ch].MPA.LACC2G0[set].xn

Linear acceleration; acceleration of the second section (G0 movements). 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

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0 to 200 000.0000 mm/min.
Possible values (2): From 0 to 7874.01575 inches/min.
Possible values (3): From 0 to 36 000 000.0000 degrees/min.
Possible values (4): From 0 to 100 000.0000 rpm.
Default value: 3000 mm/min / 118.11023 inch/min / 1 080 000 degrees/min / 3000 rpm.
Associated variable: (V.)[ch].MPA.LFEEDG0[set].xn

Linear acceleration; rate of change of acceleration (movements in G0). 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.

2.14.14 Trapezoidal and square sine acceleration (G0 movements).

ACCELG0

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0.0010 to 600 000 000.0000 mm/s².
Possible values (2): From 0.00004 to 23 622 047.2440 inches/s².
Possible values (3): From 0.0010 to 600 000 000.0000 degrees/s².
Default value: 4000.0000 mm/s² / 157.48031 inches/s² / 4000.0000 degrees/s².
Associated variable: (V.)[ch].MPA.ACCELG0[set].xn

Trapezoidal and square sine acceleration (G0 movements). See axis machine parameters DECELG0.

DECELG0

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0.0010 to 600 000 000.0000 mm/s².
Possible values (2): From 0.00004 to 23 622 047.2440 inches/s².
Possible values (3): From 0.0010 to 600 000 000.0000 degrees/s².
Default value: 4000.0000 mm/s² / 157.48031 inches/s² / 4000.0000 degrees/s².
Associated variable: (V.)[ch].MPA.DECELG0[set].xn

Trapezoidal and square sine deceleration (movements in G0). Parameters ACCELG0 and DECELG0 set the acceleration values when it is trapezoidal or square-sine (bell shaped) (parameter SLOPETYPE). The CNC applies the parameters ACCELG0 and DECELG0 to the movements in G0 when the parameter G0ACDCJERK = Yes.

ACCJERKG0

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0.0010 to 6E+011 mm/s³.
Possible values (2): From 0.00004 to 2.362E+010 inches/s².
Possible values (3): From 0.0010 to 6E+011 mm/s³.
Default value: 55 000.000 mm/s² / 2165.35433 inch/s² / 55 000.000 degrees/s².
Associated variable: (V.)[ch].MPA.ACCJERKG0[set].xn

Acceleration jerk (G0 movements). See axis machine parameters DECJERKG0.

DECJERKG0

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0.0010 to 6E+011 mm/s³.
Possible values (2): From 0.00004 to 2.362E+010 inches/s².
Possible values (3): From 0.0010 to 6E+011 mm/s³.
Default value: 55 000.000 mm/s² / 2165.35433 inch/s² / 55 000.000 degrees/s².
Associated variable: (V.)[ch].MPA.DECJERKG0[set].xn

Deceleration jerk (G0 movements). 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 the parameters ACCJERKG0 and DECELJERKG0 to the movements in G0 when the parameter G0ACDCJERK=Yes. The CNC ignores these parameters in threading movements (G33) and with HSC FAST.

2.14.15 Home search.

DECINPUT

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Yes / No.
Default value: Yes.
Associated variable: (V.)[ch].MPA.DECINPUT[set].xn

The axis/spindle has a home switch. 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.

- If the linear axis has coded I0s, the DECINPUT parameter must be set to "Yes" if the axis has a safety switch so that the axis does not run out of travel limits when searching. The axis performs the reference search by means of the coded flags, but if you press the microphone before finishing, it reverses the search direction.
- If the linear axis does not have coded I0s, the DECINPUT parameter must be set to "Yes". In homing, the axis searches for the microphone, reverses the search direction, and when it stops pressing the microphone it starts searching for I0.
- If the rotary axis has coded I0s, and is of "module" type, the DECINPUT parameter must have value "No". The axis performs homing by means of coded flags (the rotary axis can rotate endlessly).
- If the rotary axis has coded I0s, and is of "linear-like" type, the DECINPUT parameter must be set to "Yes". The axis performs the reference search by means of the coded flags, but if you press the microphone before finishing, it reverses the search direction.
- If the rotary axis does not have coded I0s, the DECINPUT parameter must be set to "Yes".
- On heads with I0 microphone, the head passes twice over the microphone. 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.

2.14.16 Configuration of the HSC mode.

HSC

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

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).
MAXERROR	Position error.

HSC | CORNERACC

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0 to 100 000.0000 mm/s².
Possible values (2): From 0 to 3937.00787 inch/s².
Possible values (3): From 0 to 100 000,0000 degrees/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 (corners). If this parameter is set with a ·0· value, the maximum acceleration of the axis is respected.

HSC | CURVACC

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0 to 100 000.0000 mm/s².
Possible values (2): From 0 to 3937.00787 inch/s².
Possible values (3): From 0 to 100 000,0000 degrees/s².
Default value: 0.
Associated variable: (V.)[ch].MPA.CURVAACC[set].xn

Maximum contouring acceleration permitted. If this parameter is set with a ·0· value, the maximum acceleration of the axis is respected.

HSC | CORNERJERK

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0 to 1E+009 mm/s³.
Possible values (2): From 0 to 39 370 078.74016 inches/s².
Possible values (3): From 0 to 1E+009 degrees/s³.
Default value: 0.
Associated variable: (V.)[ch].MPA.CORNERJERK[set].xn

Maximum Jerk permitted at the corners. If this parameter is set with a ·0· value, the maximum jerk of the axis is respected.

HSC | CURVJERK

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0 to 1E+009 mm/s³.
Possible values (2): From 0 to 39 370 078.74016 inches/s².
Possible values (3): From 0 to 1E+009 degrees/s³.
Default value: 0.
Associated variable: (V.)[ch].MPA.CURVJERK[set].xn

Maximum contouring Jerk permitted. If this parameter is set with a ·0· value, the maximum jerk of the axis is respected.

HSC | FASTACC

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0 to 100 000.0000 mm/s².
Possible values (2): From 0 to 3937.00787 inch/s².
Possible values (3): From 0 to 100 000,0000 degrees/s².
Default value: 0.
Associated variable: (V.)[ch].MPA.FASTACC[set].xn

Maximum acceleration permitted at the corners (FAST mode). If this parameter is set with a ·0· value, the maximum acceleration of the axis is respected.

HSC | MAXERROR

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.007 87 inches.
Possible values (3): From 0 to 99 999.9999 degrees.
Default value: 0.1000 mm / 0.00394 inches / 0.1000 degree.
Associated variable: (V.)[ch].MPA.MAXERROR[set].xn

Maximum axis position error in HSC when working outside the plane/trihedron.

2.14.17 Home search.

The home search process depends on the characteristics of the machine.

REFFEED1

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0 to 200 000.0000 mm/min.
Possible values (2): From 0 to 7873,992 inches/min.
Possible values (3): From 0 to 200 000,0000 degrees/min.
Possible values (4): From 0 to 100 000.0000 rpm.
Default value: 1000.0000 mm/min / 39.37001 inches/min / 1000.0000 degrees/min / 100.0000 rpm.
Associated variable: (V.)[ch].MPA.REFFEED1[set].xn

Fast home searching feedrate. See axis machine parameters REFFEED2.

REFFEED2

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0 to 200 000.0000 mm/min.
Possible values (2): From 0 to 7873,992 inches/min.
Possible values (3): From 0 to 200 000,0000 degrees/min.
Possible values (4): From 0 to 100 000.0000 rpm.
Default value: 100.0000 mm/min / 3.93700 inches/min / 100.0000 degrees/min / 10.0000 rpm.
Associated variable: (V.)[ch].MPA.REFFEED2[set].xn

Slow home searching feedrate. 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.

POSINREF

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.

MAXDIFREF

Parameter valid for linear and rotary axes.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0.0001 to 99 999.0000 mm.
Possible values (2): From 0 to 3936.96850 inches.
Possible values (3): From 0.0001 to 99 999.0000 degrees.
Default value: 0.0200 mm / 0.00079 inches / 0.0200 degrees.
Associated variable: (V.)[ch].MPA.MAXDIFREF[set].xn

Absolute feedback (parameter ABSFEEDBACK=Total/One Lap).

Maximum position difference, between the coordinate stored by the CNC and the one read by the encoder, to re-reference the axis (REFPOIN mark). If the difference is less than MAXDIFREF, the CNC starts with the new position calculated by the encoder and sets the PLC flag REFPOIN = 1. If the difference is greater than the MAXDIFREF, the CNC will start up using the position saved by the CNC at shutdown and display error 3753. With this error, the user decides on the validity of the dimension.

- The coordinate is valid if the position of the machine coincides with the “current value” field of the error. The user must clear the error by pressing [ENTER]+[RESET]; pressing only [RESET] will not clear the error.
- The coordinate is valid if the position of the machine coincides with the “expected value” field. The user must clear the error by pressing [ENTER]+[RESET] and execute a G174 in MDI with the expected value. The user must then restart the CNC to assume the dimension correctly.

Incremental feedback (parameter ABSFEEDBACK=No).

Maximum position difference between the coordinate stored by the CNC and the controller to re-reference the axis (REFPOIN flag). If after an error or a drop of the Sercos ring, the difference is greater than MAXDIFREF, the CNC sets the PLC flag REFPOIN(axis)=0. If the difference is less than MAXDIFREF, the CNC does not set the PLC flag REFPOIN(axis)=1. If MAXDIFREF is equal to 0, the CNC always disables the REFPOIN(axis) flag.

2.14.18 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

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: OFF / Standard.
Default value: Standard.
Associated variable: (V.)[ch].MPA.FLWEMONITOR[set].xn

Type of monitoring of the following error (axis lag).

Value.	Meaning.
OFF	The following error is not monitored, thus no error message will be issued.
Standard	The CNC constantly monitors the following error and prompts an error when it exceeds the value set by the parameter MAXFLWE if the axis is moving, or MINFLWE if the axis has stopped.

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 completed, this monitoring must be enabled.

MINFLWE

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0.0010 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Possible values (3): From 0.0010 to 99 999.9999 degrees.
Default value: 1.0000 mm / 0.03937 inches / 1.0000 degree.
Associated variable: (V.)[ch].MPA.MINFLWE[set].xn

With FLWEMONITOR=Standard, 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 (POSLIMIT to NEGLIMIT).

MAXFLWE

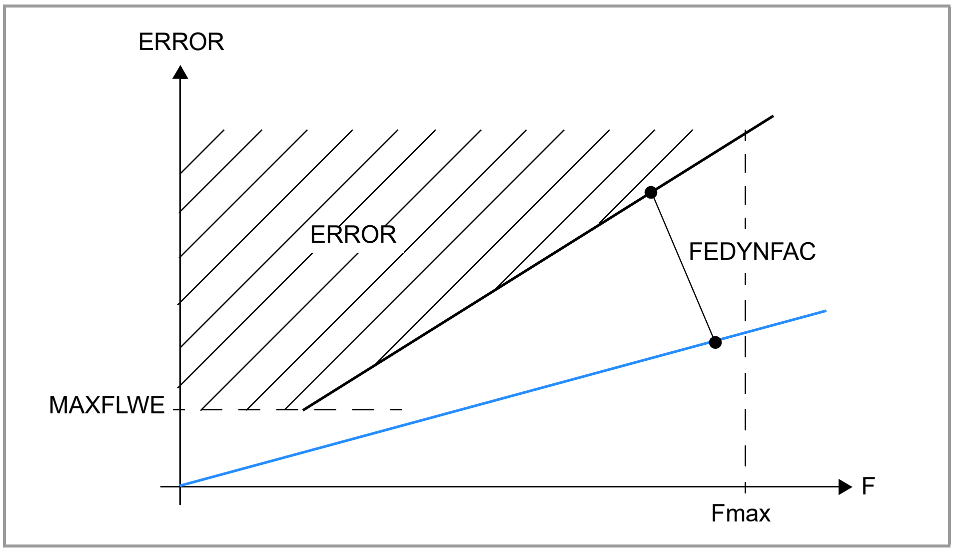
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0.0010 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Possible values (3): From 0.0010 to 99 999.9999 degrees.
Default value: 30.0000 mm / 1.18110 inches / 30.0000 degrees.
Associated variable: (V.)[ch].MPA.MAXFLWE[set].xn

With FLWEMONITOR=Standard, this parameter indicates the maximum amount of following error allowed when the axis is moving.

FEDYNFAC

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drives and Sercos drives.
Possible values: from 0 to 100 %.
Default value: 50.
Associated variable: (V.)[ch].MPA.FEDYNFAC[set].xn

With FLWEMONITOR=Standard, 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 from which following error value on, dynamic monitoring begins.


ERROR
ERROR
FEDYNFAC
MAXFLWE
F
Fmax

2.14.19 Axis in in-position zone.

INPOMAX

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 1 000 000 milliseconds.
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

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 1 000 000 milliseconds.
Default value: 0.
Associated variable: (V.)[ch].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.

2.14.20 Emergency stop.

EMERGRAMPENA

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.EMERGRAMPENA[set].xn

Enabling the emergency stop ramp when the "enable speed" signal of the PLC drops (mark SPENA).

EMERGDECMAX

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0.0010 to 600 000 000.0000 mm/s² or degrees/s².
Possible values (2): From 0.00004 to 23 622 047.24409 inches/s².
Default value: 3000 mm/s² / 118,11024 inches/s² / 3000 degrees/s².
Associated variable: (V.)[ch].MPA.EMERGDECMAX[set].xn

"Maximum deceleration of the emergency ramp"

EMERGVELMAX

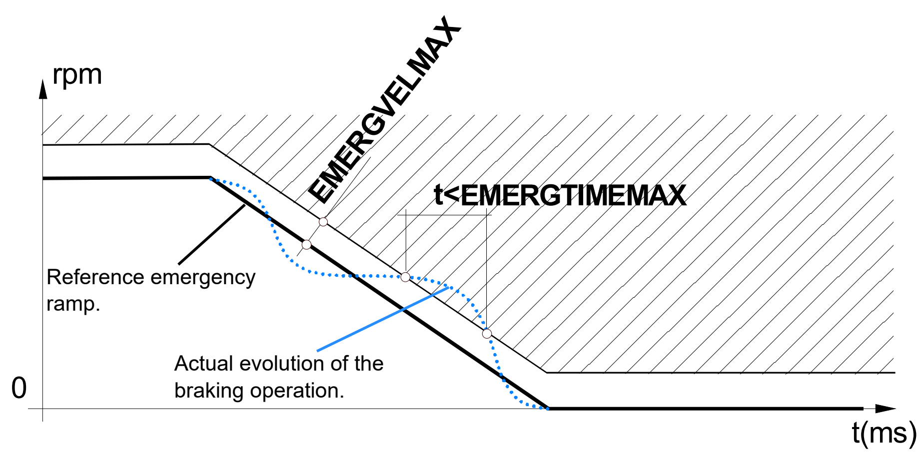
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: from 0 to 100 %.
Default value: 50 %.
Associated variable: (V.)[ch].MPA.EMERGVELMAX[set].xn

Maximum speed threshold for the emergency ramp monitoring, expressed as a percentage. If the axis exceeds this threshold for a period longer than the time given in EMERGTIMEMAX, it will prompt the corresponding error.

EMERGTIMEMAX

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: 50 %.
Associated variable: (V.)[ch].MPA.EMERGTIMEMAX[set].xn

Maximum time that the parameterized speed threshold set in EMERGVELMAX can be exceeded during emergency ramp monitoring. If it is set to 0, the emergency stop ramp monitoring is disabled.



For negative speeds, consider a figure symmetrical to this one with respect to the time axis

2.14.21 Axis lubrication.

DISTLUBRI

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0 to 2 000 000 000 mm.
Possible values (2): From 0 to 78 739 920 inches.
Possible values (3): From 0 to 2 000 000 000 degrees.
Default value: 0 (no lubrication).
Associated variable: (V.)[ch].MPA.DISTLUBRI[set].xn

Distance to move to lubricate the axis. 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".

The PLC reads this parameter in mm instead of doing it in tenths of a micron (0.0001 mm).

2.14.22 Module configuration (rotary axes and spindle).

MODUPLIM

Parameter valid for rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Within ±99 999,9999º.
Default value: 360.
Associated variable: (V.)[ch].MPA.MODUPLIM[set].xn

Upper limit of the module. See machine axis parameter MODLOWLIM.

MODLOWLIM

Parameter valid for rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Within ±99 999,9999º.
Default value: 0.
Associated variable: (V.)[ch].MPA.MODLOWLIM[set].xn

Lower limit of the module. The CNC takes these parameters into consideration only when AXISMODE = Module.

For a reading within ±180°, set MODUPLIM = 180º and MODLOWLIM = -180º.

MODNROT

Parameter valid for rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 1 to 32 767 turns.
Default value: 1.
Associated variable: (V.)[ch].MPA.MODNROT[set].xn

Module error. Number of turns. See machine axis parameter MODERR.

MODERR

Parameter valid for rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Between ±32 767.
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.

2.14.23 Spindle speed.

SZERO

Parameter valid for spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 100 000 rpm.
Default value: 1.
Associated variable: (V.)[ch].MPA.SZERO[set].xn

Speed considered "0 rpm". 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

Parameter valid for spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Positive / Negative.
Default value: Positive.
Associated variable: (V.)[ch].MPA.POLARM3[set].xn

Sign of the velocity command for M3. See axis machine parameters POLARM4.

POLARM4

Parameter valid for spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Positive / Negative.
Default value: Negative.
Associated variable: (V.)[ch].MPA.POLARM4[set].xn

Lathe and mill models.

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.

Laser model.

Parameters POLARM3 and POLARM indicate the direction of the velocity command for functions M3 and M4 respectively. Both parameters must be set to "Positive".

2.14.24 Identifying motors and feedbacks.

REFERENCEID

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Any motor defined in the "Motors" branch.
Default value: None.
Associated variable: (V.)[ch].MPA.REFERENCEID[set].xn

Motor identifier. The names that appear in the list correspond to those defined in the motor table.

CURRFBID

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Any feedback defined in the "Feedbacks" branch.
Default value: None.
Associated variable: (V.)[ch].MPA.CURRFBID[set].xn

Feedback identifier of the current loop. The names that appear in the list correspond to those defined in the feedback table.

SPEEDFBID

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Any feedback defined in the "Feedbacks" branch.
Default value: None.
Associated variable: (V.)[ch].MPA.SPEEDFBID[set].xn

EtherCAT regulators.

The regulator closes the speed loop, therefore it is not necessary to define this parameter. If it is defined, the position of the loop will be used for both feedbacks (motor and direct).

POSITIONFBID

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Any feedback defined in the "Feedbacks" branch.
Default value: None.
Associated variable: (V.)[ch].MPA.POSITIONFBID[set].xn

EtherCAT regulators.

If SPEEDFBID is not defined, the loop position will work with the feedback defined in POSITIONFBID (motor or direct feedback). If SPEEDFBID and POSITIONFBID is defined, the position look will be used in both feedbacks.

2.14.25 Delay estimate at the drive.

AXDELAY

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.

Value.	Meaning.
Standard.	For Fagor drives, that do not require any delay compensation. The CNC automatically compensates for the delay differences between the axes of the channel.
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.

2.14.26 Compensations tables.

NCOMPTABLEIDS

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 20.
Default value: 0.
Associated variable: (V.)[ch].MPA.NCOMPTABLEIDS[set].xn

Number of ballscrew or cross compensation tables.

COMPTABLEID

This parameter shows the table to assign the compensation tables to the axis. The table has a parameter for each compensation table.

Parameter.	Meaning.
COMPTABLEID n	Identifier for the ballscrew or cross compensation table.

COMPTABLEID | COMPTABLEID n

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Any compensation defined in the "Compensation" branch.
Default value: None.
Associated variable: (V.)[ch].MPA.COMPTABLEID[set].xn

Identifier for the ballscrew or cross compensation table. Compensations can be defined by the parameters in any order. The CNC first applies the ballscrew compensation and then the cross compensation, while taking into consideration the dependencies between them. For example, if the coordinate for X is compensated with Y, and the coordinate for Y with Z, the CNC will first apply the YZ compensation and then the XY compensation. Compensations must be measured coherently in this order.

2.14.27 Torque saturation.

OVERLOADTL

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Between 0 and 20,000.
Default value: 200.
Associated variable: (V.)[ch].MPA.OVERLOADTL[set].xn

Allowable torque saturation time before promting error.

OVERLOADTHRESHOLD

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values (1): From 0 to 72 000 000.0000 mm/min.
Possible values (2): From 0 to 2 834 645.66929 inches/min.
Possible values (3): From 0 to 72 000 000.0000 degrees/min.
Default value: 0.
Associated variable: (V.)[ch].MPA.OVERLOADTHRESHOLD[set].xn

Maximum torque saturation detection speed. If set to 0, the CNC selects the speed automatically.

2.14.28 Delay time to remove torque after emergency stop.

DROFFDELAY

Possible values: From 0 to 10000 ms.
Default value: 200 ms.
Associated variable: (V.)[ch].MPA.DROFFDELAY[set].xn

Time the motor remains with torque after emergency stop by Sercos III. In uncompensated axes with locking brake, this parameter prevents the Z-axis from dropping if the mains voltage drops, an error occurs or the machine is switched off. For the management to be effective, the velocity loop must be in the drive (parameter DRV_OPMODE).

- If the parameter is different from 0 and a Sercos III error occurs, the CNC executes an emergency stop at reference speed 0, with torque and without emergency ramp management. Once stopped, the engine remains at torque for the time defined in this parameter.
- If the parameter is 0 and a Sercos III error occurs, the CNC does not execute an emergency stop. The CNC disables the axis with the Sercos III error and the motor stops due to friction.

2.14.29 Runaway protection.

OVERRUN

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Yes / No.
Default value: Yes.
Associated variable: (V.)[ch].MPA.OVERRUN[set].xn

Enables shaft anti-runaway protection.

2.14.30 DRC parameter setting.

DRC

DRC parameter setting.

DRC | FBACKCHANGEMODE

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Do not change / Pair.
Default value: Do not change.
Associated variable: (V.)[ch].MPA.FBACKCHANGEMODE[set].xn

How to address loop feedback when making a set change.

DRC | VEL_PROGAIN

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Within ±21 474 836.4700 Hz.
Default value: 50.0000 Hz.
Associated variable: (V.)[ch].MPA.VEL_PROGAIN[set].xn

Velocity loop bandwidth.

DRC | VEL_INTTIME

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 2 147 483.6470 milliseconds.
Default value: 6 milliseconds.
Associated variable: (V.)[ch].MPA.VEL_INTTIME[set].xn

Velocity loop integral.

DRC | VEL_INT_THRESHOLD

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values:
Default value: 0
Associated variable: (V.)[ch].MPA.VEL_INT_THRESHOLD[set].xn

Velocity threshold to disable the integral action of the velocity PI. If the velocity is less than the set value for this parameter, the CNC disables the integral action of the velocity PI. If set to 0, the integral action of the velocity PI is not disabled.

DRC | VEL_REFTIME

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 10 000.0000 milliseconds.
Default value: 0 milliseconds.
Associated variable: (V.)[ch].MPA.VEL_REFTIME[set].xn

Time constant of the velocity reference model.

DRC | VEL_OBS_FREQ

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 21 474 836.4700 Hz.
Default value: 1 Hz.
Associated variable: (V.)[ch].MPA.VEL_OBS_FREQ[set].xn

Cutoff frequency of the low passing filter of the velocity estimator.

DRC | VEL_OBS_DAMPING

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 21 474 836.4700.
Default value: 1.
Associated variable: (V.)[ch].MPA.VEL_OBS_DAMPING[set].xn

Damping factor of the low passing filter of the velocity estimator.

DRC | VEL_OBS_ENA

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.VEL_OBS_ENA[set].xn

Enabling the velocity estimator (observer) within the velocity loop.

Value.	Meaning.
Yes	The CNC closes the velocity loop with the estimated motor speed.
No	The CNC closes the velocity loop with the real motor speed.

DRC | VEL_FB_FILT_FREQ

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 21 474 836.4700 Hz.
Default value: 0 Hz.
Associated variable: (V.)[ch].MPA.VEL_FB_FILT_FREQ[set].xn

Cutoff frequency of the first order low-pass filter inserted into the velocity feedback.

DRC | INERTIA_FACTOR

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 21 474 836,4700 %.
Default value: 0 %.
Associated variable: (V.)[ch].MPA.INERTIA_FACTOR[set].xn

Inertia ratio (Jload/Jmotor).

DRC | ST_FRCOMP_POS

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Within ±21 474 836.4700 Nm.
Default value: 0 Nm.
Associated variable: (V.)[ch].MPA.ST_FRCOMP_POS[set].xn

Constant friction compensation in the positive direction of the velocity. It is a constant value for all the positive reference speeds.

DRC | ST_FRCOMP_NEG

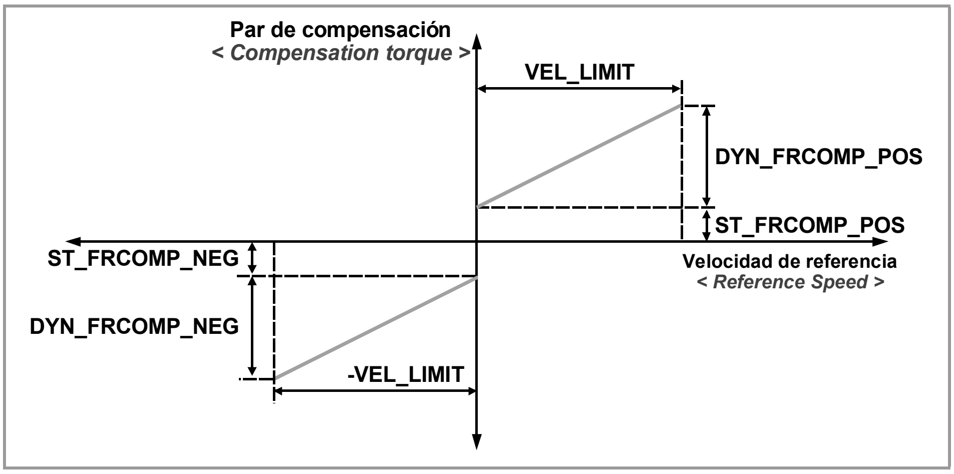
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Within ±21 474 836.4700 Nm.
Default value: 0 Nm.
Associated variable: (V.)[ch].MPA.ST_FRCOMP_NEG[set].xn

Constant friction compensation in the negative direction of the velocity. It is a constant value for all the negative reference speeds.

DRC | DYN_FRCOMP_POS

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Within ±21 474 836.4700 Nm.
Default value: 0 Nm.
Associated variable: (V.)[ch].MPA.DYN_FRCOMP_POS[set].xn

Compensation of the dynamic friction in the positive direction of the velocity. It is the value of the compensation with the reference speed equal to VEL_LIMIT. It is directly proportional to other positive reference speeds.



DRC | DYN_FRCOMP_NEG

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: Within ±21 474 836.4700 Nm.
Default value: 0 Nm.
Associated variable: (V.)[ch].MPA.DYN_FRCOMP_NEG[set].xn

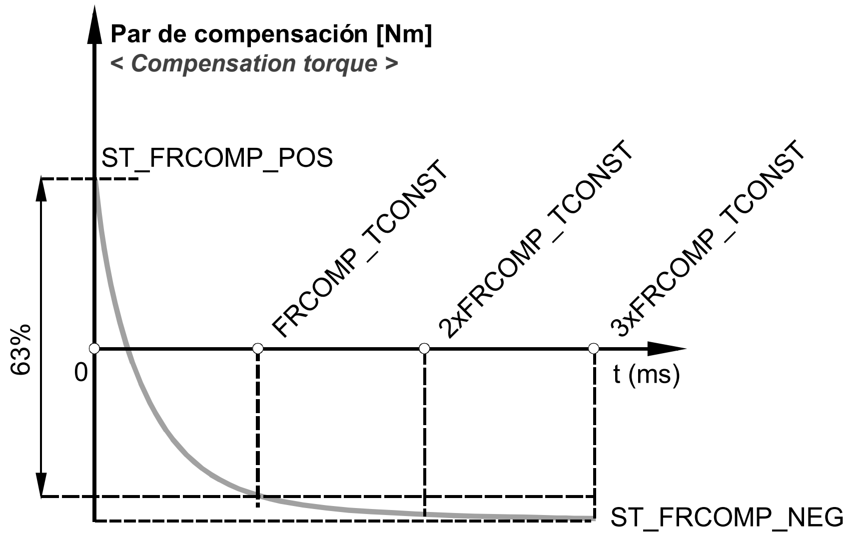
Compensation of the dynamic friction in the negative direction of the velocity. It is the value of the compensation with the reference speed equal to -VEL_LIMIT. It is directly proportional to other negative reference speeds. It is set as an absolute value, i.e. in positive, although the compensation has a negative value.

DRC | FRCOMP_TCONST

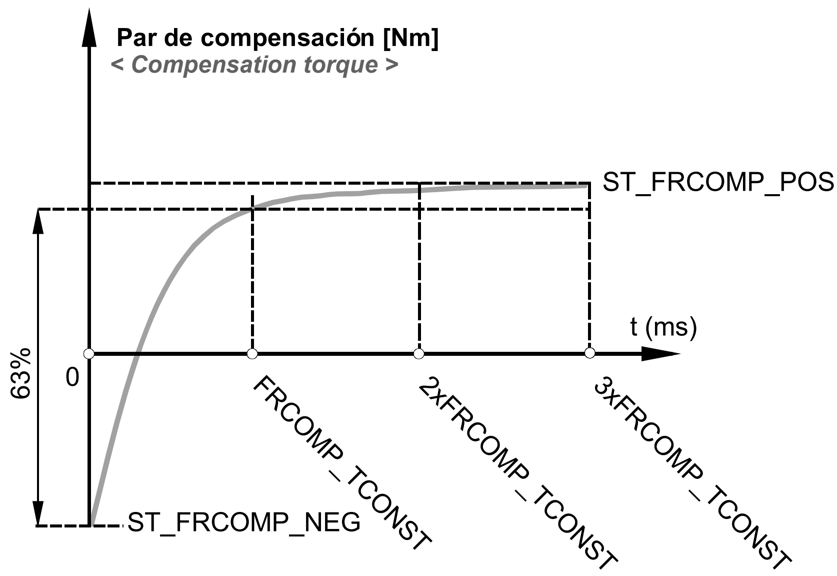
Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 2 147 483.6470 milliseconds.
Default value: 0 milliseconds.
Associated variable: (V.)[ch].MPA.FRCOMP_TCONST[set].xn

Time constant of the friction compensation. Before applying the friction compensation, it goes through a low-pass filter to improve the friction behavior in the velocity direction changes. The constant friction suddenly changes when changing the sign of the reference speed. When it goes through the filter, it smoothens the compensation torque without jerking the system and improving friction behavior. A 0 value cancels the friction compensations.

Torque compensation when going from a positive speed value to a negative speed value.



Torque compensation when going from a negative speed value to a positive speed value.



- Between 0 and FRCOMP_TCONST it sets 63 % of torque compensation.
- Between 0 and 2×FRCOMP_TCONST it sets 87 % of torque compensation.
- Between 0 and 3×FRCOMP_TCONST it sets 95 % of torque compensation.

DRC | FRCOMP_HYST

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 200 000 rpm.
Default value: 0 rpm.
Associated variable: (V.)[ch].MPA.FRCOMP_HYST[set].xn

Friction compensation hysteresis. If set to 0, the drive will internally determine an approximate fixed hysteresis amplitude value using VEL_LIMIT/10000 to compensate for the friction torque.

DRC | FRCOMP_ENABLE

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.FRCOMP_ENABLE[set].xn

Enabling friction compensation.

DRC | TORQUELIM

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.TORQUELIM[set].xn

Torque limit.

DRC | CURRENT_FILT_TYPE_1
..
DRC | CURRENT_FILT_TYPE_4

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 2.0000.
Default value: 0.
Associated variable: (V.)[ch].MPA.CURRENT_FILT_TYPE_1[set].xn

Type current command filter.

Value.	Meaning.
0	Filter disabled.
1	Low passing filter.
2	Notch filter.

DRC | CURRENT_FILT_FREQ_1
..
DRC | CURRENT_FILT_FREQ_4

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 2 147 483.6470 Hz.
Default value: 1000 Hz.
Associated variable: (V.)[ch].MPA.CURRENT_FILT_FREQ_1[set].xn

Filter cutoff frequency.

DRC | CURRENT_FILT_DAMP_1
..
DRC | CURRENT_FILT_DAMP_4

Parameter valid for linear and rotary axes and spindles.
Parameter valid for analog drive, Sercos and simulated.
Possible values: From 0 to 2 147 483.6470.
Default value: 1000.
Associated variable: (V.)[ch].MPA.CURRENT_FILT_DAMP_1[set].xn

The meaning of this parameter depends on the selected current command filter (parameter CURRENT_FILT_TYPE_1 to CURRENT_FILT_TYPE_4).

- If it is a low-pass filter (CURRENT_FILT_TYPE_n=1), this parameter indicates the filter damping factor, in thousandths.
- If it is a notch filter (CURRENT_FILT_TYPE_n=2), this parameter indicates the width of the cutoff frequency, in Hertz.

2.15 Machine parameters for JOG mode.

2.15.1 Configure the jog keys.

JOGKEYDEF
JOGKEYBD2DEF
...
JOGKEYBD8DEF

This parameter shows the table to configure the jog keys. Each table shows the following machine parameters to configure it.

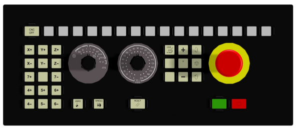
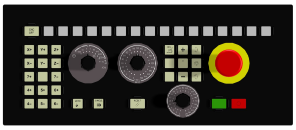
Parameter.	Meaning.
JOGKEYDEF n	Axis and moving direction of each jog key.

JOGKEYDEF | JOGKEYDEF 1
JOGKEYDEF | ···
JOGKEYDEF | JOGKEYDEF 15

Possible values: 1- to 10+, 1 to 10, R, +, -.
Default value: 1+, 2+, 3+, 1-, 2-, 3-, 7-, R, 7+,4+, 5+, 6+, 4-, 5-, 6.-
Associated variable: (V.)MPMAN.JOGKEYDEF[jk]

Axis and moving direction of each jog key. Each one of these parameter sets the function of each JOG key. The number of jog keys available depends on the type of keyboard that may sometimes have up to 15 keys. 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.

OP-PANEL OP-PANEL+SPDL RATE	OP-PANEL-329	QC-C65☐-10K
 		
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15	1 2 3 7 8 9 4 5 6 USERKEYS

2. CONNECTIONS AND MACHINE PARAMETERS.
Machine parameters for JOG mode.

CNCelite
8058 8060
8065 8070

REF: 2210

The values below define key behavior.

Value.	Meaning.
1-, 2-, 3-, .. 10- 1+, 2+, 3+, .. 10+	Keys to define the axis and the jogging direction. The parameter is set with a value between 1- and 10+ (signed). The sign indicates the positive direction (+) or the negative direction (-) of movement and the number corresponds to the logic axis, according to parameter AXISNAME.
1 .. 10	Keys to define the axis to be jogged. The parameter is set with a value between 1 and 10 (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.
R	Rapid key. The parameter is set with the "R" value.

2.15.2 Configure the user keys as jog keys.

USERKEYDEF
USERKEYBD2DEF
..
USERKEYBD8DEF

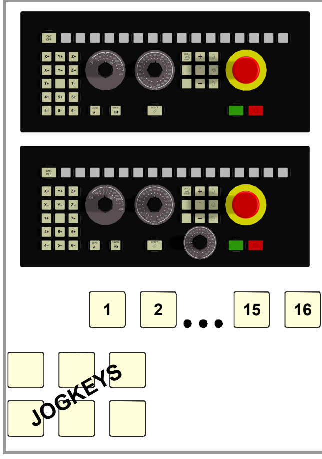
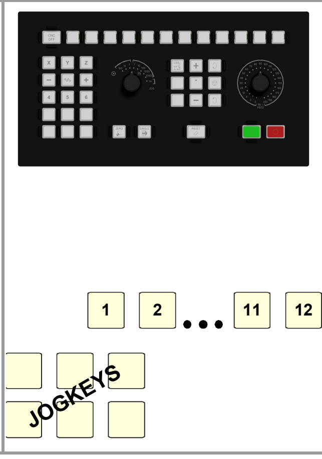
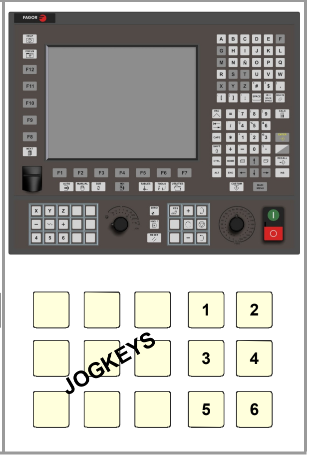
This parameter shows the table to configure the user keys as jog keys. Each table shows the following machine parameters to configure it.

Parameter.	Meaning.
USERKEYDEF n	Axis and moving direction of each jog key.

USERKEYDEF | USERKEYDEF 1
USERKEYDEF | ···
USERKEYDEF | USERKEYDEF 16

Possible values: 1- to 10+, 1 to 10, R, +, -.
Default value: -.
Associated variable: (V.)MPMAN.USERKEYDEF[uk]

Axis and moving direction of each user key. Each one of these parameter sets the function of each user key. The number of user keys available depends on the type of keyboard that may sometimes have up to 16 keys. 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.

OP-PANEL OP-PANEL+SPDL RATE	OP-PANEL-329	QC-C65□-10K
		

The meaning of this parameter is similar to that of machine parameter JOGKEYDEF. To define the behavior of each key, assign to them one of the following values:

Value.	Meaning.
1-, 2-, 3-, .. 10- 1+, 2+, 3+, .. 10+	Keys to define the axis and the jogging direction. The parameter is set with a value between 1- and 10+ (signed). The sign indicates the positive direction (+) or the negative direction (-) of movement and the number corresponds to the logic axis, according to parameter AXISNAME.
1 .. 10	Keys to define the axis to be jogged. The parameter is set with a value between 1 and 10 (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.
R	Rapid key. The parameter is set with the "R" value.

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.

2.15.3 Behavior of the JOG keys.

JOGTYPE

Possible values: Pressed axis / Maintained axis.
Default value: Pressed axis.
Associated variable: (V.)MPMAN.JOGTYPE

Behavior of the JOG keys. This parameter is applied when the jog keyboard has different keys to select the axis and the moving direction. In this case, jogging an axis requires activating both the axis key and the moving direction.

- 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].

2.15.4 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.

  Disk graduated with 100 lines.

  Pulses per turn: 100

  Resolutions (X): 0,001, 0,01, 0,1

- Individual handwheel for the Y axis.

  Disk graduated with 100 lines.

  Pulses per turn: 200

  Resolutions (Y): 0,001, 0,01, 0,1

- General handwheel for the rest of the axes (Z, A).

  Disk graduated with 100 lines.

  Pulses per turn: 100

  Resolutions (Z): 0,001, 0,01, 0,1

  Resolutions (A): 0,01, 0,1, 1

Parameter setting:

Handwheel associated with X.

Machine parameter.	Value.	Meaning.
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

Handwheel associated with Y.

Machine parameter.	Value.	Meaning.
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

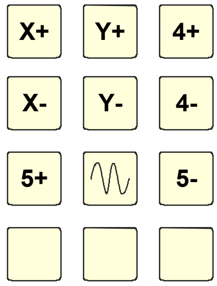
Handwheel associated with the rest of the axes (Z + A).

Machine parameter.	Value.	Meaning.
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	Axis A. Resolution 0.01
A - MPGRESOL 10	0.1	Axis A. Resolution 0.1
A - MPGRESOL 100	1	Axis A. Resolution 1

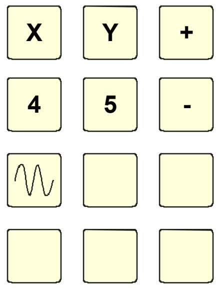
Setting the jog keys.

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.

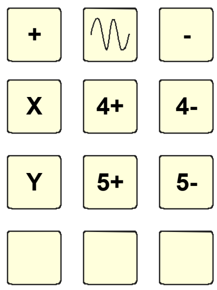
Example 1: JOG-PANEL keyboard.

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	- - -	

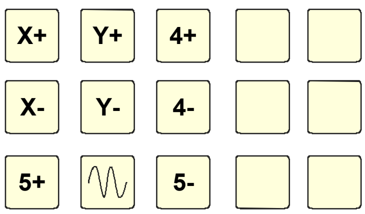
Example 2: JOG-PANEL keyboard.

Jog keyboard.	JOGKEYDEF	Key.	Value.
	1	[X]	1
	2	[Y]	2
	3	[+]	+
	4	[4]	3
	5	[5]	4
	6	[-]	-
	7	[R]	R
	8 - 15	- - -	

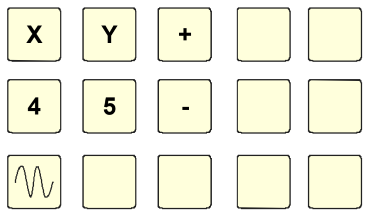
Example 3: JOG-PANEL keyboard.

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	- - -	

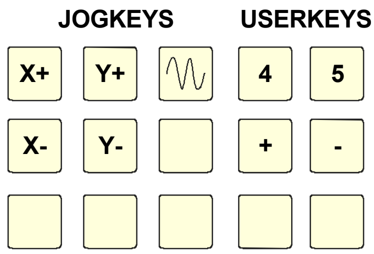
Example 4: LCD-10K keyboard.

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.
	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	- - -	

2.16 Machine parameters for the M function table.

2.16.1 M function table.

MTABLESIZE

Possible values: from 0 to 200.
Default value: 50.
Associated variable: (V.)MPM.MTABLESIZE

This parameter indicates the number of M functions in the table. They can be either new or existing control functions. The table is used to set the type of synchronization (executed function confirmation) and associate a subroutine with the M functions.

Keep in mind that some M functions, besides what is indicated in this table, have a special meaning for the CNC.

Function.	Meaning.
M00	Program stop.
M01	Conditional program stop.
M02	End of program.
M03	Start the spindle clockwise.
M04	Start the spindle counterclockwise.
M05	Stop the spindle.
M06	Tool change.
M08	Enable coolant.
M09	Disable Coolant.
M17	End of a global or local subroutine.
M19	Spindle orientation.
M29	End of a global or local subroutine.
M30	End of program.
M41	Selects gear ·1·.
M42	Selects gear ·2·.
M43	Selects gear ·3·.
M44	Selects gear ·4·.

DATA

This parameter shows the table to set the M functions. The following fields must be defined for each M function.

DATA	
MNUM	M function number.
SYNCHTYPE	Type of synchronism.
MTIME	Estimated time to execute the M function.
MPROGNAME	Name of subroutine associated with M function.
MPLC	Send the M function to the PLC during block search.
COMMENT	Description of the M function.

DATA | MNUM

Possible values: from 0 to 65 535.
Associated variable: (V.)MPM.MNUM[pos]

M function number.

DATA | SYNCHTYPE

Possible values: Without synchronization / Before-Before / Before-After / After-After.
Default value: Before-Before.
Associated variable: (V.)MPM.SYNCHTYPE[pos]

Type of synchronism. Since the M functions can be programmed in a block together with the axis, the CNC must be indicated when the function is sent to the PLC and when it is synchronized (function executed confirmation).

Value.	Meaning.
Without synchronization.	M function not synchronized. For example, 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 been turned on.
Before - Before.	The M function is sent to the PLC and synchronized before the movement. For example, for functions M03 and M04 to start the spindle, it is convenient to execute and synchronize them 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. For example, for function M05 for the spindle, it is convenient to execute and synchronize it after the movement.

DATA | MTIME

Possible values: From 0 to 1 000 000 milliseconds.
Default value: 0 milliseconds.
Associated variable: (V.)MPM.MTIME[pos]

Estimated time to execute the M function. In EDISIMU work mode, there is an option that can calculate 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.

DATA | MPROGNAME

Possible values: any text with up to 64 characters.
Default value: Without subroutine.
Associated variable: (V.)MPM.MPROGNAME[pos]

Name of subroutine associated with M function. The M functions may have an associated subroutine that the CNC will execute instead of the function. To execute the M function, it must be included in the subroutine. To send the M function to the PLC, it must be programmed into 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).

The name of the subroutine can have 64 characters and must be saved in the folder ..\Mtb \Sub.

i

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..

DATA | MPLC

Possible values: Yes / No.
Default value: Yes.
Associated variable: (V.)MPM.MPLC[pos]

The block search can be used restore the program history up to a specific block. This field indicates whether the M function must be sent to the PLC during block search. All M functions set in the table will be sent out to the PLC or not depending on the setting of this field; the rest of M functions will be sent or not depending on the setting of machine parameter FUNPLC.

MPLC.	Meaning.
Yes.	The functions are sent out to the PLC during block search as they are being read. The subroutine associated with the M functions is executed when the M is sent out to the PLC.
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.

In order to send the M function to the PLC, it must not have an associated subroutine, and if it does, the M function must be programmed into the subroutine.

DATA | COMMENT

Possible values: 255 characters.

Default value: -.

Associated variable: (V.)MPM.COMMENT[pos]

This field can associate a short description with the M function. 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 ..\Mtb\Data\Lang.

This field is for information only; it is not used by the CNC.

2.17 Machine parameters for kinetics.

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.

2.17.1 Kinematics configuration.

NKIN

Possible values: From 0 to 6.
Default value: 0.
Associated variable: (V.)MPK.NKIN

Number of kinematics defined. 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

This parameter shows the table to define the kinematics. The kinematics preset by Fagor (parameter TYPE=1..99) have the following machine parameters to configure them.

Parameter.	Meaning.
TYPE	Kinematics type.
HEADREF	Machine reference position HEAD/TOOL.
TDATA 1 .. TDATA 100	Numeric parameter in decimal format.
TDATA_I 1 .. TDATA_I 100	Numeric parameter in integer format.

To configure the OEM kinematics (parameter TYPE=100..105), use the following machine parameters.

Parameter.	Meaning.
TYPE	Kinematics type.
NKINAX	Number of axes of the kinematics.
PARAM_D_SIZE	Number of numeric parameters in decimal format.
TDATA n	Numeric parameter in decimal format.
PARAM_I_SIZE	Number of numerical parameters in integer format.
TDATA_I n	Numeric parameter in integer format.
AUXCTE_SIZE	Size of the area of auxiliary variables.
KINDATA_SIZE	Size of the area for general purpose data.

KINEMATIC n | TYPE

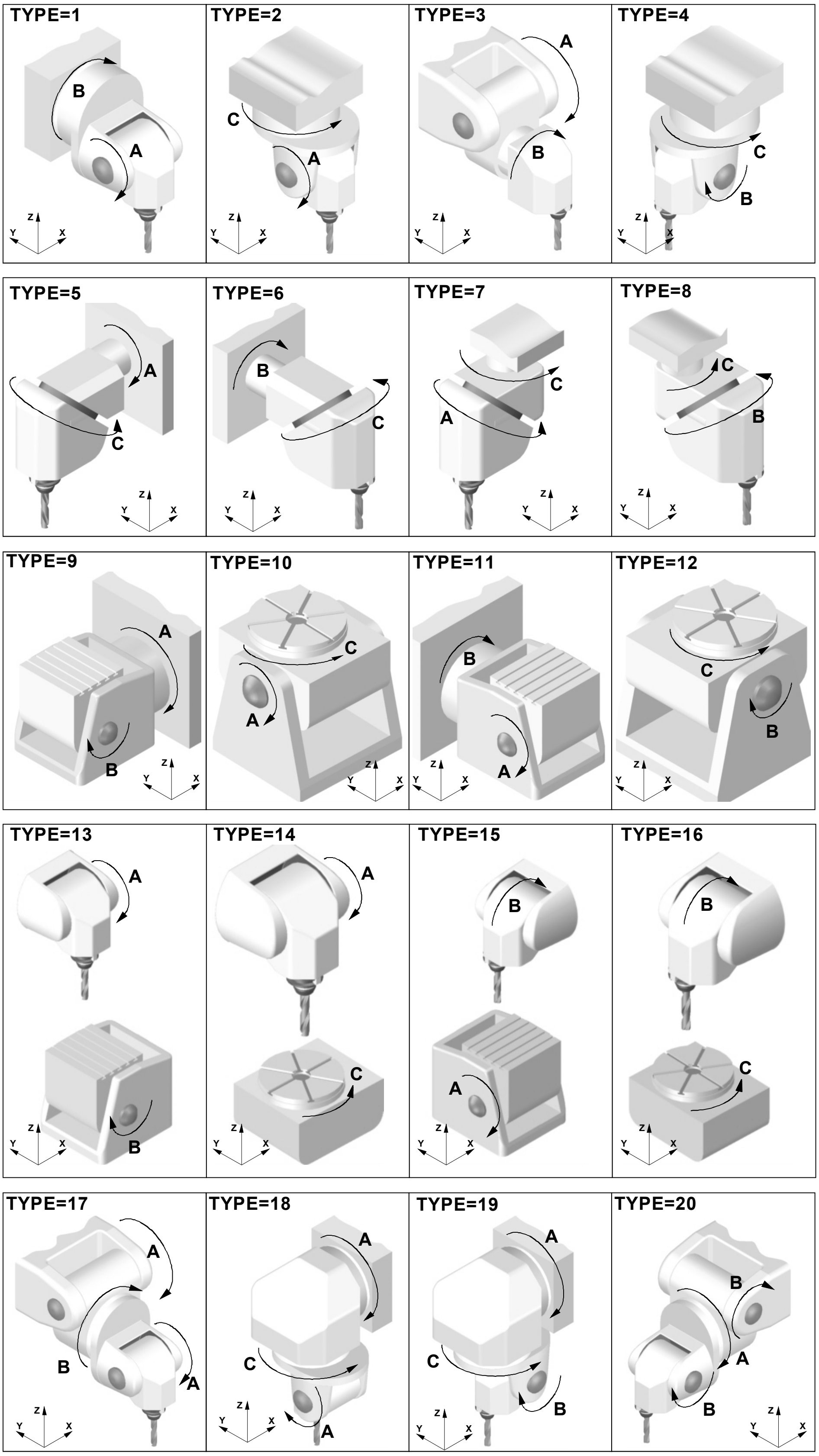
Possible values (1): From 0 to 99 (kinematics predefined by Fagor).
Possible values (2): From 0 to 105 (OEM kinematics).
Default value: 0 (there are no kinematics).
Associated variable: (V.)MPK.TYPE

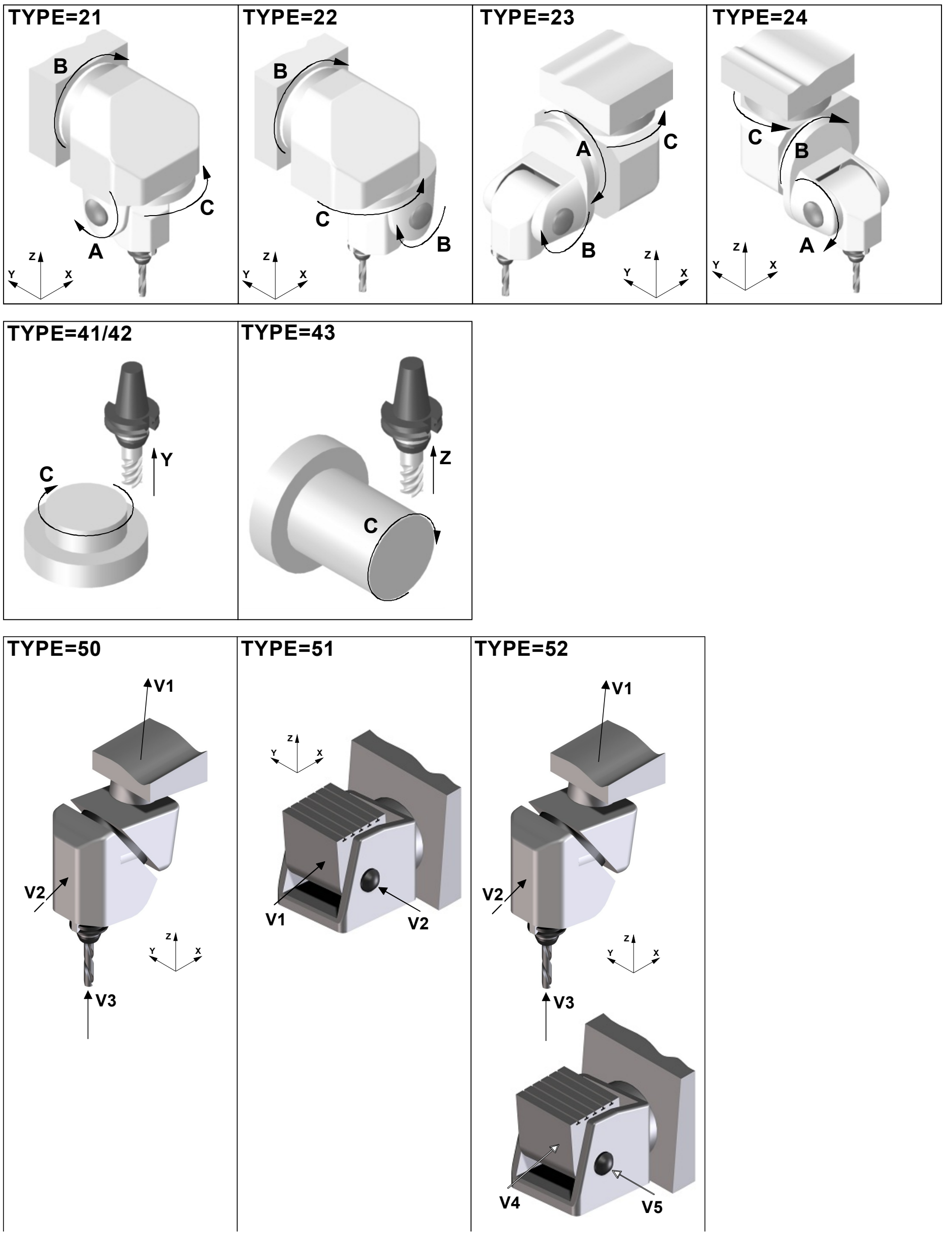
Kinematics type. Types 1 to 99 are for the predefined Fagor kinematics and types 100 to 105 are for the kinematics integrated into the CNC by the OEM.

Kinematics predefined by Fagor.

Value.	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.

Value.	Kinematics type.
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 = 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.
TYPE = 50	Vectorial definition of spindle kinematics.
TYPE = 51	Vectorial definition of table kinematics.
TYPE = 52	Vectorial definition of spindle-table kinematics.

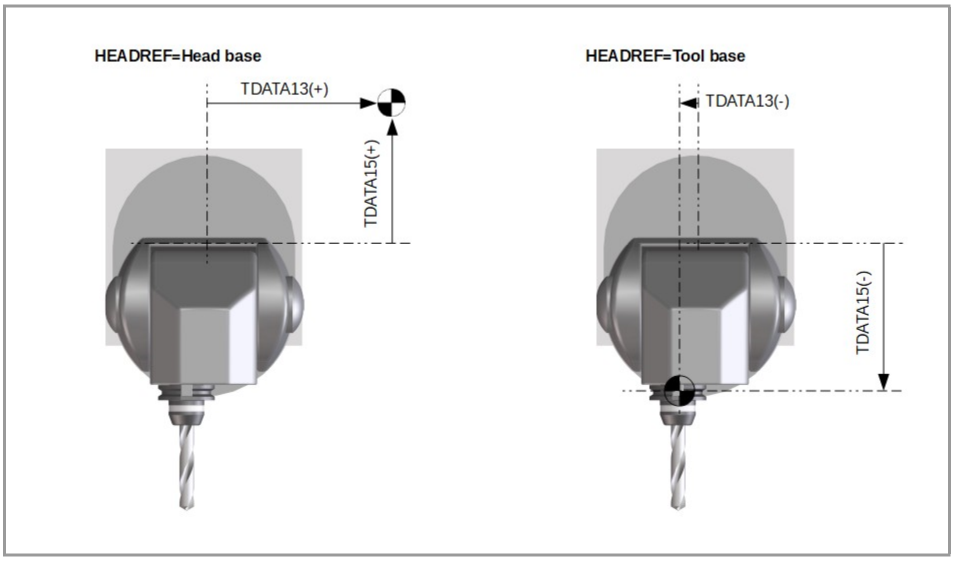

TYPE=1
TYPE=2
TYPE=3
TYPE=4
TYPE=5
TYPE=6
TYPE=7
TYPE=8
TYPE=9
TYPE=10
TYPE=11
TYPE=12
TYPE=13
TYPE=14
TYPE=15
TYPE=16
TYPE=17
TYPE=18
TYPE=19
TYPE=20


TYPE=21
TYPE=22
TYPE=23
TYPE=24
TYPE=41/42
TYPE=43
TYPE=50
TYPE=51
TYPE=52
A
B
C
Y
Z
V1
V2
V3
V4
V5

KINEMATIC n | HEADREF

Possible values: Tool base / Head base / Auto tool base.
Default value: Tool base.
Associated variable: (V.)MPK.HEADREF

This parameter enables you to place the reference point of the kinematics at the point of the tool holder, following calibration (Tool Base) or activation (Auto Tool Base).



HEADREF	Meaning.
Tool base.	The kinematics calibration places the reference point of the kinematics on the point of the tool holder. From the parameters TDATA13 to TDATA15 (non-vectorial kinematics) or TDATA1 to TDATA3 (vectorial kinematics), the calibration calculates the new point of reference of the kinematics. This option modifies the value of these parameters in the table. This is the recommended option when the machine has a single spindle that is calibrated regularly.
Head base.	The kinematics calibration does not modify the reference point of the kinematics, used to define the kinematics in the machine parameters. This is the recommended when the machine has a spindle change.
Auto tool base.	On activating the kinematics, the CNC places the point of reference of the kinematics on the point of the tool holder. From the parameters TDATA13 to TDATA15 (non-vectorial kinematics) or TDATA1 to TDATA3 (vectorial kinematics), the CNC calculates the new point of reference of the kinematics. The value of these parameters is not changed on the parameters table.

KINEMATIC n | TDATA 1
..
KINEMATIC n | TDATA 100

Possible values: Between ±999 999 999.0000
Default value: 0.
Associated variable: (V.)MPK.TDATAkin[nb]
Associated variable: (V.)MPK.MAXOFTDATAkin[nb]

Numeric parameter in decimal format. The following fields must be defined for each parameter.

DATA	
VALUE	To be defined by the OEM. Value of the parameter.
MAX OFFSET	To be defined by the OEM. Maximum offset that may be defined by the user.
OFFSET	To be defined by the user. Offset of the parameter.

On the "Value" column, the OEM sets the parameter value that may be modified by the user using the "OFFSET" column or its associated variable.

On the "MAX OFFSET" column, the OEM determines the maximum offset that the user may set. This column will always be a positive value and the offset set by the user will be a value between "- MAX OFFSET" and "+ MAX OFFSET". If he sets a maximum offset of zero, the user will not be able to set any offset.

The offset values set by the user will be added to its corresponding TDATA parameter. The new value will be effective when activating the kinematics (#KIN ID) or RTCP.

KINEMATIC n | TDATA_I 1
..
KINEMATIC n | TDATA_I 100

Possible values: Between ±2 147 483 647
Default value: 0.
Associated variable: (V.)MPK.TDATA_Ikin[nb]
Associated variable: (V.)MPK.MAXOFTDATA_Ikin[nb]

Numeric parameter in integer format. The following fields must be defined for each parameter.

DATA	
VALUE	To be defined by the OEM. Value of the parameter.
MAX OFFSET	To be defined by the OEM. Maximum offset that may be defined by the user.
OFFSET	To be defined by the user. Offset of the parameter.

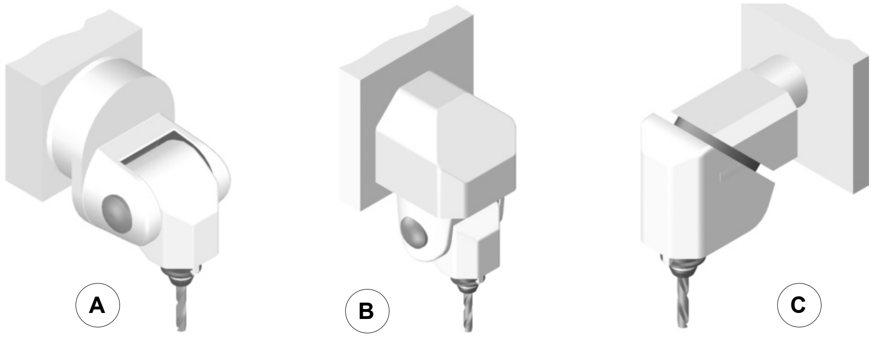
On the "Value" column, the OEM sets the parameter value that may be modified by the user using the "OFFSET" column or its associated variable.

On the "MAX OFFSET" column, the OEM determines the maximum offset that the user may set. This column will always be a positive value and the offset set by the user will be a value between "- MAX OFFSET" and "+ MAX OFFSET". If he sets a maximum offset of zero, the user will not be able to set any offset.

The offset values set by the user will be added to its corresponding TDATA parameter. The new value will be effective when activating the kinematics (#KIN ID) or RTCP.

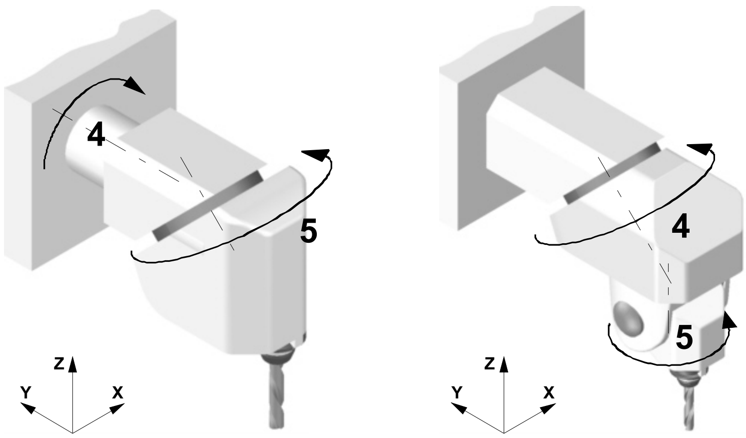
2.17.2 Definition of the spindle kinetics (types 1 through 8)

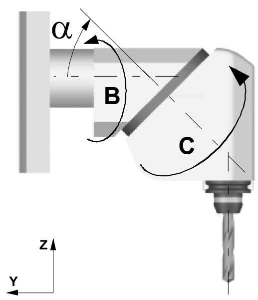
These kinematics can control spherical, orthogonal and angular spindles.



A Spherical spindle. B Orthogonal spindle. C Angular spindle.

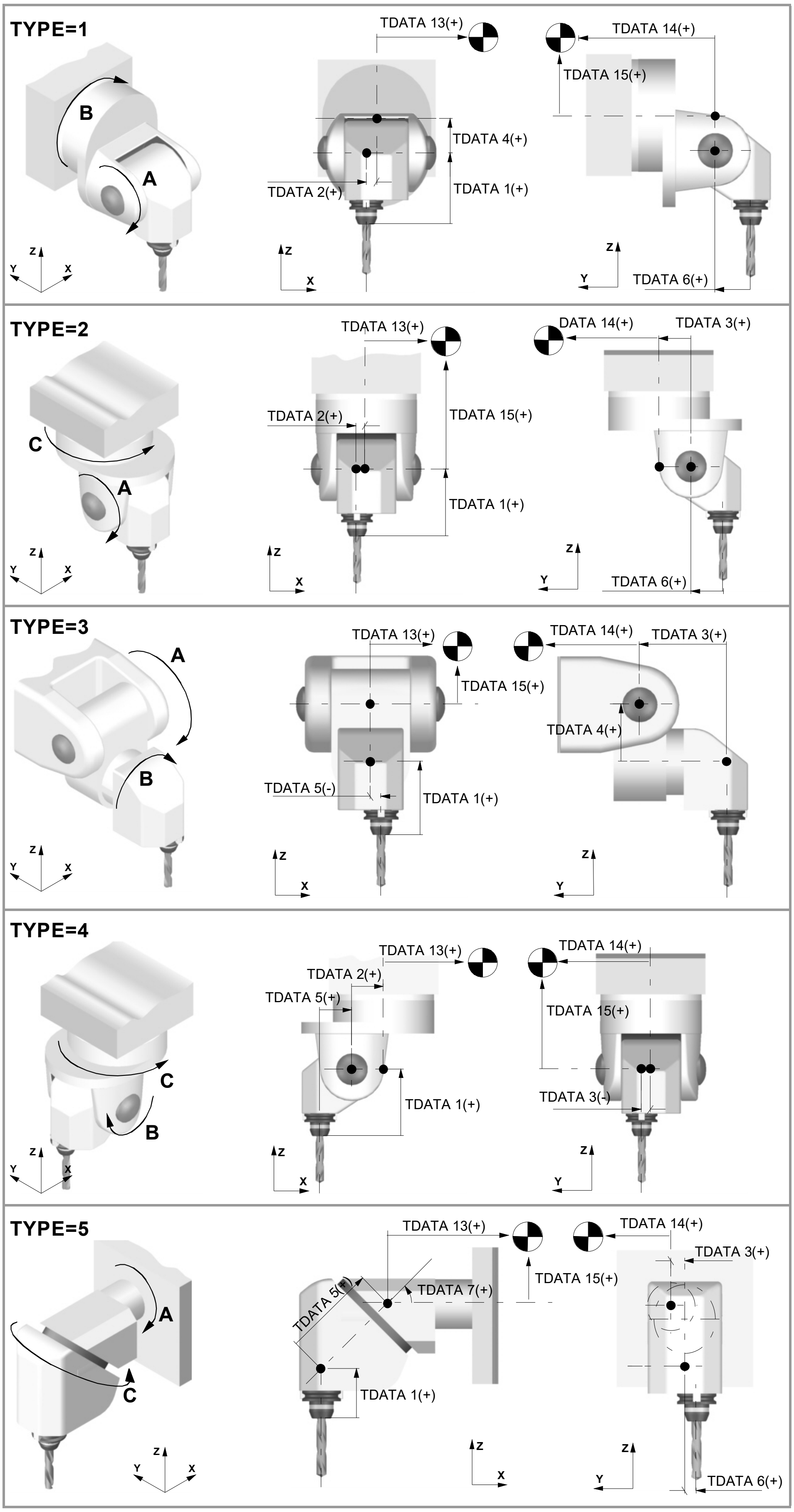
When having an angular spindle head, the main rotary axis (4) must rotate around one of the main axes (X, Y, Z) and the secondary or dragged axis (5) will form a particular angle. The left figure meets this condition, whereas in the right one the main rotary axis (4) does not rotate around the Y axis (it forms an angle with it).

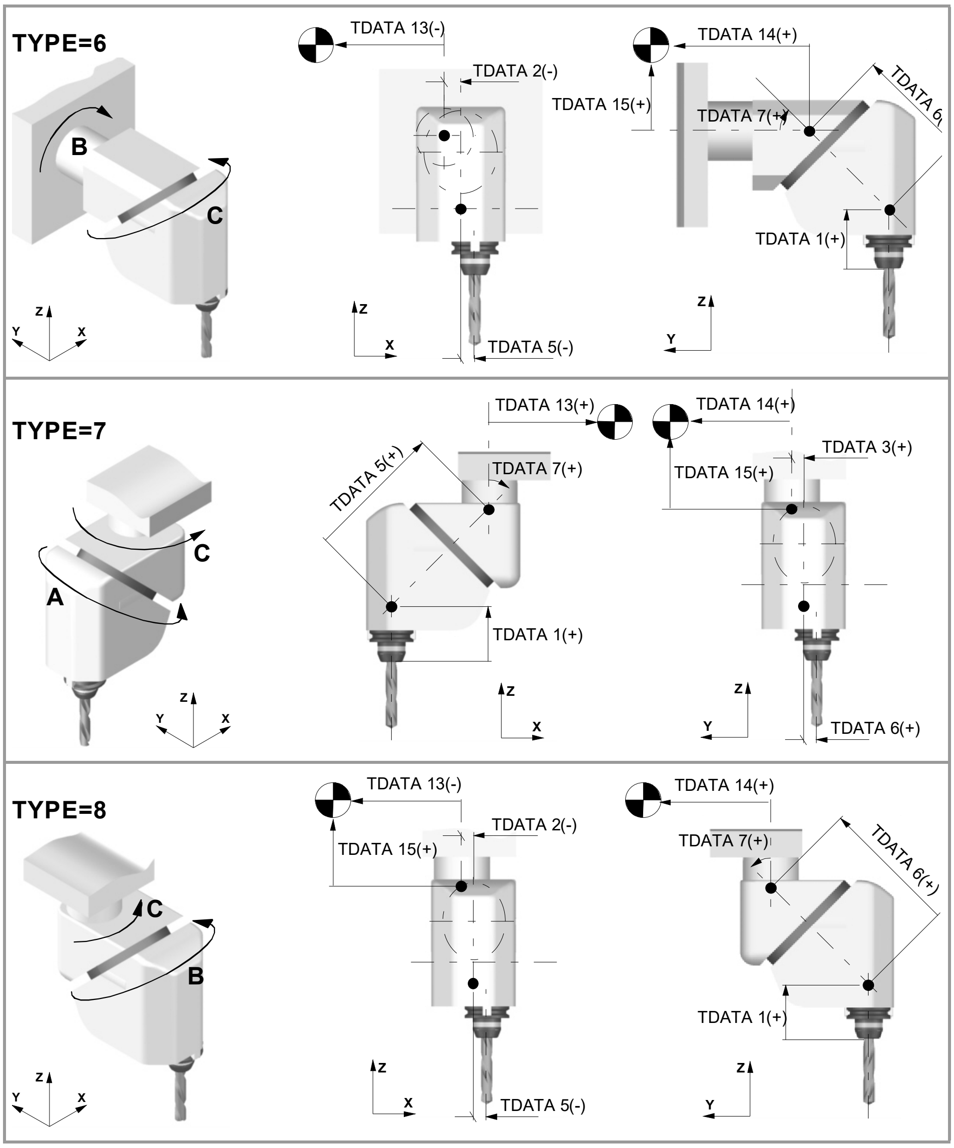




From now on, all the explanations will assume that the main axes are X Y and Z and their associated rotary axes A, B and C.

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 (TYPE 1..8)

Possible values: Between ±999 999 999.0000
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[1]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[1]

Distance between the tip of the quill and the secondary rotary axis along the Z axis. It is not necessary to define it in all the kinematics.

TDATA2 (TYPE 1..8)

Possible values: Between ±999 999 999.0000
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[2]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[2]

Distance between the secondary and main rotary axis along the X axis. It is not necessary to define it in all the kinematics.

TDATA3 (TYPE 1..8)

Possible values: Between ±999 999 999.0000
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[3]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[3]

Distance between the secondary rotary axis and the main axis along the Y axis. It is not necessary to define it in all the kinematics.

TDATA4 (TYPE 1..8)

Possible values: Between ±999 999 999.0000
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[4]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[4]

Distance between the secondary rotary axis and the main axis along the Z axis. It is not necessary to define it in all the kinematics.

TDATA5 (TYPE 1..8)

Possible values: Between ±999 999 999.0000
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[5]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[5]

Distance between the tool axis and the secondary rotary axis along the X axis. It is not necessary to define it in all the kinematics.

TDATA6 (TYPE 1..8)

Possible values: Between ±999 999 999.0000
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[6]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[6]

Distance between the tool axis and the secondary rotary axis along the Y axis. It is not necessary to define it in all the kinematics.

TDATA7 (TYPE 1..8)

Possible values: Within ±99 999,9999º.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[7]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[7]

Angle between the main and secondary rotary axes on rotary spindles. It is not necessary to define it in all the kinematics.

TDATA8 (TYPE 1..8)

Possible values: Within ±99 999,9999º.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[8]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[8]

Rest position of the main rotary axis. The rest position is when the tool is perpendicular to the work plane (parallel to the longitudinal axis).

TDATA9 (TYPE 1..8)

Possible values: Within ±99 999,9999º.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[9]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[9]

Rest position of the secondary rotary axis. The rest position is when the tool is perpendicular to the work plane (parallel to the longitudinal axis).

TDATA10 (TYPE 1..8)

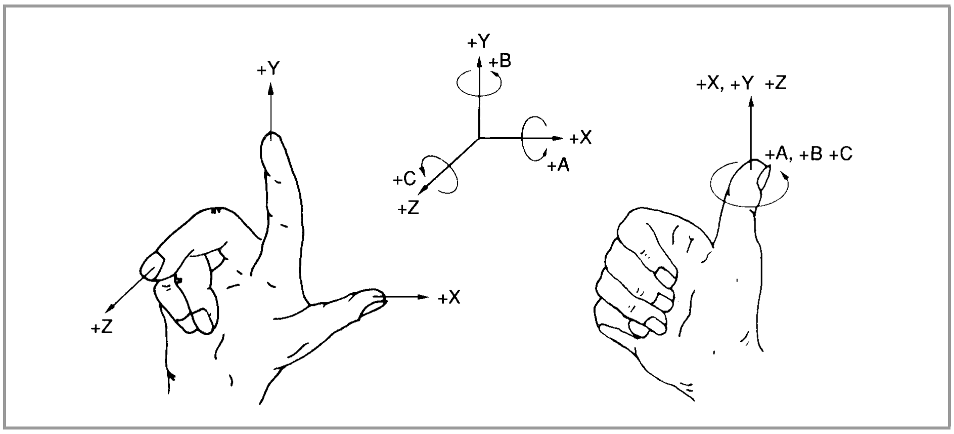
Possible values: 0 (by the standard DIN 66217) / 1 (opposite to the standard DIN 66217).
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[10]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[10]

Turning direction of the main rotary axis. DIN 66217 standard can be easily remembered using the right-hand rule. The positive turning direction of an axis is determined by the direction of your fingers, by closing them around the linear axis while your thumb points toward the positive direction of the linear axis.

TDATA11 (TYPE 1..8)

Possible values: 0 (by the standard DIN 66217) / 1 (opposite to the standard DIN 66217).
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[11]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[11]

Turning direction of the secondary rotary axis. 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 (TYPE 1..8)

Possible values: 0 to 3.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[12]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[12]

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 (TYPE 1..8)

Possible values: Between ±999 999 999.0000
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[13]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[13]

Distance between the main rotary axis and the reference point, along the X axis.

TDATA14 (TYPE 1..8)

Possible values: Between ±999 999 999.0000
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[14]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[14]

Distance between the main rotary axis and the reference point, along the Y axis.

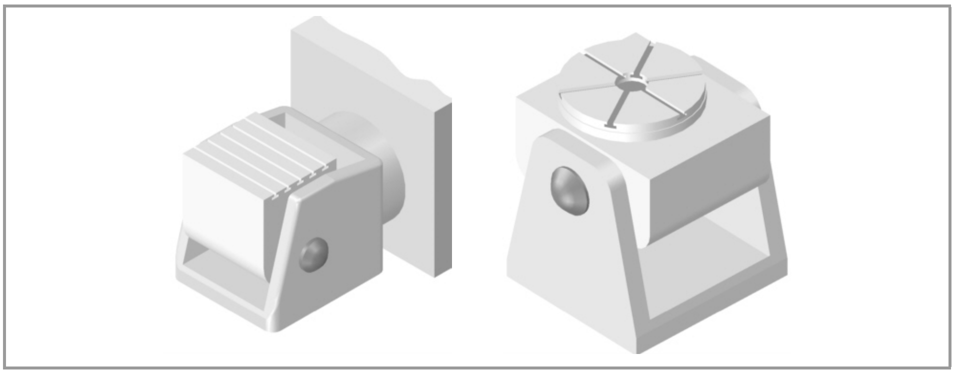
TDATA15 (TYPE 1..8)

Possible values: Between ±999 999 999.0000
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[15]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[15]

Distance between the main rotary axis and the reference point, along the Z axis.

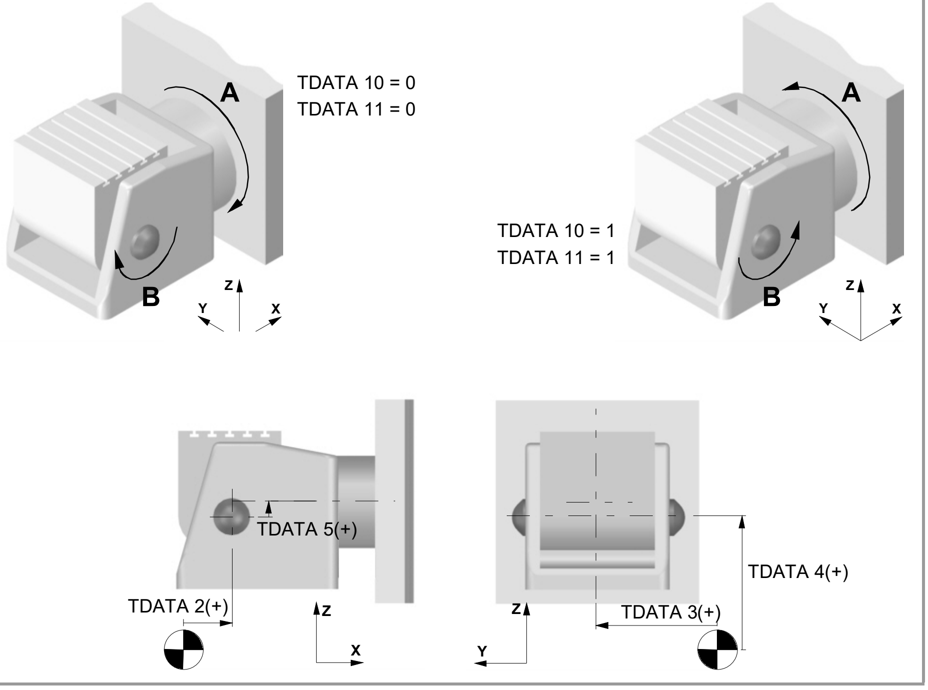
2.17.3 Definition of the table kinetics (types 9 through 12).

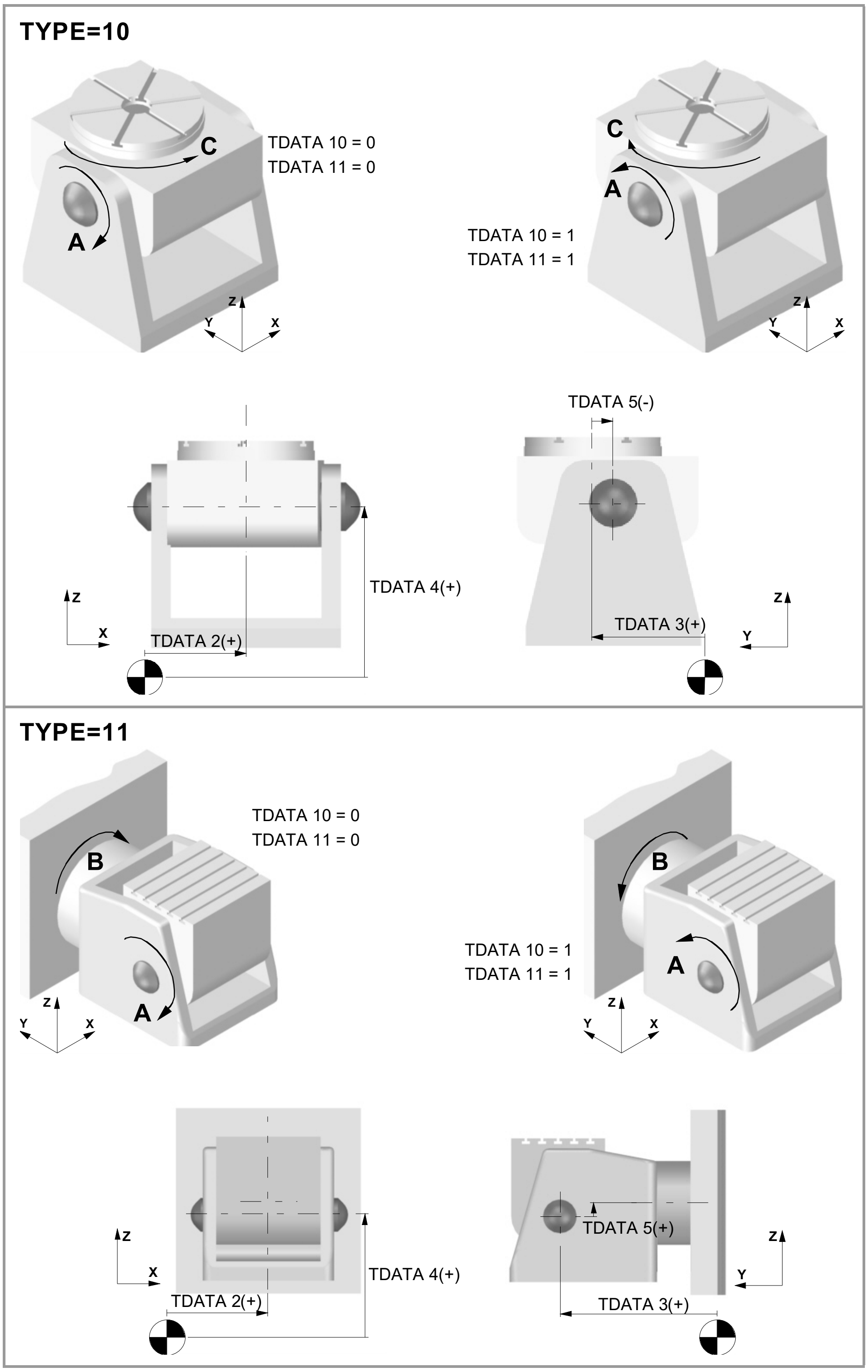
These kinematics can control the following types of rotary tables.

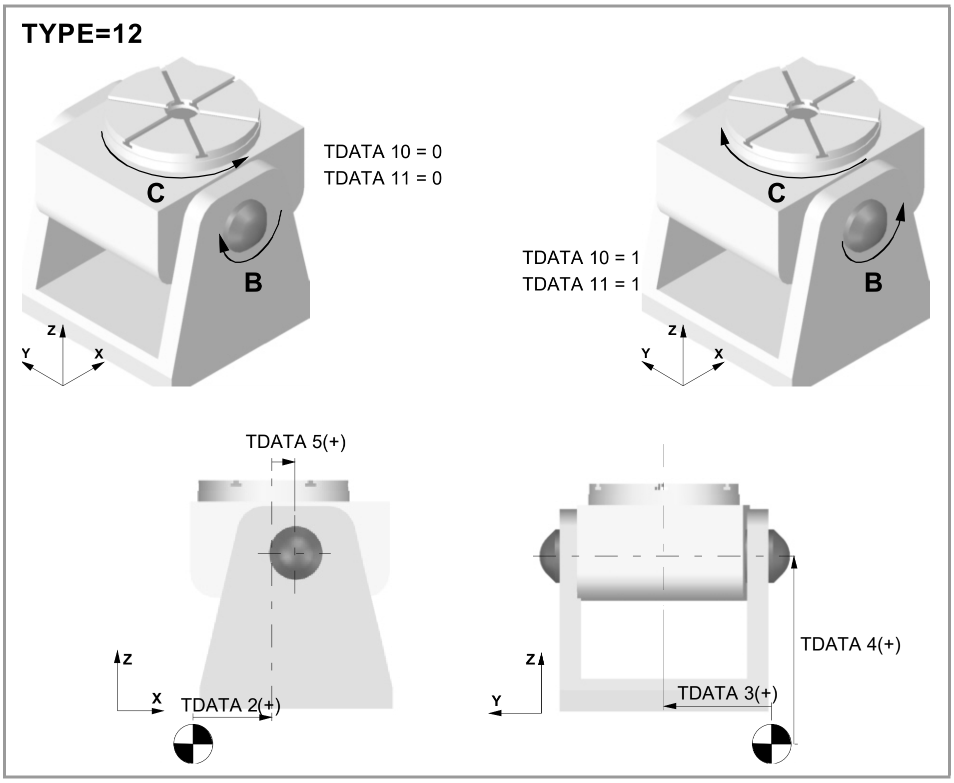


The dimensions may be defined with a positive or negative value. The (+) sign in the illustrations indicates that the direction is assumed as positive.

TYPE=9




TYPE=10
C
A
TDATA 10 = 0
TDATA 11 = 0
TDATA 10 = 1
TDATA 11 = 1
TDATA 5(-)
TDATA 4(+)
TDATA 2(+)
TDATA 3(+)
TYPE=11
B
A
TDATA 10 = 0
TDATA 11 = 0
TDATA 10 = 1
TDATA 11 = 1
TDATA 5(+)
TDATA 4(+)
TDATA 2(+)
TDATA 3(+)



TDATA2 (TYPE 9..12)

Possible values: Between ±999 999 999.0000
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[2]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[2]

Distance from the secondary rotary axis to the machine reference point or the intersection with the primary axis along the X axis.

TDATA3 (TYPE 9..12)

Possible values: Between ±999 999 999.0000
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[3]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[3]

Distance from the secondary rotary axis to the machine reference point or the intersection with the primary axis along the Y axis.

TDATA4 (TYPE 9..12)

Possible values: Between ±999 999 999.0000
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[4]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[4]

Distance from the secondary rotary axis to the machine reference point or the intersection with the primary axis along the Z axis.

TDATA5 (TYPE 9..12)

Possible values: Between ±999 999 999.0000
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[5]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[5]

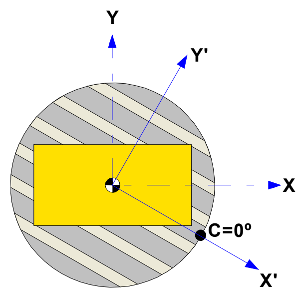
Distance between the secondary and the main rotary tables.

TDATA7 (TYPE 9..12)

Possible values: 0 (machine zero) / 1 (part zero).
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[7]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[7]

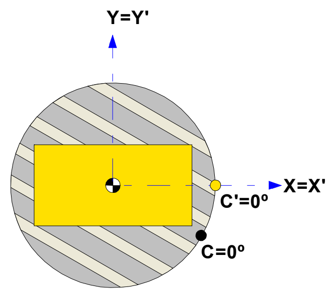
Origin for applying the RTCP. This parameter indicates whether the CNC applies the RTCP referred to machine zero or to part zero. For this type of kinematics it is recommended to set this parameter with a value of ·1·.

TDATA7=0



C=0° (machine coordinates)

TDATA7=1



C=0° (machine coordinates)
C'=0° (part coordinates)

TDATA8 (TYPE 9..12)

Possible values: Within ±99 999,9999º.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[8]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[8]

Rest position of the main rotary axis. The rest position is when the tool is perpendicular to the work plane (parallel to the longitudinal axis).

TDATA9 (TYPE 9..12)

Possible values: Within ±99 999,9999º.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[9]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[9]

Rest position of the secondary rotary axis. The rest position is when the tool is perpendicular to the work plane (parallel to the longitudinal axis).

TDATA10 (TYPE 9..12)

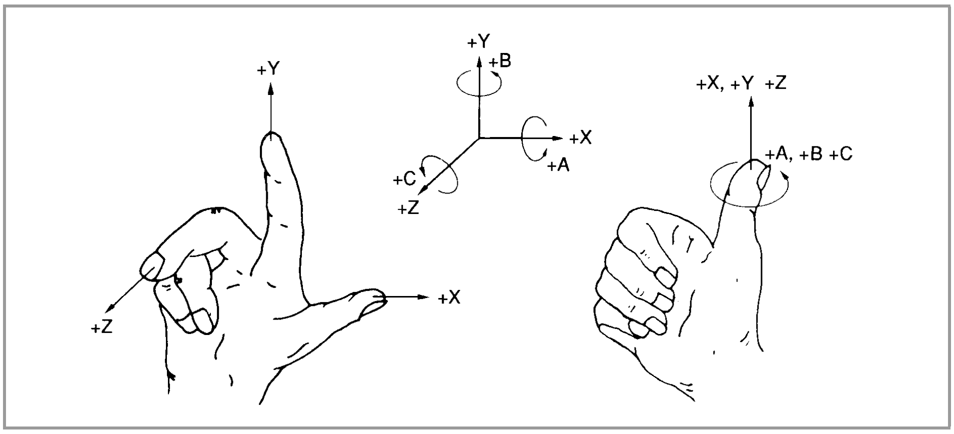
Possible values: 0 (by the standard DIN 66217) / 1 (opposite to the standard DIN 66217).
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[10]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[10]

Turning direction of the main rotary axis.

TDATA11 (TYPE 9..12)

Possible values: 0 (by the standard DIN 66217) / 1 (opposite to the standard DIN 66217).
Default value: 1.
Associated variable (1): (V.)MPK.TDATAkin[11]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[11]

Turning direction of the secondary rotary axis. 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.



TDATA12 (TYPE 9..12)

Possible values: 0 to 3.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[12]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[12]

This parameter indicates whether the rotary axes are manual or servo-controlled.

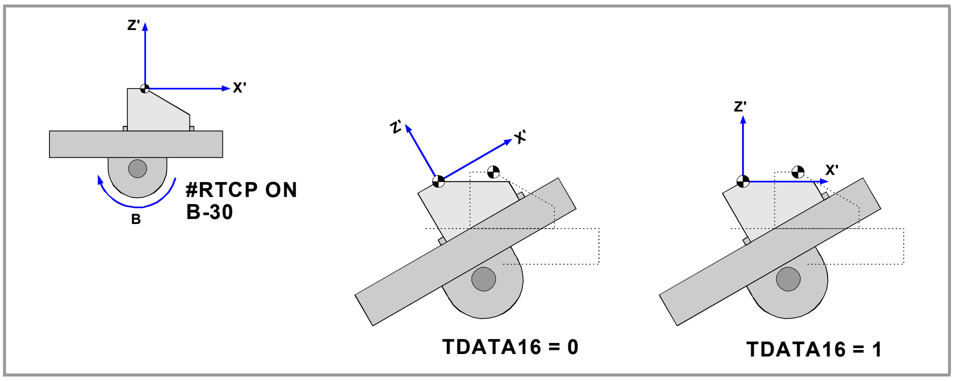
TDATA12	Meaning.
0	The two axes are servo-controlled.
1	Main axis manual and secondary axis servo-controlled.
2	Main axis servo-controlled and secondary axis manual.
3	The two axes are manuals.

TDATA16 (TYPE 9..12)

Possible values: 0 / 1.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[16]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[16]

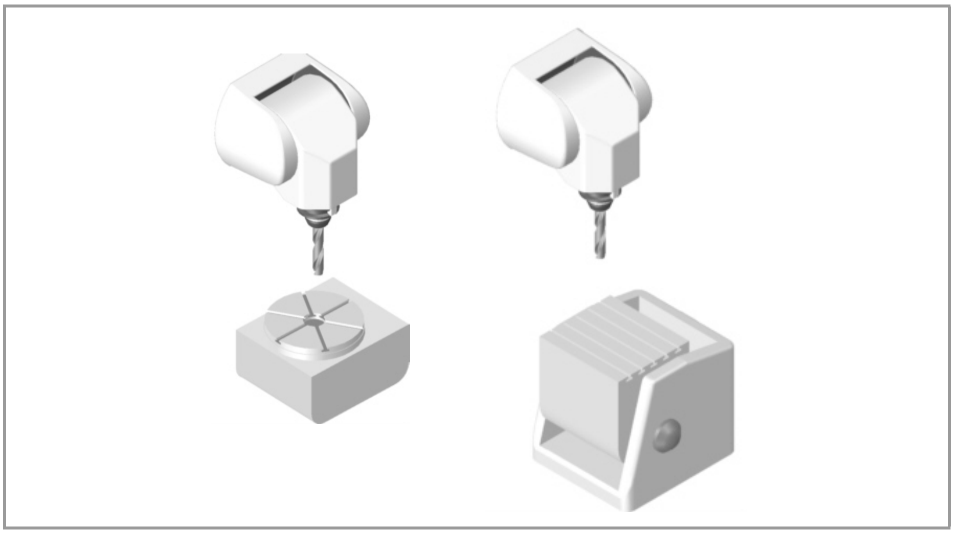
Rotate the part coordinate system when rotating the table. Having the RTCP mode active, this parameter determines whether the part coordinate system is fixed on the part and rotates with it or it stays parallel to the machine coordinate system.

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.



2.17.4 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.

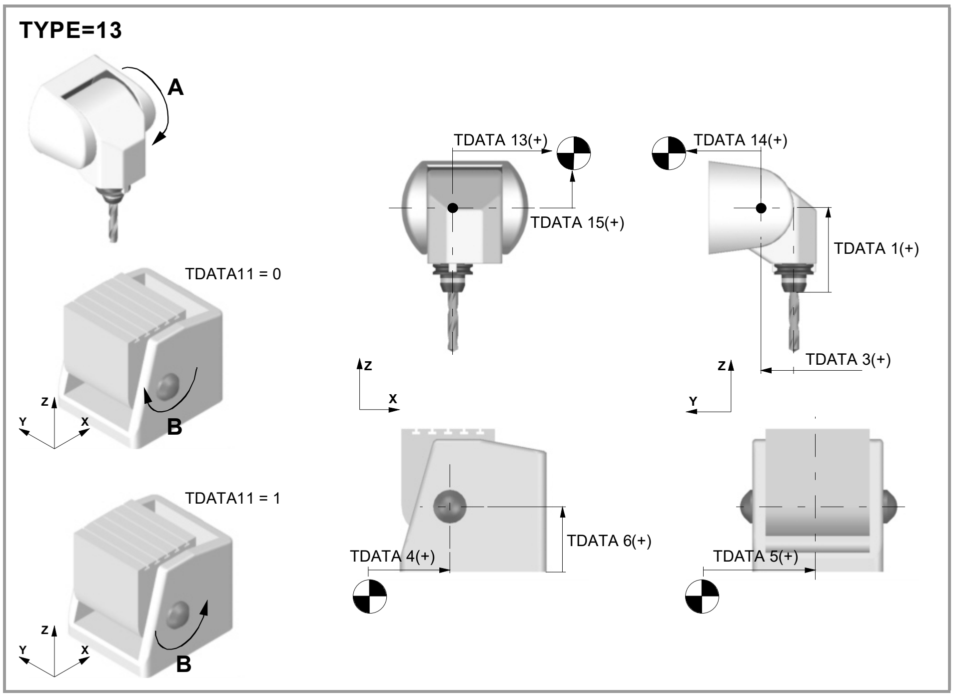


The order of the axes in the channel where the kinematics is applied is:

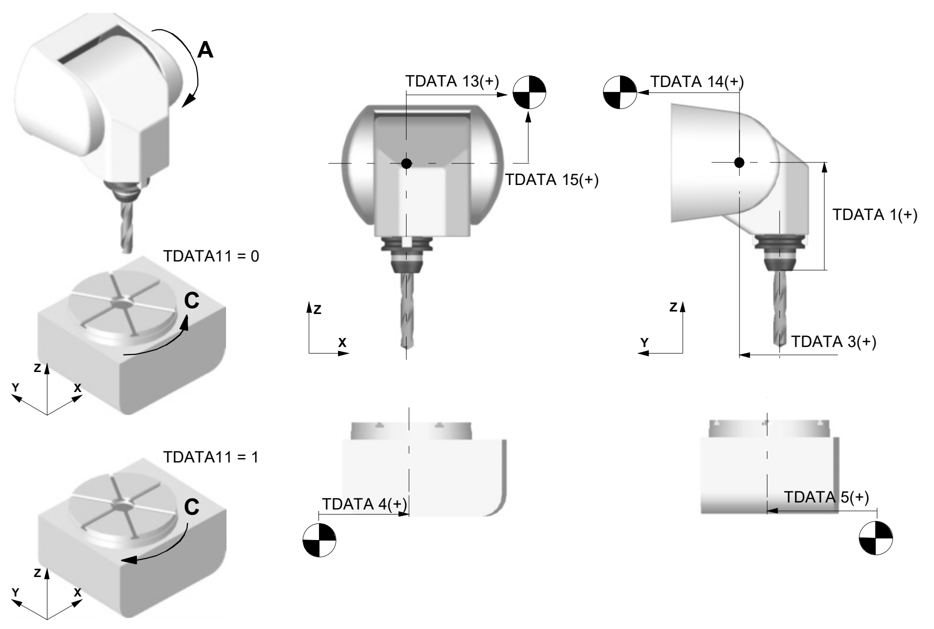
- The first two axes correspond to the work plane.
- The third axis corresponds to the tool axis.
- The fourth axis corresponds to the rotary axis of the spindle.
- The fifth axis corresponds to the rotary axis of the table.

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.

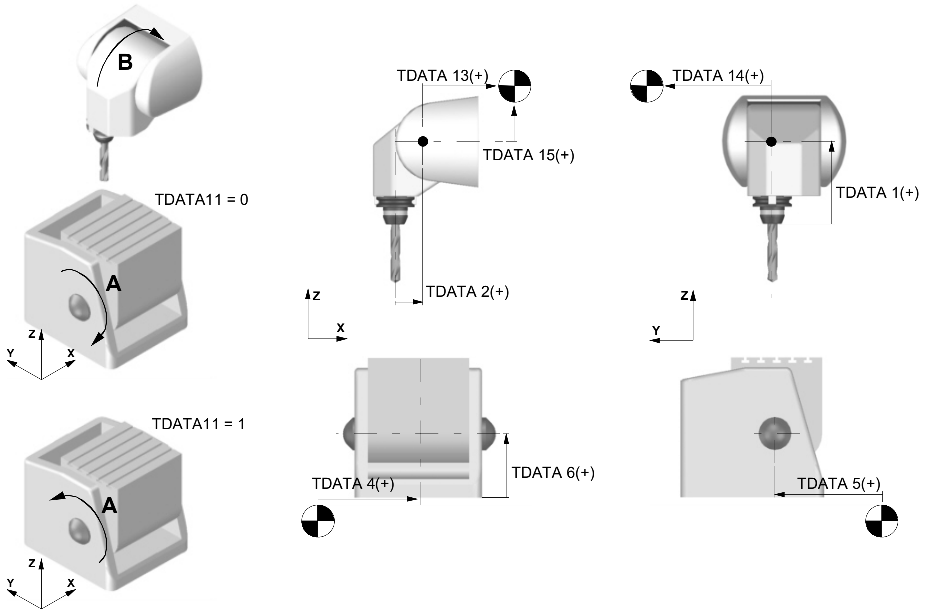
The dimensions may be defined with a positive or negative value. The (+) sign in the illustrations indicates that the direction is assumed as positive.



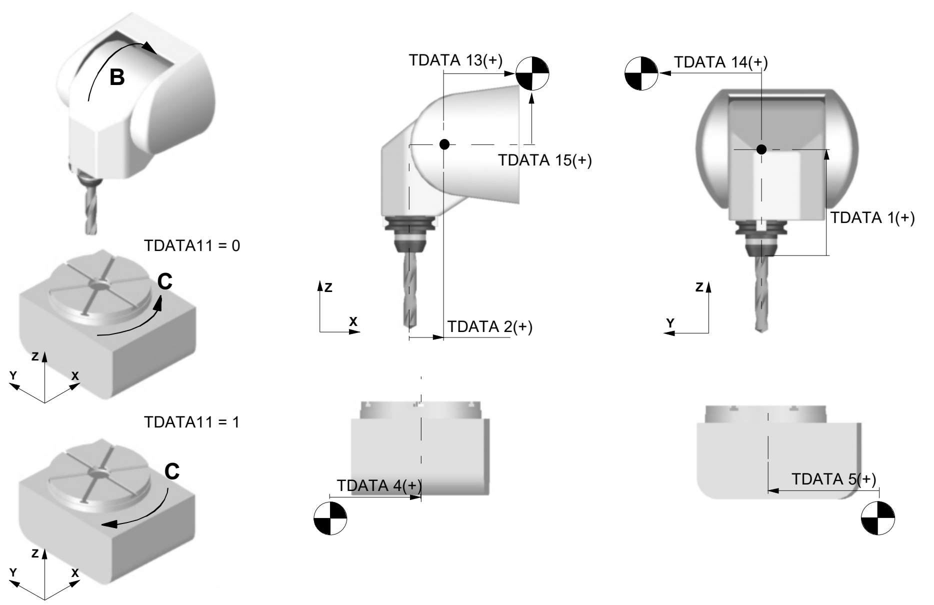
TYPE=14



TYPE=15



TYPE=16



TDATA1 (TYPE 13..16)

Possible values: Between ±999 999 999.0000
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[1]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[1]

Distance between the tip of the quill and the spindle rotary axis along the Z axis. It is not necessary to define it in all the kinematics.

TDATA2 (TYPE 13..16)

Possible values: Between ±999 999 999.0000
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[2]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[2]

Distance between the tool axis and the rotary axis of the spindle, according to the X axis. It is not necessary to define it in all the kinematics.

TDATA3 (TYPE 13..16)

Possible values: Between ±999 999 999.0000
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[3]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[3]

Distance between the tool axis and the spindle rotary axis along the Y axis. It is not necessary to define it in all the kinematics.

TDATA4 (TYPE 13..16)

Possible values: Between ±999 999 999.0000
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[4]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[4]

Distance from the rotary axis of the table to the machine reference point along the X axis. It is not necessary to define it in all the kinematics.

TDATA5 (TYPE 13..16)

Possible values: Between ±999 999 999.0000
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[5]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[5]

Distance from the rotary axis of the table to the machine reference point along the Y axis. It is not necessary to define it in all the kinematics.

TDATA6 (TYPE 13..16)

Possible values: Between ±999 999 999.0000
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[6]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[6]

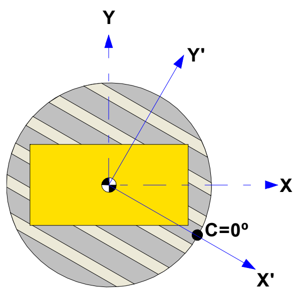
Distance from the rotary axis of the table to the machine reference point along the Z axis. It is not necessary to define it in all the kinematics.

TDATA7 (TYPE 13..16)

Possible values: 0 (machine zero) / 1 (part zero).
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[7]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[7]

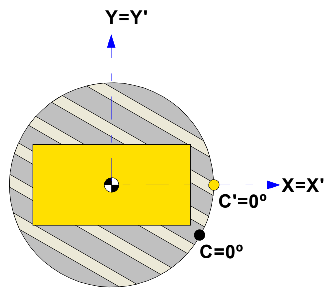
This parameter indicates whether the CNC applies the RTCP referred to machine zero or to part zero.

TDATA7=0



C=0º (machine coordinates)

TDATA7=1



C=0º (machine coordinates)
C'=0º (part coordinates)

TDATA8 (TYPE 13..16)

Possible values: Within ±99 999,9999º.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[8]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[8]

Rest position of the main rotary axis. The rest position is when the tool is perpendicular to the work plane (parallel to the longitudinal axis).

TDATA9 (TYPE 13..16)

Possible values: Within ±99 999,9999º.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[9]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[9]

Rest position of the secondary rotary axis. The rest position is when the tool is perpendicular to the work plane (parallel to the longitudinal axis).

TDATA10 (TYPE 13..16)

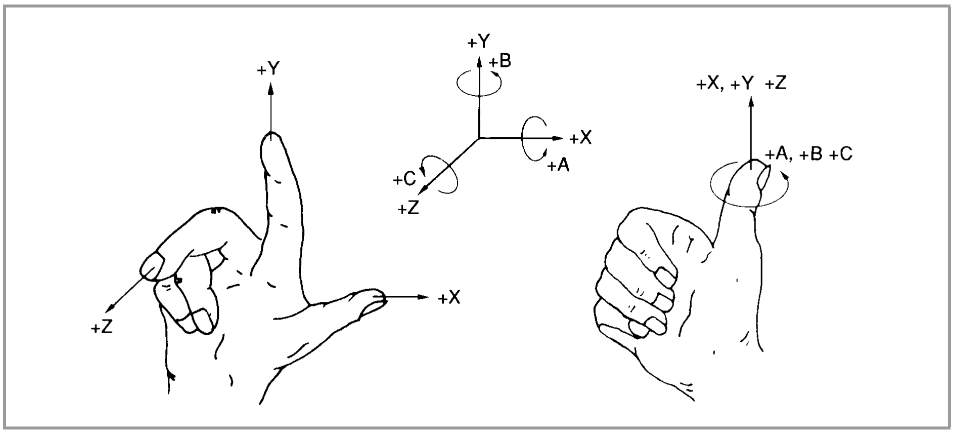
Possible values: 0 (by the standard DIN 66217) / 1 (opposite to the standard DIN 66217).
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[10]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[10]

Turning direction of the main rotary axis.

TDATA11 (TYPE 13..16)

Possible values: 0 (by the standard DIN 66217) / 1 (opposite to the standard DIN 66217).
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[11]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[11]

Turning direction of the secondary rotary axis. 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.



TDATA12 (TYPE 13..16)

Possible values: 0 to 3.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[12]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[12]

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 (TYPE 13..16)

Possible values: Between ±999 999 999.0000
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[13]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[13]

Distance defining the spindle placement, from the rotary axis, along the X axis.

TDATA14 (TYPE 13..16)

Possible values: Between ±999 999 999.0000
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[14]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[14]

Distance defining the spindle placement, from the rotary axis, along the Y axis.

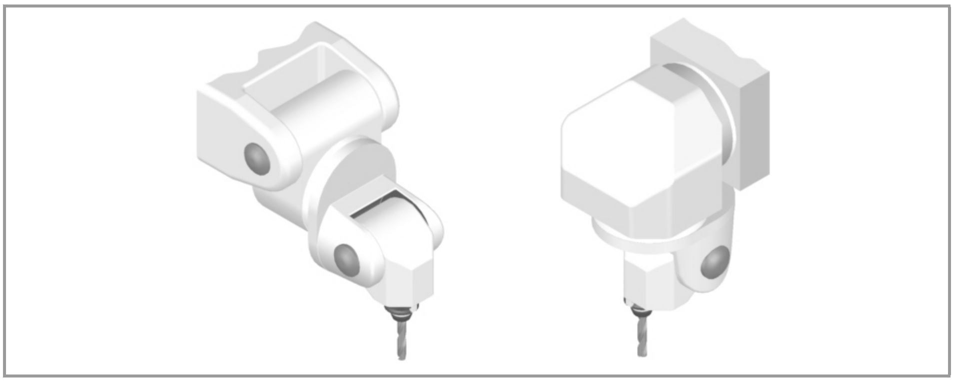
TDATA15 (TYPE 13..16)

Possible values: Between ±999 999 999.0000
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[15]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[15]

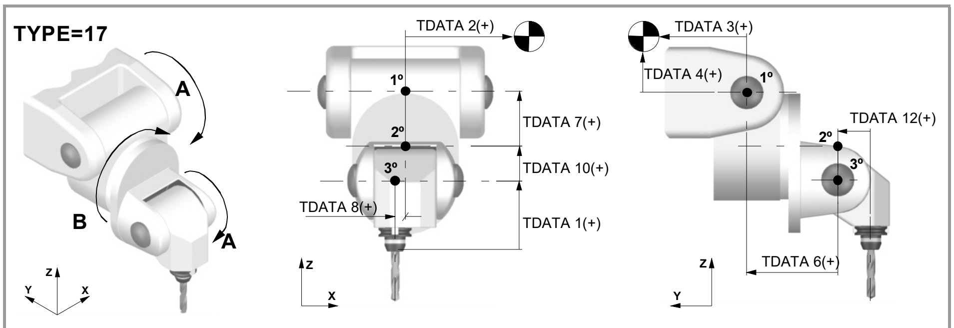
Distance defining the spindle placement, from the rotary axis, along the Z axis.

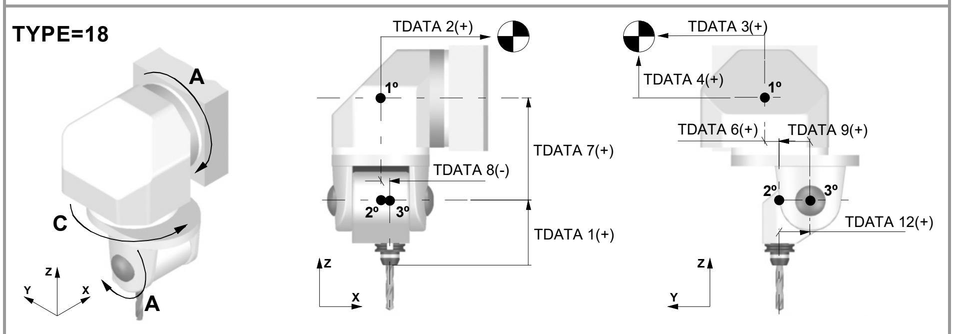
2.17.5 Definition of the spindle kinetics (types 17 through 24)

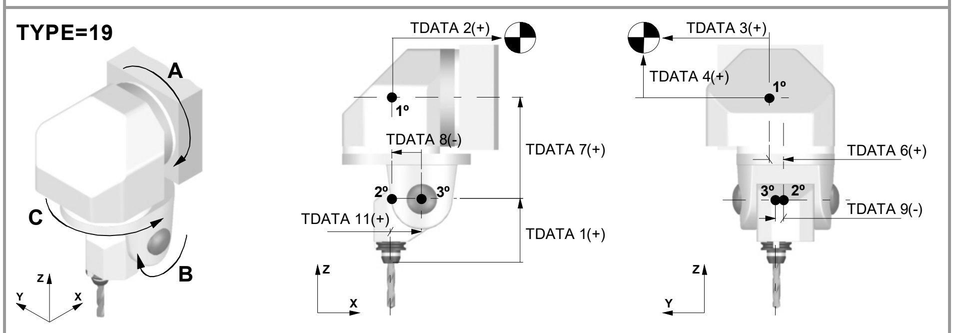
These kinematics can control orthogonal spindles with three rotary axes.

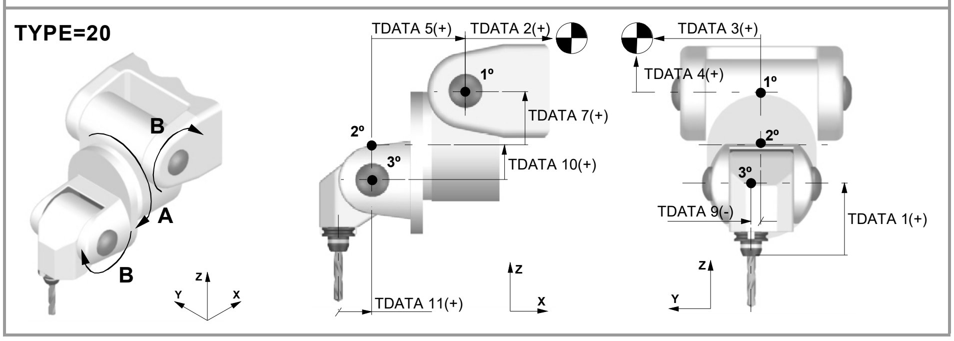


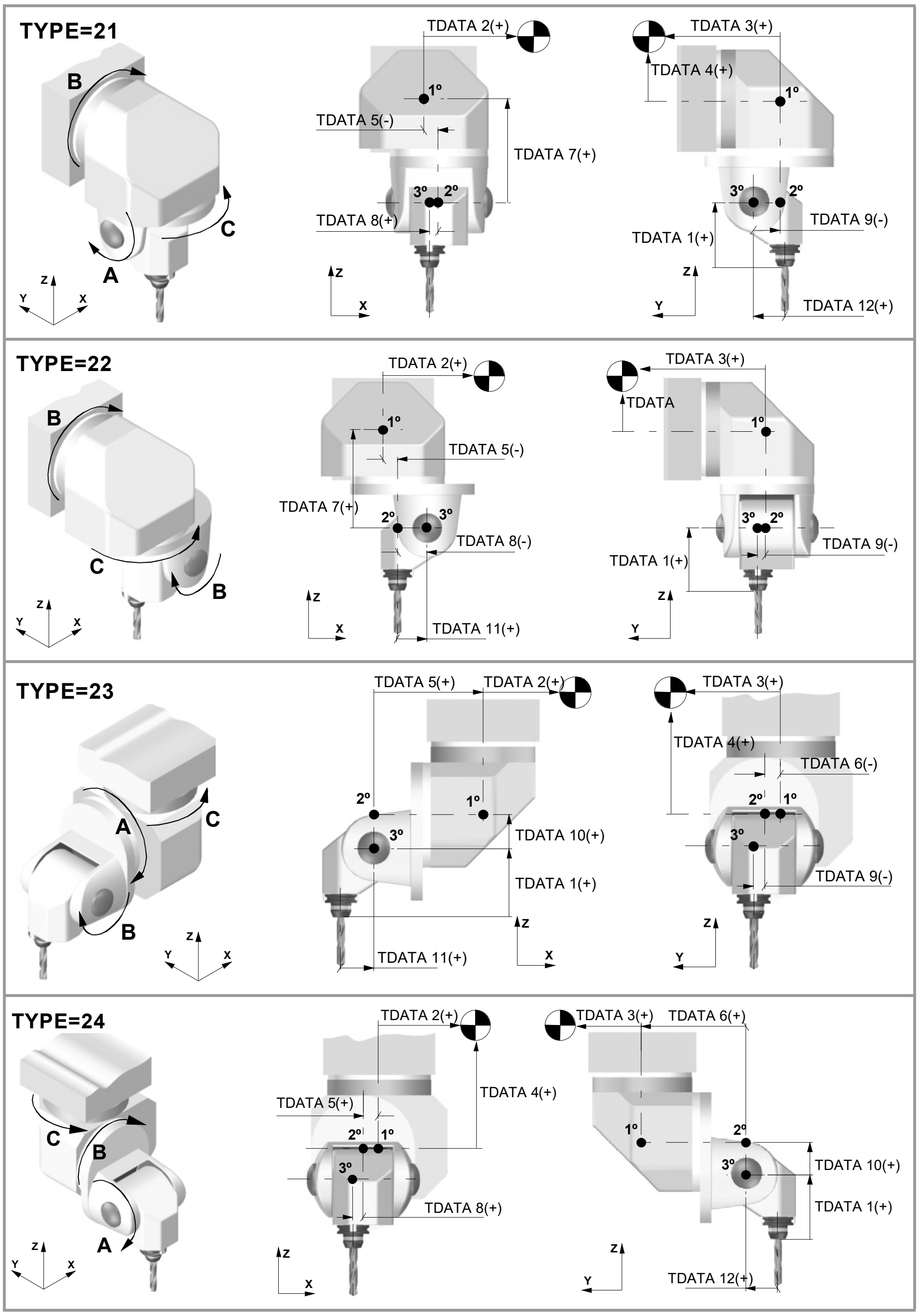
They may be defined with a positive or negative value. The (+) sign in the illustrations indicates that the direction is assumed as positive, numbers 1st, 2nd and 3rd indicate the rotation centers. The main rotary axis is the rotary axis of the spindle that when it rotates it drags the other two rotation 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.











TDATA1 (TYPE 17..24)

Possible values: Between ±999 999 999.0000
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[1]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[1]

Distance between the tip of the quill and the third rotation center along the Z axis.

TDATA2 (TYPE 17..24)

Possible values: Between ±999 999 999.0000
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[2]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[2]

Indicates the distance of the spindle from the first rotation center to the machine reference point along the X axis.

TDATA3 (TYPE 17..24)

Possible values: Between ±999 999 999.0000
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[3]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[3]

Indicates the distance of the spindle from the first rotation center to the machine reference point along the Y axis.

TDATA4 (TYPE 17..24)

Possible values: Between ±999 999 999.0000
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[4]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[4]

Indicates the distance of the spindle from the first rotation centre to the machine reference point along the Z axis.

TDATA5 (TYPE 17..24)

Possible values: Between ±999 999 999.0000
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[5]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[5]

Distance from the second rotation center to the first rotation center along the X axis.

TDATA6 (TYPE 17..24)

Possible values: Between ±999 999 999.0000
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[6]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[6]

Distance from the second rotation center to the first rotation center along the Y axis.

TDATA7 (TYPE 17..24)

Possible values: Between ±999 999 999.0000
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[7]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[7]

Distance from the second rotation center to the first rotation center along the Z axis.

TDATA8 (TYPE 17..24)

Possible values: Between ±999 999 999.0000
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[8]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[8]

Distance from the third rotation center to the second rotation center along the X axis.

TDATA9 (TYPE 17..24)

Possible values: Between ±999 999 999.0000
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[9]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[9]

Distance from the third rotation center to the second rotation center along the Y axis.

TDATA10 (TYPE 17..24)

Possible values: Between ±999 999 999.0000
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[10]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[10]

Distance from the third rotation center to the second rotation center along the Z axis.

TDATA11 (TYPE 17..24)

Possible values: Between ±999 999 999.0000
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[11]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[11]

Distance from the shaft of the tool to the third rotation center along the X axis.

TDATA12 (TYPE 17..24)

Possible values: Between ±999 999 999.0000
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[12]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[12]

Distance from the shaft of the tool to the third rotation center along the Y axis.

TDATA13 (TYPE 17..24)

Possible values: Between ±999 999 999.0000º.
By default: 0º.
Associated variable (1): (V.)MPK.TDATAkin[13]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[13]

Rest position of the main rotary axis. The rest position is when the tool is perpendicular to the work plane (parallel to the longitudinal axis).

TDATA14 (TYPE 17..24)

Possible values: Between ±999 999 999.0000º.
By default: 0º.
Associated variable (1): (V.)MPK.TDATAkin[14]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[14]

Rest position of the secondary rotary axis. The rest position is when the tool is perpendicular to the work plane (parallel to the longitudinal axis).

TDATA15 (TYPE 17..24)

Possible values: Between ±999 999 999.0000º.
By default: 0º.
Associated variable (1): (V.)MPK.TDATAkin[15]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[15]

Rest position of the third rotary axis. The rest position is when the tool is perpendicular to the work plane (parallel to the longitudinal axis).

TDATA16 (TYPE 17..24)

Possible values: 0 (by the standard DIN 66217) / 1 (opposite to the standard DIN 66217).
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[16]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[16]

Turning direction of the main rotary axis.

TDATA17 (TYPE 17..24)

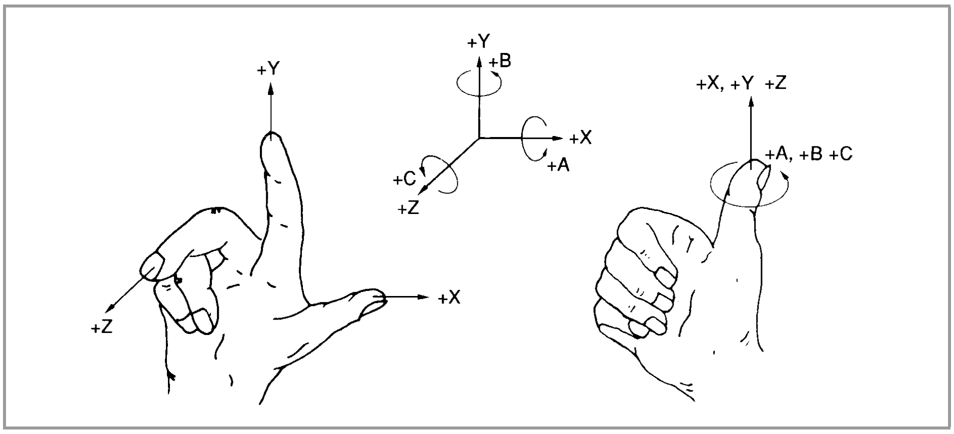
Possible values: 0 (by the standard DIN 66217) / 1 (opposite to the standard DIN 66217).
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[17]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[17]

Turning direction of the secondary rotary axis.

TDATA18 (TYPE 17..24)

Possible values: 0 (by the standard DIN 66217) / 1 (opposite to the standard DIN 66217).
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[18]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[18]

Turning direction of the third rotary axis. 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 (TYPE 17..24)

Possible values: 0 to 7.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[19]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[19]

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.

2.17.6 Definition of the C axis kinematics (Types 41 through 42).

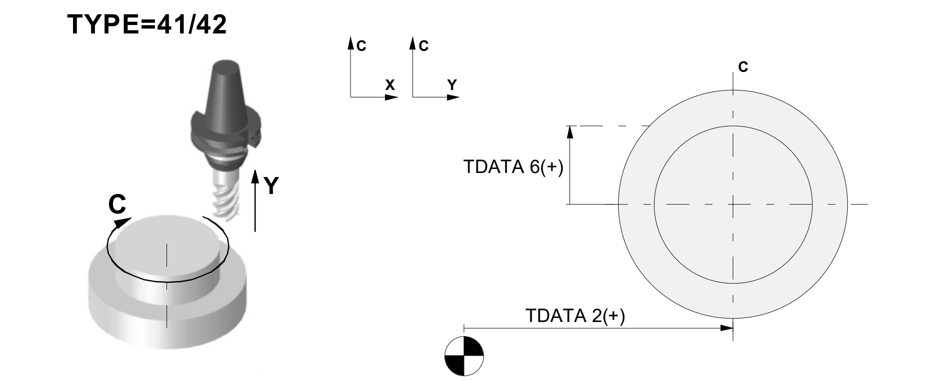
In this type of kinematics, one must define the physical location of the rotary axis with respect to the linear axes. If these kinematics are defined, it assumes that the rotary axis coincides with the linear axis (e.g. the spindle of a lathe).

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 (TYPE 41/42)

Possible values: Between ±999 999 999.0000
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[2]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[2]

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 (TYPE 41/42)

Possible values: Between ±999 999 999.0000º.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[4]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[4]

Angular offset of the rotary axis.

TDATA5 (TYPE 41/42)

Possible values: 0 / 1.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[5]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[5]

Position of the rotation axis.

Value.	Meaning.
0	The rotation axis is in part-zero.
1	The position of the rotation axis is indicated by TDATA2.

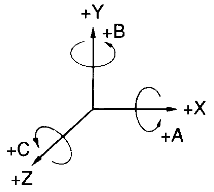
TDATA6 (TYPE 41/42)

Possible values: Between ±999 999 999.0000
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[6]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[6]

Tool misalignment with the C axis. 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 (TYPE 41/42)

Possible values: 0 (by the standard DIN 66217) / 1 (opposite to the standard DIN 66217).
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[10]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[10]



Turning direction of the rotary axis. The diagram shows the positive turning direction of the rotary axes according to DIN 66217.

2.17.7 Definition of the C axis kinematics (type 43).

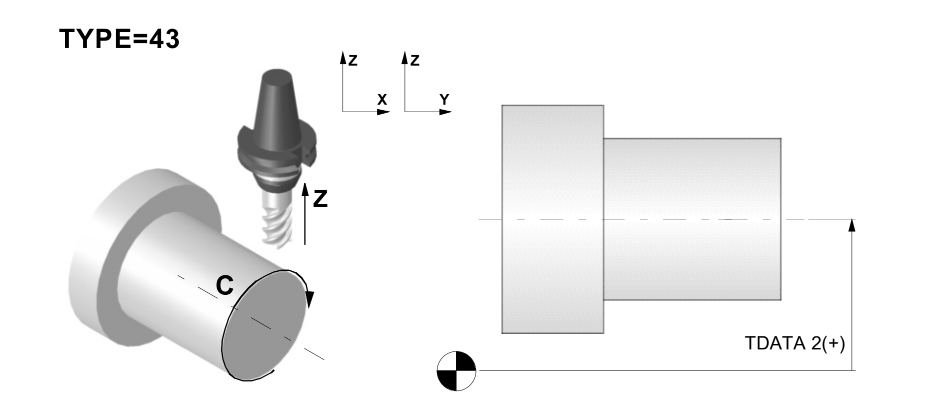
In this type of kinematics, one must define the physical location of the rotary axis with respect to the linear axes. If these kinematics are defined, it assumes that the rotary axis coincides with the linear axis (e.g. the spindle of a lathe).

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 (TYPE 43)

Possible values: Between ±999 999 999.0000
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[2]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[2]

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.

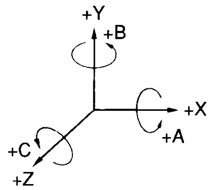
TDATA4 (TYPE 43)

Possible values: Between ±999 999 999.0000º.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[4]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[4]

Angular offset of the rotary axis.

TDATA10 (TYPE 43)

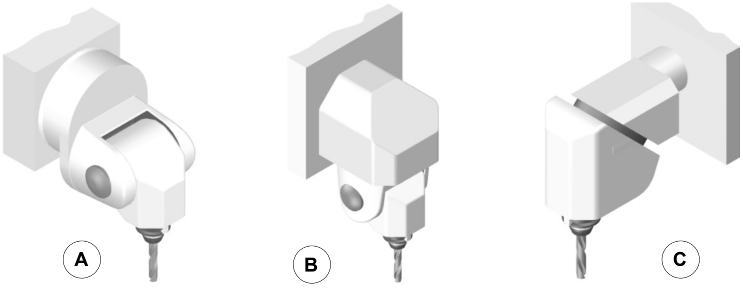
Possible values: 0 (by the standard DIN 66217) / 1 (opposite to the standard DIN 66217).
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[10]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[10]



Turning direction of the rotary axis. The diagram shows the positive turning direction of the rotary axes according to DIN 66217.

2.17.8 Vectorial definition of spindle kinematics (type 50).

This kinematics may be used to control any type of spindle that has a maximum of two rotary axes.

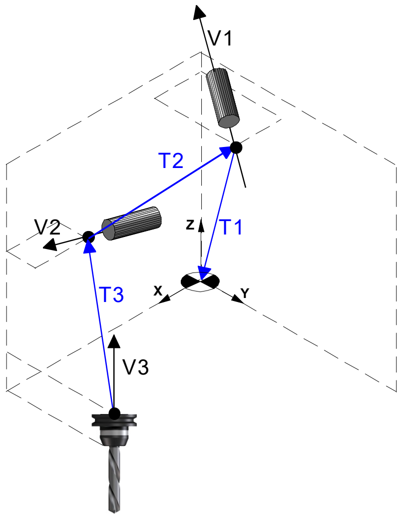


A Spherical spindle. B Orthogonal spindle. C Angular spindle.

Vectorial definition of the kinematics.

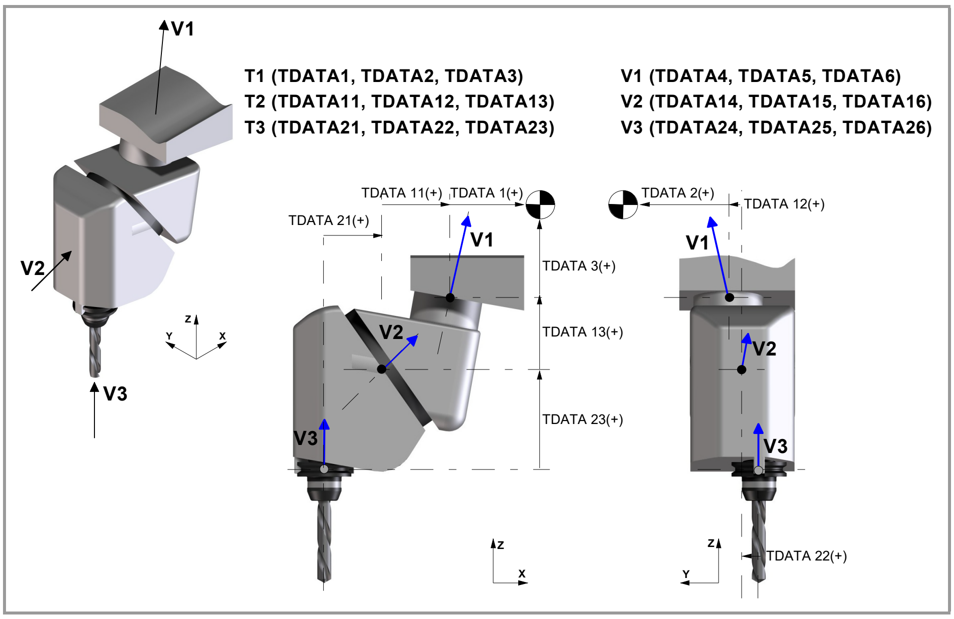
This type of kinematics may be used to define the rotary axes that make it up using vectors. Each rotary axis of the kinematics is defined with a translation vector (position of the axis) and a direction vector (direction of the axis). The position of the tool holder is defined the same way, with a translation vector and a direction vector.

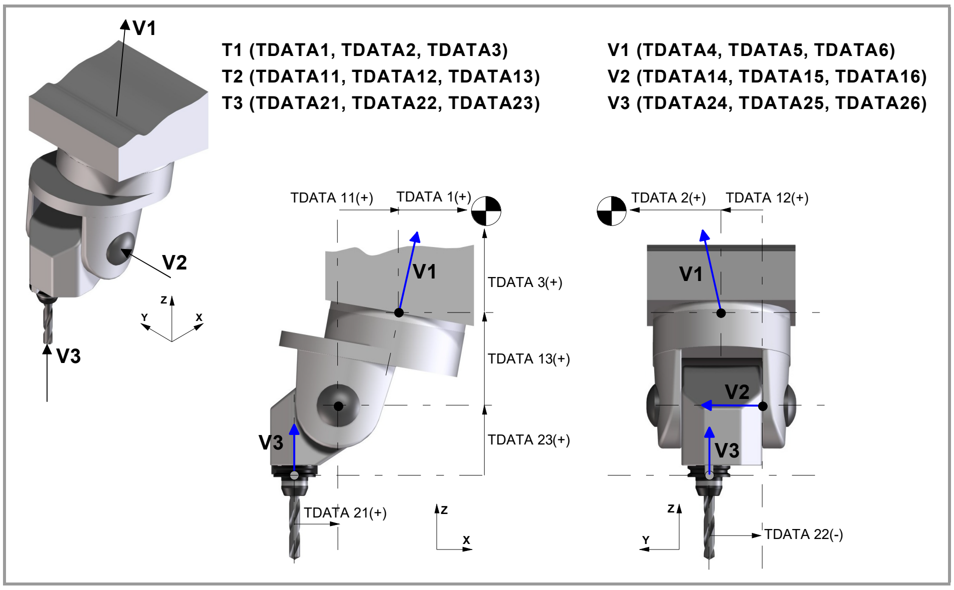
The translation vector may have its origin anywhere on the rotation axis of the rotary axis (except the translation value of the tool holder, whose origin is at its base). This reference point will be the same for all the translation vector associated with that axis.



T1: Translation vector of the first rotary axis.
T2: Translation vector of the second rotary axis.
T3: Translation vector of the tool holder.

V1: Direction vector of the first rotary axis.
V2: Direction vector of the second rotary axis.
V3: Direction vector of the tool holder.





TDATA1 (TYPE 50)

Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[1]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[1]

Translation vector of the main rotary axis. Distance from the reference point of the main rotary axis to the machine reference point (X axis).

TDATA2 (TYPE 50)

Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[2]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[2]

Translation vector of the main rotary axis. Distance from the reference point of the main rotary axis to the machine reference point (Y axis).

TDATA3 (TYPE 50)

Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[3]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[3]

Translation vector of the main rotary axis. Distance from the reference point of the main rotary axis to the machine reference point (Z axis).

TDATA4 (TYPE 50)

Possible values: From 0 to 99 999.9999.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[4]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[4]

Direction vector of the main rotary axis. Component of the direction vector of the main rotary axis (X axis).

TDATA5 (TYPE 50)

Possible values: From 0 to 99 999.9999.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[5]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[5]

Direction vector of the main rotary axis. Component of the direction vector of the main rotary axis (Y axis).

TDATA6 (TYPE 50)

Possible values: From 0 to 99 999.9999.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[6]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[6]

Direction vector of the main rotary axis. Component of the direction vector of the main rotary axis (Z axis).

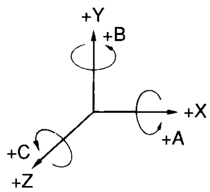
TDATA7 (TYPE 50)

Possible values: Within ±99 999,9999º.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[7]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[7]

Rest position of the main rotary axis. The rest position is when the tool is perpendicular to the work plane (parallel to the longitudinal axis).

TDATA8 (TYPE 50)

Possible values: 0 (by the standard DIN 66217) / 1 (opposite to the standard DIN 66217).
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[8]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[8]



Turning direction of the main rotary axis. The diagram shows the positive turning direction of the rotary axes according to DIN 66217.

TDATA9 (TYPE 50)

Possible values: 0 (servo-controlled) / 1 (manual).
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[9]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[9]

This parameter indicates whether the main rotary axis is manual or servo-controlled.

TDATA11 (TYPE 50)

Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[11]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[11]

Translation vector of the secondary rotary axis. Distance from the reference point of the secondary rotary axis to the reference point of the main rotary axis (X axis).

TDATA12 (TYPE 50)

Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[12]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[12]

Translation vector of the secondary rotary axis. Distance from the reference point of the secondary rotary axis to the reference point of the main rotary axis (Y axis).

TDATA13 (TYPE 50)

Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[13]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[13]

Translation vector of the secondary rotary axis. Distance from the reference point of the secondary rotary axis to the reference point of the main rotary axis (Z axis).

TDATA14 (TYPE 50)

Possible values: From 0 to 99 999.9999.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[14]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[14]

Direction vector of the secondary rotary axis. Component of the direction vector of the secondary rotary axis (X axis).

TDATA15 (TYPE 50)

Possible values: From 0 to 99 999.9999.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[15]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[15]

Direction vector of the secondary rotary axis. Component of the direction vector of the secondary rotary axis (Y axis).

TDATA16 (TYPE 50)

Possible values: From 0 to 99 999.9999.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[16]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[16]

Direction vector of the secondary rotary axis. Component of the direction vector of the secondary rotary axis (Z axis).

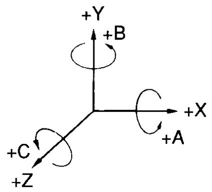
TDATA17 (TYPE 50)

Possible values: Within ±99 999,9999º.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[17]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[17]

Rest position of the secondary rotary axis. The rest position is the one where the direction vector of the rotary axes of the kinematics have been defined.

TDATA18 (TYPE 50)

Possible values: 0 (by the standard DIN 66217) / 1 (opposite to the standard DIN 66217).
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[18]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[18]



Turning direction of the secondary rotary axis. The diagram shows the positive turning direction of the rotary axes according to DIN 66217.

TDATA19 (TYPE 50)

Possible values: 0 (servo-controlled) / 1 (manual).
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[19]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[19]

This parameter indicates whether the secondary rotary axis is manual or servo-controlled.

TDATA21 (TYPE 50)

Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[21]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[21]

Translation vector of the tool holder. Distance from the base of the tool holder to the reference point of the secondary rotary axis (X axis).

TDATA22 (TYPE 50)

Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[22]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[22]

Translation vector of the tool holder. Distance from the base of the tool holder to the reference point of the secondary rotary axis (Y axis).

TDATA23 (TYPE 50)

Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[23]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[23]

Translation vector of the tool holder. Distance from the base of the tool holder to the reference point of the secondary rotary axis (Z axis).

TDATA24 (TYPE 50)

Possible values: From 0 to 99 999.9999.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[24]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[24]

Direction vector of the tool holder. Component of the direction vector of the tool holder (X axis).

TDATA25 (TYPE 50)

Possible values: From 0 to 99 999.9999.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[25]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[25]

Direction vector of the tool holder. Component of the direction vector of the tool holder (Y axis).

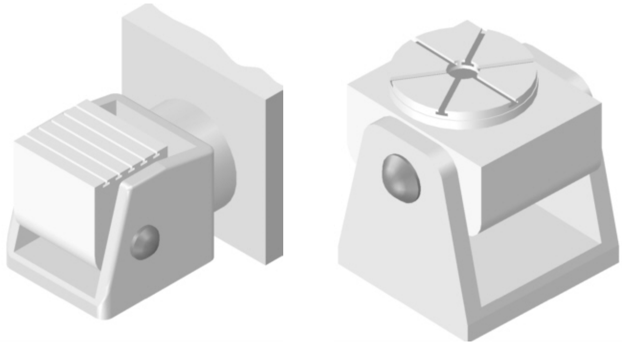
TDATA26 (TYPE 50)

Possible values: From 0 to 99 999.9999.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[26]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[26]

Direction vector of the tool holder. Component of the direction vector of the tool holder (Z axis).

2.17.9 Vectorial definition of table kinematics (type 51).

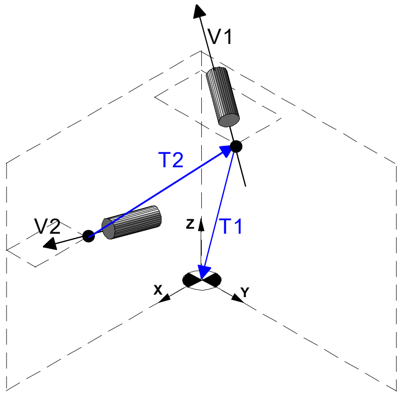
This kinematics may be used to control any type of table that has a maximum of two rotary axes.



Vectorial definition of the kinematics.

This type of kinematics may be used to define the rotary axes that make it up using vectors. Each rotary axis of the kinematics is defined with a translation vector (position of the axis) and a direction vector (direction of the axis).

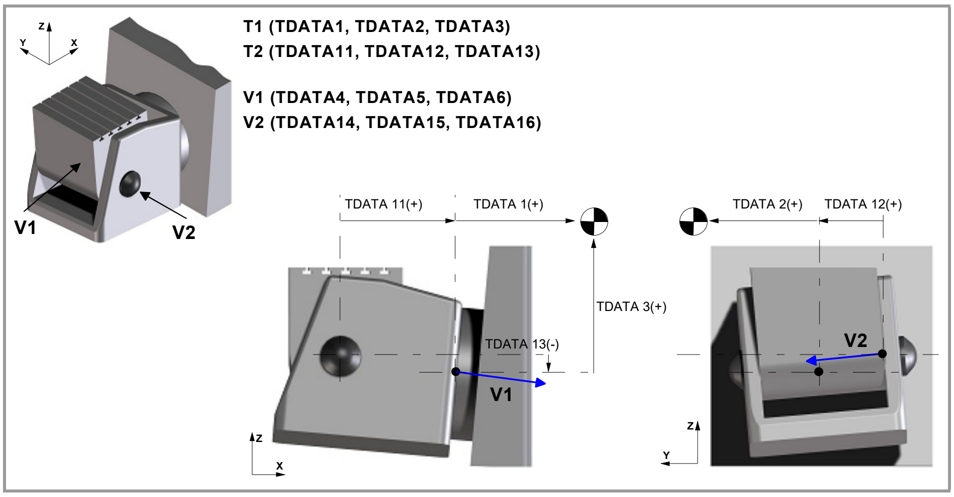
The translation vector may have its origin anywhere on the rotation axis of the rotary axis. This reference point will be the same for all the translation vector associated with that axis.

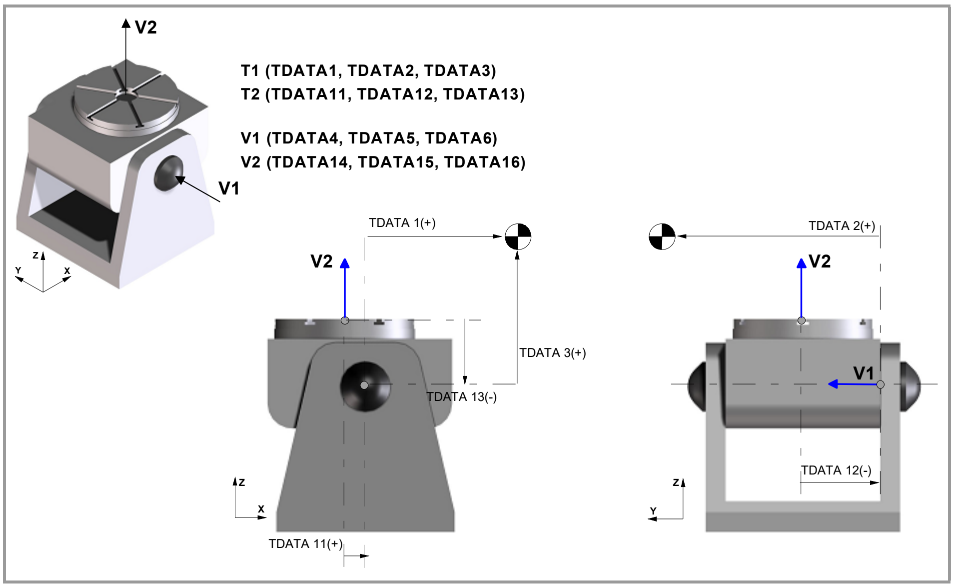


From now on, all the explanations assume that the main axes are X Y Z.

T1: Translation vector of the first rotary axis.
T2: Translation vector of the second rotary axis.

V1: Direction vector of the first rotary axis.
V2: Direction vector of the second rotary axis.





TDATA1 (TYPE 51)

Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[1]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[1]

Translation vector of the main rotary axis. Distance from the reference point of the main rotary axis to the machine reference point (X axis).

TDATA2 (TYPE 51)

Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[2]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[2]

Translation vector of the main rotary axis. Distance from the reference point of the main rotary axis to the machine reference point (Y axis).

TDATA3 (TYPE 51)

Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[3]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[3]

Translation vector of the main rotary axis. Distance from the reference point of the main rotary axis to the machine reference point (Z axis).

TDATA4 (TYPE 51)

Possible values: From 0 to 99 999.9999.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[4]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[4]

Direction vector of the main rotary axis. Component of the direction vector of the main rotary axis (X axis).

TDATA5 (TYPE 51)

Possible values: From 0 to 99 999.9999.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[5]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[5]

Direction vector of the main rotary axis. Component of the direction vector of the main rotary axis (Y axis).

TDATA6 (TYPE 51)

Possible values: From 0 to 99 999.9999.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[6]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[6]

Direction vector of the main rotary axis. Component of the direction vector of the main rotary axis (Z axis).

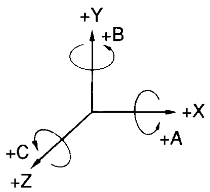
TDATA7 (TYPE 51)

Possible values: Within ±99 999,9999º.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[7]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[7]

Rest position of the main rotary axis. The rest position is when the tool is perpendicular to the work plane (parallel to the longitudinal axis).

TDATA8 (TYPE 51)

Possible values: 0 (by the standard DIN 66217) / 1 (opposite to the standard DIN 66217).
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[8]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[8]



Turning direction of the main rotary axis. The diagram shows the positive turning direction of the rotary axes according to DIN 66217.

TDATA9 (TYPE 51)

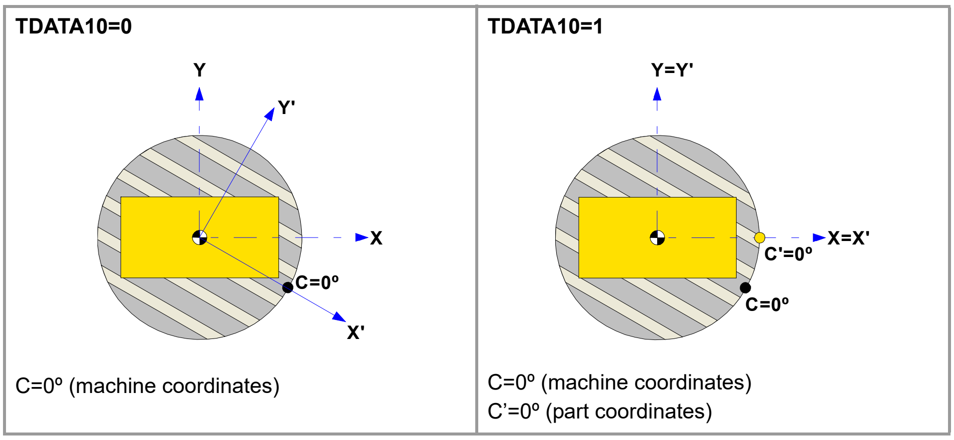
Possible values: 0 (servo-controlled) / 1 (manual).
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[9]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[9]

This parameter indicates whether the main rotary axis is manual or servo-controlled.

TDATA10 (TYPE 51)

Possible values: 0 (machine zero) / 1 (part zero).
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[10]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[10]

Origin for applying the RTCP. This parameter indicates whether the CNC applies the RTCP referred to machine zero or to part zero.



TDATA11 (TYPE 51)

Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[11]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[11]

Translation vector of the secondary rotary axis. Distance from the reference point of the secondary rotary axis to the reference point of the main rotary axis (X axis).

TDATA12 (TYPE 51)

Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[12]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[12]

Translation vector of the secondary rotary axis. Distance from the reference point of the secondary rotary axis to the reference point of the main rotary axis (Y axis).

TDATA13 (TYPE 51)

Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[13]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[13]

Translation vector of the secondary rotary axis. Distance from the reference point of the secondary rotary axis to the reference point of the main rotary axis (Z axis).

TDATA14 (TYPE 51)

Possible values: From 0 to 99 999.9999.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[14]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[14]

Direction vector of the secondary rotary axis. Component of the direction vector of the secondary rotary axis (X axis).

TDATA15 (TYPE 51)

Possible values: From 0 to 99 999.9999.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[15]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[15]

Direction vector of the secondary rotary axis. Component of the direction vector of the secondary rotary axis (Y axis).

TDATA16 (TYPE 51)

Possible values: From 0 to 99 999.9999.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[16]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[16]

Direction vector of the secondary rotary axis. Component of the direction vector of the secondary rotary axis (Z axis).

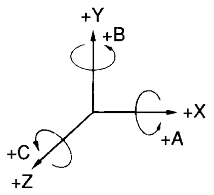
TDATA17 (TYPE 51)

Possible values: Within ±99 999,9999º.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[17]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[17]

Rest position of the secondary rotary axis. The rest position is the one where the direction vector of the rotary axes of the kinematics have been defined.

TDATA18 (TYPE 51)

Possible values: 0 (by the standard DIN 66217) / 1 (opposite to the standard DIN 66217).
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[18]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[18]



Turning direction of the secondary rotary axis. The diagram shows the positive turning direction of the rotary axes according to DIN 66217.

TDATA19 (TYPE 51)

Possible values: 0 (servo-controlled) / 1 (manual).
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[19]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[19]

This parameter indicates whether the secondary rotary axis is manual or servo-controlled.

TDATA20 (TYPE 51)

Possible values: 0 (machine zero) / 1 (part zero).
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[20]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[20]

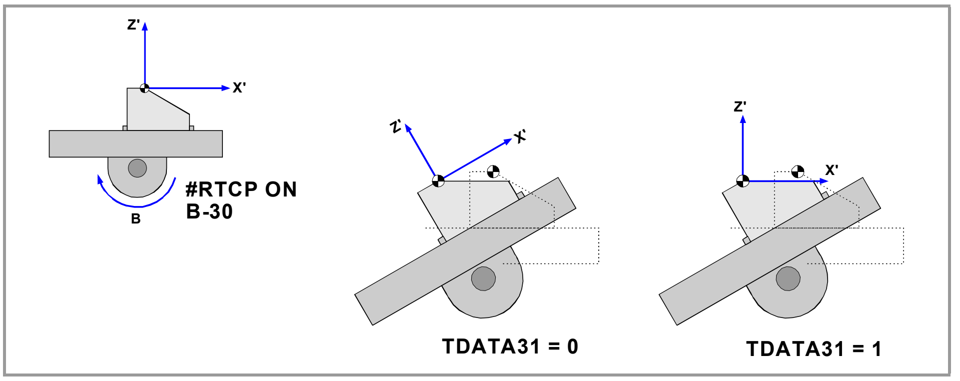
Origin for applying the RTCP. See parameter TDATA10.

TDATA31 (TYPE 51)

Possible values: 0 (yes) / 1 (no).
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[31]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[31]

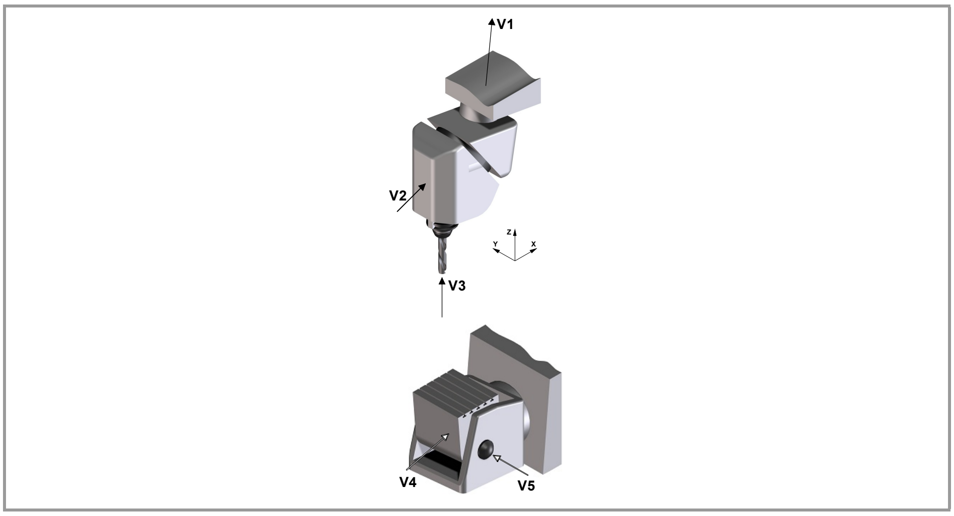
Rotate the part coordinate system when rotating the table. Having the RTCP mode active, this parameter determines whether the part coordinate system is fixed on the part and rotates with it or it stays parallel to the machine coordinate system.

Value.	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.



2.17.10 Vectorial definition of spindle-table kinematics (type 52).

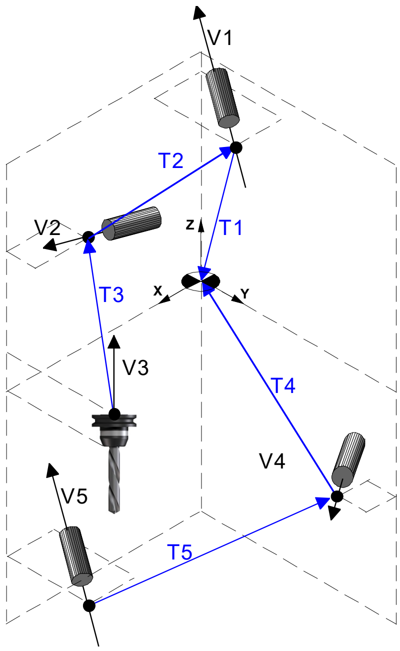
This kinematics may be used to control any type of kinematics consisting of a spindle and a table, having a maximum of two rotary axes on the spindle and two on the table.



Vectorial definition of the kinematics.

This type of kinematics may be used to define the rotary axes that make it up using vectors. Each rotary axis of the kinematics is defined with a translation vector (position of the axis) and a direction vector (direction of the axis). The position of the tool holder is defined the same way, with a translation vector and a direction vector.

The translation vector may have its origin anywhere on the rotation axis of the rotary axis (except the translation value of the tool holder, whose origin is at its base). This reference point will be the same for all the translation vector associated with that axis.



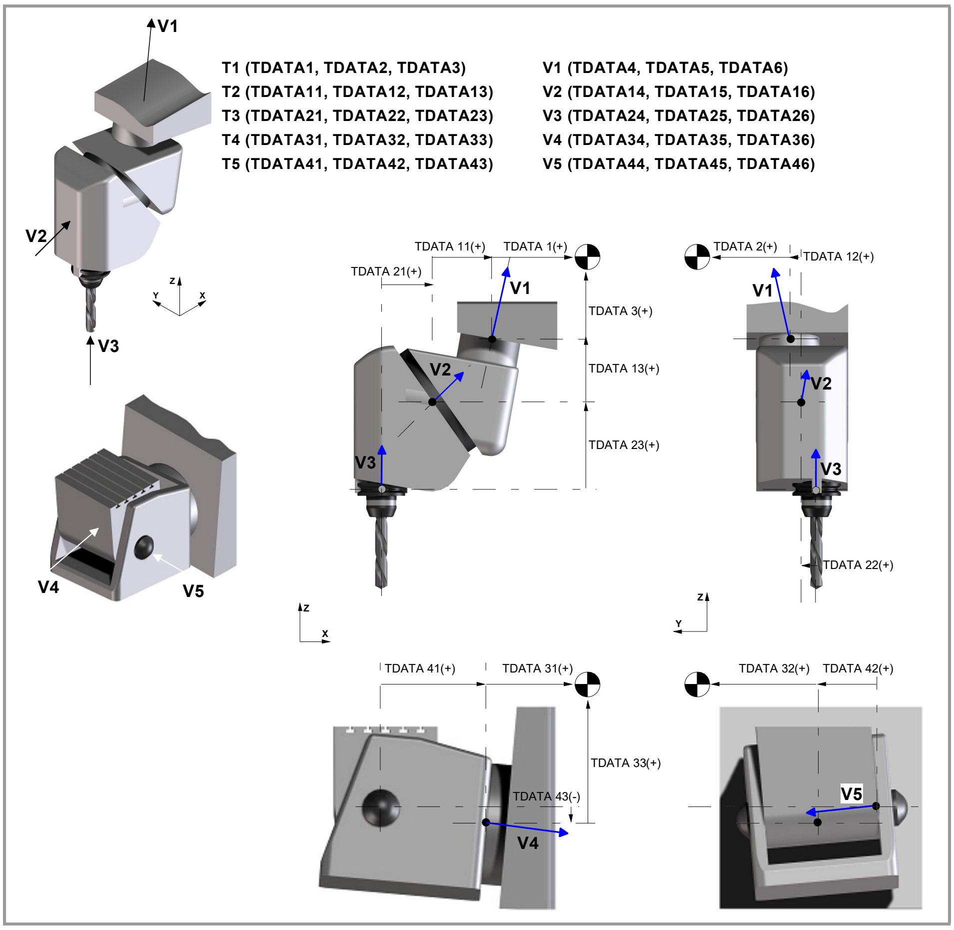
Spindle:
T1: Translation vector of the first rotary axis.
T2: Translation vector of the second rotary axis.
T3: Translation vector of the tool holder.
Table:
T4: Translation vector of the third rotary axis.
T5: Translation vector of the fourth rotary axis.

Spindle:
V1: Direction vector of the first rotary axis.
V2: Direction vector of the second rotary axis.
V3: Direction vector of the tool holder.
Table:
V4: Direction vector of the third rotary axis.
V5: Direction vector of the fourth rotary axis.

Definition of the axes that do not exist in the kinematics.

When one of the axes of the kinematics is not physically on the machine, its parameters are set as follows.

- Translation vector of the axis: 0,0,0
- Direction vector of the axis: 0,0,1
- Rest position: 0º
- Manual or servo-controlled rotary axis: 1 (manual)
- The variable of the corresponding component: V.G.POSROTn = 0



TDATA1 (TYPE 52)

Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[1]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[1]

Translation vector of the main rotary axis of the spindle. Distance from the reference point of the main rotary axis to the machine reference point (X axis). If the axis does not exist in the kinematics, it must be defined as TDATA1=0.

TDATA2 (TYPE 52)

Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[2]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[2]

Translation vector of the main rotary axis of the spindle. Distance from the reference point of the main rotary axis to the machine reference point (Y axis). If the axis does not exist in the kinematics, it must be defined as TDATA2=0.

TDATA3 (TYPE 52)

Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[3]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[3]

Translation vector of the main rotary axis of the spindle. Distance from the reference point of the main rotary axis to the machine reference point (Z axis). If the axis does not exist in the kinematics, it must be defined as TDATA3=0.

TDATA4 (TYPE 52)

Possible values: From 0 to 99 999.9999.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[4]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[4]

Direction vector of the main rotary axis of the spindle. Component of the direction vector of the main rotary axis (X axis). If the axis does not exist in the kinematics, it must be defined as TDATA4=0.

TDATA5 (TYPE 52)

Possible values: From 0 to 99 999.9999.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[5]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[5]

Direction vector of the main rotary axis of the spindle. Component of the direction vector of the main rotary axis (Y axis). If the axis does not exist in the kinematics, it must be defined as TDATA5=0.

TDATA6 (TYPE 52)

Associated variable (1): (V.)MPK.TDATAkin[6]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[6]
Possible values: From 0 to 99 999.9999.
Default value: 0.

Direction vector of the main rotary axis of the spindle. Component of the direction vector of the main rotary axis (Z axis). If the axis does not exist in the kinematics, it must be defined as TDATA6=0.

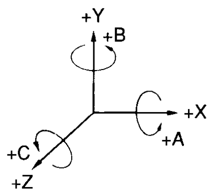
TDATA7 (TYPE 52)

Possible values: Within ±99 999,9999º.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[7]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[7]

Rest position of the main rotary axis of the spindle. The rest position is when the tool is perpendicular to the work plane (parallel to the longitudinal axis). If the axis does not exist in the kinematics, it must be defined as TDATA7=0.

TDATA8 (TYPE 52)

Possible values: 0 (by the standard DIN 66217) / 1 (opposite to the standard DIN 66217).
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[8]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[8]



Turning direction of the main rotary axis of the spindle. The diagram shows the positive turning direction of the rotary axes according to DIN 66217.

TDATA9 (TYPE 52)

Possible values: 0 (servo-controlled) / 1 (manual).
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[9]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[9]

This parameter indicates whether the main rotary axis is manual or servo-controlled. If the axis does not exist in the kinematics, it must be defined as TDATA9=1.

TDATA11 (TYPE 52)

Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[11]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[11]

Translation vector of the secondary rotary axis of the spindle. Distance from the reference point of the secondary rotary axis to the reference point of the main rotary axis (X axis). If the axis does not exist in the kinematics, it must be defined as TDATA11=0.

TDATA12 (TYPE 52)

Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[12]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[12]

Translation vector of the secondary rotary axis of the spindle. Distance from the reference point of the secondary rotary axis to the reference point of the main rotary axis (Y axis). If the axis does not exist in the kinematics, it must be defined as TDATA12=0.

TDATA13 (TYPE 52)

Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[13]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[13]

Translation vector of the secondary rotary axis of the spindle. Distance from the reference point of the secondary rotary axis to the reference point of the main rotary axis (Z axis). If the axis does not exist in the kinematics, it must be defined as TDATA13=0.

TDATA14 (TYPE 52)

Possible values: From 0 to 99 999.9999.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[14]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[14]

Direction vector of the secondary rotary axis of the spindle. Component of the direction vector of the secondary rotary axis (X axis). If the axis does not exist in the kinematics, it must be defined as TDATA14=0.

TDATA15 (TYPE 52)

Possible values: From 0 to 99 999.9999.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[15]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[15]

Direction vector of the secondary rotary axis of the spindle. Component of the direction vector of the secondary rotary axis (Y axis). If the axis does not exist in the kinematics, it must be defined as TDATA15=0.

TDATA16 (TYPE 52)

Possible values: From 0 to 99 999.9999.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[16]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[16]

Direction vector of the secondary rotary axis of the spindle. Component of the direction vector of the secondary rotary axis (Z axis). If the axis does not exist in the kinematics, it must be defined as TDATA16=0.

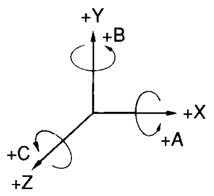
TDATA17 (TYPE 52)

Possible values: Within ±99 999,9999º.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[17]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[17]

Rest position of the secondary rotary axis of the spindle. The rest position is when the tool is perpendicular to the work plane (parallel to the longitudinal axis). If the axis does not exist in the kinematics, it must be defined as TDATA17=0.

TDATA18 (TYPE 52)

Possible values: 0 (by the standard DIN 66217) / 1 (opposite to the standard DIN 66217).
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[18]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[18]



Turning direction of the secondary rotary axis of the spindle. The diagram shows the positive turning direction of the rotary axes according to DIN 66217.

TDATA19 (TYPE 52)

Possible values: 0 (servo-controlled) / 1 (manual).
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[19]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[19]

This parameter indicates whether the secondary rotary axis is manual or servo-controlled. If the axis does not exist in the kinematics, it must be defined as TDATA19=1.

TDATA21 (TYPE 52)

Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[21]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[21]

Translation vector of the tool holder. Distance from the base of the tool holder to the reference point of the secondary rotary axis (X axis).

TDATA22 (TYPE 52)

Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[22]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[22]

Translation vector of the tool holder. Distance from the base of the tool holder to the reference point of the secondary rotary axis (Y axis).

TDATA23 (TYPE 52)

Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[23]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[23]

Translation vector of the tool holder. Distance from the base of the tool holder to the reference point of the secondary rotary axis (Z axis).

TDATA24 (TYPE 52)

Possible values: From 0 to 99 999.9999.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[24]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[24]

Direction vector of the tool holder. Component of the direction vector of the tool holder (X axis).

TDATA25 (TYPE 52)

Possible values: From 0 to 99 999.9999.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[25]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[25]

Direction vector of the tool holder. Component of the direction vector of the tool holder (Y axis).

TDATA26 (TYPE 52)

Possible values: From 0 to 99 999.9999.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[26]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[26]

Direction vector of the tool holder. Component of the direction vector of the tool holder (Z axis).

TDATA31 (TYPE 52)

Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[31]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[31]

Translation vector of the main rotary axis of the table. Distance from the reference point of the main rotary axis to the machine reference point (X axis). If the axis does not exist in the kinematics, it must be defined as TDATA31=0.

TDATA32 (TYPE 52)

Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[32]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[32]

Translation vector of the main rotary axis of the table. Distance from the reference point of the main rotary axis to the machine reference point (Y axis). If the axis does not exist in the kinematics, it must be defined as TDATA32=0.

TDATA33 (TYPE 52)

Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[33]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[33]

Translation vector of the main rotary axis of the table. Distance from the reference point of the main rotary axis to the machine reference point (Z axis). If the axis does not exist in the kinematics, it must be defined as TDATA33=0.

TDATA34 (TYPE 52)

Possible values: From 0 to 99 999.9999.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[34]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[34]

Direction vector of the main rotary axis of the table. Component of the direction vector of the main rotary axis (X axis). If the axis does not exist in the kinematics, it must be defined as TDATA34=0.

TDATA35 (TYPE 52)

Possible values: From 0 to 99 999.9999.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[35]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[35]

Direction vector of the main rotary axis of the table. Component of the direction vector of the main rotary axis (Y axis). If the axis does not exist in the kinematics, it must be defined as TDATA35=0.

TDATA36 (TYPE 52)

Possible values: From 0 to 99 999.9999.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[36]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[36]

Direction vector of the main rotary axis of the table. Component of the direction vector of the main rotary axis (Z axis). If the axis does not exist in the kinematics, it must be defined as TDATA36=0.

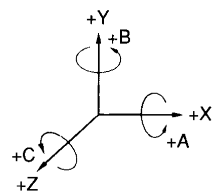
TDATA37 (TYPE 52)

Possible values: Within ±99 999,9999º.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[37]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[37]

Rest position of the main rotary axis of the table. The rest position is the one where the direction vector of the rotary axes of the kinematics have been defined. If the axis does not exist in the kinematics, it must be defined as TDATA37=0.

TDATA38 (TYPE 52)

Possible values: 0 (by the standard DIN 66217) / 1 (opposite to the standard DIN 66217).
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[38]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[38]



Turning direction of the main rotary axis of the table. The diagram shows the positive turning direction of the rotary axes according to DIN 66217.

TDATA39 (TYPE 52)

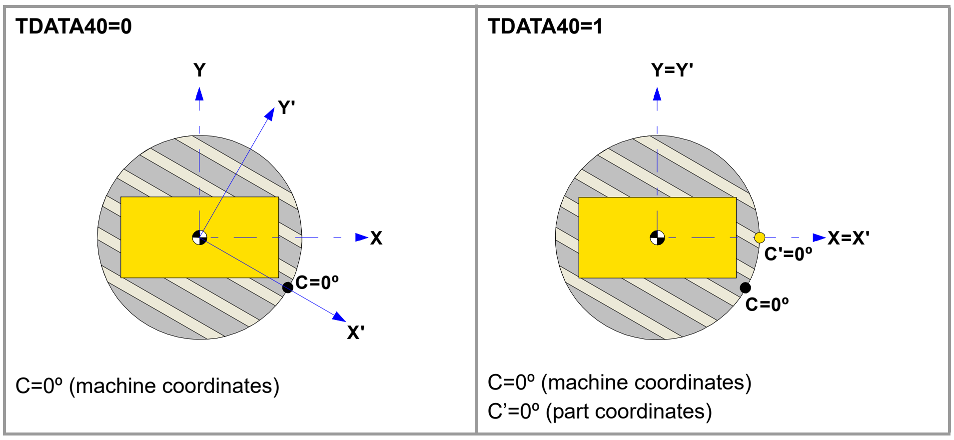
Possible values: 0 (servo-controlled) / 1 (manual).
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[39]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[39]

This parameter indicates whether the rotary axis is manual or servo-controlled. If the axis does not exist in the kinematics, it must be defined as TDATA39=1.

TDATA40 (TYPE 52)

Possible values: 0 (machine zero) / 1 (part zero).
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[40]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[40]

Origin for applying the RTCP. This parameter indicates whether the CNC applies the RTCP referred to machine zero or to part zero.



TDATA41 (TYPE 52)

Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[41]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[41]

Translation vector of the secondary rotary axis of the table. Distance from the reference point of the secondary rotary axis to the reference point of the main rotary axis (X axis). If the axis does not exist in the kinematics, it must be defined as TDATA41=0.

TDATA42 (TYPE 52)

Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[42]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[42]

Translation vector of the secondary rotary axis of the table. Distance from the reference point of the secondary rotary axis to the reference point of the main rotary axis (Y axis). If the axis does not exist in the kinematics, it must be defined as TDATA42=0.

TDATA43 (TYPE 52)

Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[43]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[43]

Translation vector of the secondary rotary axis of the table. Distance from the reference point of the secondary rotary axis to the reference point of the main rotary axis (Z axis). If the axis does not exist in the kinematics, it must be defined as TDATA43=0.

TDATA44 (TYPE 52)

Possible values: From 0 to 99 999.9999.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[44]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[44]

Direction vector of the secondary rotary axis of the table. Component of the direction vector of the secondary rotary axis (X axis). If the axis does not exist in the kinematics, it must be defined as TDATA44=0.

TDATA45 (TYPE 52)

Possible values: From 0 to 99 999.9999.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[45]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[45]

Direction vector of the secondary rotary axis of the table. Component of the direction vector of the secondary rotary axis (Y axis). If the axis does not exist in the kinematics, it must be defined as TDATA45=0.

TDATA46 (TYPE 52)

Possible values: From 0 to 99 999.9999.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[46]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[46]

Direction vector of the secondary rotary axis of the table. Component of the direction vector of the secondary rotary axis (Z axis). If the axis does not exist in the kinematics, it must be defined as TDATA46=0.

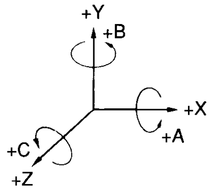
TDATA47 (TYPE 52)

Possible values: Within ±99 999,9999º.
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[47]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[47]

Rest position of the secondary rotary axis of the table. The rest position is the one where the direction vector of the rotary axes of the kinematics have been defined. If the axis does not exist in the kinematics, it must be defined as TDATA7=0.

TDATA48 (TYPE 52)

Possible values: 0 (by the standard DIN 66217) / 1 (opposite to the standard DIN 66217).
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[48]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[48]



Turning direction of the secondary rotary axis of the table. The diagram shows the positive turning direction of the rotary axes according to DIN 66217.

TDATA49 (TYPE 52)

Possible values: 0 (servo-controlled) / 1 (manual).
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[49]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[49]

This parameter indicates whether the rotary axis is manual or servo-controlled. If the axis does not exist in the kinematics, it must be defined as TDATA49=1.

TDATA50 (TYPE 52)

Possible values: 0 (machine zero) / 1 (part zero).
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[50]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[50]

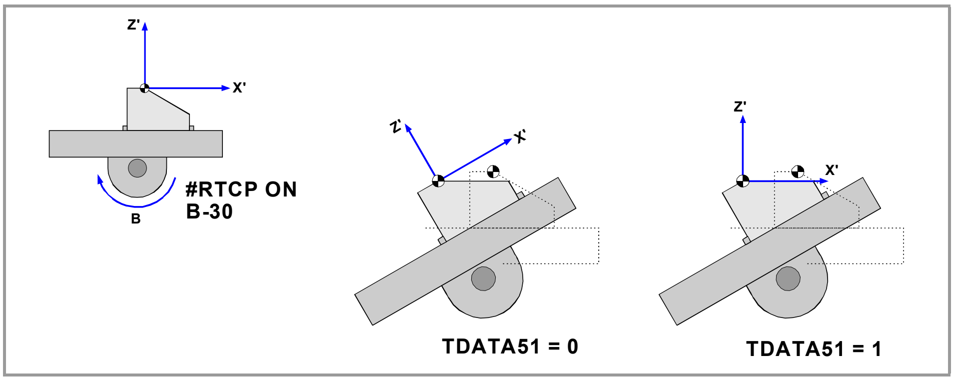
Origin for applying the RTCP. See parameter TDATA40.

TDATA51 (TYPE 52)

Possible values: 0 (yes) / 1 (no).
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[51]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[51]

Rotate the part coordinate system when rotating the table. 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.

Value.	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.



TDATA52 (TYPE 52)

Possible values: 0 (yes) / 1 (no).
Default value: 0.
Associated variable (1): (V.)MPK.TDATAkin[52]
Associated variable (2): (V.)MPK.MAXOFTDATAkin[52]

Apply full RTCP. This parameter determines whether the CNC applies the RTCP to the whole kinematics or only to the spindle side.

Value.	Meaning.
0	Full RTCP, table + spindle.
1	RTCP considering only the spindle side.

2.17.11 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

Possible values: From 0 to 8.
Default value: 0.
Associated variable: (V.)MPK.NKINAX[kin]

Number of axes of the kinematics.

PARAM_D_SIZE

Possible values: From 0 to 100.
Default value: 0.
Associated variable: (V.)MPK.PARAM_D_SIZE[kin]

Number of parameters in decimal format There are 100 kinematics predefined by Fagor. For OEM kinematics, the number of parameters may be configured.

TDATA1
..
TDATA100

Parameters for configuring the kinematics (decimal format).

PARAM_I_SIZE

Possible values: From 0 to 100.
Default value: 0.
Associated variable: (V.)MPK.PARAM_I_SIZE[kin]

Number of parameters in integer format. There are 100 kinematics predefined by Fagor. For OEM kinematics, the number of parameters may be configured.

TDATA_I1
..
TDATA_I100

Parameters for configuring the kinematics (integer format).

AUXCTE_SIZE

Possible values: From 0 to 1.000 bytes.
Default value: 0.
Associated variable: (V.)MPK.AUXCTE_SIZE[kin]

Size of the area of auxiliary variables.

KINDATA_SIZE

Possible values: From 0 to 100 000 bytes.
Default value: 0.
Associated variable: (V.)MPK.KINDATA_SIZE[kin]

Size of the area for general purpose data.

2.17.12 Tables for redefining kinematics of type 1..16 as vectorial.

For kinematics types 1 to 16, if the kinematics calibration cycle does not correctly calibrate any vector, it is recommended to redefine the kinematics as vector (type 50 to 52) and repeat the calibration.

Adapting a type 1 kinematics to a type 50 kinematics.

Vector kinematics parameter (type 50).	Equivalent value/parameter in type 1 kinematics.
TYPE	50
TDATA1	TDATA13
TDATA2	TDATA14
TDATA3	TDATA15
TDATA4	0
TDATA5	1
TDATA6	0
TDATA7	TDATA8
TDATA8	TDATA10
TDATA9	0, if TDATA12=0 or TDATA12=2 1, if TDATA12=1 or TDATA12=3.
TDATA11	TDATA2
TDATA12	0
TDATA13	TDATA4
TDATA14	1
TDATA15	0
TDATA16	0
TDATA17	TDATA9
TDATA18	TDATA11
TDATA19	0, if TDATA12=0 or TDATA12=1 1, if TDATA12=2 or TDATA12=3.
TDATA21	0
TDATA22	TDATA6
TDATA23	TDATA1
TDATA24	0
TDATA25	0
TDATA26	1

Adapting a type 2 kinematics to a type 50 kinematics.

Vector kinematics parameter (type 50).	Equivalent value/parameter in type 2 kinematics.
TYPE	50
TDATA1	TDATA13
TDATA2	TDATA14
TDATA3	TDATA15
TDATA4	0
TDATA5	0
TDATA6	1
TDATA7	TDATA8
TDATA8	TDATA10
TDATA9	0, if TDATA12=0 or TDATA12=2 1, if TDATA12=1 or TDATA12=3.
TDATA11	TDATA2
TDATA12	TDATA3
TDATA13	0
TDATA14	1
TDATA15	0
TDATA16	0
TDATA17	TDATA9
TDATA18	TDATA11
TDATA19	0, if TDATA12=0 or TDATA12=1 1, if TDATA12=2 or TDATA12=3.
TDATA21	0
TDATA22	TDATA6
TDATA23	TDATA1
TDATA24	0
TDATA25	0
TDATA26	1

Adapting a type 3 kinematics to a type 50 kinematics.

Vector kinematics parameter (type 50).	Equivalent value/parameter in type 3 kinematics.
TYPE	50
TDATA1	TDATA13
TDATA2	TDATA14
TDATA3	TDATA15
TDATA4	1
TDATA5	0
TDATA6	0
TDATA7	TDATA8
TDATA8	TDATA10
TDATA9	0, if TDATA12=0 or TDATA12=2 1, if TDATA12=1 or TDATA12=3.
TDATA11	0
TDATA12	0
TDATA13	TDATA4
TDATA14	0
TDATA15	1
TDATA16	0
TDATA17	TDATA9
TDATA18	TDATA11
TDATA19	0, if TDATA12=0 or TDATA12=1 1, if TDATA12=2 or TDATA12=3.
TDATA21	TDATA5
TDATA22	TDATA3
TDATA23	TDATA1
TDATA24	0
TDATA25	0
TDATA26	1

Adapting a type 4 kinematics to a type 50 kinematics.

Vector kinematics parameter (type 50).	Equivalent value/parameter in type 4 kinematics.
TYPE	50
TDATA1	TDATA13
TDATA2	TDATA14
TDATA3	TDATA15
TDATA4	0
TDATA5	0
TDATA6	1
TDATA7	TDATA8
TDATA8	TDATA10
TDATA9	0, if TDATA12=0 or TDATA12=2 1, if TDATA12=1 or TDATA12=3.
TDATA11	TDATA2
TDATA12	TDATA3
TDATA13	0
TDATA14	0
TDATA15	1
TDATA16	0
TDATA17	TDATA9
TDATA18	TDATA11
TDATA19	0, if TDATA12=0 or TDATA12=1 1, if TDATA12=2 or TDATA12=3.
TDATA21	TDATA5
TDATA22	0
TDATA23	TDATA1
TDATA24	0
TDATA25	0
TDATA26	1

Adapting a type 5 kinematics to a type 50 kinematics.

Vector kinematics parameter (type 50).	Equivalent value/parameter in type 5 kinematics.
TYPE	50
TDATA1	TDATA13
TDATA2	TDATA14
TDATA3	TDATA15
TDATA4	1
TDATA5	0
TDATA6	0
TDATA7	TDATA8
TDATA8	TDATA10
TDATA9	0, if TDATA12=0 or TDATA12=2 1, if TDATA12=1 or TDATA12=3.
TDATA11	TDATA5*COS(TDATA7)
TDATA12	TDATA3
TDATA13	TDATA5*SIN(TDATA7)
TDATA14	COS(TDATA7)
TDATA15	0
TDATA16	SIN(TDATA7)
TDATA17	TDATA9
TDATA18	TDATA11
TDATA19	0, if TDATA12=0 or TDATA12=1 1, if TDATA12=2 or TDATA12=3.
TDATA21	0
TDATA22	TDATA6
TDATA23	TDATA1
TDATA24	0
TDATA25	0
TDATA26	1

Adapting a type 6 kinematics to a type 50 kinematics.

Vector kinematics parameter (type 50).	Equivalent value/parameter in type 6 kinematics.
TYPE	50
TDATA1	TDATA13
TDATA2	TDATA14
TDATA3	TDATA15
TDATA4	0
TDATA5	1
TDATA6	0
TDATA7	TDATA8
TDATA8	TDATA10
TDATA9	0, if TDATA12=0 or TDATA12=2 1, if TDATA12=1 or TDATA12=3.
TDATA11	TDATA2
TDATA12	TDATA6*COS(TDATA7)
TDATA13	TDATA6*SIN(TDATA7)
TDATA14	0
TDATA15	COS(TDATA7)
TDATA16	SIN(TDATA7)
TDATA17	TDATA9
TDATA18	TDATA11
TDATA19	0, if TDATA12=0 or TDATA12=1 1, if TDATA12=2 or TDATA12=3.
TDATA21	TDATA5
TDATA22	0
TDATA23	TDATA1
TDATA24	0
TDATA25	0
TDATA26	1

Adapting a type 7 kinematics to a type 50 kinematics.

Vector kinematics parameter (type 50).	Equivalent value/parameter in type 7 kinematics.
TYPE	50
TDATA1	TDATA13
TDATA2	TDATA14
TDATA3	TDATA15
TDATA4	0
TDATA5	0
TDATA6	1
TDATA7	TDATA8
TDATA8	TDATA10
TDATA9	0, if TDATA12=0 or TDATA12=2 1, if TDATA12=1 or TDATA12=3.
TDATA11	TDATA5*SIN(TDATA7)
TDATA12	TDATA3
TDATA13	TDATA5*COS(TDATA7)
TDATA14	SIN(TDATA7)
TDATA15	0
TDATA16	COS(TDATA7)
TDATA17	TDATA9
TDATA18	TDATA11
TDATA19	0, if TDATA12=0 or TDATA12=1 1, if TDATA12=2 or TDATA12=3.
TDATA21	0
TDATA22	TDATA6
TDATA23	TDATA1
TDATA24	0
TDATA25	0
TDATA26	1

Adapting a type 8 kinematics to a type 50 kinematics.

Vector kinematics parameter (type 50).	Equivalent value/parameter in type 8 kinematics.
TYPE	50
TDATA1	TDATA13
TDATA2	TDATA14
TDATA3	TDATA15
TDATA4	0
TDATA5	0
TDATA6	1
TDATA7	TDATA8
TDATA8	TDATA10
TDATA9	0, if TDATA12=0 or TDATA12=2 1, if TDATA12=1 or TDATA12=3.
TDATA11	TDATA2
TDATA12	TDATA6*SIN(TDATA7)
TDATA13	TDATA6*COS(TDATA7)
TDATA14	0
TDATA15	SIN(TDATA7)
TDATA16	COS(TDATA7)
TDATA17	TDATA9
TDATA18	TDATA11
TDATA19	0, if TDATA12=0 or TDATA12=1 1, if TDATA12=2 or TDATA12=3.
TDATA21	TDATA5
TDATA22	0
TDATA23	TDATA1
TDATA24	0
TDATA25	0
TDATA26	1

Adapting a type 9 kinematics to a type 51 kinematics.

Vector kinematics parameter (type 51).	Equivalent value/parameter in type 9 kinematics.
TYPE	51
TDATA1	0
TDATA2	-TDATA3
TDATA3	-(TDATA5+TDATA4)
TDATA4	1
TDATA5	0
TDATA6	0
TDATA7	TDATA8
TDATA8	TDATA10
TDATA9	0, if TDATA12=0 or TDATA12=2 1, if TDATA12=1 or TDATA12=3.
TDATA10	TDATA7
TDATA11	-TDATA2
TDATA12	0
TDATA13	TDATA5
TDATA14	0
TDATA15	1
TDATA16	0
TDATA17	TDATA9
TDATA18	TDATA11
TDATA19	0, if TDATA12=0 or TDATA12=1 1, if TDATA12=2 or TDATA12=3.
TDATA20	TDATA7
TDATA31	TDATA16

Adapting a type 10 kinematics to a type 51 kinematics.

Vector kinematics parameter (type 51).	Equivalent value/parameter in type 10 kinematics.
TYPE	51
TDATA1	0
TDATA2	-(TDATA5+TDATA3)
TDATA3	-TDATA4
TDATA4	1
TDATA5	0
TDATA6	0
TDATA7	TDATA8
TDATA8	TDATA10
TDATA9	0, if TDATA12=0 or TDATA12=2 1, if TDATA12=1 or TDATA12=3.
TDATA10	TDATA7
TDATA11	-TDATA2
TDATA12	TDATA5
TDATA13	0
TDATA14	0
TDATA15	0
TDATA16	1
TDATA17	TDATA9
TDATA18	TDATA11
TDATA19	0, if TDATA12=0 or TDATA12=1 1, if TDATA12=2 or TDATA12=3.
TDATA20	TDATA7
TDATA31	TDATA16

Adapting a type 11 kinematics to a type 51 kinematics.

Vector kinematics parameter (type 51).	**Equivalent value/parameter in type 11 kinematics.**
TYPE	51
TDATA1	-(TDATA2)
TDATA2	0
TDATA3	-(TDATA5+TDATA4)
TDATA4	0
TDATA5	1
TDATA6	0
TDATA7	TDATA8
TDATA8	TDATA10
TDATA9	0, if TDATA12=0 or TDATA12=2 1, if TDATA12=1 or TDATA12=3.
TDATA10	TDATA7
TDATA11	0
TDATA12	-TDATA3
TDATA13	TDATA5
TDATA14	1
TDATA15	0
TDATA16	0
TDATA17	TDATA9
TDATA18	TDATA11
TDATA19	0, if TDATA12=0 or TDATA12=1 1, if TDATA12=2 or TDATA12=3.
TDATA20	TDATA17
TDATA31	TDATA16

Adapting a type 12 kinematics to a type 51 kinematics.

Vector kinematics parameter (type 51).	Equivalent value/parameter in type 12 kinematics.
TYPE	51
TDATA1	-(TDATA5+TDATA2)
TDATA2	0
TDATA3	-TDATA4
TDATA4	0
TDATA5	1
TDATA6	0
TDATA7	TDATA8
TDATA8	TDATA10
TDATA9	0, if TDATA12=0 or TDATA12=2 1, if TDATA12=1 or TDATA12=3.
TDATA10	TDATA7
TDATA11	TDATA5
TDATA12	-TDATA3
TDATA13	0
TDATA14	0
TDATA15	0
TDATA16	1
TDATA17	TDATA9
TDATA18	TDATA11
TDATA19	0, if TDATA12=0 or TDATA12=1 1, if TDATA12=2 or TDATA12=3.
TDATA20	TDATA17
TDATA31	TDATA16

Adapting a type 13 kinematics to a type 52 kinematics.

Vector kinematics parameter (type 52).	Equivalent value/parameter in type 13 kinematics.
TYPE	52
TDATA1	TDATA13
TDATA2	TDATA14
TDATA3	TDATA15
TDATA4	0
TDATA5	0
TDATA6	0
TDATA7	0
TDATA8	0
TDATA9	1
TDATA11	0
TDATA12	0
TDATA13	0
TDATA14	1
TDATA15	0
TDATA16	0
TDATA17	TDATA8
TDATA18	TDATA10
TDATA19	0
TDATA21	0
TDATA22	TDATA3
TDATA23	TDATA1
TDATA24	0
TDATA25	0
TDATA26	1

Vector kinematics parameter (type 52).	Equivalent value/parameter in type 13 kinematics.
TDATA31	-TDATA4
TDATA32	0
TDATA33	-TDATA6
TDATA34	0
TDATA35	1
TDATA36	0
TDATA37	TDATA9
TDATA38	TDATA11
TDATA39	0
TDATA40	To be determined.
TDATA41	0
TDATA42	0
TDATA43	0
TDATA44	0
TDATA45	0
TDATA46	0
TDATA47	0
TDATA48	0
TDATA49	1
TDATA50	To be determined.
TDATA51	To be determined.
TDATA52	To be determined.

Adapting a type 14 kinematics to a type 52 kinematics.

Vector kinematics parameter (type 52).	Equivalent value/parameter in type 14 kinematics.
TYPE	52
TDATA1	TDATA13
TDATA2	TDATA14
TDATA3	TDATA15
TDATA4	0
TDATA5	0
TDATA6	0
TDATA7	0
TDATA8	0
TDATA9	1
TDATA11	0
TDATA12	0
TDATA13	0
TDATA14	1
TDATA15	0
TDATA16	0
TDATA17	TDATA8
TDATA18	TDATA10
TDATA19	0
TDATA21	0
TDATA22	TDATA3
TDATA23	TDATA1
TDATA24	0
TDATA25	0
TDATA26	1

Vector kinematics parameter (type 52).	Equivalent value/parameter in type 14 kinematics.
TDATA31	-TDATA4
TDATA32	-TDATA5
TDATA33	0
TDATA34	0
TDATA35	0
TDATA36	1
TDATA37	TDATA9
TDATA38	TDATA11
TDATA39	0
TDATA40	To be determined.
TDATA41	0
TDATA42	0
TDATA43	0
TDATA44	0
TDATA45	0
TDATA46	0
TDATA47	0
TDATA48	0
TDATA49	1
TDATA50	To be determined.
TDATA51	To be determined.
TDATA52	To be determined.

Adapting a type 15 kinematics to a type 52 kinematics.

Vector kinematics parameter (type 52).	Equivalent value/parameter in type 15 kinematics.
TYPE	52
TDATA1	TDATA13
TDATA2	TDATA14
TDATA3	TDATA15
TDATA4	0
TDATA5	0
TDATA6	0
TDATA7	0
TDATA8	0
TDATA9	1
TDATA11	0
TDATA12	0
TDATA13	0
TDATA14	0
TDATA15	1
TDATA16	0
TDATA17	TDATA8
TDATA18	TDATA10
TDATA19	0
TDATA21	TDATA2
TDATA22	0
TDATA23	TDATA1
TDATA24	0
TDATA25	0
TDATA26	1

Vector kinematics parameter (type 52).	Equivalent value/parameter in type 15 kinematics.
TDATA31	0
TDATA32	-TDATA5
TDATA33	-TDATA6
TDATA34	1
TDATA35	0
TDATA36	0
TDATA37	TDATA9
TDATA38	TDATA11
TDATA39	0
TDATA40	To be determined.
TDATA41	0
TDATA42	0
TDATA43	0
TDATA44	0
TDATA45	0
TDATA46	0
TDATA47	0
TDATA48	0
TDATA49	1
TDATA50	To be determined.
TDATA51	To be determined.
TDATA52	To be determined.

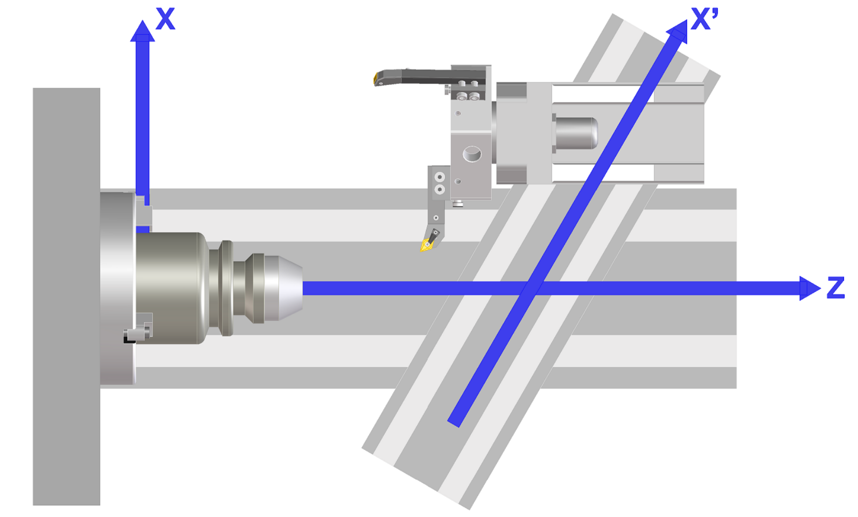
Adapting a type 16 kinematics to a type 52 kinematics.

Vector kinematics parameter (type 52).	Equivalent value/parameter in type 16 kinematics.
TYPE	52
TDATA1	TDATA13
TDATA2	TDATA14
TDATA3	TDATA15
TDATA4	0
TDATA5	0
TDATA6	0
TDATA7	0
TDATA8	0
TDATA9	1
TDATA11	0
TDATA12	0
TDATA13	0
TDATA14	0
TDATA15	1
TDATA16	0
TDATA17	TDATA8
TDATA18	TDATA10
TDATA19	0
TDATA21	TDATA2
TDATA22	0
TDATA23	TDATA1
TDATA24	0
TDATA25	0
TDATA26	1

Vector kinematics parameter (type 52).	Equivalent value/parameter in type 16 kinematics.
TDATA31	-TDATA4
TDATA32	-TDATA5
TDATA33	0
TDATA34	0
TDATA35	0
TDATA36	1
TDATA37	TDATA9
TDATA38	TDATA11
TDATA39	0
TDATA40	To be determined.
TDATA41	0
TDATA42	0
TDATA43	0
TDATA44	0
TDATA45	0
TDATA46	0
TDATA47	0
TDATA48	0
TDATA49	1
TDATA50	To be determined.
TDATA51	To be determined.
TDATA52	To be determined.

2.17.13 Configuration of angular transformations.

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. A typical case is the X axis of a lathe that for sturdiness reasons is not perpendicular to the Z axis. Programming in the Cartesian system (Z-X) movements requires activating an angular transformation 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.



X Cartesian axis.

X' Angular axis.

Z Orthogonal axis.

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.

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; if it is not active, the real coordinates of the axis well be displayed.

2.17.14 Configuration of angular transformations (parameters).

NANG

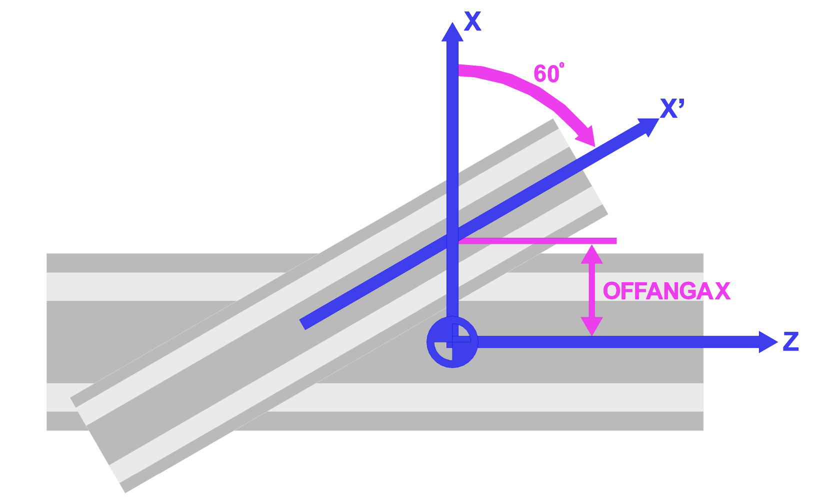
Possible values: From 0 to 14.
Default value: 0.
Associated variable: (V.)MPK.NANG

Number of angular transformations defined. 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

This parameter shows the configuration tables of the angular transformations. The following parameters must be defined for each angular transformation:

Parameter.	Meaning.
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

ANGTR | ANGAXNA

Possible values: Any axis defined in the parameter "AXISNAME".
Default value: None.
Associated variable: (V.)MPK.ANGAXNA[ang]

Name of the angular axis (incline axis).

ANGTR | ORTAXNA

Possible values: Any axis defined in the parameter "AXISNAME".
Default value: None.
Associated variable: (V.)MPK.ORTAXNA[ang]

Name of the Cartesian axis used in the angular transformation.

ANGTR | ANGANTR

Possible values: Within ±360.0000º.
Default value: 30º.
Associated variable: (V.)MPK.ANGANTR[ang]

Angle between the Cartesian angle and the angular axis it is associated with. Positive angle when the angular axis is turning clockwise and negative in the other direction. If its value is 0º, there is no need to do an angular transformation.

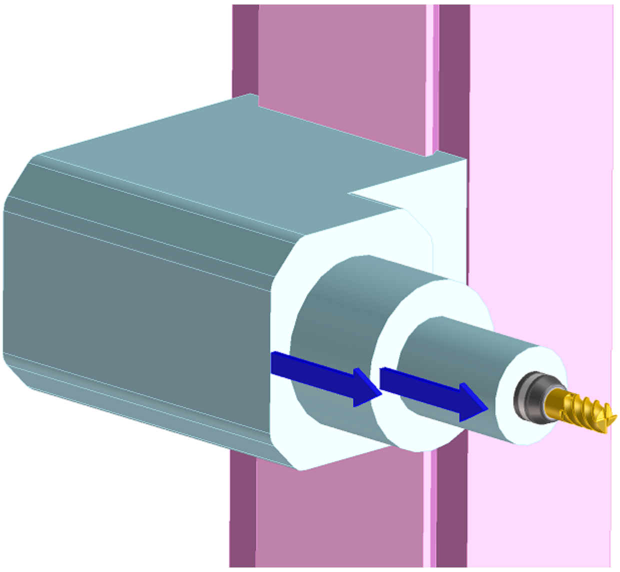
ANGTR | OFFANGAX

Possible values (1): Between ±99 999.9999 mm.
Possible values (2): Within ±3937.00787 inches.
Default value: 0.
Associated variable: (V.)MPK.OFFANGAX[ang]

Offset of the origin of the angular transformation. Distance between machine zero and the origin of the coordinate system of the incline axis.

2.17.15 Kinematics of parallel axes.

This kinematics allows you to work with axes that share physical direction and act on the tool. The CNC does not assume any kinematics after power-up; the activation of the kinematics is carried out from the part program by means of the #PARALAX instruction. The kinematics remain active after a reset or M30.



NPARAL
Number of parallel axis kinematics.
Possible values: From 0 to 6.
Default value: 0.
Associated variable: (V.)MPK.NPARAL

Up to 6 parallel axis kinematics can be customised.

PARALLEL n
Table of kinematics of parallel axes.

Displays the configuration tables of the parallel axis kinematics. The following parameters must be defined for each kinematics:

Parameter.	Meaning.
PARALAX	Main axis of the transformation.
ADDAX	Parallel additive axis.
ADDORG	Distance between the machine zeroes of both axes.
PARALINI	Status of parallel kinematics at start-up.

PARALAX
Main axis of the transformation.
Possible values: Any axis defined in the parameter "AXISNAME".
Default value: -.
Associated variable: (V.)MPK.PARALAX[nb]
Parameter included in the PARALLEL n table.

Main axis of the transformation.

ADDAX
Parallel additive axis.

Possible values: Any axis defined in the parameter "AXISNAME".
Default value: -.
Associated variable: (V.)MPK.ADDAX[nb]
Parameter included in the PARALLEL n table.

Parallel additive axis.

ADDORG
Distance between the machine zeroes of both axes.

Possible values: ±214748.3647 mm /
Default value: 0
Associated variable: (V.)MPK.ADDORG[nb]
Parameter included in the PARALLEL n table.

Distance between the machine zeroes of both axes, from the main axis to the additive axis.

PARALINI
Status of parallel kinematics at start-up.

Possible values: PARALAX OFF / PARALAX VIS.
Default value: PARALAX OFF.
Associated variable: (V.)MPK.PARALINI[nb]
Parameter included in the PARALLEL n table.

Status of parallel kinematics at start-up.

Value.	Meaning.
PARALAX OFF	Kinematics deactivated.
PARALAX VIS.	Active kinematics in display mode. The CNC updates the dimension of the master axis with the movement made by the additive axis, so that the master axis always shows the distance from the tool to the workpiece.

•

2.18 Machine parameters; frequency filters.

ORDER

Possible values: From 0 to 50.
Default value: 0 (the filter is not applied).
Associated variable: (V.)MPFIL.ORDER.nb

Filter order. The down ramp is dampened down; the higher the order the greater the drop.

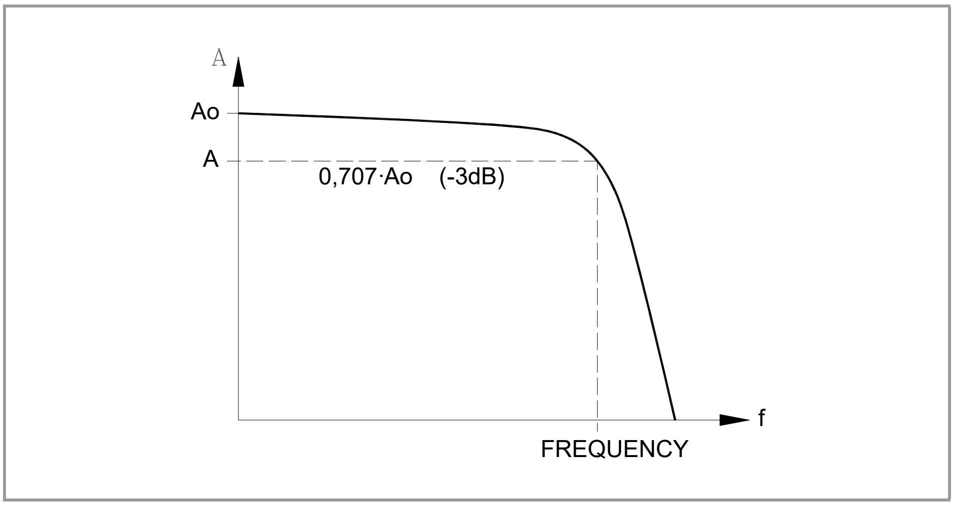
TYPE

Possible values: Low pass / Band rejection / Fagor low pass / File.
Default value: Low Passing.
Associated variable: (V.)MPFIL.TYPE.nb

Filter type. 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 filters.

Low-pass filters limit the band pass at a certain frequency. 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.

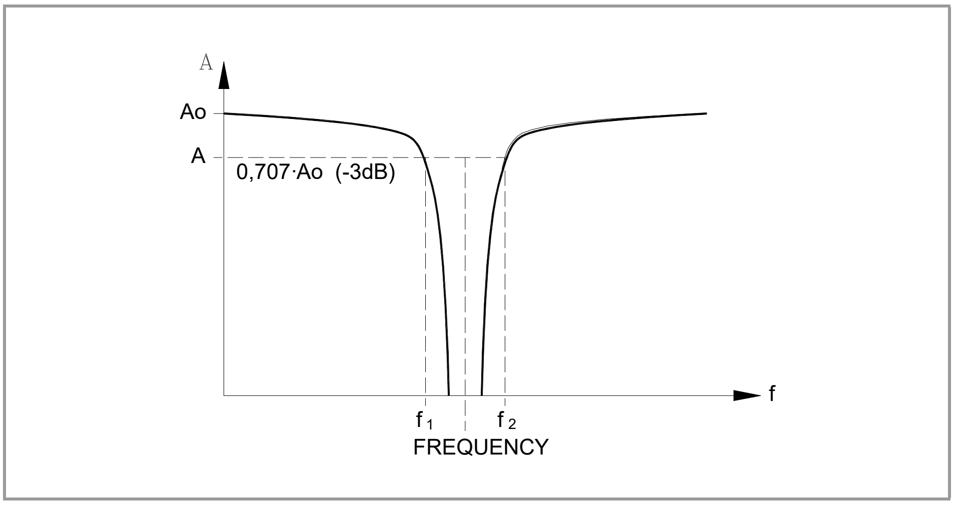


For the same frequency, the low-pass filter needs less order than the Fagor low-pass filter to have the same level of filtering and, as a consequence, causes less delay.

- The low-pass filter inserts a non-constant phase shift variable 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.

Anti-resonance filter (notch filter).

The band-rejection filter (notch filter) limits a zone of the band pass. 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.



File.

The filter characteristics are defined in a file (parameter FILTERFILE).

FREQUENCY

Possible values: From 0 to 500.0 Hz.
Default value: 30.0 Hz.
Associated variable: (V.)MPFIL.FREQUENCY.nb

Cutoff or center frequency

- 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

Possible values: From 0 to 100.0
Default value: 1.0
Associated variable: (V.)MPFIL.NORWIDTH.nb

Standardized bandwidth. 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

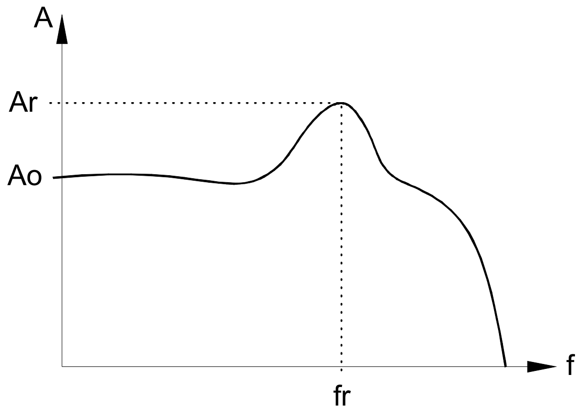
$$\text{NORBWIDTH} = \frac{\text{FREQUENCY}}{(f_2 - f_1)}$$

SHARE

Possible values: from 0 to 100 %.
Default value: 100 %.
Associated variable: (V.)MPFIL.SHARE.nb

Percentage of signal going through the filter. This value must be equivalent to the percentage overshooting of the resonance because it has to make up for it.

Example of a calculation for a particular response of the machine.



SHARE=100(Ar-Ao)/Ar

FILTERFILE

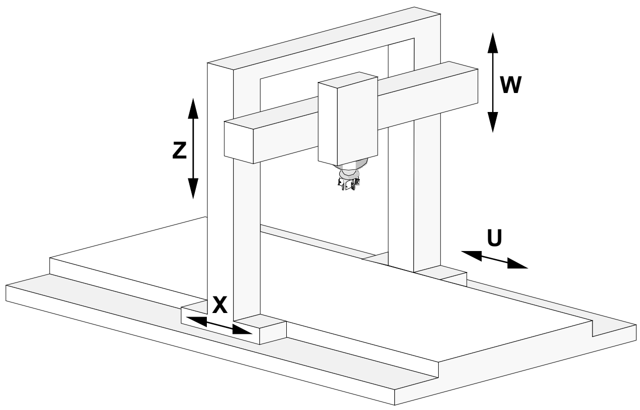
Possible values: any text with up to 64 characters.
Associated variable: (V.)MPFIL.FILTERFILE.nb

File containing the filter definition. Filters must be saved in the file Mtb\Data.

2.19 Machine parameters; gantry axes.

Gantry axes are any two axes that, due to the way the machine is built, must move together 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 (parameter AXISTYPE) and drives must be of the same type.
- The channel changing permission (parameter AXISEXCH) must be the same in both axes.
- Both axes and drives must have the same software limits (same POSLIMIT and NEGLIMIT 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 (parameter 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.

MASTERAXIS

Possible values: Any axes defined in AXISNAME.
Associated variable: (V.)MPGTY.MASTERAXIS.nb

Gantry master axis.

SLAVEAXIS

Possible values: Any axes defined in AXISNAME.
Associated variable: (V.)MPGTY.SLAVEAXIS.nb

Slave axis of the gantry.

WARNCOUPE

Possible values (1): From 0.0001 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Possible values (3): From 0.0001 to 99 999.9999 degrees.
Default value: 1.0000 mm / 0,03937 inches / 1.0000 degree.
Associated variable: (V.)MPGTY.WARNCOUPE.nb

Maximum difference allowed between the following error of the master axis and slave axis before prompting 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

Possible values (1): From 0.0001 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Possible values (3): From 0.0001 to 99 999.9999 degrees.
Default value: 1.0000 mm / 0,03937 inches / 1.0000 degree.
Associated variable: (V.)MPGTY.MAXCOUPE.nb

Maximum difference allowed between the following error of the master axis and slave axis before prompting a warning.

DIFFCOMP

Possible values: Yes / No.
Default value: Yes.
Associated variable: (V.)MPGTY.DIFFCOMP.nb

This parameter enables the correction of the position difference between the master and the slave axes after homing. On axes with absolute feedback, if the DIFFCOMP parameter is set to "Yes", the CNC will correct the coordinate difference when enabling the axis (rising edge of the SERVOxnON flag of the master or slave 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

Possible values (1): From 0 to 99 999.9999 mm.
Possible values (2): From 0 to 3937.00787 inches.
Possible values (3): From 0 to 99 999.9999 degrees.
Default value: 0.
Associated variable: (V.)MPGTY.MAXDIFF.nb

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.

2.20 Machine parameters; tandem axes.

2.20.1 Configuring a tandem (spindle) axis.

MASTERAXIS

Possible values: Any axis defined in the parameter AXISNAME or spindle defined in SPDLNAME.
Default value: - .
Associated variable: (V.)MPTDM.MASTERAXIS.nb

Axis or master spindle of the tandem. 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

Possible values: Any axis defined in the parameter AXISNAME or spindle defined in SPDLNAME.
Default value: - .
Associated variable: (V.)MPTDM.SLAVEAXIS.nb

Axis or slave spindle of the tandem. The slave motor only provides torque. The CNC closes the position loop only with the position of the master axis of the tandem.

TORQDIST

Possible values: from 1 to 99 %.
Default value: 50 %.
Associated variable: (V.)MPTDM.TORQDIST.nb

Torque distribution. This parameter sets the torque percentage supplied by each motor to obtain the total necessary torque in the tandem. The parameter indicates the total torque percentage to be provided by the master motor. The difference between the value of this parameter and 100% is the percentage that the slave motor will apply. If the motors are identical and they're both supposed to output the same torque, this parameter should be set to 50 %.

PRELOAD

Possible values: Within ±100 %.
Default value: 0 (disable the preload).
Associated variable: (V.)MPTDM.PRELOAD.nb

Preload between both motors. 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. If is set to 0, there is no preload between both motors.

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

Possible values: From 0 to 65 535 milliseconds.
Default value: 0 (disable filter).
Associated variable: (V.)MPTDM.PRELFITI.nb

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. Setting it to 0 disables the filter.

TPROGAIN

Possible values: from 0 to 100 %.
Default value: 0 (no proportional gain applied).
Associated variable: (V.)MPTDM.TPROGAIN.nb

Proportional gain for the tandem. The proportional controller generates an output proportional to the torque error between the two motors.

Proportional gain.

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

[S] Speed.
[Kp] Proportional gain.
[T.err] Torque error between motors.

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

[S.max] Maximum velocity.
[T.nom] Rated torque.

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

[T.mst] Torque of the master motor.
[T.slv] Torque of the slave motor.
[PRELOAD] Preload between both motors (machine parameter PRELOAD).

Example.
For a tandem axis with a maximum speed of 2000 rpm and a nominal torque of 20 Nm, the parameter TPROGAIN is set to 10%.
Kp = (2000 rpm / 20 Nm) × 0,1 = 10 rpm/Nm.

TINTTIME

Possible values: From 0 to 65 535 milliseconds.
Default value: 0 (no integral gain applied).
Associated variable: (V.)MPTDM.TINTIME.nb

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

Integral gain.

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

[S] Speed.
[Ki] Integral gain.
[T.err] Torque error between motors.

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

[Kp] Proportional gain.

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

[T.mst] Torque of the master motor.
[T.slv] Torque of the slave motor.
[PRELOAD] Preload between both motors (machine parameter PRELOAD).

TCOMPLIM

Possible values: from 0 to 100 %.
Default value: 0 (disable the tandem axis).
Associated variable: (V.)MPTDM.TCOMPLIM.nb

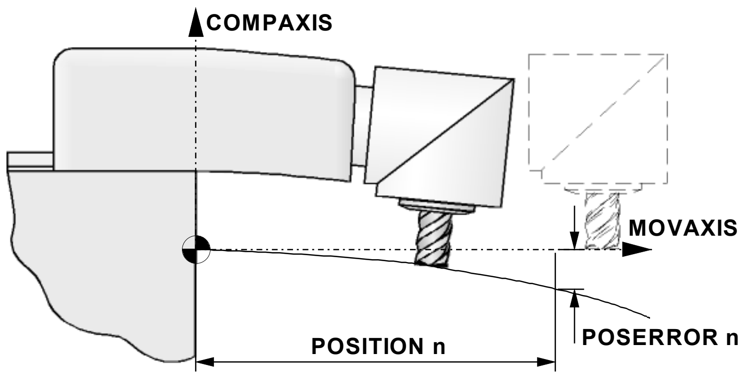
Torque compensation limit. This parameter sets the maximum torque compensation introduced by the tandem. This parameter refers to the master motor. It is defined as percentage of the maximum speed of the master motor. If programmed with a "0" value, the output of the tandem control will be zero, thus disabling the tandem. The torque compensation limit also applies to the integral.

2.21 Machine parameters; compensations.

Even though the CNC always activates the compensation tables defined by it by default, these tables may be disabled via the PLC using the mark DISCROSS.

- Ballscrew compensation is used when the movement of an axis generates position errors along the same axis.
- Cross compensation is used when the movement of one axis causes variations in the position of another axis.

Cross compensation.



The CNC applies the cross compensations, while taking into account the dependencies between them. For example, if the coordinate for X is compensated with Y, and the coordinate for Y with Z, the CNC will first apply the YZ compensation and then the XY compensation. The compensation tables must be defined consistently in this order.

Import 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 compensation tables.

MOVEAXIS

Possible values: Any axes defined in AXISNAME.

Master axis. Axis whose movement causes variations.

COMPAXIS

Possible values: Any axes defined in AXISNAME.

Compensated axis. If the master and compensated axis are the same, then this is defined as a ballscrew compensation. If the master and compensated axis are not the same, then this is defined as a cross compensation.

NPOINTS

Possible values: From 0 to 1000.
Default value: 0 (there is no table).
Associated variable: (V.)MPCMP.NPOINTS.tbl

Number of points in the table.

TYPE

Possible values: Real / Theoretical.
Default value: Real.
Associated variable: (V.)MPCMP.TYPE.tbl

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

BIDIR

Possible values: Yes / No.
Default value: No.
Associated variable: (V.)MPCMP.BIDIR.tbl

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

Possible values: Yes / No.
Default value: No.
Associated variable: (V.)MPCMP.REFNEED.tbl

This parameter indicates whether the involved axes must be referenced or not before applying the compensation.

DATA

This parameter shows the table to define the compensation values. For each compensation point, the following fields must be defined.

Parameters.	Meaning.
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.

The CNC only lets access this table when parameters are MOVEAXIS, COMPAXIS and NPOINTS set. Parameter NEGERROR only if the table has been defined with bidirectional compensation (parameter BIDIR).

DATA | POSITION

Possible values (1): Between ±99 999.9999 mm.
Possible values (2): Within ±3937.00787 inches.
Possible values (3): Between ±99 999.9999 degrees.
Default value: 0.
Associated variable: (V.)MPCMP.POSITION[pt].tbl

Position of the master axis at each compensation point. 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.

DATA | POSERROR

Possible values (1): Between ±99 999.9999 mm.
Possible values (2): Within ±3937.00787 inches.
Possible values (3): Between ±99 999.9999 degrees.
Default value: 0.
Associated variable: (V.)MPCMP.POSERROR[pt].tbl

This parameter indicates the error to be compensated on the axis COMPAXIS when the master axis moves in a 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, whose position is indicated in parameter REFVALUE, must have ·0· error.

DATA | NEGERROR

Possible values (1): Between ±99 999.9999 mm.
Possible values (2): Within ±3937.00787 inches.
Possible values (3): Between ±99 999.9999 degrees.
Default value: 0.
Associated variable: (V.)MPCMP.NEGERROR[pt].tbl

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

The machine reference point, whose position is indicated in parameter REFVALUE, must have ·0· error.

2.22 Machine parameters for the magazine.

2.22.1 Tool magazine configuration.

NTOOLMZ

Possible values: From 0 to 2.
Default value: 1.
Associated variable: (V.)TM.NTOOLMZ

Number of 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

Possible values: Yes / No.
Default value: No.
Associated variable: (V.)TM.MZGROUND[mz]

Ground tools. They are the ones not located in the magazine. When programming them, the CNC requests them to be inserted in the spindle.

MAGAZINE n

This parameter shows the tables to define the magazine data. Each table shows the following machine parameters to configure it.

Parameters.	Meaning.
STORAGE	Parameters related to storage.
MANAGEMENT	Magazine management related parameters.

2.22.2 Magazine data.

MAGAZINE n | STORAGE

This parameter shows or hides the magazine data. Each table shows the following machine parameters to configure it.

Parameter.	Meaning.
SIZE	Size of the tool magazine (number of pockets).
RANDOM	Random magazine

MAGAZINE n | STORAGE | SIZE

Possible values: From 0 to 1000.
Default value: 20.
Associated variable: (V.)TM.MZSIZE[mz]

Magazine size (number of positions), not counting the ground tools.

MAGAZINE n | STORAGE | RANDOM

Possible values: Yes / No.
Default value: No.
Associated variable: (V.)TM.MZRANDOM[mz]

Random magazine. This parameter indicates whether the tools must always occupy the same position (non-random) or they may occupy any position (random).

2.22.3 Tool magazine management.

MAGAZINE n | MANAGEMENT

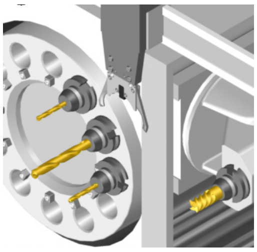
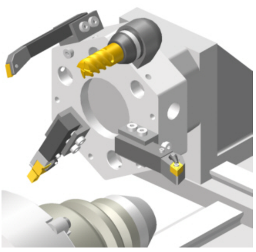
This parameter shows the data to configure the magazine management. Each table shows the following machine parameters to configure it.

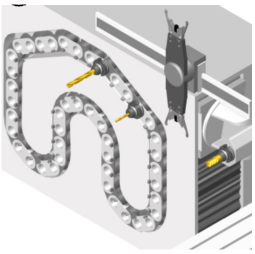
Parameter.	Meaning.
TYPE	Type of magazine.
CYCLIC	Cyclic tool changer.
OPTIMIZE	Optimizing management.
RESPECTSIZES	In random magazine, look for pockets of the same size.
M6ALONE	Action when executing an M06 without selecting a T.

MAGAZINE n | MANAGEMENT | TYPE

Possible values: Asynchronous / Synchronous / Turret / Synchronous + 2 Arms / Synchronous + 1 Arm.
Default value: Synchronous.
Associated variable: (V.)TM.MZTYPE[mz]

Type of magazine.

Value.	Meaning.
Asynchronous. 	The magazine is 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) During machining, the magazine selects the new tool. The changer arm takes the tool from the magazine and carries it to the spindle. (2) The CNC finalizes the movement of the axes. (3) The second claw (holder) of the changer arm removes the tool from the spindle and changes the tool. (4) The CNC resumes with the program execution and the changer arm returns to the magazine to leave the tool.
Synchronous. 	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) The CNC finalizes the movement of the axes. (2) The magazine approaches the spindle to grab the tool. (3) The magazine selects the new tool and inserts it into the spindle. (4) The magazine is removed from the spindle. (5) The CNC resumes program execution.
Turret. 	Typical magazine for lathes. The tool cannot be changed while the part is being machined. The tool change is carried out as follows: (1) The CNC finalizes the movement of the axes. (2) The magazine moves away from the spindle. (3) The magazine rotates to put the new tool into the work position. (4) The CNC resumes program execution.

Value.	Meaning.
Synchronous + 1 arms. Synchronous + 2 arms. 	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) During machining, the magazine selects the new tool. (2) The CNC finalizes the movement of the axes. (3) The changer arm takes a tool with each holder (magazine and spindle) and makes the change. (4) The changer arm returns to the magazine to leave the tool. (5) The CNC resumes program execution.

MAGAZINE n | MANAGEMENT | CYCLIC

Possible values: Yes / No.
Default value: Yes.
Associated variable: (V.)TM.MZCYCLIC[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. 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).

MAGAZINE n | MANAGEMENT | OPTIMIZE

Possible values: Yes / No.
Default value: Yes.
Associated variable: (V.)TM.MZOPTIMIZED[mz]

Optimizing management. Optimization only works when executing a program; In MDI mode, this parameter is ignored and all the blocks are executed.

Value.	Meaning.
Yes	When programming several Ts in a row without an M06, the magazine only selects the tools involved in the change.
No	When programming several Ts in a row without an M06, the magazine selects all the programmed tools even if they are not involved in the change.

MAGAZINE n | MANAGEMENT | RESPECTSIZES

Possible values: Yes / No.
Default value: No.
Associated variable: (V.)TM.RESPECTSIZES

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.

MAGAZINE n | MANAGEMENT | M6ALONE

Possible values: Nothing / Issue a warning / Issue an error.
Default value: Show an error.
Associated variable: (V.)TM.MZM6ALONE[mz]

The M06 function implies a tool change. This parameter indicates what happens when executing an M06 without having selected a tool first.

2.23 Machine parameters for HMI (Interface).

2.23.1 HMI type.

HMIELITE

Possible values: Yes / No.
Default value: No.

Start the CNC with HMIelite.

2.23.2 Main window dimensions and resolution.

WINDOW

This parameter shows the data to configure the main window. The table has the following machine parameters.

Parameter.	Meaning.
POSX	X coordinate of the top left corner.
POSY	Y coordinate of the top left corner.
RESOLUTION	Window resolution.
WIDTH	Width of the window.
HEIGHT	Height of the window.

WINDOW | POSX

Default value: 0.
Parameter included in the WINDOW table.

Consult the machine parameter of HMI POSY.

WINDOW | POSY

Default value: 0.
Parameter included in the WINDOW table.

The POSX and POSY parameters define the X and Y coordinate to position the upper left corner of the interface. The interface can be moved by clicking with the mouse on the icon showing the program status of the active channel. Double click on this icon and the interface is positioned in the upper left corner (POSX=0 POSY=0).

WINDOW | RESOLUTION

Possible values: CUSTOM / 10" / 15" / 19" / 21" LANDSCAPE / 21" PORTRAIT.
Default value: 10".
Parameter included in the WINDOW table.

Screen resolution. The CUSTOM option enables the width (WIDTH parameter) and the height (HEIGHT parameter) of the window to be customized. The predefined sizes are set to the following resolutions.

RESOLUTION. (value)	Interface size (pixels).
CUSTOM	This option enables the width (WIDTH parameter) and the height (HEIGHT parameter) of the screen to be customized.
10"	800 × 600.
15"	1024 × 768.
19"	1280 × 1024.
21" LANDSCAPE Horizontal widescreen.	1300 × 1080. The interface does not cover the entire screen, allowing room for a virtual keyboard.
21" PORTRAIT Vertical widescreen.	1080 × 1400. The interface does not cover the entire screen, allowing room for a virtual keyboard.

WINDOW | WIDTH
Width of the window.

Possible values: From 800 to 1680 pixels.
Default value: 800 pixels.
Parameter included in the WINDOW table.

Consult the machine parameter of HMI HEIGHT.

WINDOW | HEIGHT
Height of the window.

Possible values: From 480 to 1050 pixels.
Default value: 600 pixels.
Parameter included in the WINDOW table.

The WIDTH and HEIGHT parameters define the window size. These parameters can only be modified when the RESOLUTION parameter has the value "CUSTOM".

2.23.3 Customizing the softkeys.

HMENUNSOFTKEY

Possible values: From 7 to 12.
Default value: 7.

Number of softkeys on the horizontal menu.

VMENUNSOFTKEY

Possible values: From 5 to 12.
Default value: 5.

Number of softkeys on the vertical menu.

SOFTKEYMETRICS

Possible values: Standard / Advanced.
Default value: Standard.

This parameter makes it possible to optimize the width of the vertical softkeys and the height of the horizontal softkeys, so that the space they occupy is more proportionate.

Value.	Meaning.
Standard.	The interface maintains a proportional value between the height and width of the softkeys.
Advanced.	The interface optimizes the space occupied by the softkeys. Recommended option for panoramic monitors, where the interface occupies the entire screen.

2.23.4 Interface setting.

HMITYPE

Possible values: Standard / Classic / Advanced.
Default value: Standard.

HMI type. 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.	Default operating way for the 8070. 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.

SFTYPE

Possible values: Standard / Popup.
Default value: Standard.

Softkey tree type.

Value.	Meaning.
Standard.	Softkey tree based on menus and submenus, i.e. there are different softkey levels within the same work mode.
Popup.	Softkey tree based on popup menus so there is only one softkey level. There are no softkey submenu, which simplifies the operation.

VMENU

Possible values: Left / Right.
Default value: Right.

Position of the vertical softkey-menu. 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

Possible values: Select a language among those available.
Default value: English.

Work language.

ENGLISH	SPANISH	ITALIAN	GERMAN
FRENCH	BASQUE	PORTUGUESE	CHINESE
RUSSIAN	CZECK	KOREAN	DUTCH
POLISH	TURKISH		

KEYBOARDLANG

Possible values: Select a language from the list.
Default value: None (disable language selection).

Keyboard language. Each time the parameter changes value, after validating the machine parameters, the keyboard must be registered using the Disk Monitor application ("Register devices" option).

FFORMAT

Possible values: 3 characters.
Default value: 5,2.

Display format for the programmed F. 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 the original configuration (the one defined by the Fguim application).

SFORMAT

Possible values: 3 characters.
Default value: 5,1.

Display format for the programmed S. 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 the original configuration (the one defined by the Fguim application).

MMINCHSOFTKEY

Possible values: Yes / No.
Default value: No.

This parameter indicates whether or not the CNC shows the softkey for toggling between mm and inches. This parameter is linked to the MMInchSoftkey input of the mmc8070.ini file,

2.23.5 Setting the user key ([CUSTOM] key).

USERKEY

This table shows the parameters to configure the [CUSTOM] key. The table has the following machine parameters.

Parameter.	Meaning.
UKFUNCTION	Function of the key.
COMPONENT	Access a component of the CNC.
APPLICATION	Execute a PC application.

USERKEY | UKFUNCTION

Possible values: Windows / Component / Application / CNC OFF / Nothing.
Default value: Windows.

Function of the [CUSTOM] key. This key can perform one of the following functions.

Value.	Meaning.
Windows	Minimize the CNC.
Component	Accessing a component (work mode) of the CNC. It is only possible to define components that do not have an access key; for example, the PLC.
Application	Run a program or application; for example, the configuration tool Fguim.exe.
CNC OFF	Turn off the CNC application.
Nothing.	Disable the key.

USERKEY | COMPONENT

Possible values: Diagnosis / PLC / Machine parameters / TUNING.
Default value: -.

This parameter shows the list of components (work modes) of the CNC that have no preset key on the keyboard. Besides these components, it will also display the components created with the tool FGUIM.

USERKEY | APPLICATION

Possible values: Any program or application.
Default value: -.

When setting parameter USERKEY=Application, this parameter allows to select the application The whole application path must be indicated; for example ..\Fagor\Release\Fguim.exe.

For a laser model, it is recommended to use this key to open the "Lantek Expert Inside" application for the nesting; C:\Lantek\Expert\Expert.exe.

2.23.6 Configuring the change key ([NEXT] key).

CHANGEKEY

This table shows the parameters to configure the [NEXT] key. The table has the following machine parameters.

Parameter.	Meaning.
CKFUNCTION	Function of the key.
SYSMENUMODE	Behavior of the system menu.
SYSHMENU	Horizontal system menu.
SYSVMENU	Vertical system menu.

CHANGEKEY | CKFUNCTION

Possible values: Next page / Next channel / Menu.
Default value: Next page.

Function of the [NEXT] key. This key can perform one of the following functions.

Value.	Meaning.
Next page.	The key selects the next page of the active component (work mode).
Next channel.	The key selects the next channel.
Menu.	The key shows the list of channels and pages on the softkey menus.

CHANGEKEY | SYSMENUMODE

Possible values: Volatile / Fixed.
Default value: Volatile.

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.

CHANGEKEY | SYSHMENU

Possible values: Disabled / Pages / Channels / Components.
Default value: Disabled.

Content of the horizontal softkey menu.

Value.	Meaning.
Disabled.	The menu will be disabled.
Screens.	The menu shows the pages of the active component.
Channels.	The menu shows the available channels.
Components.	The menu shows the components or work modes of the CNC.

CHANGEKEY | SYSVMENU

Possible values: Disabled / Pages / Channels / Components.
Default value: Disabled.

Content of the vertical softkey menu.

Value.	Meaning.
Disabled.	The menu will be disabled.
Screens.	The menu shows the pages of the active component.
Channels.	The menu shows the available channels.
Components.	The menu shows the components or work modes of the CNC.

2.23.7 Setting the menu key ([Main-Menu] key).

MAINMENUKEY

This table shows the parameters to configure the [Main-Menu] key. The table has the following machine parameters.

Parameter.	Meaning.
MMKFUNCTION	Function of the key.
MMKCOMPONENT	Access a component of the CNC.
MMKAPPLICATION	Execute a PC application.

MAINMENUKEY | MMKFUNCTION

Possible values: Default / Component / Application / Nothing.
Default value: By default.

Function of the [Main-Menu] key. This key can perform one of the following functions.

Value.	Meaning.
Default	Accessing the main menu of the CNC.
Component	Accessing a component (work mode) of the CNC; for example, the PLC.
Application	Run a program or application; for example, the configuration tool Fguim.exe.
Nothing.	Disable the key.

MAINMENUKEY | MMKCOMPONENT

Possible values: Choose a component from the list.
Default value: -.

This parameter shows the list of available components (work modes).

MAINMENUKEY | MMKAPPLICATION

Possible values: Any program or application.
Default value: -.

This parameter selects a program or application.

2.23.8 Setting the escape key ([ESC] key).

ESCKEY

This table shows the parameters to configure the [ESC] key. The table has the following machine parameters.

Parameter.	Meaning.
EKFUNCTION	Function of the key.
NPREVIOUS	Maximum number of previous components stored.

ESCAPEKEY | EKFUNCTION

Possible values: Previous menu / Previous component / Previous menu-component.
Default value: Previous menu.

Function of the [ESC] key. This key can perform one of the following functions.

Value.	Meaning.
Previous menu.	Every time the [ESC] key is pressed, the CNC goes up on the menu until it reaches the main menu of the component. From then on, the CNC changes to the main menu.
Previous component.	Every time the [ESC] key is pressed, the CNC goes up on the menu until it reaches the main menu of the component. From then on, the CNC toggles between the previous last work modes (parameter NPREVIOUS). The [ESC] key does not change to the main menu. The CNC does not change softkey menu of the previous work modes.
Previous menu and component.	Every time the [ESC] key is pressed, the CNC goes up on the menu until it reaches the main menu of the component. From then on, the CNC toggles between the previous last work modes (parameter NPREVIOUS). The [ESC] key does not change to the main menu. In each component, the [ESC] key first goes up on the menu until it reaches the main menu of the component.

ESCAPEKEY | NPREVIOUS

Possible values: From 1 to 5.
Default value: 1.

Number of components stored by the CNC to be displayed when pressing the escape key.

2.23.9 Simulated jog keyboard.

SIMJOGPANEL

Possible values: Yes / No.
Default value: No.

This parameter indicates whether the simulated operator panel is available or not. The simulated operator panel is laid over the CNC and is used to simulate the jog keys and work mode access keys. This operator panel may be required when working with telediagnosis (CNC remote control).

The simulated operator panel can be enabled and disabled using the [CTRL][J] keys.

2.23.10 CNC shut down.

WINEXIT

Possible values: Yes / No.
Default value: No.

Exit Windows when closing the CNC application.

2.23.11 Graphics configuration.

GRAPHTYPE

On the lathe model, this parameter shows the table to define the configuration of the graphics in each channel.

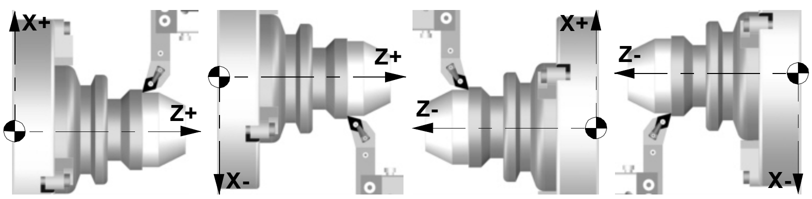
Parameter.	Meaning.
GRAPHTYPECH n	Type of graphics in each channel.

GRAPHTYPE | GRAPHTYPECH n

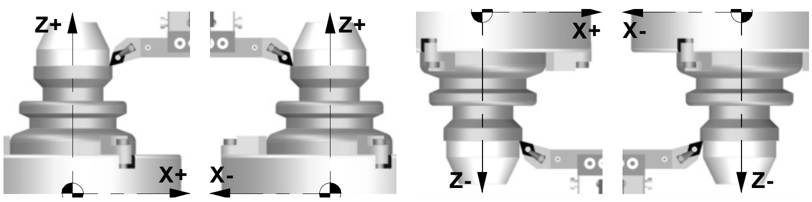
Possible values: Horizaontal or vertical lathe; X+ Z+ / X- Z+ / X+ Z- / X- Z-.
Default value: Horizontal X+ Z+.

Type of graphics in each 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.

2.23.12 Validate and save machine parameters.

MPMANAGEMENT

Possible values: Validate and save / Validate and consult / Consult.
Default value: Consult.

Validation management and backing up machine parameters.

Value.	Meaning.
Validate and save.	When exiting the table, validate and save the machine parameters. This option validates all tables.
Validate and consult.	When exiting the table, validate the machine parameters and prompts the user to save them or not.
Consult.	When exiting the table, ask the user to validate and save the machine parameters.

2.23.13 Maximum number of visible errors in the error window.

NERROR

Possible values: From 1 to 100.
Default value: 5.

Number of errors to show in the error window; if there are more errors, the CNC shows the last NERROR errors.

2.23.14 Reserved parameters (exclusive use of Fagor).

DIAGPSW

(Reserved)

2.24 OEM machine parameters.

2.24.1 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

This table offers 1000 generic parameters that the OEM can use as machine parameters. The table has the following machine parameters.

Parameter.	Meaning.
SIZE	Number of OEM parameters.
DATA	List of OEM parameters.

MTBPAR | SIZE

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.

MTBPAR | DATA

This parameter shows the table of OEM parameters. The following fields must be defined for each parameter.

Field.	Meaning.
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.

MTBPAR | DATA | VALUE

Possible values: Between ±99 999.9999.
Default value: 0.
Associated variable: (V.)MTB.P[nb]
Associated variable: (V.)MTB.PF[nb]

Value of the parameter. If the parameter is affected by the change of units (INCHES field), the value is entered in the table in the active units in the CNC (general 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. 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.

MTBPAR | DATA | MODE

Possible values: Read / Write.
Default value: Read.

The access to the parameters from the variables may be read-only or read-write. The value may always be written directly into the table, regardless of the value assigned to this field.

In Setup mode, the variables can modify the read-only MTBPAR parameters using a part program.

Setup mode. Variables can write parameters using write and read-only permission. These new values are kept after the CNC has been restarted or shutdown.

User Mode. Variables can only write parameters using write permission. These new values are kept after the CNC has been restarted or shutdown.

MTBPAR | DATA | INCHES

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.

MTBPAR | DATA | COMMENT

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 ..\Mtb\Data\Lang.

2.24.2 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

This table shows the tables to define the electronic cams. Each table shows the following machine parameters to configure it.

Parameter.	Meaning.
NCAMDATA	Number of electronic cams.
DATA	List of electronic cams.

CAMTABLE | NCAMDATA

Possible values: From 0 to 16.
Default value: 0.

This parameter sets the number of electronic cams of the system.

CAMTABLE | DATA

This parameter shows the table to define the electronic cams. The table has one parameter for each cam.

Parameter.	Meaning.
CAM n	Axis name.

CAMTABLE | DATA | CAM n

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 product documentation to obtain further information regarding the requirements and operation of the electronic cam.

PART 3.
PLC.

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 (CNCTIME=4) and a frequency of 2 cycles (PRGFREQ=2), the PLC executes the main module every 4×2=8 ms.

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 2 147 483 647 ms and never less than the loop time (parameter CNCTIME). 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, etc).
3. The PLC executes the main program (PRG).
4. The PLC updates the internal CNC variables (EMERGEN, STOP, FEEDHOL, etc) 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-M10240).
- Messages (MSG1-MSG1024).
- Errors (ERR1-ERR1024)
- Clocks (CLK).
- Registers (R1-R4096).
- 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 358.

Local inputs (LI1-LI16) and local outputs (LO1-LO8).

The central unit has 8 local digital inputs and 16 local digital outputs. 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-M10240).

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 M10240.

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-R4096).

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 R4096. 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 360.

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 370.

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, etc.
- Modifiable signals: _EMERGEN, _STOP, _FEEDHOL, etc.

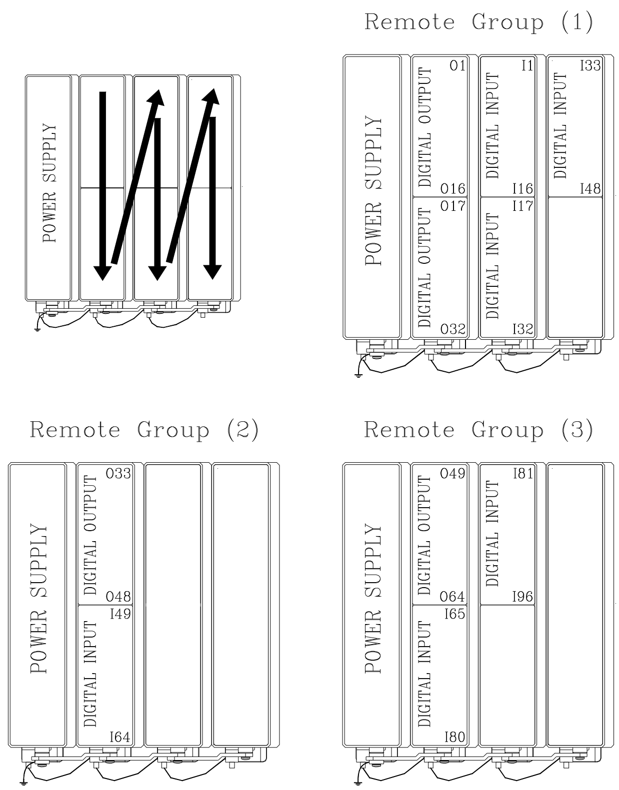
See chapter *"6 Logic CNC inputs and outputs."*.

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.

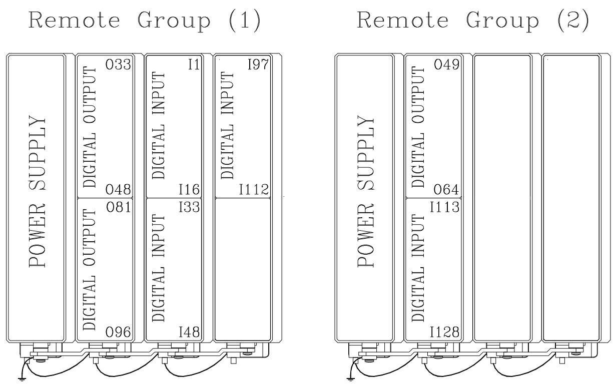


Group ·1·	Group ·2·	Group ·3·
O1...O32	O33...O48	O49...O64
I1...I48	I49...I64	I65...I96

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.

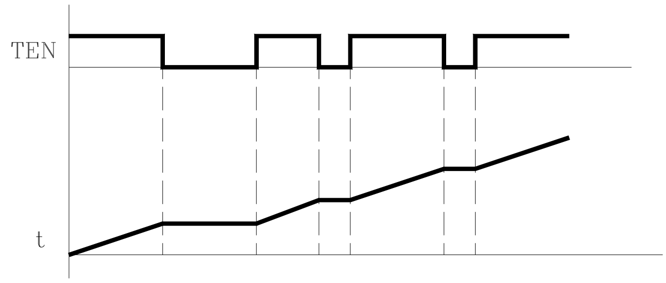
All the timers have a status output "T" and the inputs: TEN, TRS, TG1, TG2, TG3 and TG4. It is also possible to check, at any time, the elapsed time t since the timer was triggered.

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.



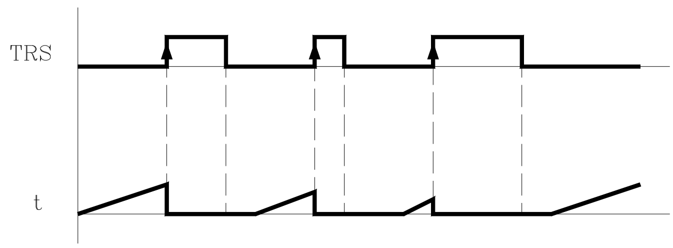
By default, every time a timer is triggered, the PLC sets this input high (=1).

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.

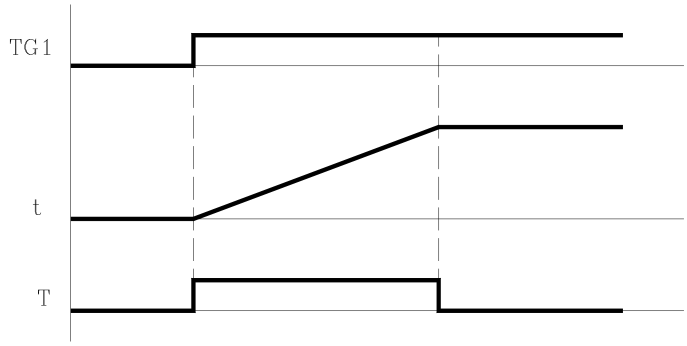
In arithmetic and comparison instructions, it refers to the elapsed time.

```
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.

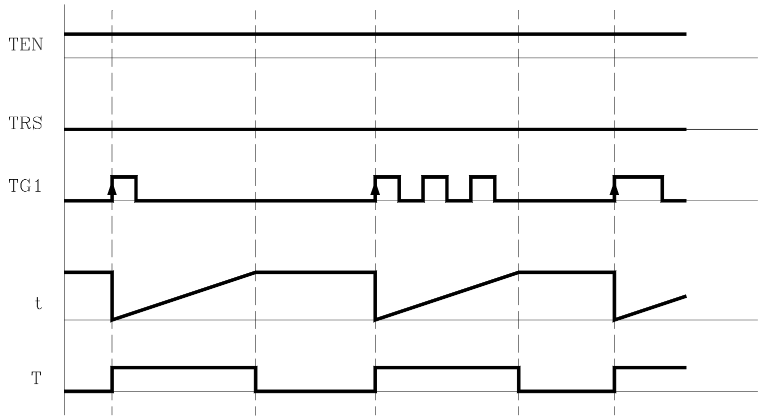
3.5.1 Monostable mode. TG1 input.

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.

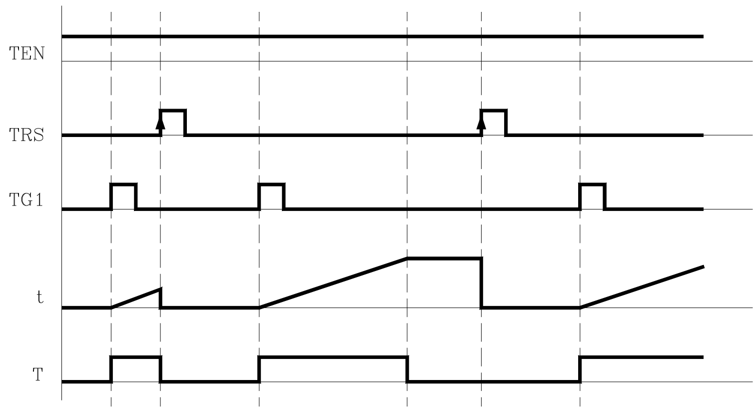


Any changes at the TG1 input (up or down-flank) while timing, has no effect.

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.

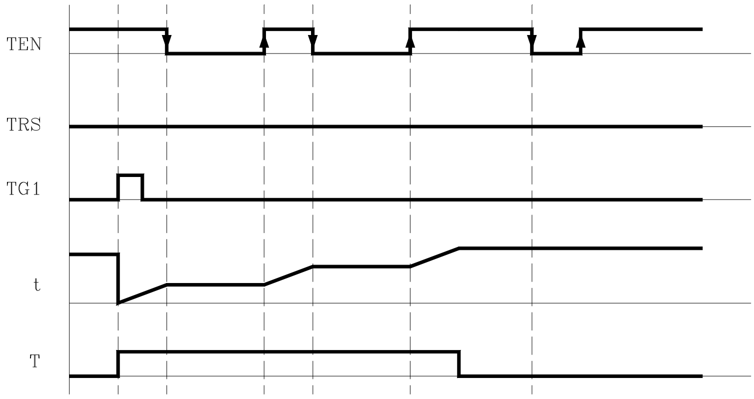
If an up-flank occurs at the TRS input while timing or after it, the PLC resets the timer setting its status output low (T=0) and resetting its timing (t=0).



Since the timer is reset, its trigger input must be activated again to turn it back on.

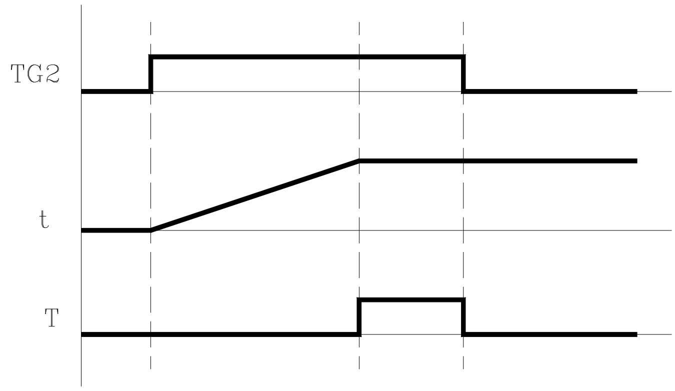
Operation of the TEN input in this mode.

If once the timer has been activated, TEN = 0, the PLC interrupts the timing and TEN must be set to "1" to resume timing.



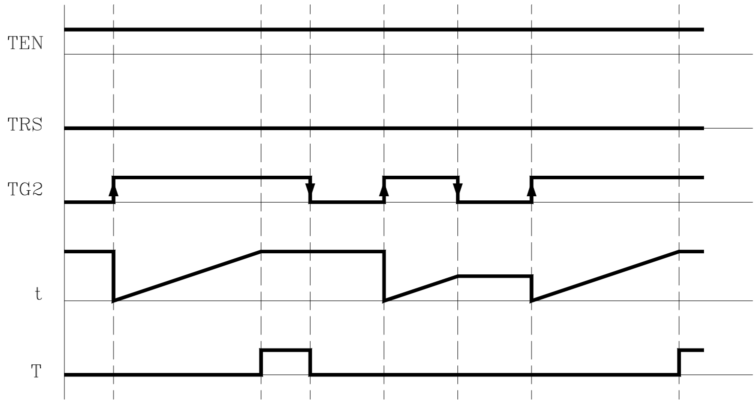
3.5.2 Delayed activation mode. TG2 input.

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.

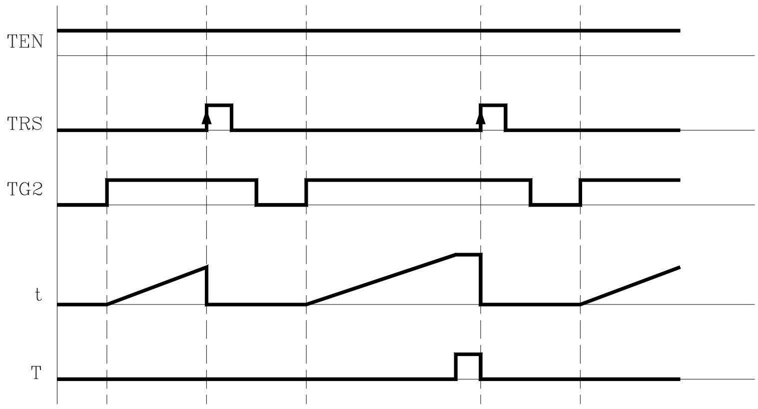


The elapsed time (t) is maintained until a new up-flank occurs at trigger input TG2.

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.

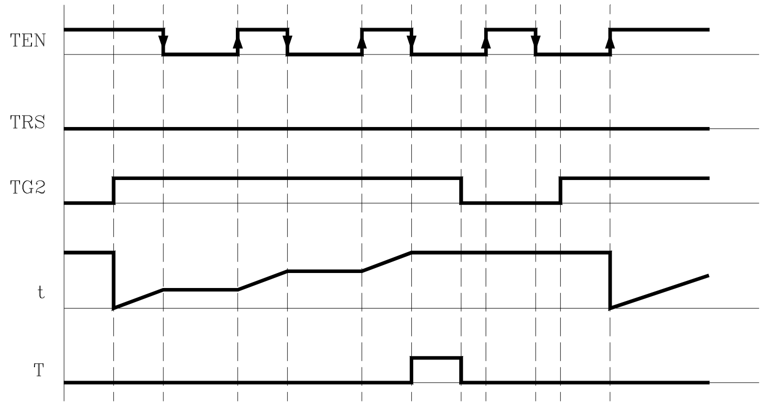
If an up-flank occurs at the TRS input while timing or after it, the PLC resets the timer setting its status output low (T=0) and resetting its timing (t=0).



Since the timer is reset, its trigger input must be activated again to turn it back on.

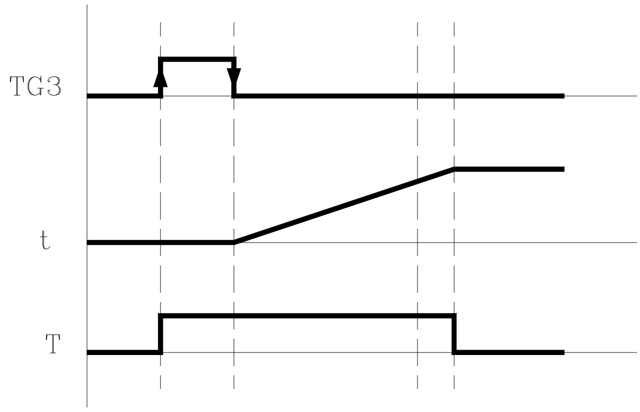
Operation of the TEN input in this mode.

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.

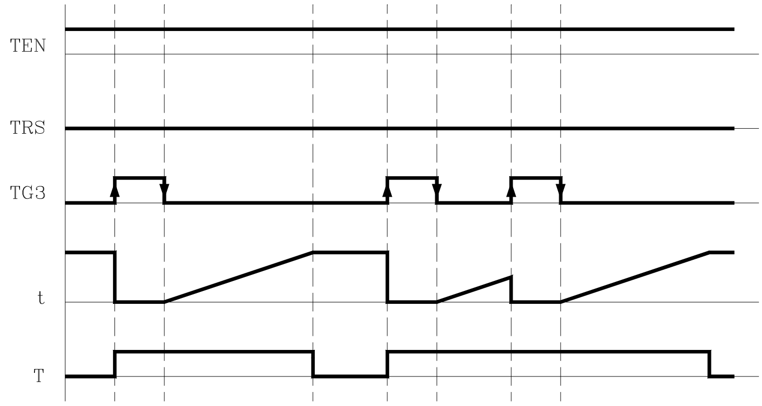
This operating mode is used to apply a delay between the deactivation of trigger input TG3 and that of the "T" output of the timer. The time delay is set by the time constant.



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).

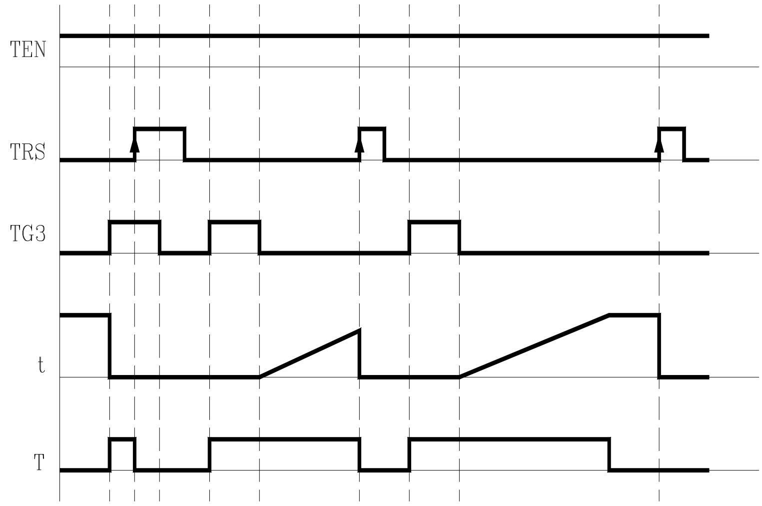


The elapsed time "t" is maintained until a new up-flank occurs at input TG3.

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.

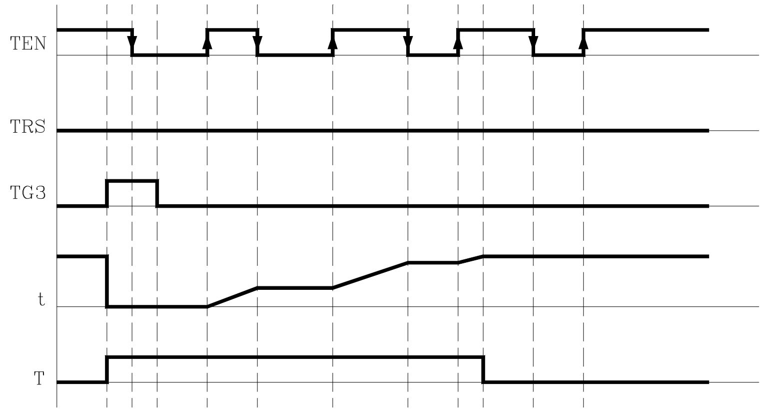
If an up-flank occurs at the TRS input while timing or after it, the PLC resets the timer setting its status output low (T=0) and resetting its timing (t=0).



Since the timer is reset, its trigger input must be activated again to turn it back on.

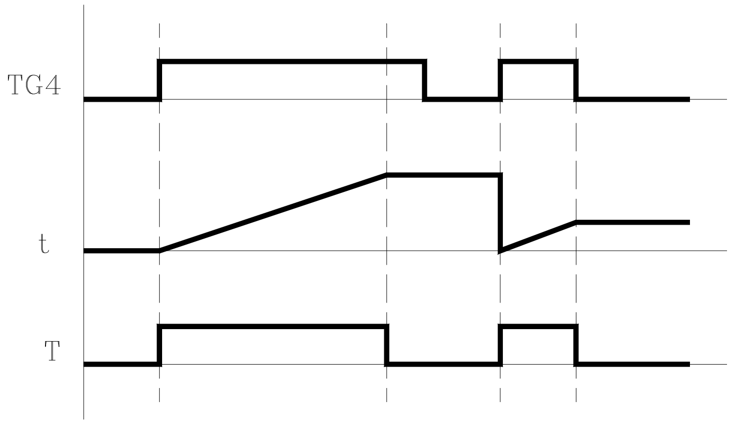
Operation of the TEN input in this mode.

If once the timer has been activated, TEN = 0, the PLC interrupts the timing and TEN must be set to "1" to resume timing.



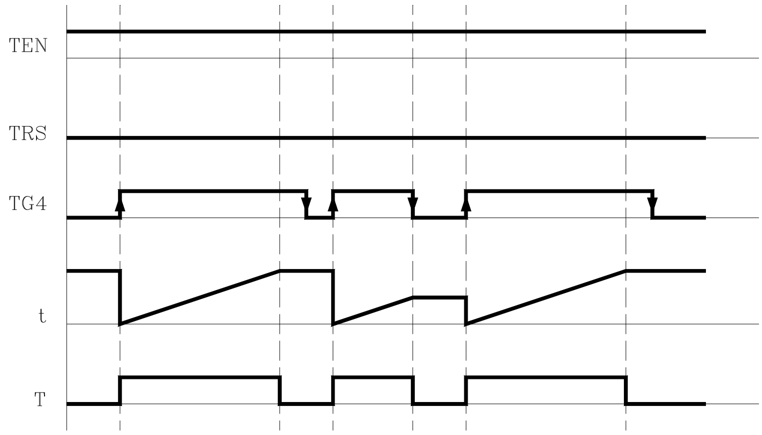
3.5.4 Signal limiting mode. TG4 Input.

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).



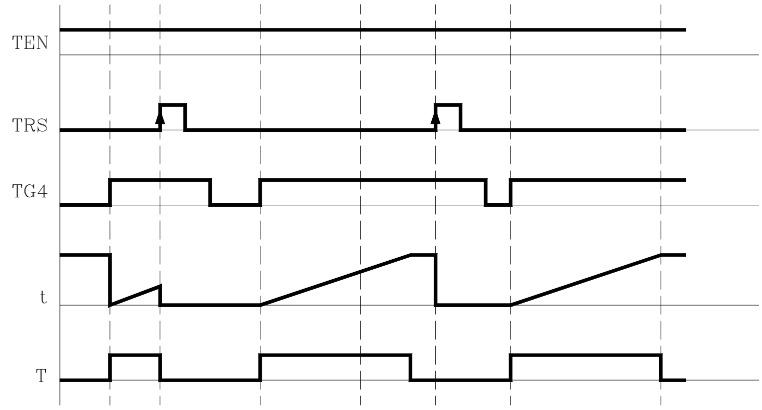
The elapsed time (t) is maintained until a new up-flank takes place at input TG4.

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.

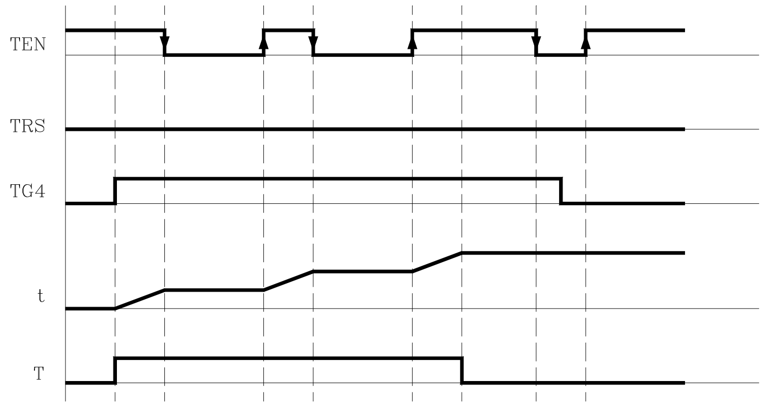
If an up-flank occurs at the TRS input while timing or after it, the PLC resets the timer setting its status output low (T=0) and resetting its timing (t=0).



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.

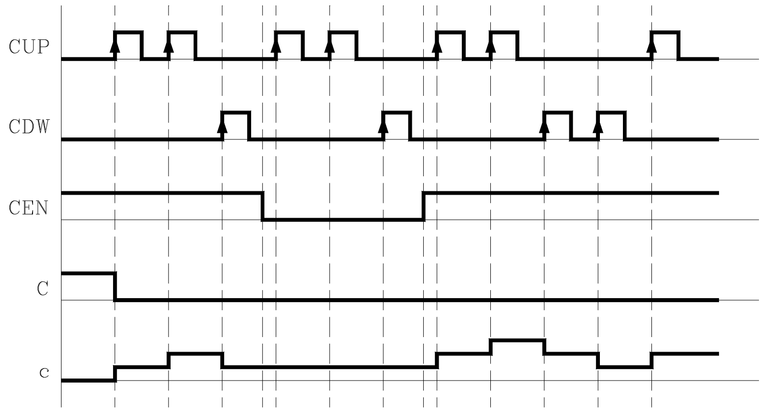
```
I3 = CDW 20
```
Every time an up-flank occurs at I3, the count of the C10 counter decreases one unit.

(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.

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 the directive instruction that defines it (CY1, PRG, PE) and end with the directing instruction END. See *"3.2 Modular structure of the PLC program."* on page 353.

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 = O2
```

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 directive allows you to define an unlimited number of symbols for use in the PLC program or in a PLC subroutine in C language.

PDEF The PDEF directing instruction can define up to 200 symbols and can be used in the PLC program, a PLC subroutine in C language, part program (using variables) or 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.

Every time the PLC compiles a program with subroutines in C language, it generates the file plc_pdef.h with the #defines of all the symbols defined in the PLC program. The file is saved in the folder ..\MTB\PLC\Project.

```
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-M10240)
- Messages (MSG1-MSG1024)
- Errors (ERR1-ERR1024)
- Clocks (CLK)
- Registers (R1-R4096)
- Register bits (B0-B31 R1-R4096)
- 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 [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··10240
	MSG1··1024
	ERR1··1024
	B0··31 R1··4096
	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··4096	EQ	R1··4096
	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 1/10240
= 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 1/10240
	MSG 1/1024
	ERR 1/1024
	B 0/31 R 1/4096
	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
	M1/10240	M1/10240			24
	MSG1/1024	MSG1/1024			20
	ERR1/1024	ERR1/1024			16
	T1/512	R1/4096			12
	C1/256	R CNC-PLC			8
	R1/4096				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).

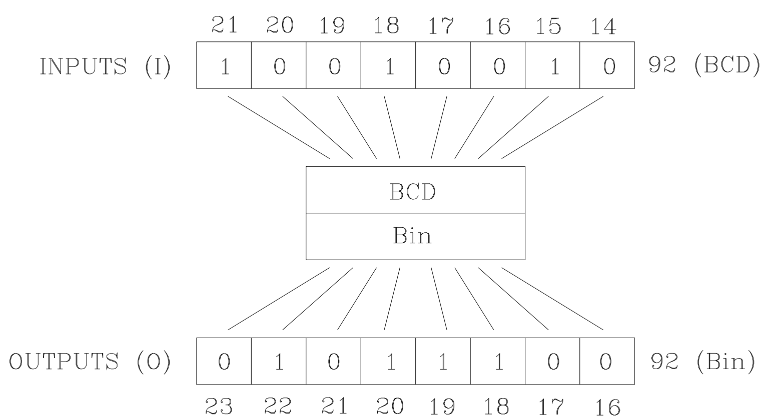
MOV	I12	M100	0032	binary to binary in 32 bits.
MOV	O21	R100	0012	binary to binary in 12 bits.
MOV	C22	O23	0108	binary to BCD in 8 bits.
MOV	T10	M112	1020	BCD to binary in 20 bits.

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.



4. PLC PROGRAMMING. Action instructions.

FAGOR AUTOMATION

CNCelite 8058 8060 8065 8070

REF: 2210

= NGU R 1/4096
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 1/4096
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	R1/4096	R1/4096	R1/4096
= 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	R1/4096	R1/4096	R1/4096
OR	R CNC-PLC	R CNC-PLC	R CNC-PLC
XOR	#	#	

Examples with R200 = B1001 0010 and R201 = B0100 0101.

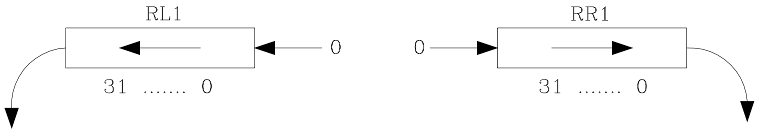
() = AND	R200	R201	R202	; R202 = B0
() = OR	R200	R201	R203	; R203 = B11010111
() = XOR	R200	R201	R204	; R204 = B11010111
() = AND	B1111	R201	R205	; R205 = B00000101
() = OR	R200	B1111	R206	; R206 = B10011111
() = XOR	B1010	B1110	R207	; R207 = B00000100

= 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	R1/4096	R1/4096	R1/4096
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
	M1/10240	1/10240
	MSG1/1024	1/1024
	ERR1/1024	1/1024
	T1/512	1/512
	C1/256	1/256
	R1/4096	1/4096

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	R1/4096	M1/10240
	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.

4.4.7 Instructions for reading and writing variables.

= CNCRD
= CNCWR
Reading (CNCRD) and writing (CNCWR) of CNC variables.

These actions allow the reading (CNCRD) and writing (CNCWR) of CNC variables from their mnemonic. The CNCRD action loads the contents of the variable into the register and the CNCWR action loads the contents of the 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).

The programming format is as follows:

```
CNCRD(variable, register, mark<,registerError>)
CNCWR(register, variable, mark<,registerError>)
```

Argument.	Meaning.
variable	Mnemonic of the variable.
log	Log in which the value read (CNCRDID) or the value to be written (CNCWRID) is stored.
mark	Flag indicating whether the read/write was successful (0) or unsuccessful (1).
registroError	Optional. If the read/write fails, this log stores the error number. If registroError is not programmed and the instruction generates an error, the CNC will display the error window.

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 variable mnemonic contains arrays, for example (V.)G.GUP[i], these may also be defined by an integer, a register or a symbol defined by DEF or PDEF.

```
... = MOV 153 R101
CNCWR (G.GUP[R101], R10, M1000)
DEF PARAM 153
CNCRD (G.GUP[PARAM], R10, M1000)
```

Error handling in the reading of variables.

The registroError argument sets the PLC behaviour on read errors.

- If registroError is not programmed, reading a non-existent variable (e.g. the dimension of an axis that does not exist), gives an error. 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".
- If registroError is programmed, the PLC does not give an error, the record takes the value 12345678 and registroError saves the generated error.

```
DFU B0KEYBD1 = CNCRD(A.POS.V, R100, M100)
    - If the V-axis does not exist, the PLC gives an error.
DFU B0KEYBD1 = CNCRD(A.POS.V, R100, M100, R200)
    - If the V-axis does not exist, the PLC does not give an error.
    - The R100 log takes the value 12345678.
    - The R200 log stores the error number; 46 (non-existent axis).
```

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.

= CNCRDID
= CNCWRID
Reading (CNCRD) and writing (CNCWR) of CNC variables.

These actions allow the reading (CNCRDID) and writing (CNCWRID) of CNC variables from their identifier. The CNCRDID action loads the contents of the variable into the register and the CNCWRID action loads the contents of the 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 CNCRDID to read the variables of the arithmetic parameters and those of the OEM, it returns the value multiplied by 10000 (float mode reading).

The programming format is as follows:

```
CNCRDID(idVar,channel,axisLog,idxArray1,idxArray2,register,mark<,registerError>)
CNCWRID(register,idVar,channel,axisLog,idxArray1,idxArray2,mark<,registerError>)
```

Argument.	Meaning.
idVar	Define associated with the variable. (number, define de in fagor_var.h, log).
channel	Channel programmed in the variable (0 if not relevant). (number, DEF/PDEF symbol, log).
ejeLog	Logical number of the axis programmed in the variable (0 if not relevant). (number, DEF/PDEF symbol, log).
idxArray1	First array index of the variable (-1 if not relevant). (number, DEF/PDEF symbol, log).
idxArray2	Second array index of the variable (-1 if not relevant). (number, DEF/PDEF symbol, log).
log	Log in which the value read (CNCRDID) or the value to be written (CNCWRID) is stored.
mark	Flag indicating whether the read/write was successful (0) or unsuccessful (1).
registroError	Optional. If the read/write fails, this log stores the error number. If registroError is not programmed and the instruction generates an error, the CNC will display the error window.

Syntax of the variables in commands CNCRD and CNCWR.

The variable identifiers are defined in the ..\Fagor\PLC\inc_sys\fagor_var.h. file. The channel number, logical axis and variable arrays are defined by arguments. These arguments may be an integer, a register or a symbol defined by DEF or PDEF.

```
CNCRDID (1404,0,0,0,0,R10,M1000)
    (Reading of the FREAL variable)
... = MOV 2 R12
CNCRDID (1404,R12,0,0,0,R10,M1000)
    (Reading of the variable FREAL from channel 2)
```

Error handling in the reading of variables.

The registroError argument sets the PLC behaviour on read errors.

- If registroError is not programmed, reading a non-existent variable (e.g. the dimension of an axis that does not exist), gives an error. 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".
- If registroError is programmed, the PLC does not give an error, the record takes the value 12345678 and registroError saves the generated error.

```
DFU B0KEYBD1 = CNCRDID(1827,0,5,0,0,R100,M100)
    - If the V-axis does not exist, the PLC gives an error.
DFU B0KEYBD1 = CNCRDID(1827,0,5,0,0,R100,M100,R200)
    - If the V-axis does not exist, the PLC does not give an error.
    - The R100 log takes the value 12345678.
    - The R200 log stores the error number; 46 (non-existent axis).
```

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 = CNCRDID (1617,0,0,0,0,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> = CNCRDID (1404,0,0,0,0,R12,M12)
  CPS R12 GT 2000 = ...
  ```

  No need to wait before consulting the data because synchronous variables are resolved immediately.

4.4.8 Instructions for executing a CNC block.

= 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 MOVEABS, MOVEADD and MOVEINF 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)
```

4.4.9 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.

4.4.10 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.

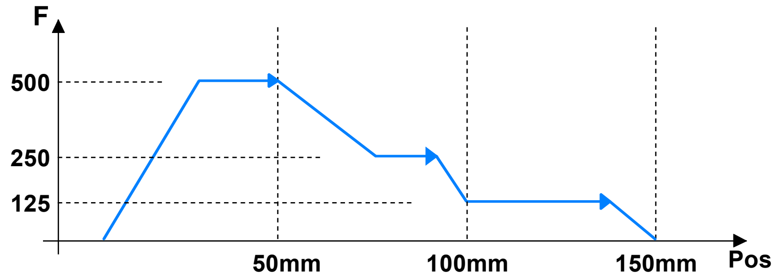
Moving direction. It is used with MOVE INF to execute an endless (infinite) movement until the axis limit is reached or until the movement is interrupted.

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)
```



4.4.11 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. Optional parameters are indicated between the <> characters.

```
FOLLOW ON (master, slave, <Nnratio>, <Ddratio>, synctype)
TFOLLOW ON (master, slave, <Nnratio>, <Ddratio>, synctype)
FOLLOW OFF (slave)
```

Parameter.	Meaning.
master	Name of the master axis.
slave	Name of the slave axis.
Nnratio	Optional. Numerator of the gear ratio. Turns of the slave axis.
Ddratio	Optional. 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, N3, D1, VEL)
TFOLLOW ON (A1, Z, 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.

4.4.12 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	
#SELECT PROBE [1] G100 X100	·· = TOUCHPROBE (X,1)	Wrong.
#SELECT PROBE [1] G100 X100	·· = TOUCHPROBE (Z,1)	Wrong.
#SELECT PROBE [1] G100 X100	·· = TOUCHPROBE (X,2)	Right.
#SELECT PROBE [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 Considerations for some functions.

4.5.1 Work zones.

The work zones may be defined, enabled and disabled from the PLC using the command CNCWR, writing the corresponding variables.

Variable.	Meaning.
[ch].A.ZONELIMITTOL.xn	Safety distance of the limits of the work zones.
[ch].G.ZONEST[k]	Status of work zone [K]. (0=Disabled zone). (1=Work zone enabled as no-entry). (2=Work zone enabled as no-exit).
[ch].A.ZONELOWLIM[k].xn	Lower limit of zone [k].
[ch].A.ZONEUPLIM[k].xn	Upper limit of zone [k].
[ch].G.ZONECIRAX1[k]	Logic axis corresponding to the first coordinate of the center of zone [k].
[ch].G.ZONECIRAX2[k]	Logic axis corresponding to the second coordinate of the center of zone [k].
[ch].G.ZONECIR1[k]	Center coordinate of zone [k] along the first axis that defines the circular zone.
[ch].G.ZONECIR2[k]	Center coordinate of zone [k] along the second axis that defines the circular zone.
[ch].G.ZONER[k]	Radius of zone [k] (circular zone).
[ch].G.ZONETOOLWATCH[k]	Monitor the tool tip or the tool base. (0=Monitor the tool tip). (1=Monitor the tool base). (2=Monitor the tool tip and the tool base).

Syntax of the variables.

·ch· Channel number.

·k· Zone number.

·xn· Name, logic number or index of the axis.

[2].G.ZONEST[1]	Channel ·2·. Zone 1.
A.ZONEUPLIM[1].Z	Z axis. Zone 1.
A.ZONEUPLIM[1].4	Axis with logic number ·4·. Zone 1.
[2].A.ZONEUPLIM[1].1	Axis with index ·1· in the channel ·2·. Zone 1.

Example of how to activate a work zone from the PLC.

The following example shows how to define a forbidden zone of the X axis the area between position 0 and 100 mm (1000000 tenth-thousands of a mm).

```
<condition> = MOV 0 R1 = CNCWR(R1, A.ZONELOWLIM[1].X, M1)
          = MOV 1000000 R1 = CNCWR(R1, A.ZONEUPLIM[1].X, M1)
          = MOV 1 R1 = CNCWR(R1, G.ZONEST[1], M1)
```

The coordinates will be given in ten-thousandths if they are in mm or hundred-thousandths if they are in inches.

Synchronize changes in the zones from the part-program and from the PLC.

When making changes in the work zone from the PLC while executing a part-program, it must be borne in mind that the CNC has already prepared the program blocks immediately after the one being executed; therefore, checking the new limits and/or the new status may not be correct for some blocks after the change. In order to ensure proper synchronization of the changes between the part-program and the PLC, these changes should be executed via interruption subroutines.

4.6 Summary programming commands.

Resources available at the PLC.

Inputs (I1··I1024)

Outputs (O1··O1024)

Marks (M1··M10240)

Message marks (MSG1··MSG1024)

Error marks (ERR1··ERR1024)

Clocks (CLK)

CNC-PLC marks

Timers (T1··T512)

Counters (C1··C256)

Registers (R1··R4096)

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··4096).

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··10240	Marks.
MSG1··1024	Messages.
ERR1··1024	Errors.
T1··512	Timers (status).
C1··256	Counters (status).
B0··31 R1··4096	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 1/10240
= 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··4096	Complements the bits of a register.
= NGS R1··4096	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 internal variables (mnemonic).
=CNCRDID	Reading internal variables (identifier).
=CNCWR	Writing internal mnemonic variables.
=CNCWRID	Writing internal variables (identifier).
= 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:

- The control of logic inputs and outputs from the CNC by means of an exchange of information between both systems, which is done periodically and by means of 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 418.

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 can have two channels and both channels can execute a part program in parallel with the others. 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 three spindles, it is possible to program in the same block 6 non-spindle-related M functions, the startup of all three 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.

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* - MFUNC2*. 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.

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 functions and 7 H functions can be programmed in the same block. The treatment of the auxiliary H functions is similar to that of 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* command 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 418.

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 can have two channels and both channels can execute a part program in parallel with the others. 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.

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* - HFUNC2*. 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.

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 418.

5.3.1 Special considerations with the multi-spindle option and channels.

The CNC may have three 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 SFUN3. 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
1000	550	$FFFFFFFF

Commands SP1FUN* - SP3FUN*. 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.

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.

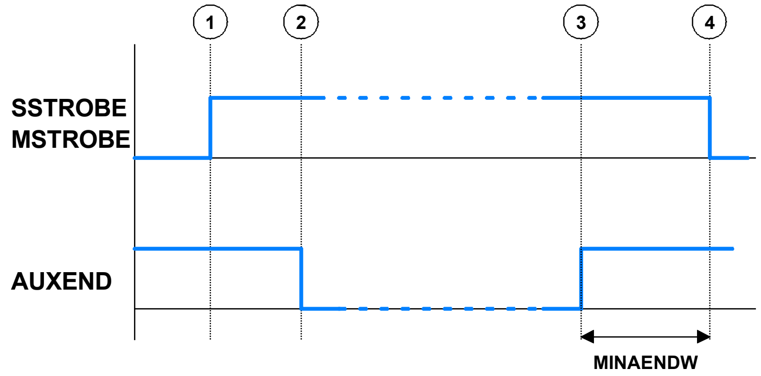
The CNC assigns the values of the S programmed in each spindle to registers SFUN1 through SFUN3.

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 SFUN3 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.

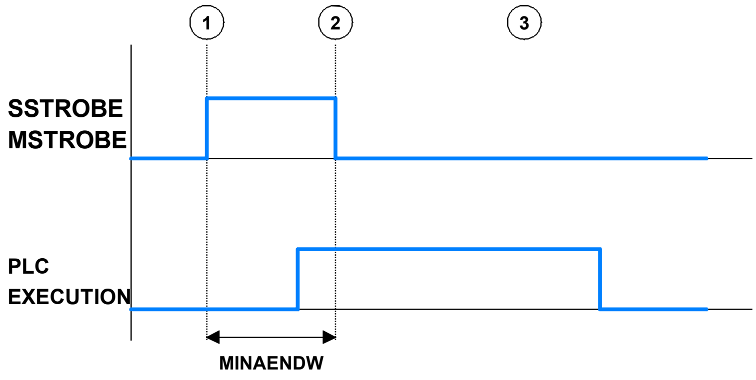
3 After selecting the requested S, the PLC must activate the AUXEND mark to let the CNC know that the execution has ended.

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.

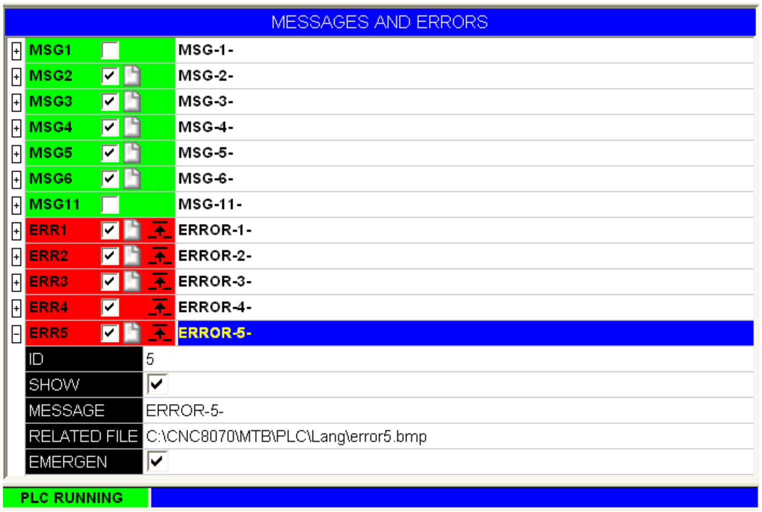
5.5 Displaying PLC errors and messages.

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.

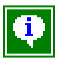


For more information on how to edit this table, refer to the operating manual.

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.

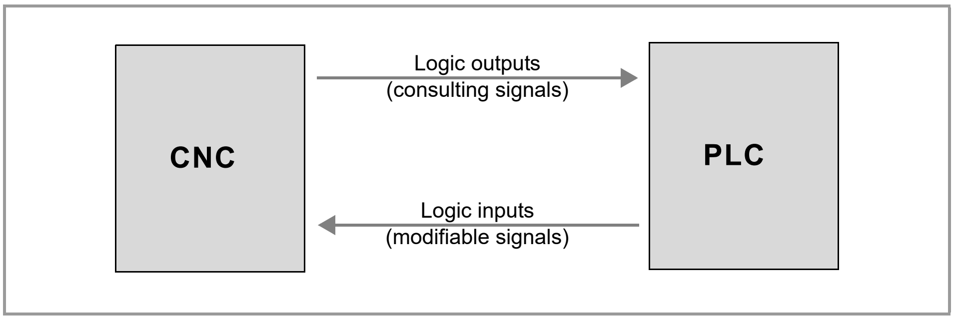
For example, the "associated file" will be defined as follows.

```
C:\FagorCNC\MTB\PLC\LANG\OEM.txt#27
```

LOGIC CNC INPUTS AND OUTPUTS.

6

The CNC's physical inputs and outputs are the set of system inputs and outputs governed by the PLC and communicate with external units through CNC connectors, remote modules, etc. The CNC also has a number of logic inputs and outputs to exchange information with the PLC and this way the PLC can access specific internal information from the CNC. The CNC sends information to the PLC through the logic outputs (consulting signals) and receives information from the PLC through the logic inputs (modifiable signals).



Identifying and activating signals.

Each of these logic inputs and outputs may be referred to using its associated mnemonic. Mnemonics are represented as follows.

Mnemoni.	Meaning.
CNCREADY	Mnemonic for system signals.
START STARTC1 STARTC2	Mnemonics for signals per channel, magazine, etc. Each channel and warehouse has its own mnemonic. In this case, the START and STARTC1 mnemonics are equivalent, but to make the program more comprehendible, it is recommended that START be used for a single-channel system and STARTC1 for a multi-channel system.
ENABLE(axis)	Mnemonics for axis and spindle signals; replace the text (axis) with the name of the spindle or the name or logical number of the axis. For example; ENABLEX, ENABLE2, ENABLES.

All the mnemonics refer to their associated variable, it being necessary to use the NOT operator to refer to its negation. For example, NOT CNCREADY, NOT _STOP, etc.

Activating the signals (active high/active low).

Mnemonics that begin with the "_" character indicate that the signal is active low (=0); the others are active high (=1).

Mnemoni.	Meaning.
_STOP	This mark is active low (=0).
CNCREADY	This mark is active high (=1).

6.1 General consulting signals.

_ALARM
_ALARMC1
..
_ALARMC4

This mark is active low (=0).
Each mark corresponds to a channel (_ALARM and _ALARMC1 are equivalent).

The CNC channel activates (=0) this mark when there is an alarm, an emergency or when the PLC activates an emergency signal (_EMERGEN). The CNC channel deactivates (=1) this mark when the warning has been removed and the cause of the alarm or emergency has been resolved. There is no digital output associated with this mark.

When moving in automatic mode, if the CNC detects a collision in a positioning or withdrawal movement inside a probing cycle (or even outside of it), it stops the movement, displays the corresponding error message and opens the emergency relay and activates (=0) this mark.

The following example shows how to associate the mark _ALARM to output O1.

```
_ALARM AND (rest of conditions) = O1
```

If there are no errors, output O1 will be high (=1).

ADVINPOS
ADVINPOSC1
..
ADVINPOSC4

This mark is active high (=1).
Each mark corresponds to a channel (ADVINPOS and ADVINPOSC1 are equivalent).

The CNC channel activates (=1) this mark a certain amount of time before the axes reach position. This anticipation time is defined in the parameter ANTIME. If the duration of the movement is less than the duration defined by the parameter ANTIME, the channel actives (=1) this mark immediately. If the ANTIME parameter has been set to 0, the mark will always be active.

This mark can be used for a punching press that uses an eccentric cam for its 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.

AUTOMAT
AUTOMATC1
..
AUTOMATC4

This mark is active high (=1).
Each mark corresponds to a channel (AUTOMAT and AUTOMATC1 are equivalent).

The CNC channel activates (=1) the mark when the automatic mode is selected.

BLKSEARCH
BLKSEARCHC1
..
BLKSEARCHC4

This mark is active high (=1).
Each mark corresponds to a channel (BLKSEARCH and BLKSEARCHC1 are equivalent).

The CNC channel activates (=1) this mark when the “block search” option is selected in automatic mode.

BUSESRDY

This mark is active high (=1).

This flag indicates the status of the buses (Sercos and EtherCAT). The CNC activates (=1) this flag when the Sercos (SERCOSRDY) and EtherCAT (ETHRDY) buses have been initialized correctly. The CNC disables (=0) this flag if either bus has not been initialized correctly.

CAXIS
CAXISC1
..
CAXISC4

This mark is active high (=1).
Each mark corresponds to a channel (CAXIS and CAXISC1 are equivalent).

The CNC channel activates (=1) this mark when the spindle is working as a C axis. This mark remains active when any of the functions #CAX, #FACE or #CYL are active.

CNCREADY

This mark is active high (=1).

This mark indicates the CNC status. The CNC deactivates (=0) this mark when the CNC is in an error state (red window status), otherwise it is active (=1). Include this mark in the PLC routine 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.
```

COLLISIONACTIV
COLLISIONACTIVC1
..
COLLISIONACTIVC4

This mark is active high (=1).
Each mark corresponds to a channel (COLLISIONACTIV and COLLISIONACTIV1 are equivalent).

The CNC activates (=1) this mark when the FCAS option is active. The CNC deactivates (=0) this mark when the FCAS option is not active. In order to keep the collision control active, apart from acting on the COLLISIONOFF mark, the machine configuration (xca file) must be loaded and the HD graphics must be selected. Collision control is not active if the STD graphics (Fagor standard graphics) are selected.

CSS
CSSC1
..
CSSC4

This mark is active high (=1).
Each mark corresponds to a channel (CSS and CSSC1 are equivalent).

The CNC channel activates (=1) this mark when constant surface speed is selected (G96).

DINDISTC1C2/DINDISTC1C3/DINDISTC1C4
DINDISTC2C1/DINDISTC2C3/DINDISTC2C4
DINDISTC3C1/DINDISTC3C2/DINDISTC3C4
DINDISTC4C1/DINDISTC4C2/DINDISTC4C3

This mark is active high (=1).
Each mark corresponds to a channel.

These marks are associated with the dynamic distribution of machining operations between channels (instruction #DINDIST), for the option of synchronized identical passes. Apart form using these marks, this option also uses those associated with synchronizing independent axes.

During the roughing operation of the cycle, the CNC channel activates (=1) 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 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.

The CNC channel deactivates (=0) these 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.

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.

DINDISTYPEC1
··
DINDISTYPEC4

This mark is active high (=1).
Each mark corresponds to a channel.

These marks are associated with the dynamic distribution of machining operations between channels (instruction #DINDIST) and indicates the active distribution type. The CNC activates (=1) the mark when the option of synchronized identical passes is active (#DINDIST[1]). The CNC deactivates (=0) the mark when the option for distribution of passes between channels is active (#DINDIST[0]).

DM00
DM00C1
··
DM00C4

This mark is active high (=1).
Each mark corresponds to a channel (DM00 and DM00C1 are equivalent).

The CNC channel indicates the status of the function M00 in these marks. The CNC activates (=1) the mark if the function is active, otherwise it deactivates (=0) it.

DM01
DM01C1
··
DM01C4

This mark is active high (=1).
Each mark corresponds to a channel (DM01 and DM01C1 are equivalent).

The CNC channel indicates the status of the M01 function in these marks. The CNC activates (=1) the mark if the function is active, otherwise it deactivates (=0) it.

DM02
DM02C1
··
DM02C4

This mark is active high (=1).
Each mark corresponds to a channel (DM02 and DM02C1 are equivalent).

The CNC channel indicates the status of the M02 function in these marks. The CNC activates (=1) the mark if the function is active, otherwise it deactivates (=0) it.

DM03
DM03SP1
··
DM03SP6

This mark is active high (=1).
Each mark corresponds to a spindle (DM03 and DM03SP1 are equivalent).

The CNC channel indicates the status of the M03 function of the spindle. The CNC activates (=1) the mark if the function is active, otherwise it deactivates (=0) it.

DM04
DM04SP1
··
DM04SP6

This mark is active high (=1).
Each mark corresponds to a spindle (DM04 and DM04SP1 are equivalent).

The CNC channel indicates the status of the M04 function of the spindle. The CNC activates (=1) the mark if the function is active, otherwise it deactivates (=0) it.

DM05
DM05SP1
··
DM05SP6

This mark is active high (=1).
Each mark corresponds to a spindle (DM05 and DM05SP1 are equivalent).

The CNC channel indicates the status of the M05 function of the spindle. The CNC activates (=1) the mark if the function is active, otherwise it deactivates (=0) it.

DM06
DM06C1
··
DM06C4

This mark is active high (=1).
Each mark corresponds to a channel (DM06 and DM06C1 are equivalent).

The CNC channel indicates the status of the function M06 in these marks. The CNC activates (=1) the mark if the function is active, otherwise it deactivates (=0) it.

DM08
DM08C1
··
DM08C4

This mark is active high (=1).
Each mark corresponds to a channel (DM08 and DM08C1 are equivalent).

The CNC channel indicates the status of the M08 function in these marks. The CNC activates (=1) the mark if the function is active, otherwise it deactivates (=0) it.

DM09
DM09C1
··
DM09C4

This mark is active high (=1).
Each mark corresponds to a channel (DM09 and DM09C1 are equivalent).

The CNC channel indicates the status of the M09 function in these marks. The CNC activates (=1) the mark if the function is active, otherwise it deactivates (=0) it.

DM19
DM19SP1
··
DM19SP6

This mark is active high (=1).
Each mark corresponds to a spindle (DM19 and DM19SP1 are equivalent).

The CNC channel indicates the status of the M19 function of the spindle. The CNC activates (=1) the mark if the function is active, otherwise it deactivates (=0) it.

DM30
DM30C1
..
DM30C4

This mark is active high (=1).
Each mark corresponds to a channel (DM30 and DM30C1 are equivalent).

The CNC channel indicates the status of the M30 function in these marks. The CNC activates (=1) the mark if the function is active, otherwise it deactivates (=0) it.

DM41
DM41SP1
..
DM41SP6

This mark is active high (=1).
Each mark corresponds to a spindle (DM41 and DM41SP1 are equivalent).

The CNC channel indicates the status of the M41 function of the spindle. The CNC activates (=1) the mark if the function is active, otherwise it deactivates (=0) it.

DM42
DM42SP1
..
DM42SP6

This mark is active high (=1).
Each mark corresponds to a spindle (DM42 and DM42SP1 are equivalent).

The CNC channel indicates the status of the M42 function of the spindle. The CNC activates (=1) the mark if the function is active, otherwise it deactivates (=0) it.

DM43
DM43SP1
..
DM43SP6

This mark is active high (=1).
Each mark corresponds to a spindle (DM43 and DM43SP1 are equivalent).

The CNC channel indicates the status of the M43 function of the spindle. The CNC activates (=1) the mark if the function is active, otherwise it deactivates (=0) it.

DM44
DM44SP1
..
DM44SP6

This mark is active high (=1).
Each mark corresponds to a spindle (DM44 and DM44SP1 are equivalent).

The CNC channel indicates the status of the M44 function of the spindle. The CNC activates (=1) the mark if the function is active, otherwise it deactivates (=0) it.

DMCPWRHOLDC1
..
DMCPWRHOLDC4

This mark is active high (=1).
Each mark corresponds to a channel.

With the DMC active, the CNC channel activates (=1) this mark if all the power values read over 2 seconds exceed the power to achieve multiplied by the DMCPOWERFACTOR parameter.

DMCPWRPEAKC1
..
DMCPWRPEAKC4

This mark is active high (=1).
Each mark corresponds to a channel.

With the DMC active, the CNC channel activates (=1) this mark if the power read exceeds the power to achieve (target power + no-load power) in the defined percentage in the DMCPEAKSIZE parameter.

ETHRDY

This mark is active high (=1).

This mark indicates the status of the EtherCAT bus. The CNC activates (=1) this flag when the bus has been initialized correctly. The CNC deactivates (=0) this flag if there is a communication error, or when resetting the bus.

FHOUT
FHOUTC1
..
FHOUTC4

This mark is active high (=1).
Each mark corresponds to a channel (FHOUT and FHOUTC1 are equivalent).

The CNC channel activates (=1) this mark when the execution of the part program is interrupted, either by a stop (_STOP) or by a feed-hold (_FEEDHOL) by the PLC.

FOCUS1CH
..
FOCUS8CH

Each register corresponds to an HMI.

In a system where there are several HMIs, these registers indicate the active channel for each one. FOCUS1CH corresponds to the first HMI, FOCUS2CH to the second and so on.

The FOCUSnCH and KEYBDnCH registers always assign a keyboard to the active HMI channel.

```
() = MOV FOCUS1CH KEYBD1CH
    Keyboard 1 is always assigned to the active channel of HMI 1.
```

FREEC1
..
FREEC4

This mark is active high (=1).
Each mark corresponds to a channel.

The CNC channel activates (=1) this mark to indicate to the PLC that it is ready to accept a new block, sent using the CNCEX command.

FWAITLIM

This mark is active high (=1).

The PLC activates (=1) this mark when it stops the execution of the program due to a #FOLLOW[WAITINLIMIT], so that the programmed position falls within the limits (laser model).

HFUN1..HFUN7
HFUN1C1..HFUN7C1
..
HFUN1C4..HFUN7C4

There are seven registers for each channel (HFUN1/MFUN7 and HFUN1C1/MFUN7C1 are equivalent); one register for each H function.

The CNC channel uses these registers to indicate to the PLC the auxiliary H functions selected for execution. Each channel can have up to 7 H functions in a block. If all the registers are not used, the CNC assigns $FFFFFFFF to the unused ones (those with the highest numbers).

If functions H10 and H20 are programmed in the first channel and functions H30 and H40 in the second channel, the CNC will transfer the following data.

```
HFUN1C1=10
HFUN2C1=20
HFUN3C1=$FFFFFFFF
...
HFUN7C1=$FFFFFFFF

HFUN1C2=30
HFUN2C2=40
HFUN3C2=$FFFFFFFF
...
HFUN7C2=$FFFFFFFF
```

If the H50 function is subsequently executed on the first channel, the CNC will send the following information to the PLC.

```
HFUN1C1=50
HFUN2C1=$FFFFFFFF
HFUN3C1=$FFFFFFFF
...
HFUN7C1=$FFFFFFFF
```

Commands HFUNC1* - HFUNC2*. 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.

HFUNC1*	For channel 1. They can also be programmed as HFUN*.
HFUNC2*	For channel 2.

Example for detecting H10 in channel 1.

```
CPS HFUNC1* EQ 10 = (action)
```

If the H10 function has been programmed, the command HFUNC1* will return a "1", otherwise it will be "0".

HSTROBE
HSTROBEC1
..
HSTROBEC4

This mark is active high (=1).
Each mark corresponds to a channel (HSTROBE and HSTROBEC1 are equivalent).

The CNC channel activates (=1) this mark high to indicate to the PLC that it must execute the auxiliary H functions indicated in registers HFUN1 through HFUN7.

INCYCLE
INCYCLEC1
..
INCYCLEC4

This mark is active high (=1).
Each mark corresponds to a channel (INCYCLE and INCYCLEC1 are equivalent).

The CNC activates (=1) this mark when executing a block or moving an axis.

- During the execution of a program, the CNC channel activates (=1) this mark at the start of the execution and it is kept until the CNC completes the part program, until the [STOP] key on the operator panel is pressed or the mark _STOP is activated (=0).
- In MDI/MDA mode or in block-by-block execution mode, the CNC channel deactivates (=0) this mark after the block execution has completed.
- In manual mode, by jogging the axes, the CNC channel keeps this mark active (=1) while holding any of these keys.

INPOSI
INPOSIC4
··
INPOSIC2

This mark is active high (=1).
Each mark corresponds to a channel (INPOSI and INPOSIC1 are equivalent).

The CNC channel activates (=1) this mark when all its active axes and spindles have reached their position, with the exception of the independent axes programmed from the PLC. The INPOSI mark remains active during the movement of independent axes.

An axis is in position when it stays within the in-position zone (window) (parameter INPOSW) for a time period indicated by parameter INPOSTIME.

INTEREND
INTERENDC1
··
INTERENDC4

This mark is active high (=1).
Each mark corresponds to a channel (INTEREND and INTERENDC1 are equivalent).

The CNC channel activates (=1) this mark when the theoretical movement of the axes has been completed (when it no longer outputs the velocity command). The INTEREND mark, like the ADVINPOS mark, can be used to activate mechanisms before the axes reach their position.

KINCOMP1
··
KINCOMP6

This mark is active high (=1).

The PLC activates (=1) this mark when the kinematics compensation (COMPID parameter) is active. Each kinematic has an associated compensation (COMPID parameter). KINCOMP1 is associated with COMPID 1, KINCOMP2 with COMPID2 and so on. For example, if the first kinematic (#KIN ID[1]) has the third compensation (COMPID=3) associated with it, when this kinematic is activated, the PLC activates the KINCOMP3 flag.

MANUAL
MANUALC1
··
MANUALC4

This mark is active high (=1).
Each mark corresponds to a channel (MANUAL and MANUALC1 are equivalent).

The CNC channel activates (=1) the mark when the jog mode is selected.

MDI
MDIC1
··
MDIC4

This mark is active high (=1).
Each mark corresponds to a channel (MDI and MDIC1 are equivalent).

The CNC channel activates (=1) the mark when the MDI/MDA mode is selected. If MDI/MDA mode is selected from automatic mode, the channel keeps the AUTOMAT mark active. If MDI/MDA mode is selected from manual mode, the channel keeps the MANUAL mark active.

MFUN1..MFUN7
MFUN1C1..MFUN7C1
..
MFUN1C4..MFUN7C4

There are seven registers for each channel (MFUN1/MFUN7 and MFUN1C1/MFUN7C1 are equivalent); one register for each H function.

The CNC channel uses these registers to indicate to the PLC the auxiliary H functions selected for execution. Each channel can have up to 7 M functions in a block. If all the registers are not used, the CNC assigns $FFFFFFFF to the unused ones (those with the highest numbers).

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=100
MFUN2C1=135
MFUN3C1=$FFFFFFFF
...
MFUN7C1=$FFFFFFFF

MFUN1C2=88
MFUN2C2=75
MFUN3C2=$FFFFFFFF
...
MFUN7C2=$FFFFFFFF
```

If the M88 function is subsequently executed on the first channel, the CNC will send the following information to the PLC.

```
MFUN1C1=88
MFUN2C1=$FFFFFFFF
MFUN3C1=$FFFFFFFF
...
MFUN7C1=$FFFFFFFF
```

Commands MFUNC1* - MFUNC2*. 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*	For channel 1. They can also be programmed as MFUN*.
MFUNC2*	For channel 2.

Example for detecting M04 in channel 1.

```
CPS MFUNC1* EQ 4 = (action)
```

If the M04 function has been programmed, the command HFUNC1* will return a “1”, otherwise it will be “0”.

MMCWDG

This mark is active high (=1).

This mark indicates the status of the operating system. The CNC channel is activated (=1) when the operating system locks up and it is deactivated (=0) when the CNC operating system is working properly. It is recommended to include this mark in the PLC routine to enable the emergencies when the operating system locks up.

MSTROBE
MSTROBEC1
..
MSTROBEC4

This mark is active high (=1).
Each mark corresponds to a channel (MSTROBE and MSTROBEC1 are equivalent).

The CNC channel activates (=1) this mark high to indicate to the PLC that it must execute the auxiliary M functions indicated in registers MFUN1 through MFUN7 of the channel.

OVERTEMP

This mark is active high (=1).

This mark indicates the temperature status of the CNC. The mark will be deactivated (=0) when the CNC temperature is correct. The CNC checks the unit temperature every minute; if in three samples in a row where the temperature exceeds 60 ºC (140 ºF), the CNC activates (=1) this mark and prompts a W169 warning. 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 temperature of the unit does not exceed 65 ºC (149 ºF) and, if it does, it inhibits the [START] and prompts an E173 error. 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 CNC is at risk. Turn the CNC off to avoid possible damage.

PROBE
PROBEC1
..
PROBEC4

This mark is active high (=1).
Each mark corresponds to a channel (PROBE and PROBEC1 are equivalent).

The CNC channel activates (=1) this mark when executing a probing movement (G100).

PSWSET

This mark is active high (=1).

The CNC activates (=1) this mark when there is an OEM password.

PW1STAF
PW1STAS

This mark is active high (=1).

The CNC uses these marks to indicate the status of the first RPS (QC-RPS module) defined by the connection tree.

PW1STAF	PW1STAS	Meaning.
0	0	The RPS is in the initial state with no bus voltage; there is an error or there is no RPS.
0	1	The RPS is in the process of charging the DC power bus (softstart).
1	0	RPS powered at the DC Bus. The RPS can be enabled.
1	1	PRS enabled.

PW2STAF
PW2STAS

This mark is active high (=1).

The CNC uses these marks to indicate the status of the second RPS (QC-RPS module) defined by the connection tree.

PW2STAF	PW2STAS	Meaning.
0	0	The RPS is in the initial state with no bus voltage; there is an error or there is no RPS.
0	1	The RPS is in the process of charging the DC power bus (softstart).
1	0	RPS powered at the DC Bus. The RPS can be enabled.
1	1	PRS enabled.

RAPID
RAPIDC1
..
RAPIDC4

This mark is active high (=1).
Each mark corresponds to a channel (RAPID and RAPIDC1 are equivalent).

The CNC channel activates (=1) this mark when executing a rapid traverse movement (G0).

READYC1
..
READYC4

This mark is active high (=1).
Each mark corresponds to a channel.

This mark indicates the status of the CNC channel. The CNC channel deactivates (=0) this mark when the CNC is in an error state (red window status), otherwise it is active (=1).

RESETOUT
RESETOUTC1
..
RESETOUTC4

This mark is active high (=1).
Each mark corresponds to a channel (RESETOUT and RESETOUTC1 are equivalent).

CNC channel reset. The CNC channel activates (=1) this mark when the reset has completed. This mark stays active for a time period set in parameter MINAENDW. The channel uses this mark whenever performing a reset, either from the [RESET] key on the operator panel or when the PLC activates the RESETIN mark.

RETRACT
RETRACTC1
..
RETRACTC4

This mark is active high (=1).
Each mark corresponds to a channel (RETRACT and RETRACTC1 are equivalent).

The CNC uses this mark when it withdraws the axes from the part after interrupting a threading operation ([STOP] key or PLC mark _FEEDHOL). The CNC channel activates (=1) this mark when it begins to withdraw the axes and until it completes the movement.

The withdrawal of the axes is enabled in parameter RETRACTTHREAD and it is applied to electronic threading (G33/G34) and the threading canned cycles of the -T- model, both ISO and conversational.

RETRAENDC1
..
RETRAENDC4

This mark is active high (=1).
Each mark corresponds to a channel.

The CNC channel activates (=1) this mark when it 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 activates (=1) this mark to indicate to the PLC that all possible blocks have been executed and that it must deactivate (=0) its RETRACE mark.

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 also finalises after executing an M30 or a reset, or when the PLC deactivates (=0) the RETRACE mark of the channel.

RIGID
RIGIDC1
..
RIGIDC4

This mark is active high (=1).
Each mark corresponds to a channel (RIGID and RIGIDC1 are equivalent).

The CNC channel activates (=1) this mark when executing an rigid tapping block (G63). Any tapping (G84 in milling machines and G83 in lathes) set as rigid tapping activates this mark (=1) during the whole tapping operation, including the dwell at the bottom of the thread.

SBOUT
SBOUTC1
..
SBOUTC4

This mark is active high (=1).
Each mark corresponds to a channel (RAPID and SBOUTC1 are equivalent).

The CNC activates (=1) this mark when the "single block" mode is selected. If the channel is in automatic mode, the AUTOMAT mark will also be active (=1). If the channel is in manual mode, the manual mark will also be active (=1).

SERCOSRDY

This mark is active high (=1).

This mark indicates the status of the Sercos bus. The CNC activates (=1) this mark when the Sercos bus has been initialized properly. All the drive status consultation processes (DRSTAF (axis) and DRSTAS (axis)) must be conditioned by this mark.

SFUN1
...
SFUN6

Each register corresponds to a spindle.

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).

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=1000
SFUN2=550
SFUN3=$FFFFFFFF
```

SPN1C1..SPN7C1
..
SPN1C4..SPN7C4

There are seven registers for each channel; one register for every M function programmed in the channel.

The CNC 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 CNC assigns $FFFFFFFF to the unused ones (those with the highest numbers). If an M function is programmed in the block without a reference to the spindle, the PLC will assume it is the master spindle of the channel.

If the next block is programmed in the first channel ...

```
M3.S1 S1=1000 M4.S2 S2=500
; The spindle S1 turns clockwise at 1000 rpm.
; The spindle S2 turns counterclockwise at 500 rpm.
```

... the CNC will send the following information to the PLC.

```
SPN1C1=1
SPN2C1=2
SPN3C1=$FFFFFFFF
...
SPN7C1=$FFFFFFFF
MFUN1C1=3
MFUN2C1=4
MFUN3C1=$FFFFFFFF
...
MFUN7C1=$FFFFFFFF
```

Commands SP1FUN* - SP3FUN*. 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.

Example of determining whether the first spindle has received an M5 function from any channel.

```
CPS SP1FUN* EQ 5 = (action)
```

If the M5 function has been programmed, the command HFUNC1* will return a “1”, otherwise it will be “0”.

SSTROBE
SSTROBEC1
..
SSTROBEC4

This mark is active high (=1).
Each mark corresponds to a channel (SSTROBE and SSTROBEC1 are equivalent).

The CNC channel activates (=1) this mark 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.

START
STARTC1
..
STARTC4

This mark is active high (=1).
Each mark corresponds to a channel (START and STARTC1 are equivalent).

The CNC channel activates (=1) this mark when the [START] key is pressed on the operator panel. If the rest of the conditions are met (hydraulic, safety, etc.), the PLC must activate (=1) the CYSTART mark in order for the program to start running.

```
START AND (rest of conditions) = CYSTART
```

TANGACTIVC1
··
TANGACTIVC4

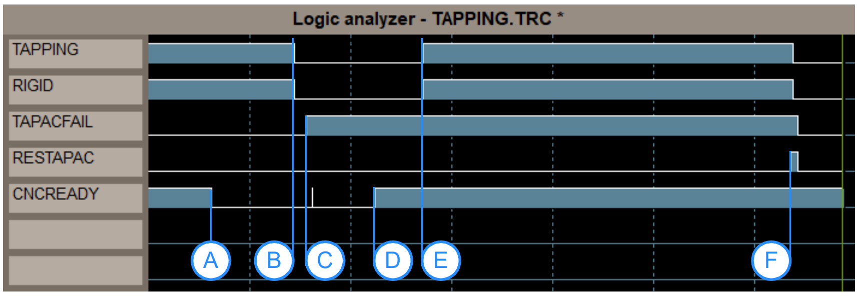
This mark is active high (=1).
Each mark corresponds to a channel.

The CNC channel activates (=1) this mark when the tangential control is active. This mark is not initialized when tangential control is frozen (suspended).

TAPACFAIL

This mark is active high (=1).

The CNC activates (=1) this marker when there is a tapping withdrawal pending, because the voltage has dropped or an error or a reset has occurred during the performance of a tapping operation. The CNC deactivates it (=0) when the tool removal process is finished (the user has pressed the softkey). This marker is also deactivated when the PLC activates the RESTAPAC marker.



(A) Run error during rigid tapping.
(B) CNC Reset.
(C) Error 3105. Tapping interrupted error.
(D) [SHIFT]+[RESET].
(E) Manual removal of the tool.
(F) "Completed" softkey.

TAPPING
TAPPINGC1
··
TAPPINGC4

This mark is active high (=1).
Each mark corresponds to a channel (TAPPING and TAPPINGC1 are equivalent).

The CNC channel activates (=1) this mark when executing a tapping canned cycle. The mark stays active during a possible programmed dwell 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.

Tapping set as rigid threads use the RIGID mark.

THREAD
THREADC1
··
THREADC4

This mark is active high (=1).
Each mark corresponds to a channel (THREAD and THREADC1 are equivalent).

The CNC channel activates (=1) this mark when executing an electronic threading block (G33).

WAITOUTC1
..
WAITOUTC4

This mark is active high (=1).
Each mark corresponds to a channel.

These marks are applied to channel synchronization. The CNC channel activates (=1) this mark 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.

ZERO
ZEROC1
..
ZEROC4

This mark is active high (=1).
Each mark corresponds to a channel (ZERO and ZEROC1 are equivalent).

The CNC channel activates (=1) this mark when searching home (G74).

6.2 Consulting signals for axes and spindles in closed loop (M19 or G63).

ACTFBACK(axis)

This mark is active high (=1).

The CNC uses this mark when the system has dual feedback (parameters SPEEDFBID and POSITIONFBID not equal). The CNC activates (=1) this mark when working with direct feedback (POSITIONFBID parameter) and deactivates (=0) it when working with motor feedback (SPEEDFBID parameter). The feedback type may be swapped from the PLC using the mark FBACKSEL(axis).

BRKFB(axis)

Not available since v2.00.01.
This mark is active high (=1).

Brake status The activation time of this mark must be at least 10 ms for the PLC to consider it as valid.

BRKOC(axis)

Not available since v2.00.01.
This mark is active high (=1).

The CNC deactivates (=0) this mark if the brake is powered. The CNC activates (=1) this mark when there is a short circuit in the brake output or if the power supply fails.

DIR(axis)

This mark is active high (=1).

The CNC activates (=1) this mark to indicate that the axis is moving in a negative direction and deactivates (=0) it when it moves in a positive direction. When the axis is stopped, it keeps the last value.

DRSTAF(axis)
DRSTAS(axis)

This mark is active high (=1).

The CNC uses these marks when communicating with the drive via Sercos and they indicate the drive's status.

DRSTAF(axis)	DRSTAS(axis)	Meaning.
0	0	The drive is turned off or it does not exist.
0	1	The internal check of the drive after powering up is correct. Drive without power at the DC bus. The drive cannot be enabled; but it is possible to provide power to the drives' power supply.
1	0	Drive with power at the DC bus. The drive is ready to provide torque. The drive may be enabled.
1	1	The drive is enabled. The “drive enable” and “speed enable” inputs are activated and the drive is working properly.

ENABLE(axis)

This mark is active high (=1).

The CNC activates (=1) this mark to move the axis, so that the PLC can enable the axis if necessary. This mark stays active until the axis gets in position, even if the theoretical movement has not finished; i.e. until the CNC activates the INPOS(axis) mark. If the PLC stops the movement of the axes (mark _FEEDHOL=0) the ENABLE(axis) signal stays active (=1).

The CNC also activates (=1) this mark for the movements of the independent axis, in closed loop spindle movements (for example a M19 or G74) or when a spindle enters a 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.

HIRTHON(axis)

This mark is active high (=1).

This mark is related to Hirth axes. The CNC activates (=1) this mark when the axis is working as a Hirth axis and deactivates (=0) it if the axis is working as a normal rotary or linear axis.

A Hirth axis is a rotary axis that can only be positioned at specific positions, every so many degrees. Both linear and rotary axes can be Hirth axes. The machine parameter HIRTH indicates whether the axis can work as a 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).

INPOS(axis)

This mark is active high (=1).

The CNC activates (=1) this mark when the corresponding axis or spindle is in position and there is no movement request (ENABLE(axis)=0). 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.

LOPEN(axis)

This mark is active high (=1).

The CNC activates (=1) this mark when the axis position loop is open.

LUBR(axis)

This mark is active high (=1).

This mark, together with the LUBRENA (axis) and LUBROK (axis) marks should be used to lubricate the axes. The CNC activates (=1) this mark when the corresponding axis or spindle must be lubricated. The parameter DISTLUBRI indicates the distance the axis must move before being lubricated.

MATCH(axis)

This mark is active high (=1).

The CNC activates (=1) this mark if the Hirth axis is properly positioned.

MAXDIFF(axis)

This mark is active high (=1).

This mark is used on gantry axes. The CNC activates (=1) 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 parameter MAXDIFF(axis).

PARK(axis)

This mark is active high (=1).

The CNC activates (=1) this mark when parking an axis or spindle. 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.

REFPOIN(axis)

This mark is active high (=1).

The CNC activates (=1) the mark when the axis or the spindle has been homed.

Total absolute feedback (parameter ABSFEEDBACK=Total).

The mark REFPOIN(axis) is always active (=1). A home search is not required.

Absolute feedback in one turn (parameter ABSFEEDBACK=One turn).

The CNC activates (=1) the mark REFPOIN(axis) during startup if the axis had been homed correctly during the previous session, the CNC had recovered the axis position saved during shutdown and the position of the encoder is the same as that of the shutdown position (difference of positions less than parameter MAXDIFREF). In this case, the CNC refers to the homed axis and it is not necessarily a new home search.

No absolute feedback (parameter ABSFEEDBACK=No).

The CNC deactivates (=0) this mark when starting up and activates (=1) it after completing the machine reference homing or after setting the machine coordinate (G174). The mark stays active until turning the CNC off. For axes and spindles without absolute feedback, the CNC deactivates (=0) this mark 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.

SETCHGALLOW(axis)

This mark is active high (=1).

The CNC activates this flag when the PLC changes the parameter set of an axis, which implies a motor change (different motor registration). The CNC deactivates this flag after updating and validating the parameters.

In a star-delta configuration change, include this flag in the PLC maneuver to disable the axis/spindle during the parameter change. Enable the spindle after the contactor change and when the CNC deactivates this flag.

SPDLZERO(axis)

This mark is active high (=1).

This flag indicates whether the spindle is performing a home search. This flag is activated when the spindle is enabled (ENABLE(axis)).

TANGACT(axis)

This mark is active high (=1).

The CNC activates (=1) this mark when the tangential control is activated. The CNC deactivates (=0) this mark when freezing (suspending) or canceling the tangential control.

UNPARK(axis)

This mark is active high (=1).

The CNC activates (=1) this mark when unparking an axis or spindle. 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.

6.3 Consulting signals for the spindle in open loop (M3/M4).

GEAROK
GEAROK1
..
GEAROK6

This mark is active high (=1).
Each mark corresponds to a spindle (GEAROK and GEAROK1 are equivalent).

The CNC activates (=1) 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 routine 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.

REVOK
REVOK1
..
REVOK6

This mark is active high (=1).
Each mark corresponds to a spindle (REVOK and REVOK1 are equivalent).

For an M3 and M4, the CNC activates (=1) this mark if the actual spindle revolutions match the ones programmed; that is, if they are between the percentages set by the parameters UPSPDLIM and LOSPDLIM. The REVOK signal may be used to handle the feed-hold signal and avoid machining at lower or higher rpm than desired.

For an M5, the mark is always active (=1).

With the spindle in closed loop (M19 or G63), the CNC deactivates (=0) this mark during the movements and activates (=1) it when the spindle is in position.

SYNCHRON1
..
SYNCHRON6

This mark is active high (=1).
Each mark corresponds to a spindle.

The CNC activates (=1) this mark for the slave spindle when a synchronization has commenced using the instruction #SYNC. When activating a synchronization, the CNC activates the ENABLE(axis) signal at both spindles and waits for the SERVO(axis)ON signal (if DWELL is used).

When a spindle synchronization is active, the PLCCNTL, INHIBIT(axis) 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.

SYNCHRONP1
..
SYNCHRONP6

This mark is active high (=1).
Each mark corresponds to a spindle.

The CNC activates this mark (=1) for the slave spindle at the beginning of a synchronization in position. 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.

SYNCMASTER1
..
SYNCMASTER6

This mark is active high (=1).
Each mark corresponds to a spindle.

The CNC activates (=1) this mark for the master spindle when a synchronization has commenced using the instruction #SYNC. When activating a synchronization, the CNC activates the ENABLE(axis) signal at both spindles and waits for the SERVO(axis)ON signal (if DWELL is used).

When a spindle synchronization is active, the PLCCNTL, INHIBIT(axis) 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.

SYNCPOSI1
..
SYNCPOSI6

This mark is active high (=1).
Each mark corresponds to a spindle.

The CNC activates (=1) this mark for the slave spindle when it is synchronized in position. The CNC deactivates (=0) this mark when the maximum error for the allowed position has been exceeded, whose default value is set by the machine parameter DSYNCPOSW.

SYNSPEED1
..
SYNSPEED6

This mark is active high (=1).
Each mark corresponds to a spindle.

The CNC activates (=1) this mark on the slave spindle when it is synchronized in speed. The CNC deactivates (=0) this mark when the maximum error for the allowed speed has been exceeded, whose default value is set by the machine parameter DSYNCVELW.

6.4 Consultation signals of the independent interpolator.

FOLLOW(axis)
This mark is active high (=1).

The independent interpolator of the axis actives (=1) this mark when the axis synchronization is active (#FOLLOW).

IBUSY(axis)
This mark is active high (=1).

The independent interpolator of the axis activates (=1) this mark when there is an instruction pending to be executed.

IEND(axis)
This mark is active high (=1).

The independent interpolator of the axis activates (=1) this mark when it is done generating the theoretical movement.

IFHOUT(axis)
This mark is active high (=1).

The independent interpolator of the axis activates (=1) this mark when the execution is interrupted.

IFREE(axis)
This mark is active high (=1).

The independent interpolator of the axis activates (=1) 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.

INSYNC(axis)
This mark is active high (=1).

For synchronization movements of an independent axis and of an electronic cam, the interpolator of the axis activates (=1) this mark when synchronism has been reached. This mark stays active while maintaining synchronism.

LATCH1ACTIVE(axis)
LATCH2ACTIVE(axis)
This mark is active high (=1).

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 (=1) 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)

This mark is active high (=1).

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 (=1) 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.

MOVCMD(axis)

This mark is active high (=1).

The independent interpolator of the axis (=1) activates this mark when a positioning has been executed (#MOVE).

PROBE1ACTIVE
PROBE2ACTIVE

This mark is active high (=1).

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 (=1) 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.

6.5 Consulting logic signals; laser.

ACTIVEMATERIALON

This mark is active high (=1).

The CNC activates (=1) this mark when there is an active material file (due to the execution of #MATERIAL, #TECHTABLE, activation from HMI or recovery during start up).

ADVCUTOFF

This mark is active high (=1).

The CNC activates (=1) this mark during the execution of microjoints, when the laser reaches the distance G.ADVCUTOFFW or the time G.ADVCUTOFFT before #CUTTING OFF. If both variables are programmed, the CNC activates the mark when the laser reaches both values. The CNC deactivates (=0) this mark when starting a G0 or at the end of interpolation. The PLC operations associated with the end of the profile must be associated with this mark (turn off the laser, close the gas outlet, etc).

```
DEF R_FULLSTATUS R201
DEF R_PWMON_CNC R202 ; PWM activated by CNC
;
DEF M_PWMON_CNC M516 ; PWM activated by CNC
DEF M_CYCLE_STOP M517 ; CYCLE STOP
DEF M_CNCRD_SYN M6000
;
;***** First cycle module *****
CY1
() = MOV 0 PWMDUTY
END
;
;******* Main module *******
PRG
() = CAL PWM
END
;
;******* Subroutines *******
SUB PWM
;
()= CNCRD(G.FULLSTATUS,R_FULLSTATUS,M_CNCRD_SYN)
  = CNCRD(G.PWMON,R_PWMON_CNC,M_CNCRD_SYN) ; PWM activated by CNC
;
CPS R_FULLSTATUS EQ 1029 ; Stopped
  = M_CYCLE_STOP ; Cycle stop
;
CPS R_PWMON_CNC EQ 1 ; PWM activated by CNC
  = M_PWMON_CNC ; PWM activated by CNC
;
M_CYCLE_STOP ; CYCLE STOP
  OR ADVCUTOFF ; Mark for microjoint
  OR M333
  = SET PWMON ; PWM control from PLC with PWMDUTY=0
;
(DFU INCYCLE ; In execution
  AND (PIERCING OR CUTTING)); Piercing or cutting subroutine
  OR (NOT M_CYCLE_STOP ; CYCLE STOP
  AND NOT ADVCUTOFF) ; Mark for microjoint
  = RES PWMON ; Reset PWM control from PLC
;
END
```

PIERCING

This mark is active high (=1).

Piercing enabled. The OEM must manage this variable in the subroutine associated with piercing (by default, Piercing.fst), to indicate to the PLC when to start (value 1) and end (valor 0) the piercing operation. The instruction #PLC makes it possible to manage a PLC mark from the part or subroutine program, without interrupting the block preparation.

This mark shows the state of the variable V.PLC.PIERCING. The OEM can also use this variable in the subroutine associated with the piercing (by default, Piercing.fst) to modify the status of this mark. Entering this variable interrupts the block preparation.

CUTTING

This mark is active high (=1).

Active cutting. The OEM must manage this mark in the subroutines associated with cutting (by default, Cutting.fst / Cuttingoff.fst), to indicate to the PLC when to begin (value 1) and end (valor 0) the cutting operation. The instruction #PLC makes it possible to manage a PLC mark from the part or subroutine program, without interrupting the block preparation.

This mark shows the state of the variable V.PLC.CUTTING. The OEM can also use this variable in the subroutines associated with cutting (by default, Cuttingon.fst / Cuttingoff.fst) to modify the status of this mark. Entering this variable interrupts the block preparation.

COMVARACT

This mark is active high (=1).

The CNC activates (=1) this mark when the user validates the common parameters of the active piercing or cutting table. The PLC should disable this mark when management with these variables is complete.

CUTVARACT

This mark is active high (=1).

The CNC activates (=1) this mark when the user validates the cutting parameters of the active table. The PLC should disable this mark when management with these variables is complete.

When the subroutine Cuttingon.fst changes only the cutting type (#CUTTING ON), the CNC does not activate this mark, as the synchronization with the PLC is implemented through M functions.

The subroutine Cuttingon.fst (or the one associated with the cutting process) performs a series of operations with the variables associated with the technological tables, operations that may also be required when the user modifies the table directly (which means modifying the variables). The PLC manages the effects of these variables on the machine devices (analog inputs, digital inputs, etc).

PIRVARACT

This mark is active high (=1).

The CNC activates (=1) this mark when the user validates the piercing parameters of the active table. The PLC should disable this mark when management with these variables is complete.

When the subroutine Piercing.fst changes only the piercing type (#PIERCING), the CNC does not activate this mark, as the synchronization with the PLC is implemented through M functions.

The subroutine Piercing.fst (or the one associated with the piercing process) performs a series of operations with the variables associated with the technological tables, operations that may also be required when the user modifies the table directly (which means modifying the variables). The PLC manages the effects of these variables on the machine devices (analog inputs, digital inputs, etc).

INPOSGAP

This mark is active high (=1).

The CNC activates (=1) this mark if the gap is within the range defined by the parameters GAPMIN-GAPMAX. If the gap exceeds the range set by these parameters, the CNC deactivates the INPOSGAP mark. The GAPERRORCANCEL parameter establishes the CNC behavior when the gap exceeds the GAPMIN - GAPMAX range.

GAPERRORCANCEL	Meaning.
Yes.	The CNC cancels the out-of-range gap error and does not stop the movement of the axes.
No	The CNC returns an out-of-range gap error and stops the movement of the axes, according to the braking ramp and controlling the gap during the ramp.

INTOL

This mark is active high (=1).

The CNC activates (=1) this mark if the gap is within the tolerance limit defined by the parameter GAPTOL regarding the gap value programmed. If the axis exceeds the tolerance set in the GAPTOL parameter, the CNC deactivates the INTOL mark. The GAPTOLCANCEL parameter establishes the CNC behavior when the gap exceeds the tolerance set in GAPTOL.

GAPTOLCANCEL	Meaning.
Yes.	The CNC cancels the out-of-range gap error and does not stop the movement of the axes.
No	The CNC returns an out-of-range gap error and stops the movement of the axes, according to the braking ramp and controlling the gap during the ramp.

INPOSLIMIT

This mark is active high (=1).

The CNC activates (=1) this mark when the leap programmed with a #LEAP reaches the highest point (POSLIMIT command). The CNC deactivates (=0) this mark when the leap starts to lower from this point. If the leap does not reach the highest point due to lack of space, the CNC will not activate this mark. In the event of any error, the CNC will deactivate this mark.

RESTARTPOINT

This mark is active high (=1).

The PLC activates (=1) this flag when the CNC executes a piercing or cutting after restarting from a breakpoint.

6.6 Tool manager consulting signals.

LEAVEPOS
LEAVEPOSMZ1
··
LEAVEPOSMZ4

Each register corresponds to a magazine (LEAVEPOS and LEAVEPOSMZ1 are equivalent).

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.

MZIDC1
··
MZIDC4

Each register corresponds to a channel.

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.

NEXTPOS
NEXTPOSMZ1
··
NEXTPOSMZ4

Each register corresponds to a magazine (NEXTPOS and NEXTPOSMZ1 are equivalent).

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.

TAKEPOS
TAKEPOSMZ1
··
TAKEPOSMZ4

Each register corresponds to a magazine (TAKEPOS and TAKEPOSMZ1 are equivalent).

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.

TMINEM
TMINEMZ1
··
TMINEMZ4

Each mark corresponds to a magazine (TMINEM and TMINEMZ1 are equivalent).

The CNC activates (=1) this mark to inform the PLC that an emergency has occurred at the tool manager.

TMOPERATION
TMOPERATIONC1
··
TMOPERATIONC4

Each register corresponds to a channel (TMOPERATION and TMOPERATIONC1 are equivalent).

This register indicates the type of operation to be carried out by the tool manager.

TMOPSTROBE
TMOPSTROBEC1
..
TMOPSTROBEC4

This mark is active high (=1).
Each mark corresponds to a channel (TMOPSTROBE and TMOPSTROBEC1 are equivalent).

The CNC channel activates (=1) this mark on to let the PLC know that it must execute the operation indicated by TMOPERATION.

TWORNOUT
TWORNOUTC1
..
TWORNOUTC4

Each mark corresponds to a channel (TWORNOUT and TWORNOUTC1 are equivalent).

The CNC channel activates (=1) this mark to "tell" the PLC that the tool has been rejected because it is worn out (real life > maximum life span).

6.7 Keystroke consulting signals.

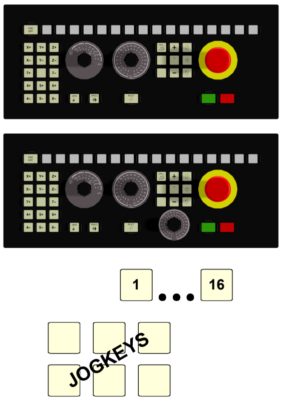
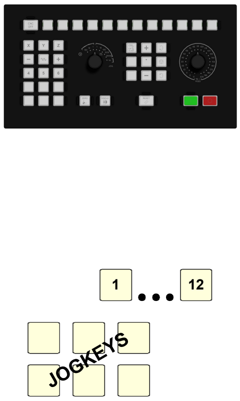
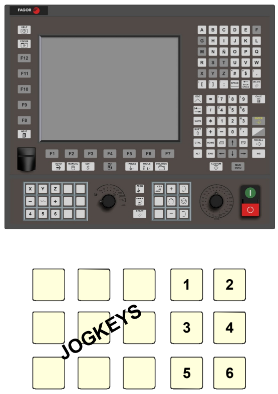
KEYBD1 / KEYBD2 / KEYBD3

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.

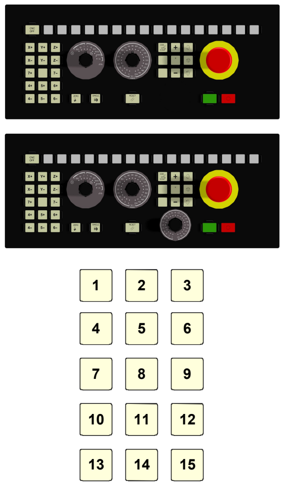
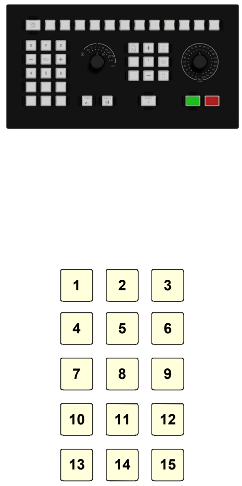
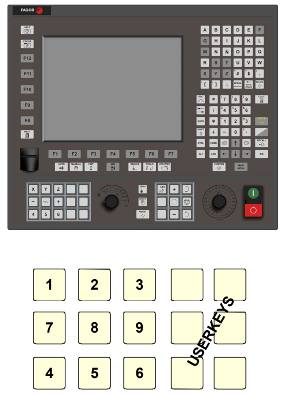
KEYBD1_1 / KEYBD2_1 / KEYBD3_1
..
KEYBD1_8 / KEYBD2_8 / KEYBD3_8

These registers indicate (bit=1) which key has been pressed on each operator panel. Registers KEYBD1_1 to KEYBD3_1 correspond to the first operator panel, KEYBD1_2 to KEYBD3_2 to the second one and so on.

Registers KEYBD1 / KEYBD1_1 to KEYBD1_8. User keys.

Bit.	OP-PANEL	OP-PANEL-329	QC-C65☐-10K
			
0	User key 1	User key 1	User key 1
1	User key 2	User key 2	User key 2
2	User key 3	User key 3	User key 3
3	User key 4	User key 4	User key 4
4	User key 5	User key 5	User key 5
5	User key 6	User key 6	User key 6
6	User key 7	User key 7	- - -
7	User key 8	User key 8	- - -
8	User key 9	User key 9	- - -
9	User key 10	User key 10	- - -
10	User key 11	User key 11	- - -
11	User key 12	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.	OP-PANEL	OP-PANEL-329	QC-C65☐-10K
			
16	Jog key 1	Jog key 1	Jog key 1
17	Jog key 2	Jog key 2	Jog key 2
18	Jog key 3	Jog key 3	Jog key 3
19	Jog key 4	Jog key 4	Jog key 4
20	Jog key 5	Jog key 5	Jog key 5
21	Jog key 6	Jog key 6	Jog key 6
22	Jog key 7	Jog key 7	Jog key 7
23	Jog key 8	Jog key 8	Jog key 8
24	Jog key 9	Jog key 9	Jog key 9
25	Jog key 10	Jog key 10	- - -
26	Jog key 11	Jog key 11	- - -
27	Jog key 12	Jog key 12	- - -
28	Jog key 13	Jog key 13	- - -
29	Jog key 14	Jog key 14	- - -
30	Jog key 15	Jog key 15	- - -
31	- - - -	- - -	- - -

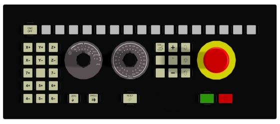
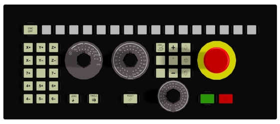
Registers KEYBD2 / KEYBD2_1 to KEYBD2_8.

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.	Key.
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

Bits.					Selector position.
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).

Bits.				Selector position.
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.

Bits.					Selector position.
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 %

KEYHBLS1
KEYHBLS2

These registers are a copy of HBLS handwheel keymap and indicate which key has been pressed (bit=1). Register KEYHBLS1 correspond to the first handwheel and KEYHBLS2 to the second one

6.8 General modifiable signals.

_EMERGEN
_EMERGENC1
..
_EMERGENC4

This mark is active low (=0).
Each mark corresponds to a channel (_EMERGEN and _EMERGENC1 are equivalent).
This mark must be defined in the PLC program.

If the PLC activates (=0) this mark, the CNC channel 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 active (=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.
```

_FEEDHOL
_FEEDHOLC1
..
_FEEDHOLC4

This mark is active low (=0).
Each mark corresponds to a channel (_FEEDHOL and _FEEDHOLC1 are equivalent).
This mark must be defined in the PLC program.

If the PLC activates (= 0) this mark, the CNC channel temporarily interrupts the movement of the axes. When the PLC deactivates (=1) this mark, the axes resume their movement. All the stops and starts of the axes are carried out with the corresponding acceleration and deceleration. If the PLC activates (=0) this mark on a motionless block, the CNC continues executing the program until it detects a motion block.

- This mark does not stop the spindle.
- This mark does not affect the independent movement of an axis (independent interpolator).
- 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 (=1). The status of the mark is not displayed if the screen does not show this text.

_STOP
_STOPC1
..
_STOPC4

This mark is active low (=0).
Each mark corresponds to a channel (_STOP and _STOPC1 are equivalent).
This mark must be defined in the PLC program.

If the PLC activates (=0) this mark, the CNC channel interrupts the execution of the part program. To resume the execution of the program, besides deactivating (=1) this mark, the PLC must activate (=1) the CYSTART mark. This mark does not interrupt spindle rotation nor does it affect the independent movement of an axis (independent interpolator), which also does not affect the [STOP] key.

The treatment received by the signal _STOP is similar to that given to the [STOP] key in the CNC. With this signal at a low logic level, all keys remain enabled.

_XFERINH
_XFERINHC1
..
_XFERINHC4

This mark is active low (=0).
Each mark corresponds to a channel (_XFERINH and _XFERINHC1 are equivalent).
This mark must be defined in the PLC program.

If the PLC activates (=0) this mark, 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 the PLC deactivates (=1) this mark, the CNC resumes the execution of the program.

Having this mark active (=0), the CNC does not allow jog movements for the axes of the channel; pressing a jog key is ignored.

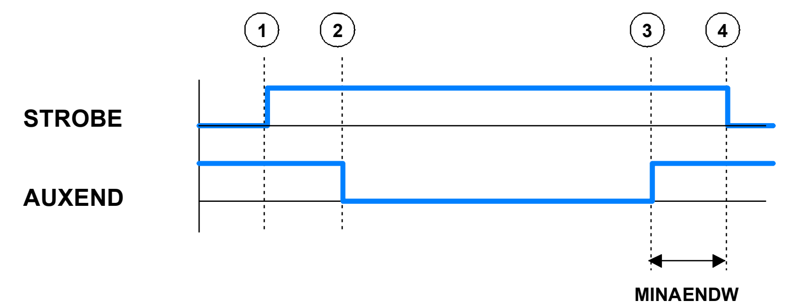
This mark always affects the movements of the independent axis programmed from the CNC; the ones programmed from the PLC depend on parameter XFITOIND. If the PLC deactivates (=1) the _XFERINH mark, the CNC channel interrupts the movements of the independent axis when they reach position and inhibits the execution of the next movement. To manage the transfer inhibit in an independent movement, the PLC also has a particular mark per axis (_IXFERINH(axis) mark).

AUXEND
AUXENDC1
..
AUXENDC4

This mark is active high (=1).
Each mark corresponds to a channel (AUXEND and AUXENDC1 are equivalent).

This mark is used when executing the auxiliary M and S functions with synchronization. The PLC deactivates (=0) this mark to indicate to the CNC that the execution has begun for the functions defined in the registers MFUN and SFUN. Once these functions have been executed, the PLC activates (=1) this mark to indicate to the CNC that it has finished. The AUXEND mark must be kept active (=1) longer than the time period established by machine parameter MINAENDW.

It works as follows:



(1) The CNC channel indicates to the PLC in the register SFUN the functions that it must execute and it activates (=1) the SSTROBE mark to start the execution.
(2) The PLC must deactivate (=0) the AUXEND mark to let the CNC know that the execution has begun.
(3) After executing the programmed functions, the PLC must activate (=1) the AUXEND mark to indicate to the CNC that it has finished. The AUXEND mark must be kept active longer than the time period established by machine parameter MINAENDW.
(4) After this time, the CNC deactivates the MSTROBE mark, thus ending the execution of the function.

BLKSKIP1
BLKSKIP1C1
··
BLKSKIP1C4

This mark is active high (=1).
Each mark corresponds to a channel (BLKSKIP1 and BLKSKIP1C1 are equivalent).

The PLC activates (=1) this mark to indicate to the CNC channel to take into account the block skip condition (/). The CNC channel will not execute the programmed blocks under this condition.

CNCOFF

This mark is active high (=1).

If the PLC activates (=1) this mark, the CNC shut-down sequence will begin. Activating this mark is the same as pressing the key combination [ALT]+[F4].

COLLISIONOFF
COLLISIONOFFC1
··
COLLISIONOFFC4

This mark is active high (=1).
Each mark corresponds to a channel (COLLISIONOFF and COLLISIONOFF are equivalent).

If the PLC activates (=1) this mark, deactivate the FCAS option (collision control). If the PLC deactivates (=0) this mark, activate the FCAS option (collision control).

CYSTART
CYSTARTC1
··
CYSTARTC4

This mark is active high (=1).
Each mark corresponds to a channel (CYSTART and CYSTARTC1 are equivalent).

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 (=1) the CYSTART mark in order for the program to start running.

```
START AND (rest of conditions) = CYSTART
```

DMCON
DMCONC1
··
DMCONC4

This mark is active high (=1).
Each mark corresponds to a channel (DMCON and DMCON1 are equivalent).

With a down flank of this mark (a change from 1 to 0), the CNC deactivates DMC. With an up flank of this mark (a change from 0 to 1), the CNC reactivates DMC.

EXRAPID
EXRAPIDC1
··
EXRAPIDC4

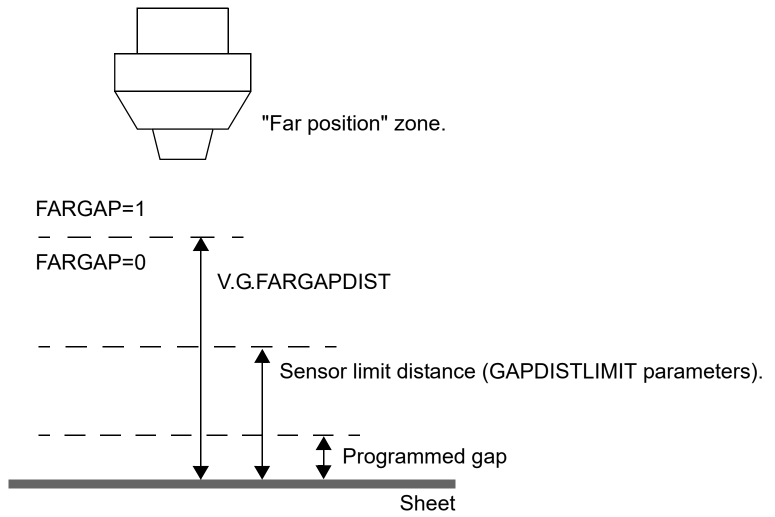
This mark is active high (=1).
Each mark corresponds to a channel (EXRAPID and EXRAPIDC1 are equivalent).

If the PLC activates (=1) 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.

FARGAP

This mark is active high (=1).

"Far position" signal provided by the gap sensor. The PLC must activate (=1) this marker when the sensor is in the far position and deactivate it (=0) when it leaves the far position. The distance from the plate to the far position point must be defined in the variable (V.)G.FARGAPDIST. The CNC uses the far position signal to apply a higher gain on the approach to the gap, on the first positioning (#GAPCTRL) and on jumps over the gap (#LEAP).



FLIMITAC

This mark is active high (=1).

If the PLC activates (=1) 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 (=0) 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
FLIMITACCHC1
··
FLIMITACCHC4

This mark is active high (=1).
Each mark corresponds to a channel (FLIMITACCHC1 and FLIMITACCHC2 are equivalent).

If the PLC activates (=1) one of these marks, 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 (= 0) one of these marks, 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.

INHIBITMPG1
··
INHIBITMPG9

This mark is active high (=1).
Each mark corresponds to a handwheel.

If the PLC turns one of these marks on (=1), 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 (V.)G.HANDP associated with the handwheel does not save the pulses sent by the handwheel.

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.

INT1··INT4
INT1C1··INT4C1
··
INT1C4··INT4C4

This mark is active high (=1).
Each mark corresponds to a channel (INT1 and INT1C1 are equivalent).

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.). If the sub-routine ends with M30, the execution of the programme also ends. If the execution of a subroutine ends with #RET, the CNC resumes the execution of the programme from the interruption point and maintains the changes made to the log 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.

KEYBD1CH
··
KEYBD8CH

This mark is active high (=1).
Each register corresponds to a keyboard.

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 an operator 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	Operator panel assigned to channel 1.
2	Operator panel assigned to channel 2.
3	Operator panel assigned to channel 3.
4	Operator panel assigned to channel 4.

An operator panel becomes disabled if the machine parameter KEYBDnCH = “Disabled keyboard” and the register KEYBDnCH=0.

LATCHM

This mark is active high (=1).

With this mark, it is possible to select how the JOG keys will work in JOG mode. If this mark is not active (=0), the axes will move while the corresponding JOG key is pressed. If this mark is active (=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).

LCOUNTALARMOFF1

This mark is active high (=1).

If the PLC activates (=1) this mark, it disables the local feedback input. If the PLC deactivates (=0) this mark, it enables the feedback alarms.

When the spindle and the C axis are parameterized in two sets for the same spindle, each with its respective feedback while sharing the input signal (although with different resolutions), the movement of the spindle can sound an alarm for the C feedback. These marks allow the C feedback alarms to be disabled (for a local feedback) when the spindle is in use, although assuming there may be a loss of feedback and that the C axis must be referenced when re-activating it.

The alarm report is conditioned to the parameter FBACKAL. No alarm is detected if the local feedback is set to non-differential TTL.

M01STOP
M01STOPC1
..
M01STOPC4

This mark is active high (=1).
Each mark corresponds to a channel (M01STOP and M01STOPC1 are equivalent).

The PLC activates (=1) this mark to "tell" the CNC channel not to ignore the conditional stops (M01).

MANRAPID
MANRAPIDC1
..
MANRAPIDC4

This mark is active high (=1).
Each mark corresponds to a channel (MANRAPID and MANRAPIDC1 are equivalent).

This mark is treated in a similar way to the Rapid traverse key. If the PLC activates (=1) this mark, 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 (=0), all jog moves are carried out at the active feedrate in jog mode.

NEXTMPGAXIS

This mark is active high (=1).

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. For an up-flank (change from 0 to 1) of this mark, the CNC behaves as follows.

- If no axis has been selected, the CNC selects the first one that is displayed.
- If there is an axis selected, the CNC 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.

NOWAITC1
..
NOWAITC4

This mark is active high (=1).
Each mark corresponds to a channel.

These marks are applied to channel synchronization. The PLC activates (=1) this mark to cancel all the synchronizations with the CNC channel.

For example, with the NOWAITC1 signal activated, the waits programmed in any channel with the instruction #WAIT, and that refer to a mark of channel 1, they finish immediately and the program execution resumes.

OVRCAN
OVRCANC1
..
OVRCANC4

This mark is active high (=1).
Each mark corresponds to a channel (OVRCAN and OVRCANC1 are equivalent).

If the PLC activates (=1) this mark, the CNC applies 100 % override to the axis feedrate regardless of the value currently selected. While the OVRCAN mark is active, the CNC channel will apply a feedrate of 100% corresponding for each work mode.

PANELOFF
PANELOFF1
..
PANELOFF8

This mark is active high (=1).
Each mark corresponds to an operator panel (PANELOFF and PANELOFF1 are equivalent).

If the PLC activates (=1) one of these marks, it disables the corresponding CAN operator panel. When the keyboard and the operator panel make up a single element, this mark only disables the operator panel.

The PANELOFF1 (or PANELOFF) mark disables the first one of the bus, the PANELOFF2 mark the second one and so on. The "address" switch determines the order of the elements in the CAN bus. The first operator panel will be the one with the lowest number and so on.

Address.	Element.	PLC mark.
0	CNC.	
1	Remote (I/O) group.	
2	Operator panel.	PANELOFF1
3	Remote (I/O) group.	
4	Remote (I/O) group.	
5	Keyboard + Operator panel.	PANELOFF2

PLCABORT
PLCABORTC1
..
PLCABORTC4

This mark is active high (=1).
Each mark corresponds to a channel (PLCABORT and PLCABORTC1 are equivalent).

If the PLC activates this mark (=1), the CNC channel aborts the CNCEX command launched from the PLC, but without initializing the conditions of the channel and keeps the channel history. 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 channel deactivates (=0) this mark and it also activates (=1) the FREE mark of the channel.

If the PLC activates (=1) the PLCABORT mark without having launched a previous CNCEX, the mark remains active until a CNCEX is executed (which will be aborted automatically) or until it deactivates (=0) the mark (RES PLCABORT).

PLCREADY

This mark is active high (=1).

This mark indicates whether the PLC is running (=1) or stopped (=0). The PLC must be operating for the CNC to allow the axes to be moved and for the spindle to turn. If the PLC deactivates (=0) this mark, the execution of the PLC program is interrupted and an error message is displayed.

PRGABORT
PRGABORTC1
..
PRGABORTC4

This mark is active high (=1).
Each mark corresponds to a channel (PRGABORT and PRGABORTC1 are equivalent).

If the PLC activates (=1) this mark, 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.

PROBE1ENA
PROBE2ENA

This mark is active high (=1).
Each mark corresponds to a probe.

These are active marks by default. These mark indicate that the probe has is enabled. When executing a G100 or G103 command, the CNC will issue an error message if the mark of the active probe (the one selected with #SELECT PROBE) is not enabled. These marks do not limit the monitoring of the safe mode.

These marks should be tested in the subroutines Sub_Probe_Tool_Begin.fst and Sub_Probe_Piece_Begin.fst so the subroutine waits until the mark is active.

PROBE1MONIT
PROBE2MONIT

This mark is active high (=1).
Each mark corresponds to a probe.

These marks are associated with the safe mode of the probe. If the mark is active (=1), the probe is in safe mode monitoring collisions. Loop level monitoring, controlling all the collision scenarios in any of the two probes. The CNC can monitor the probes connected to the local inputs and to the remote CAN inputs. The CNC can monitor both probe inputs at the same time.

These marks should be tested in the subroutines Sub_Probe_Tool_Begin.fst and Sub_Probe_Piece_Begin.fst to warn, if they are deactivated, that the probe is in non-safe mode.

PT100OFF1
..
PT100OFF20

This mark is active high (=1).
Each mark corresponds to a PT100 input probe.

The PT100 inputs are activated via the machine parameters. If the input is active in the machine parameter and there is no sensor connected, the CNC returns the corresponding error. The PLC has the following marks in order to be able to temporarily disable the sensor (for example, during a spindle change).

If the PLC activates (=1) one of these marks, the CNC disables the corresponding sensor (parameter PT100 n). The PT100OFF1 mark corresponds to the input 1 sensor, PT100OFF2 is for the input 2 sensor and so forth.

PW1ENAF

This mark is active high (=1).

The PLC uses this mark to enable (=1) or disable (=0) the first RPS power supply. By enabling the power supply, the DC power bus voltage is increased to the set value.

PW1ENAS

This mark is active high (=1).

The PLC uses this mark to activate the power enabling sequence on the machine with the first RPS. When the mark is activated (=1), the RPS will charge the DC bus and connect the power contactor to the mains in a controlled manner. If this mark is deactivated during the startup process, it is canceled. At any time this mark is deactivated, it entails a fall in the DC power bus.

PW2ENAF

This mark is active high (=1).

The PLC uses this mark to enable (=1) or disable (=0) the second RPS power supply. By enabling the power supply, the DC power bus voltage is increased to the set value.

PW2ENAS

This mark is active high (=1).

The PLC uses this mark to activate the power enabling sequence on the machine with the second RPS. When the mark is activated (=1), the RPS will charge the DC bus and connect the power contactor to the mains in a controlled manner. If this mark is deactivated during the startup process, it is canceled. At any time this mark is deactivated, it entails a fall in the DC power bus.

QWERTYOFF1
..
QWERTYOFF8

This mark is active high (=1).
Each mark corresponds to a CAN keyboard.

If the PLC activates (=1) one of these marks, it disables the corresponding CAN keyboard, including the soft keys. When the keyboard and the operator panel make up a single element, this mark only disables the keyboard. The keyboards are not enabled and disabled immediately, especially for the central unit and it may take several seconds.

There is a mark for each CAN keyboard. The QWERTYOFF1 mark disables the first one, the QWERTYOFF2 mark the second one and so on. The "address" switch determines the order of the elements in the CAN bus. The first keyboard will be the one with the lowest number and so on.

Address.	Element.	PLC mark.
0	CNC 10K.	QWERTYOFF1
1	Remote (I/O) group.	
2	Keyboard.	QWERTYOFF2
3	Remote (I/O) group.	
4	Remote (I/O) group.	
5	Keyboard + Operator panel.	QWERTYOFF3

RESETIN
RESETINC1
··
RESETINC4

This mark is active high (=1).
Each mark corresponds to a channel (RESETIN and RESETINC1 are equivalent).

Resetting the CNC. The treatment of this mark is similar to that of the [RESET] key. With an up-flank (change from 0 to 1) of this mark, the CNC channel is reset. After the reset has commenced, the PLC can deactivate (=0) this mark. The reset process is as follows.

1. The user presses the [RESET] key on the operator panel or the PLC activates (=1) the RESETIN mark.
2. The channel of the CNC assumes the initial conditions (those defined by the machine parameters).
3. The channel of the CNC activates (=1) the RESETOUT mark to indicate that the reset has finalized. This mark stays active for a time period set in parameter MINAENDW.

RESTAPAC

This mark is active high (=1).

The CNC activates (=1) this marker after pressing the softkey that ends the tool withdrawal process after an interrupted tapping. The PLC can activate this marker in order to force the end of this process, without the need to press the softkey. When this flag is activated, the CNC deactivates the TAPACFAIL mark and resets to normal conditions. After the reset, the CNC deactivates the RESTAPAC marker.

RETRACEC1
··
RETRACEC4

This mark is active high (=1).
Each mark corresponds to a channel.

If the PLC activates (=1) this mark 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 deactivates this mark, the CNC finishes the retrace function. The retrace function also ends after a M30, reset or when the CNC channel activates (=1) the RETRAEND mark.

The retrace function must be enabled in machine parameter RETRACAC.

RKIN1
··
RKIN32

These registers allow for communication between the PLC and user kinematics.

SBLOCK
SBLOCKC1
··
SBLOCKC4

This mark is active high (=1).
Each mark corresponds to a channel (SBLOCK and SBLOCKC1 are equivalent).

If the PLC activates (=1) this mark, the CNC channel activates the “block-by-block” execution mode.

SLIMITAC

This mark is active high (=1).

If the PLC activates (=1) 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 (=0) this mark, the CNC restores the programmed speed.

SYNC1
..
SYNC4

Each register corresponds to a channel.

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.

TIMERON

This mark is active high (=1).

The CNC has a timer to be used at will. If the PLC activates (=1) this mark, it enables the counter (the counter will be counting).

It times in seconds and it is enabled and disabled with this mark. To initialize and/or know its count, use the (V.)PLC.TIMER variable. This variable may be accessed 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
```

VIRTUALJOGENA

This mark is active high (=1).

The PLC uses this mark to enable (=1) or disable (=0) the virtual jog keyboard (on-screen keyboard).

VOLCOMP1
··
VOLCOMP4

This mark is active high (=1).
Each mark corresponds to a volumetric compensation.

If the PLC activates (=1) one of these marks, the CNC activates the corresponding volumetric compensation (parameter VOLCOMP). If the PLC deactivates (=0) one of these marks, the CNC deactivates the corresponding volumetric compensation.

Mark.	Meaning.
VOLCOMP1	Activate the first volumetric compensation (parameter VOLCOMP 1).
VOLCOMP2	Activate the second volumetric compensation (parameter VOLCOMP 2).
VOLCOMP3	Activate the third volumetric compensation (parameter VOLCOMP 3).
VOLCOMP4	Activate the fourth volumetric compensation (parameter VOLCOMP 4).

All volumetric compensations can be simultaneously active, as long as there are no common axes among them. The CNC applies the volumetric compensation after applying the leadscrew compensation and the cross compensation. Volumetric compensation remains active after a reset, error or end of program (M30).

6.9 Modifiable signals of the axes and spindles in closed loop (M19 or G63).

AXISPOS(axis)
AXISNEG(axis)

This mark is active high (=1).

The CNC uses these marks when operating in JOG mode. If the PLC activates (=1) one of these marks, the CNC will move the relevant axis in the indicated direction: positive (AXISPOS(axis)) or negative (AXISNEG(axis)). 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.

BRKUENA(axis)
BRKLENA(axis)

Not available since v2.00.01.
This mark is active high (=1).

The BRKUENA(axis) mark activates the B+ (24 V) brake output (X12 connector of the QC-DR module). The BRKLENA(axis) mark activates the B- (0 V) brake output (X12 connector of the QC-DR module). Both marks allow for the brake to be checked and activated.

BRKUENA(axis)	BRKLENA(axis)	BRKFB(axis)	Meaning.
0	0	1	Brake not connected (not plugged in).
0	0	0	Brake connected (plugged in).
1	0	1 (*)	Output B+ (24 V) of the correct brake.
0	1	1 (*)	Output B– (0 V) of the correct brake.
1	1	1 (*)	Brake output activated (release brake).
(*) For these cases, the BRKFB(axis) mark is set to 1 with or without the brake connected.			

DEAD(axis)

This mark is active high (=1).

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(axis) and SERVO(axis)ON 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. The PLC must activate (=1) this mark to enable the axis as a “dead axis” and deactivate (=0) it to enable the axis as a “live axis”.

- If the PLC activates (=1) this mark, 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 deactivates (=0) this mark, the CNC blends the blocks even when the axis is involved in the movement.

DECEL(axis)

This mark is active high (=1).

This mark is used during home search. The PLC activates (=1) this mark to indicate that the home switch is pressed. The CNC decelerates the axis, and switches from the rapid feedrate set by the machine parameter REFFEED1 to the slow feedrate set by machine parameter REFFEED2.

During a home search for distance-coded “I0” axes, the PLC activates (=1) this mark 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.

DIFFCOMP(axis)

This mark is active high (=1).

This mark is used on gantry axes. This flag corrects the dimension difference between the master and slave axes as follows.

- With the up-flank of SERVO(axis)ON if DIFFCOMP(axis) is active (=1).
- With the up-flank of DIFFCOMP(axis) if SERVO(axis)ON is active (=1).

For both cases, 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.

DRENA(axis)
SPENA(axis)

This mark is active high (=1).

The PLC uses these marks to enable drives when communicating via Sercos. 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(axis) 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(axis) 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.

DRO(axis)

This mark is active high (=1).

This mark, together with the SERVO(axis)ON mark, allows the axis or spindle to work as a DRO. 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.

To define an axis as a DRO, the PLC must activate (=1) the DRO(axis)mark and deactivate (=0) the SERVO(axis)ON mark. If the PLC deactivates (=0) the DRO(axis) mark again, the axis ceases to be the DRO axis and the CNC assumes the current coordinate as the position coordinate, assigning a 0 value to the following error.

FBACKSEL(axis)

This mark is active high (=1).

The CNC uses this mark when the system has dual feedback (parameters SPEEDFBID and POSITIONFBID not equal). With the up-flank of this mark, the direct feedback is switched on (POSITIONFBID parameter) and with the down-flank, the motor feedback is switched on (SPEEDFBID parameter). If the combined feedback is active (parameter FBMIXTIME), it is used with the up flank of this mark. The ACTFBACK(axis) mark indicates which is the active feedback.

When the system has dual feedback, the CNC uses the motor feedback (SPEEDFBID parameter) on power-up, when resetting the drive and when initializing the Sercos ring.

GANTRYOFF(axis)

This mark is active high (=1).

This tag refers to the gantry slave axis. This mark allows for the adjustment of the gantry and allows for the correction of the axis positions when they have crossed or when they Positioncannot be restored automatically.

If the PLC activates this mark (change from 0 to 1), the CNC will decouple the gantry axis and allow for only the master axis to be moved. For example, if the Y axis is the master and the W axis is the slave, you must program GANTRYOFFW=1 to de-couple the axes.

INHIBIT(axis)

This mark is active high (=1).

If the PLC turns this mark on (=1), the CNC inhibits any movement of the corresponding axis or spindle. This movement will resume when the PLC deactivates (=0) this mark again. If the axis or spindle is moving with other axes, all the axes will stop.

For independent axes and an electronic cam, if the PLC activates (=1) this mark, the synchronization movement is interrupted and switches 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.

LIM(axis)OFF

This mark is active high (=1).

If the PLC activates this mark (=1), the CNC ignores the software travel limits and the work zones set for the corresponding axis. If the PLC deactivates (=0) this mark, the CNC considers again the software travel limits and the work zones set for the indicated axis.

This mark makes it easier to take the tool back to the permitted zone if it had exceeded the software limits or it had invaded the forbidden zone.

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. The work zones may be set using functions G120, G121, G122, G123.

LIMITPOS(axis)
LIMITNEG(axis)

This mark is active high (=1).

If the PLC activates (=1) this mark, the CNC interprets that the corresponding axis or spindle has exceeded the positive (LIMITPOS(axis)) or negative (LIMITNEG(axis)) travel limits. The CNC halts the feedrate of the axes and spindle rotation, displaying the corresponding error message on the screen.

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.

LUBRENA(axis)

This mark is active high (=1).

The marks LUBRENA(axis), LUBR(axis) and LUBROK (axis) are used for the lubrication of the axes. The PLC activates (=1) this mark to indicate that this feature is available.

If the PLC activates (=1) this mark, the CNC acts as follows:

1. When the axis has traveled the distance set by parameter DISTLUBRI(axis), the CNC activates (=1) the LUBR(axis) mark to indicate to the PLC that it must lubricate the axis.
2. After lubricating the axis, the PLC activates (=1) the LUBROK(axis) mark.
3. The CNC deactivates (=0) the LUBR(axis) mark and resets the count to 0.
4. The PLC deactivates (=0) the LUBROK(axis) mark to resume the count.

LUBROK(axis)

This mark is active high (=1).

This mark is used to lubricate the axes. The PLC activates (=1) this mark to indicate to the CNC that it has already lubricated the axis. When the CNC deactivates (=0) the LUBR(axis) mark, the PLC must deactivate (=0) this mark, otherwise the count will always stay at 0.

PARKED(axis)

This mark is active high (=1).

The PARKED(axis), PARK(axis) and UNPARK (axis) marks are used to park and unpark axes and spindles, from either the PLC or CNC. If the PLC commences the operation, it must activate (=1) this mark to park the axis and deactivate it to unpark it. If the operation is initiated by the CNC, the PLC activates (=1) this mark when the axis is parked.

SERVO(axis)ON

This mark is active high (=1).

The PLC must activate (=1) this mark to allow the movement (unlock) of the corresponding axis. If the PLC deactivates (=0) this mark when the axis or spindle is moving, the CNC interrupts the feedrate of the axes and the spindle rotation and displays the corresponding error message on the screen.

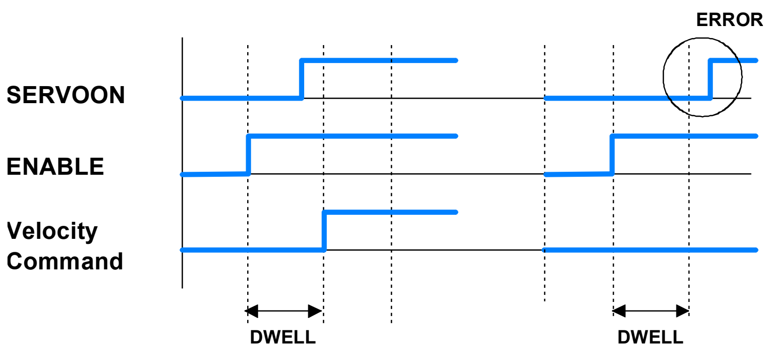
In order to be able to continuously control the axis, the SERVO(axis)ON mark must always be active (=1).

```
(there are no errors) AND (axis drive OK) = SERVOXON
```

To control the axis only during movement, the ENABLE(axis) mark must be used, which is activated (=1) by the CNC every time the axis is moved.

```
(no errors) AND (drive OK) AND ENABLEX = SERVOXON
```

If the axis moves while locked (meaning SERVOON=0), the CNC stores that displacement as axis lag (following error). To control it again (SERVO(axis)ON=1), the axis restores its position.



After activating the ENABLE(axis) mark, the CNC waits for a time period indicated by parameter DWELL only if SERVO(axis)ON is low (=0). If after this time period the SERVO(axis)ON signal is still deactivated, the CNC shows the error "axis locked up".

To control the axis only during movement, the machine parameter 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 SERVO(axis)ON 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.

TANDEMOFF(axis)

This mark is active high (=1).

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 (=1) 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 can be moved in the usual way from the operator panel, in MDI/MDA mode, etc. When moving the master axis, the CNC will not generate a setpoint for the slave axis or 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).

6.10 Modifiable spindle signals in open loop (M3/M4)

GEAR1..GEAR4
GEAR1SP1..GEAR4SP1
..
GEAR1SP6..GEAR4SP6

This mark is active high (=1).
There are four marks for each spindle (GEAR1/GEAR4 and GEAR1SP1/GEAR4SP1 are equivalent); one mark for each range.

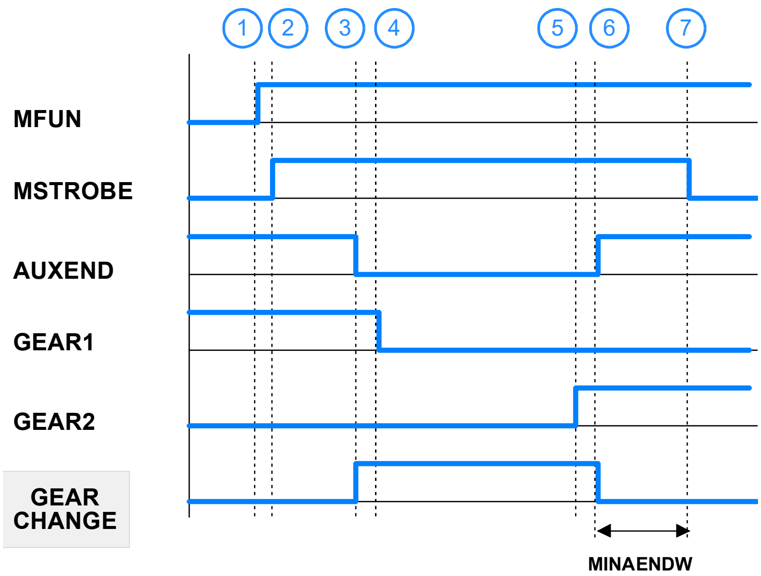
The PLC uses these marks to let the CNC know which spindle range (gear) is currently selected. The selected range will have its mark active (=1). When requesting a gearchange, the CNC informs the PLC about it using auxiliary functions: M41, M42, M43 or M44. The gear change concludes when the PLC receives 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.

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 to 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.

2 The CNC activates (=1) the MSTROBE mark. When detecting the request, the PLC sets an internal indicator (M1002).

```
DFU MSTROBE AND CPS MFUN* EQ 42 = SET M1002
```

3 The gear change begins. The PLC deactivates the AUXEND mark.

```
NOT M1002 AND <rest of conditions> \
= AUXEND \
= (starts the gear change)
```

4 The CNC removes range 1.

```
I21 = GEAR1
```

5 The CNC selects range 2.

```
I22 = GEAR2
```

6 The PLC removes the internal indicator (M1002) and activates (=1) the AUXEND mark. The GEAR2 active range indicator must be active high before acting on the AUXEND signal.

```
(GEAR change completed) = RES M1002
```

7 The gear change ends. The AUXEND mark must be kept as active high for longer that that set by parameter MINAENDW, so the CNC changes the MSTROBE mark to active low and concludes the gear change.

PLCCNTL
PLCCNTL1
··
PLCCNTL6

This mark is active high (=1).
Each mark corresponds to a spindle (PLCCNTL and PLCCNTL1 are equivalent).

When the PLC activates (=1) the PLCCNTL mark, 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.

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.

PLCM3/PLCM4/PLCM5
PLCM3SP1/PLCM4SP1/PLCM5SP1
··
PLCM3SP6/PLCM4SP6/PLCM5SP6

This mark is active high (=1).
There are four marks for each spindle (PLCM3/PLCM5 y PLCM3SP1/PLCM5SP1 are equivalent); one mark for each range.

The PLC activates these marks to indicate to the CNC that it must execute the corresponding M function at the indicated spindle (PLCM3 for M3, PLCM4 for M4 and PLCM5 for M5). 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.

SANALOG
SANALOG1
··
SANALOG6

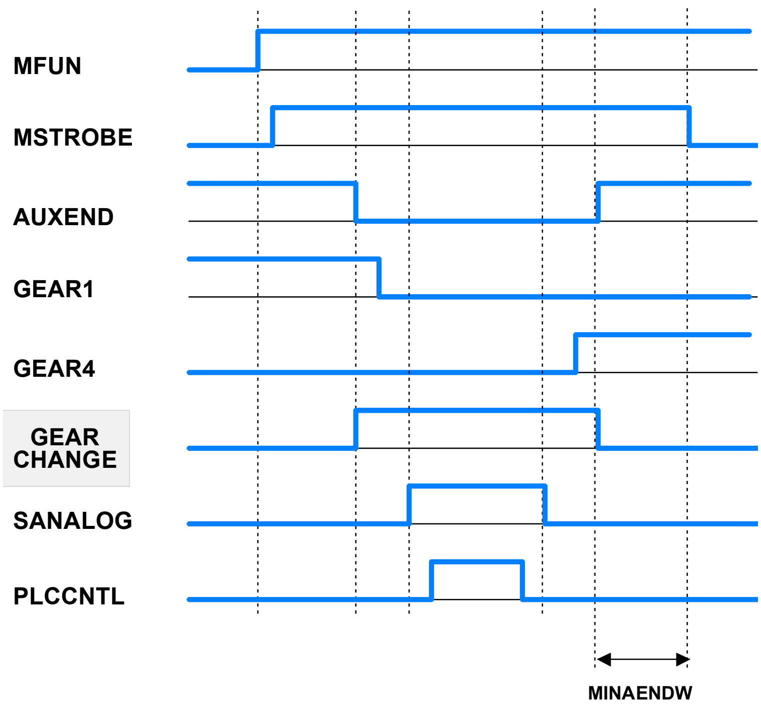
Each register corresponds to a spindle (SANALOG and SANALOG1 are equivalent).

When the spindle is controlled by the PLC, the SANALOG register determines the setpoint to be applied. 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.

Spindle type.	SANALOG units.
Analog spindle.	10 V of velocity command correspond to a value of 32767. • For 4 V, SANALOG = (4×32767) / 10 = 13107 • For -4 V, SANALOG = (-4×32767) / 10 = -13107
Sercos spindle in velocity.	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.

Example of a GEAR1 to GEAR4 change.

The spindle oscillation during a gear change is controlled by the PLC. Gear 4 is requested while gear 1 is active.



This example shows how to control the spindle oscillation during a gear change. The PLC sets SANALOG to the value of the residual setpoint and activates the PLCCNTL mark to indicate to the CNC that the spindle is controlled by the PLC. Once the gear change has completed, the PLCCNTL must be deactivated and the SANALOG set to 0.

SLIMITACSPDL
SLIMITACSPDL1
··
SLIMITACSPDL6

This mark is active high (=1).
Each mark corresponds to a spindle (SLIMITACSPDL and SLIMITACSPDL1 are equivalent).

If the PLC activates (=1) 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.

SPDLEREV
SPDLEREV1
..
SPDLEREV6

This mark is active high (=1).
Each mark corresponds to a spindle (SPDLEREV and SPDLEREV1 are equivalent).

When the PLC activates (=1) this mark on, the CNC inverts 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 M3 or M4 function is executed while the SPDLEREV mark is active (=1), the spindle will turn in the opposite direction to the one assigned for 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.

6.11 Modifiable signals of the independent interpolator.

The signal names are generic. Replace the text (axis) with the name or logic axis number.

_IXFERINH(axis)

This mark is active low (=0).

If the PLC activates this mark (=0), the independent movement ends and remains waiting until the PLC activates this mark again. The PLC also has a general mark for each channel (_XFERINH). 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.

IABORT(axis)

This mark is active high (=1).

For the movements of an independent axis, if the PLC activates (=1) this mark, the positioning block being executed (if any) stops without eliminating the other positioning blocks pending execution. This mark only affects positioning blocks; neither the pending instructions nor the synchronization movement are removed.

IRESET(axis)

This mark is active high (=1).

For movements of an independent axis, if the PLC activates (=1) this mark, it interrupts the instruction in execution and removes any instructions pending execution. For electronic-cam movements, it interrupts the cam synchronization movement switching to zero speed. The initial conditions are also set at the independent interpolator of the axis.

6.12 Modifiable logic signals; laser.

CVIEWREFRESH

This mark is active high (=1).

The PLC activates this mark to refresh the Cut_View. For example, cutting sheets longer than the table feed, in a single program, is carried out in several stages, moving the sheet between one stage and another. At the end of each stage, the Cut_View must be refreshed so that it shows the new section to be cut.

ENABLEGAP

This mark is active high (=1).

This mark is active by default. The PLC deactivates this mark to disable the active gap control. The PLC activates this mark to enable the active gap control in the CNC; if no gap control is active, this mark does nothing.

ENABLELEAP

This mark is active high (=1).

The PLC deactivates this mark to disable the active leapfrog. The PLC activates this mark to enable the active leapfrog in the CNC; if no gap control is active, this mark does nothing.

ENABLEPWM

This mark is active high (=1).

The PLC activates this flag (change from 1 to 0) to enable the active PWM on the CNC; if there is no active PWM, this flag does nothing.

ENABLEPWRDUTY

This mark is active high (=1).

This mark is active by default. This mark is associated with the PWM duty control (#PWRCTRL ON [DUTY]). The PLC deactivates this mark to disable the active duty control. The PLC activates this mark to enable the active duty control in the CNC; if no duty control is active, this mark does nothing.

If the PLC deactivates this mark, the CNC disables the duty control, setting the maximum specified in the instruction #PWMOUT.

ENABLEPWRFREQ

This mark is active high (=1).

This mark is active by default. This mark is associated with the PWM power control (#PWRCTRL ON [FREQ]). The PLC deactivates this mark to disable the active power control. The PLC activates this mark to enable the active power control in the CNC; if no power control is active, this mark does nothing.

If the PLC deactivates this mark, the CNC disable the power control, setting the maximum defined in the instruction #PWMOUT.

ENABLEPWROUT

This mark is active high (=1).

This mark is active by default. This mark is associated with the power control via an analog output associated with the spindle (#PWRCTRL ON [OUT]). The PLC deactivates this mark to disable the active power control. The PLC activates this mark to enable the active power control in the CNC; if no power control is active, this mark does nothing.

If the PLC deactivates this mark, the CNC disable the power control, setting the maximum defined in the instruction #PWRCTRL ON.

GAPCOLLISION

This mark is active high (=1).

The gap control sensor may have a collision signal connected to the CNC, which should be managed by the PLC via this mark. The PLC activates this mark to indicate that the sensor collision signal has been received. This mark is not kept in memory. When this mark is active, the CNC will display an error or not, depending on how the parameter GAPCOLLISION has been set.

LASERON

This mark is active high (=1).

The PLC activates this mark to indicate that the laser is on.

LASERREADY

This mark is active high (=1).

The PLC activates this mark to indicate that the laser is ready.

PWMDUTY

This mark is active high (=1).

PWM duty cycle, when the PWM has been activated from the PLC (1-5000 tenths of %; by default, 500).

PWMFREQ

This mark is active high (=1).

PWM frequency, when the PWM has been activated from the PLC (2-5000 Hz; by default, 0).

PWMON

This mark is active high (=1).

This mark may be used to activate and deactivate the PWM via PLC. The PWM is configured using the following registers.

Registers.	Meaning.
PWMFREQ	Frequency of the PWM (between 2 and 5000 Hz; by default, 0).
PWMDUTY	Duty cycle of the PWM (between 1 and 1000; by default, 500).

Activating the PWM via PLC has priority over activating it from the CNC.

PWM status from the CNC.	PWM status from the PLC.	PWM status.
Off.	Off.	PWM off.
Active.	Off.	PWM active from the CNC.
Off.	Active.	PWM active from the PLC.
Active.	Active.	PWM active 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.

OUTOFGAP

This mark is active high (=1).

The PLC activates this flag (change from 1 to 0) to indicate that the nozzle has once left the gap (parameters GAPMIN and GAPMAX) during the execution of the program. This flag remains active until the execution of a program is started.

OUTOFTOL

This mark is active high (=1).

The PLC activates this flag to indicate that the nozzle has once gone out of gap tolerance (parameter GAPTOL) during program execution. This flag remains active until the execution of a program is started.

SHUTTERON

This mark is active high (=1).

The PLC activates this mark to indicate that the laser source shutter is open.

6.13 Tool manager modifiable signals.

CH1TOMZ
CH1TOMZ1
..
CH1TOMZ4

This mark is active high (=1).
Each mark corresponds to a tool magazine (CH1TOMZ and CH1TOMZ1 are equivalent).

The PLC must activate (=1) this mark after taking the tool from the changer arm 1 to the magazine. Use it with an asynchronous magazine or synchronous with arm.

CH1TOSPDL
CH1TOSPDLMZ1
..
CH1TOSPDLMZ4

This mark is active high (=1).
Each mark corresponds to a tool magazine (CH1TOSPDL and CH1TOSPDLMZ1 are equivalent).

The PLC must activate (=1) this mark after taking the tool from the changer arm 1 to the spindle. Use it with an asynchronous magazine or synchronous with arm.

CH2TOMZ
CH2TOMZ1
..
CH2TOMZ4

This mark is active high (=1).
Each mark corresponds to a tool magazine (CH2TOMZ and CH2TOMZ1 are equivalent).

The PLC must activate (=1) this mark after taking the tool from the changer arm 2 to the magazine. Use it with an asynchronous magazine or synchronous with arm.

CUTTINGON
CUTTINGONC1
..
CUTTINGONC4

This mark is active high (=1).
Each mark corresponds to a channel (CUTTINGON and CUTTINGONC1 are equivalent).

The PLC must activate (=1) this mark if the tool is machining. When associating a maximum life span to a tool (monitoring), the CNC checks this mark to determine whether the tool is machining or not.

```
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
```

GRTOSPDL
GRTOSPDLC1
..
GRTOSPDLC4

This mark is active high (=1).
Each mark corresponds to a channel (GRTOSPDL and GRTOSPDLC1 are equivalent).

The PLC must activate (=1) this mark after taking the tool from the ground to the spindle. Use it with a magazine that admits ground tools.

MZPOS
MZPOSMZ1
..
MZPOSMZ4

Each register corresponds to a magazine (MZPOS and MZPOSMZ1 are equivalent).

The PLC must indicate in this register the current magazine position.

MZROT
MZROTMZ1
..
MZROTMZ4

This mark is active high (=1).
Each mark corresponds to a tool magazine (MZROT and MZROTMZ1 are equivalent).

The PLC must activate (=1) this mark when the turret has rotated. Use it with a turret type magazine.

MZTOCH1
MZTOCH1MZ1
..
MZTOCH1MZ4

This mark is active high (=1).
Each mark corresponds to a tool magazine (MZTOCH1 and MZTOCH1MZ1 are equivalent).

The PLC must activate (=1) this mark after taking the tool from the magazine to the changer arm 1. Use it with an asynchronous magazine or synchronous with arm.

MZTOSPDL
MZTOSPDLMZ1
..
MZTOSPDLMZ4

This mark is active high (=1).
Each mark corresponds to a tool magazine (MZTOSPDL and MZTOSPDLMZ1 are equivalent).

The PLC must activate (=1) this mark after taking the tool from the magazine to the spindle. Use it with a synchronous magazine without arm.

RESTMEM
RESTMEMZ1
..
RESTMEMZ4

This mark is active high (=1).
Each mark corresponds to a tool magazine (RESTMEM and RESTMEMZ1 are equivalent).

The PLC must activate (=1) this mark to cancel the emergency of the tool manager.

SETTMEM
SETTMEMZ1
..
SETTMEMZ4

This mark is active high (=1).
Each mark corresponds to a tool magazine (SETTMEM and SETTMEMZ1 are equivalent).

The PLC must activate (=1) this mark to activate the emergency of the tool manager.

SPDLTOCH1
SPDLTOCH1MZ1
..
SPDLTOCH1MZ4

This mark is active high (=1).
Each mark corresponds to a tool magazine (SPDLTOCH1 and SPDLTOCH1 are equivalent).

The PLC must activate (=1) this mark after taking the tool from the spindle to the changer arm 1. Use it with an asynchronous magazine with one changer arm.

SPDLTOCH2
SPDLTOCH2MZ1
..
SPDLTOCH2MZ4

This mark is active high (=1).
Each mark corresponds to a tool magazine (SPDLTOCH2 and SPDLTOCH2 are equivalent).

The PLC must activate (=1) this mark after taking the tool from the spindle to the changer arm 2. Use it with an asynchronous magazine or synchronous with arm.

SPDLTOGR
SPDLTOGRC1
..
SPDLTOGRC4

This mark is active high (=1).
Each mark corresponds to a channel (SPDLTOGR and SPDLTOGRC1 are equivalent).

The PLC must activate (=1) this mark after taking the tool from the spindle to ground. Use it with a magazine that admits ground tools.

SPDLTOMZ
SPDLTOMZ1
..
SPDLTOMZ4

This mark is active high (=1).
Each mark corresponds to a tool magazine (SPDLTOMZ and SPDLTOMZ1 are equivalent).

The PLC must activate (=1) this mark after taking the tool from the spindle to the magazine. Use it with a synchronous magazine without arm.

TCHANGEOK
TCHANGEOKMZ1
..
TCHANGEOKMZ4

This mark is active high (=1).
Each mark corresponds to a tool magazine (SPDLTOMZ and TCHANGEOKMZ1 are equivalent).

The PLC must activate (=1) this mark when the tool change has completed (M06).

TREJECT
TREJECTC1
..
TREJECTC4

This mark is active high (=1).
Each mark corresponds to a channel (TREJECT and TREJECTC1 are equivalent).

If the PLC activates (=1) this mark, the CNC interprets that the tool must be rejected.

6.14 Keystroke modifiable signals.

KEYLED1

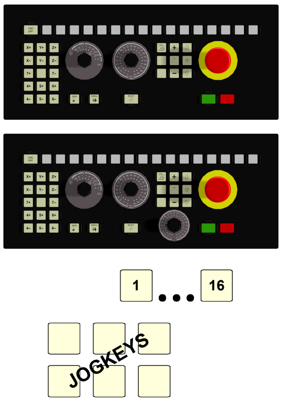
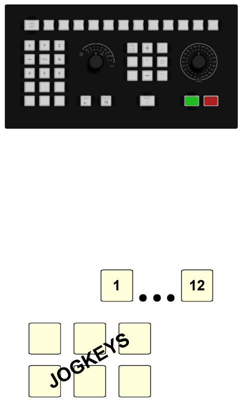
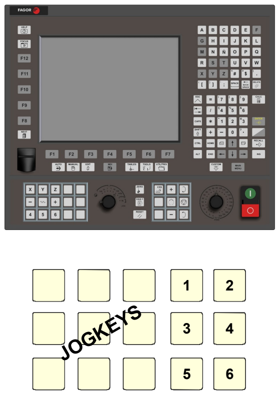
This register turns on the LED (bit=1) of the user keys on all control panels simultaneously.

KEYLED1_1
··
KEYLED1_8

These registers light up the LEDs (bit=1) of the user keys on a control panel. The KEYLED1_1 register corresponds to the first control panel, KEYLED1_2 to the second 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
```

Bit.	OP-PANEL	OP-PANEL-329	QC-C65□-10K
			
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	Not being used. The user keys do not have LEDs.	Not being used. The user keys do not have LEDs.
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		

KEYLED2

Not being used.

KEYLED2_1
..
KEYLED2_8

Not being used.

KEYLEDHBLS1
KEYLEDHBLS2

These registers turn on the HBLS handwheel key LEDs (bit=1). Register KEYLEDHBLS1 correspond to the first handwheel and KEYLEDHBLS2 to the second one

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 operator 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.

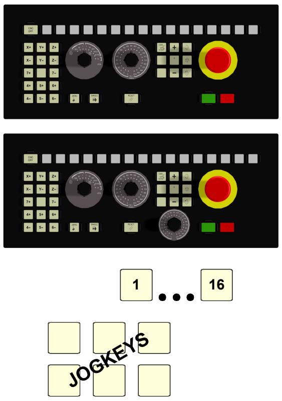
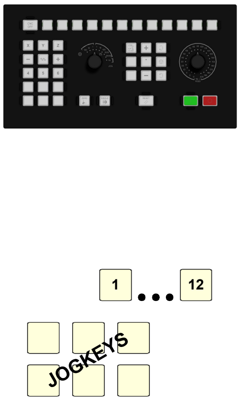
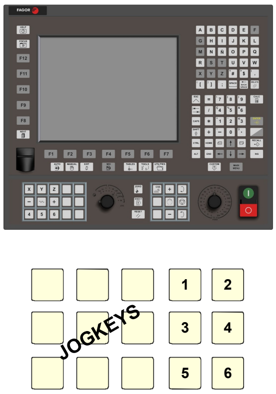
The following instruction inhibits the first jog key (bit 16) of the second keyboard.

```
( ) = B16KEYDIS1_2
```

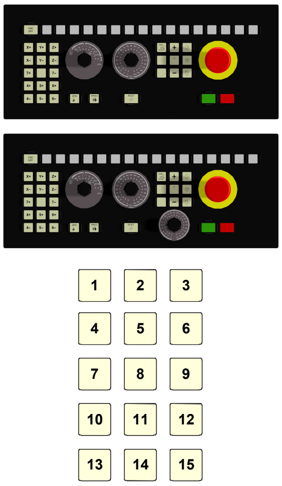
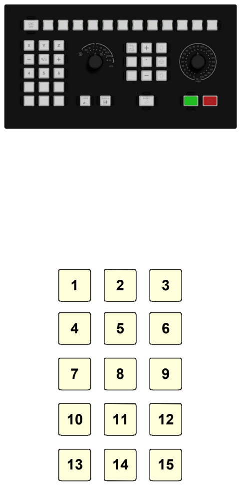
KEYDISHBLS1
KEYDISHBLS2

These registers inhibit (bit=1) the keys and the switches on the HBLS handwheel. Register KEYDISHBLS1 correspond to the first handwheel and KEYDISHBLS2 to the second one

Registers KEYDIS1 / KEYDIS1_1 to KEYDIS1_8. User keys.

Bit.	OP-PANEL	OP-PANEL-329	QC-C65□-10K
			
0	User key 1	User key 1	User key 1
1	User key 2	User key 2	User key 2
2	User key 3	User key 3	User key 3
3	User key 4	User key 4	User key 4
4	User key 5	User key 5	User key 5
5	User key 6	User key 6	User key 6
6	User key 7	User key 7	- - -
7	User key 8	User key 8	- - -
8	User key 9	User key 9	- - -
9	User key 10	User key 10	- - -
10	User key 11	User key 11	- - -
11	User key 12	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.	OP-PANEL	OP-PANEL-329	QC-C65☐-10K
			
16	Jog key 1	Jog key 1	Jog key 1
17	Jog key 2	Jog key 2	Jog key 2
18	Jog key 3	Jog key 3	Jog key 3
19	Jog key 4	Jog key 4	Jog key 4
20	Jog key 5	Jog key 5	Jog key 5
21	Jog key 6	Jog key 6	Jog key 6
22	Jog key 7	Jog key 7	Jog key 7
23	Jog key 8	Jog key 8	Jog key 8
24	Jog key 9	Jog key 9	Jog key 9
25	Jog key 10	Jog key 10	- - -
26	Jog key 11	Jog key 11	- - -
27	Jog key 12	Jog key 12	- - -
28	Jog key 13	Jog key 13	- - -
29	Jog key 14	Jog key 14	- - -
30	Jog key 15	Jog key 15	- - -
31	- - - -	- - -	- - -

Registers KEYDIS2 / KEYDIS2_1 to KEYDIS2_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	- - -

Bit.	Key.
9	- - -
10	- - -
11	ZERO
12	- - -
13	Single block
14	- - -
15	RESET
16	- - -
17/31	- - -

Registers KEYDIS3 / KEYDIS3_1 to KEYDIS3_8. Feedrate override and movement selector (handwheel, incremental jog or continuous jog).

Bit.	Selector position.	Bit.	Selector position.
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 position.	Bit.	Selector position.
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	- - -

PART 4.
CONCEPTS.

FAGOR
AUTOMATION

GENERAL SETUP.

7

7.1 Configuring the channels.

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. In a multichannel system, the channels can act independently or together; this means they can communicate, synchronize and carry out coordinated actions.

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; etc.

Operation of a channel.

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, arithmetic parameters and, if necessary, they may be synchronized via part-program. Each channel can only control its own axes and spindles, however, when using a part-program or MDI/MDA mode it can command the movements of axes or spindles from other channels.

The axes and spindles of a channel.

In order to be able to move an axis or spindle, it must be assigned to a channel. The axes and spindles of one channel can act independently or in parallel with the other 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.

7.1.1 Configuring the channels.

Configuring the number of channels.

Parameter.		Meaning.
NCHANNEL	MPG	Number of CNC channels. The maximum number of channels can vary depending on the CNC model.

[MPG]....... Machine parameters; general.

The NCHANNEL parameter must be modified directly in the parameter tree (left panel of the table), by adding or deleting the necessary channels from the "General" branch. This parameter cannot be modified directly in the table.

7.1.2 Variables.

The following variables may be accessed from (PRG) the part-program and from the MDI/MDA mode, PLC and (INT) from an external application. For each variable, the table indicates whether the access is reading (R) or writing (W). PLC access to the variables, for both reading and writing, will be synchronous. Access to the variables from the part program returns the value for the block preparation (it does not stop the preparation).

Variable.	PRG	PLC	INT
(V.)MPG.NCHANNEL Number of CNC channels. Units: -.	R	R	R

Syntax of the variables (from the part-program or MDI/MDA mode).

```
V.MPG.NCHANNEL
```

7.2 Configure the name and number of axes and spindles.

Valid names for axes and spindles.

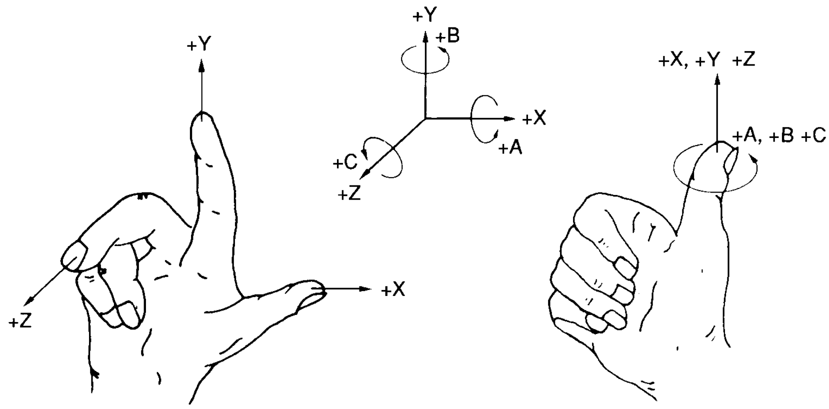
The axis name is defined by 1 or 2 characters. The first character must be one of the letters X - Y - Z - U - V - W - A - B - C. The second character is optional and will be a numerical suffix between 1 and 9. This way, the name of the spindles may be within the range X, X1…X9,...C, C1…C9. For example 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. DIN 66217.

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.



7.2.1 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·.

7.2.2 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·.

7.3 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.

7.3.1 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 Y axis.
G19	Y-Z plane. Perpendicular axis X.
G20	Any plane and/or longitudinal axis.

7.3.2 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 Y axis.
G19	Y-Z plane. Perpendicular axis X.
G20	Any plane and/or longitudinal axis.

7.3.3 Milling machine with 2 channels, 7 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).

General configuration of the channels, axes and spindles.

System.	Channel ·1·	Channel ·2·
NCHANNEL = 2		
NAXIS = 7	CHNAXIS = 3	CHNAXIS = 4
NSPDL = 2	CHNSPDL = 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	X1	Logic number ·4·.	
AXISNAME 5	Y1	Logic number ·5·.	
AXISNAME 6	Z1	Logic number ·6·.	
AXISNAME 7	W	Logic number ·7·.	
SPDLNAME 1	S1	Logic number ·8·.	Spindle with index ·1·.
SPDLNAME 2	S2	Logic number ·9·.	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 ·8·.

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 ·9·.

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.

7.3.4 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.

7.3.5 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.

7.3.6 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 Y axis.
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).

7.3.7 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.

7.3.8 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·.

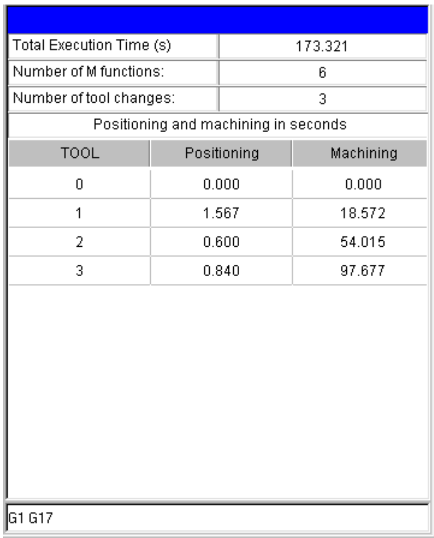
With this configuration, the work plane is always G18 and will be formed by the first two axes defined in the channel. If the X (firs axis of the channel) and Z (second axis of the channel) have been defined, the work plane will be the ZX (Z as abscissa and X as ordinate). Functions G17 and G19 are not valid. Since there are only two axes, function G20 makes no sense.

7.4 Execution time estimate.

In EDISIMU work mode, there is an option that can calculate the time required by the CNC to execute a part with the machining conditions defined by 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 or T function or a home search of one or several axes at a time.

Statistics window.

Softkey



7.4.1 Defining the estimated execution time of some functions.

H, M and T functions and home search.

Parameter.		Meaning.
REFTIME	MPG	Estimated homing time.
HTIME	MPG	Estimated time to execute an H function.
DTIME	MPG	Estimated time to execute a D function.
TTIME	MPG	Estimated time to execute a T function.
SPDLTIME	MPA	Estimated time to execute an S function.
MTIME	MPM	Estimated time to execute the M function.

[MPG] ...... Machine parameters; general.
[MPA] .... Machine parameters; axes.
[MPM] ...... Machine parameters; M function table.

7.4.2 Variables.

The following variables may be accessed from (PRG) the part-program and from the MDI/MDA mode, PLC and (INT) from an external application. For each variable, the table indicates whether the access is reading (R) or writing (W). PLC access to the variables, for both reading and writing, will be synchronous. Access to the variables from the part program returns the value for the block preparation (it does not stop the preparation).

Variable.	PRG	PLC	INT
(V.)MPG.REFTIME Estimated homing time. Units: Milliseconds.	R	R	R
(V.)MPG.HTIME Estimated time to execute an H function. Units: Milliseconds.	R	R	R
(V.)MPG.DTIME Estimated time to execute a D function. Units: Milliseconds.	R	R	R
(V.)MPG.TTIME Estimated time to execute a T function. Units: Milliseconds.	R	R	R
(V.)MPA.SPDLTIME.sn Estimated time to execute an S function. Units: Milliseconds.	R	R	R
(V.)MPM.MTIME[pos] Estimated time to execute the M function. Units: Milliseconds.	R	R	R

Syntax of the variables (from the part-program or MDI/MDA mode).

·pos· Position inside the "M" function table.

·sn· Name, logic number or index of the spindle.

V.MPG.REFTIME	
V.MPG.HTIME	
V.MPG.DTIME	
V.MPG.TTIME	
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.MPM.MTIME[12]	Position ·12· of the "M" function table.

7.5 Zero offsets.

7.5.1 Configuring zero offsets.

The zero offset may have two different looks, with or without fine setting of absolute zero offset.

- Absolute zero offset table (without fine setting of the zero offset). Each zero offset has a single value. When activating a zero offset (function G159), the CNC assumes this value as the new zero offset.

Channel 1 : Zero offsets			
Origin	X (mm)	Y (mm)	Z (mm)
PLCOF	00000.0000	00000.0000	00000.0000
G158	00054.5000	00010.0000	00000.0000
G54 (G159=1)	00000.0000	00000.0000	00000.0000
G55 (G159=2)	00000.0000	00000.0000	00000.0000
G56 (G159=3)	00000.0000	00000.0000	00000.0000

- Absolute zero offset table (with fine setting of the zero offset). Each zero offset has a coarse (or absolute) value and a fine (or incremental) value. When activating an offset (function G159), the CNC assumes as new zero offset the sum of both parts.

Channel 1 : Zero offsets			
Origin	X (mm)	Y (mm)	Z (mm)
PLCOF	00000.0000	00000.0000	00000.0000
G158	00054.5000	00010.0000	00000.0000
G54 (G159=1)	00050.0000	00000.0000	00000.0000
△	00003.0000	00000.0000	00000.0000
G55 (G159=2)	00000.0000	00000.0000	00000.0000
△	00000.0000	00000.0000	00000.0000
G56 (G159=3)	00010.0000	00000.0000	00000.0000
△	00000.0000	00000.0000	00000.0000

Table type for zero offsets.

Parameter.		Meaning.
FINEORG	MPG	Fine definition of zero offsets.

[MPG] ...... Machine parameters; general.

7.5.2 Variables.

The following variables may be accessed from (PRG) the part-program and from the MDI/MDA mode, PLC and (INT) from an external application. For each variable, the table indicates whether the access is reading (R) or writing (W). PLC access to the variables, for both reading and writing, will be synchronous. Access to the variables from the part program returns the value for the block preparation (it does not stop the preparation).

Variable.	PRG	PLC	INT
(V.)MPG.FINEORG Fine definition of zero offsets. Units: -.	R	R	R

Syntax of the variables (from the part-program or MDI/MDA mode).

```
V.MPG.NAXIS
```

FEEDBACKS.

8

8.1 Axis feedback resolution.

Feedback resolution of a linear axis with a linear encoder.

Parameter.		Meaning.
NPULSES	[MPFB]	Number of pulses per turn of the encoder; with a linear encoder, set to 0.
LINEAR_PITCH	[MPFB]	Encoder resolution (pitch).
PITCH	[MPFB]	Leadscrew pitch.

[MPFB] Feedback parameter.

Feedback resolution of a linear axis with a rotary encoder.

Parameter.		Meaning.
NPULSES	[MPFB]	Number of pulses (lines) per encoder turn.
LINEAR_PITCH	[MPFB]	Not being used.
PITCH	[MPFB]	Leadscrew pitch.

[MPFB] Feedback parameter.

Feedback resolution of a rotary axis o spindle with a rotary encoder.

Parameter.		Meaning.
NPULSES	[MPFB]	Number of pulses (lines) per encoder turn.
LINEAR_PITCH	[MPFB]	Not being used.
PITCH	[MPFB]	Degrees that the shaft rotates per encoder turn.

[MPFB] Feedback parameter.

8.2 Gear ratio.

The gear ratio between the motor and the encoder is defined by the parameters INPUTREV and OUTPUTREV. This gear ratio can also be entered directly using the parameters NPULSES and PITCH (not recommended); in this case, the parameters INPUTREV and OUTPUTREV will be set to 1. If the gear ratio is not a whole number, it is recommended to set the real values of the gear ratio for INPUTREV and OUTPUTREV instead of leaving a inaccurate number in PITCH and NPULSES

Gear ratio using a linear encoder.

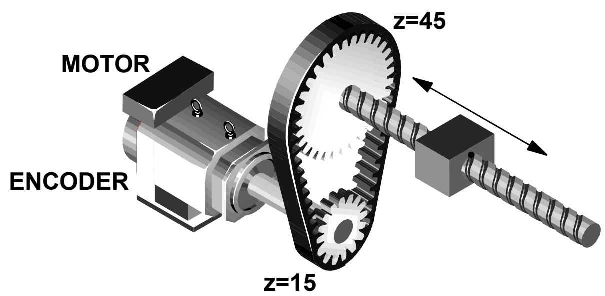
Parameter.		Meaning.
INPUTREV	[MPFB]	When using a linear encoder, the gear ratio will be 1:1 (both parameters set to 1).
OUTPUTREV	[MPFB]	

[MPFB] Feedback parameter.

Gear ratio using a rotary encoder and motor feedback.

Parameter.		Meaning.
INPUTREV	[MPFB]	Number of turns of the motor.
OUTPUTREV	[MPFB]	Number of turns of the axis moving the load.

[MPFB] Feedback parameter.

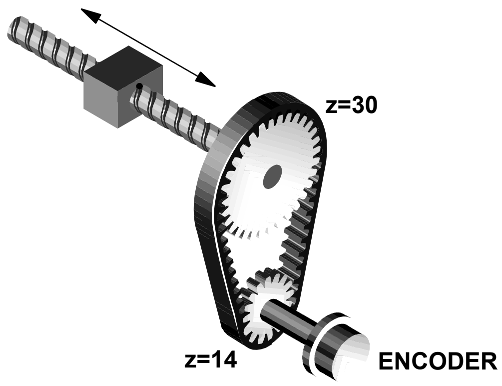


INPUTREV=15
OUTPUTREV=45

Gear ratio using a rotary encoder and direct feedback.

Parameter.		Meaning.
INPUTREV	[MPFB]	Number of turns of the encoder.
OUTPUTREV	[MPFB]	Number of turns of the axis moving the load.

[MPFB] Feedback parameter.



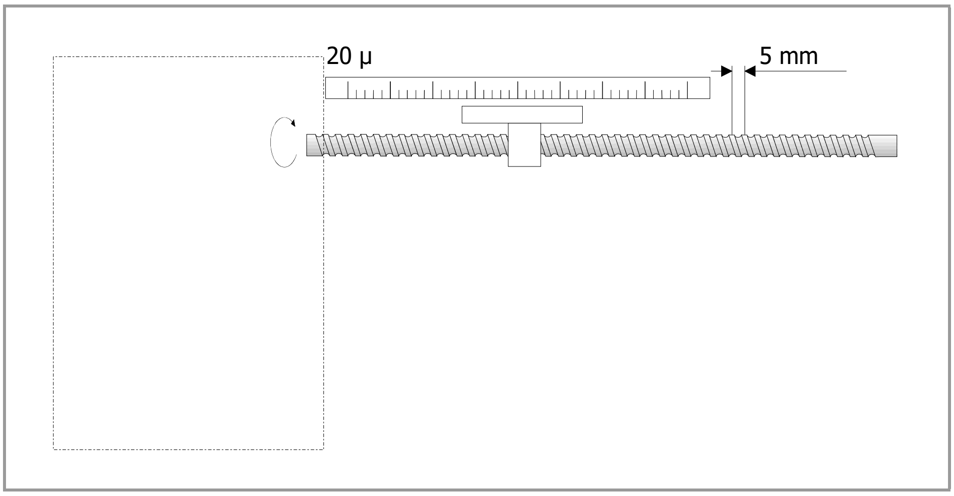
INPUTREV=14
OUTPUTREV=30

8.2.1 Feedback resolution of an axis and the gear ratio (examples).

Example 1.
Linear axis. Linear encoder. Direct feedback.

Parameter.	Meaning.
NPULSES	Number of pulses per turn of the encoder; with a linear encoder, set to 0.
LINEAR_PITCH	Encoder resolution (pitch).
PITCH	Leadscrew pitch.
INPUTREV	Gear ratio; using a linear encoder, set to 1.
OUTPUTREV	Gear ratio; using a linear encoder, set to 1.

For example, a linear axis with 5 mm pitch ballscrew and a linear encoder with a resolution of 20 µm.

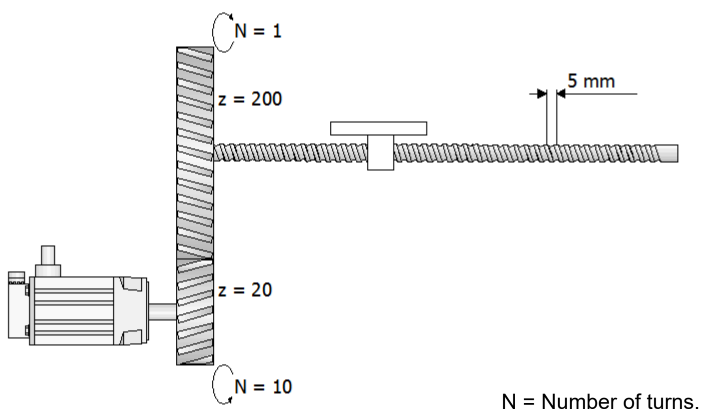


Parameter.	Value.
NPULSES	0
LINEAR_PITCH	0,020 mm.
PITCH	5 mm.
INPUTREV	1 turn.
OUTPUTREV	1 turn.

Example 2.
Linear axis. Rotary encoder. Motor feedback.

Parameter.	Meaning.
NPULSES	Number of pulses (lines) per encoder turn.
LINEAR_PITCH	Not being used.
PITCH	Leadscrew pitch.
INPUTREV	Gear ratio; number of turns of the motor.
OUTPUTREV	Gear ratio; number of turns of the ballscrew.

For example, a linear axis with a 5 mm pitch ballscrew and a rotary encoder of 90,000 pulses per turn.



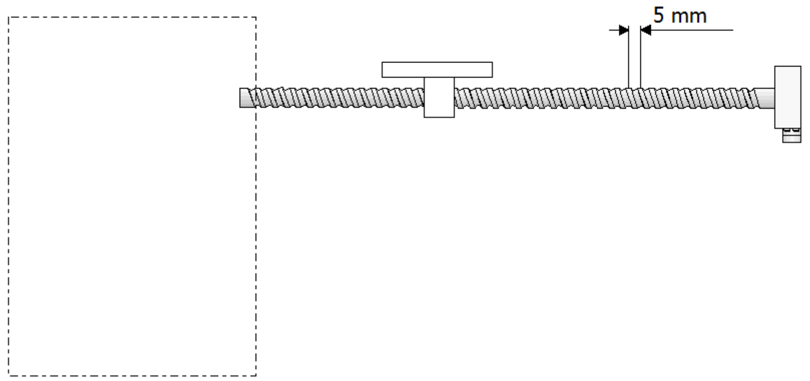
Parameter.	Value (option 1).	Value (option 2).
NPULSES	900 000 pulses/turn.	90 000 pulses/turn.
LINEAR_PITCH	Not being used.	Not being used.
PITCH	5 mm.	5 mm.
INPUTREV	1 turn.	10 turns.
OUTPUTREV	1 turn.	1 turn.

Option 1. The reduction is contemplated in the parameters NPULSES and PITCH.
Option 2. The reduction is contemplated in the parameters INPUTREV and OUTPUTREV.

Example 3.
Linear axis. Rotary encoder. Direct feedback (no reduction).

Parameter.	Meaning.
NPULSES	Number of pulses (lines) per encoder turn.
LINEAR_PITCH	Not being used.
PITCH	Leadscrew pitch.
INPUTREV	Gear ratio; number of turns of the encoder.
OUTPUTREV	Gear ratio; number of turns of the ballscrew.

For example, a linear axis with a 5 mm pitch ballscrew and a rotary encoder of 90,000 pulses per turn.

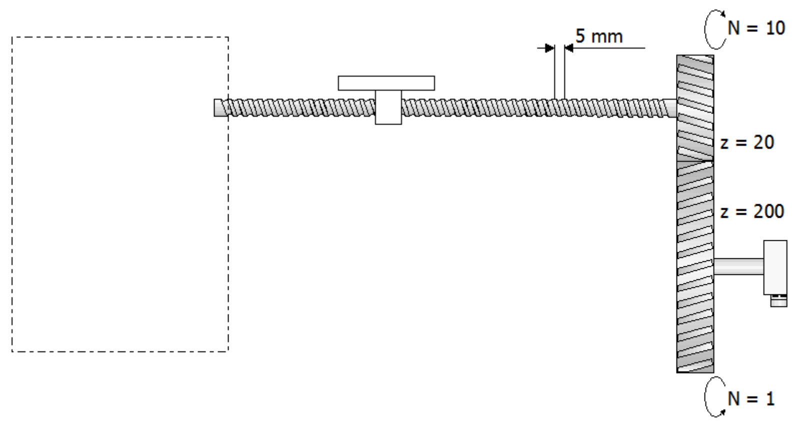


Parameter.	Value.
NPULSES	90 000 pulses/turn.
LINEAR_PITCH	Not being used.
PITCH	5 mm.
INPUTREV	1 turn.
OUTPUTREV	1 turn.

Example 4.
Linear axis. Rotary encoder. Direct feedback (with reduction).

Parameter.	Meaning.
NPULSES	Number of pulses (lines) per encoder turn.
LINEAR_PITCH	Not being used.
PITCH	Leadscrew pitch.
INPUTREV	Gear ratio; number of turns of the encoder.
OUTPUTREV	Gear ratio; number of turns of the ballscrew.

For example, a linear axis with a 5 mm pitch ballscrew and a rotary encoder of 90,000 pulses per turn.



N = Number of turns.
z = Number of teeth.

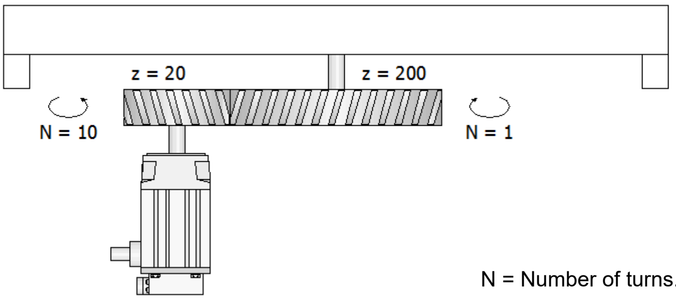
Parameter.	Value (option 1).	Value (option 2).
NPULSES	900 000 pulses/turn.	90 000 pulses/turn.
LINEAR_PITCH	Not being used.	Not being used.
PITCH	5 mm.	5 mm.
INPUTREV	1 turn.	1 turns.
OUTPUTREV	1 turn.	10 turn.

Option 1. The reduction is contemplated in the parameters NPULSES and PITCH.
Option 2. The reduction is contemplated in the parameters INPUTREV and OUTPUTREV.

Example 5.
Rotary axis. Rotary encoder. Motor feedback.

Parameter.	Meaning.
NPULSES	Number of pulses (lines) per encoder turn.
LINEAR_PITCH	Not being used.
PITCH	Degrees that the shaft rotates per encoder turn.
INPUTREV	Gear ratio; number of turns of the encoder.
OUTPUTREV	Gear ratio; number of turns of the ballscrew.

For example, a rotary axis and a rotary encoder at 36,000 pulses per turn.



N = Number of turns.
z = Number of teeth.

Parameter.	Value (option 1).	Value (option 2).
NPULSES	360 000 pulses/turn.	36 000 pulses/turn.
LINEAR_PITCH	Not being used.	Not being used.
PITCH	360 º.	36 º.
INPUTREV	1 turn.	10 turns.
OUTPUTREV	1 turn.	1 turn.

Option 1. The reduction is contemplated in the parameters NPULSES and PITCH.
Option 2. The reduction is contemplated in the parameters INPUTREV and OUTPUTREV.

8.3 Configuring the SSI protocol.

Type of SSI feedback.

Parameter.	Meaning.
SSITYPE	Type of SSI feedback. This parameter provides a list of available "Fagor" linear and rotary encoders. The "User" option allows for the customization of all the properties of the SSI communication. Depending on the type of selected feedback, some of the parameters associated with the feedback are preset and cannot be changed (a lock icon will appear alongside the parameter).

Clock frequency.

Parameter.	Meaning.
SSICLKFREQ	Frequency of the clock for SSI communication. The frequency depends on the length of the cable. • Up to 20 meters (65 feet); 600 kHz. • Up to 50 meters (164 feet); 400 kHz. • Up to 75 meters (246 feet); 300 kHz. • Up to 100 meters (328 feet); 250 kHz.

SSI package format.

Parameter.	Meaning.
SSIDATAFORMAT	SSI data format; binary code or Gray code.
SSIDATAMODE	Value of the first bit; LSB (Least Significant Bit)/MSB (Most Significant Bit).
SSIPACKFORMAT	This parameter indicates the order in which the different data types of the SSI transmission will be received. If start bits have been programmed, the CNC assumes that they will be received first
SSIDATALEN	Number of position or data bits.
SSICRCBITS	Number of bits that making up the valid transmission check (CRC, checksum, etc.) of the SSI transmission.
SSIALARMBITS	Number of alarm/acknowledge bits.

Example.
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 CRC bits.
SSIPACKFORMAT = Alarm+Data+CRC
SSICRCBITS = 5
SSIALARMBITS = 1
SSIDATABITS = 23

Digital counting (feedback) resolution.

Parameter.	Meaning.
SSIRESOL	Digital counting resolution or number of digital counting units contained in a pitch.

Example: Linear encoder.
Fagor absolute linear encoder with a pitch of 20 µm and a digital resolution of 0.1 µm.
SSIRESOL = 20 µm / 0,1 µm = 200.

Example: Rotary encoder.
Rotary encoder with 8192 pulses per turn and a ballscrew with a 10 mm pitch.
PITCH = 10 mm.
SSIRESOL = 8192.
The axis resolution will be; 10 / 8192 = 0.0012 mm.

Example: Inductosyn module.
For SSITYPE = ABSIND (Inductosyn ROT+ABS), for a 2 or 4-degree turn and depending on whether it is a high or low resolution encoder, the counting increment is 10 000 units.
PITCH = 2º or 4º.
SSIRESOL = 10 000
The resolution of the axis will be; 2º or 4º / 10 000 = 0.0002 or 0.0004 degrees.

CRC calculation.

Parameter.	Meaning.
SSICRCBITS	Number of bits that making up the valid transmission check (CRC, checksum, etc.) of the SSI transmission.
SSIDATACHECKTYPE	CRC type of the SSI feedback. 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. **Do not calculate.** Even if the CNC receives the CRC bits, they are not processed at all by the CNC and, therefore, no errors are reported when the transmission is erroneous. It is not a recommendable option and it must only be used during setup. **Fagor.** Data checking algorithm used in Fagor Automation feedback systems. **Inductosyn.** Data checking algorithm used when connecting to an Inductosyn module. **Even parity.** Even parity bit method. This method detects errors, however, it does not correct them. **Odd parity.** Odd parity bit method. This method detects errors, however, it does not correct them.

Alarm bits

Parameter.	Meaning.
SSIALARMBITS	Number of alarm bits. The feedback device can send one or more bits indicating an alarm condition or one or more acknowledgment bits if the transmission is successful.
SSIALARMLEVEL	Value that the alarm/acknowledge bits for an error condition to occur. This parameter only makes sense when SSIALARMBITS is other than 0

Example 1.

The encoder indicates an error condition using 2 bits, where the first is a 1 and the second a 0.

SSIALARMBITS = 2

SSIALARMLEVEL = 2.

Example 2.

The encoder has a correct acknowledgement transmission bit and indicates it with a logical 0.

SSIALARMBITS = 1

SSIALARMLEVEL = 1. Because a 0 means that everything is OK.

8.4 Configuring a Sensorless system.

Thanks to this application, a SENSORLESS motor can achieve the same dynamic performance as the asynchronous motor with motor feedback except at speeds under 30 rpm.

A Sensorless system controls the speed of an asynchronous motor without motor feedback (without position/speed feedback). The current loop carries out all the necessary calculations to obtain the motor speed, based on the current signals. The Quercus servo drive system will use this information to recreate a real feedback.

This feature is mostly for low-end non-Fagor motors and it may be used on both spindle motors and axis motors. The interest in this application is usually based on costs, since sensorless motors are less expensive as they do not have a motor feedback or feedback cable. Also, it allows the user to troubleshoot possible poor feedback performance.

Parameter setting for a Sensorless motor.

Identify the asynchronous motor as Sensorless motor.

Parameter.		Meaning.
MOT_CTRL_TYPE	[MOT]	Type of control of the asynchronous motor.

[MOT] motor parameter.

Setting up a sensorless feedback.

Creating a “virtual” feedback that is not physically associated with any connector, since it will be the current loop that will calculate the motor position. Feedback parameters, such as INPUTREV, OUTPUTREV and INVERT, can be used for velocity and position loops.

Parameter.		Meaning.
FB_MODEL	[MPFB]	List of available protocols and encoders.

[MPFB] Feedback parameter.

Associating an axis and spindle to the motor and feedback.

Parameter.		Meaning.
REFERENCEID	[SPDL]	Motor identifier.
CURRFBID	[SPDL]	Feedback identifier of the current loop.

[SPDL] Axis or spindle parameter.

Setup.

The following example shows how to define the S spindle as a Sensorless system.

- Make a new motor available, for example MOTOR_S, and adjust the following parameter.

Parameter.		Value.
MOT_CTRL_TYPE	[MOT]	Sensorless.

[MOT] motor parameter.

- Make a new feedback available, for example ENC_SENSORLESS, and adjust the following parameter.

Parameter.		Value.
FB_MODEL	[MPFB]	Sensorless.

[MPFB] Feedback parameter.

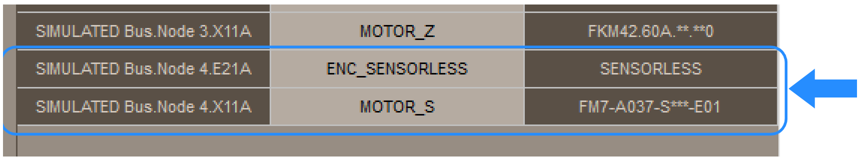
- Adjust the following parameters and the spindle set.

Parameter.		Value.
REFERENCEID	[SPDL]	MOTOR_S.
CURRFBID	[SPDL]	ENC_SENSORLESS
SPEEDFBID	[SPDL]	Assigns a feedback with a feedback device or leaves the parameter unset, in which case the spindle assumes the feedback selected in CURRFBID.
POSFBID	[SPDL]	Assigns a feedback with a feedback device or leaves the parameter unset, in which case the spindle assumes the feedback selected in CURRFBID.

[SPDL] ..... Spindle parameter.

- In the connection table, associate the motor (MOTOR_S) to its physical connector.
- The feedback (ENC_SENSORLESS) does not have to be associated with any physical connector. When it is sensorless, the system associates REFERENCEID and CURRFBID; in this example, MOTOR_S with ENC_SENSORLESS.

Connections table.

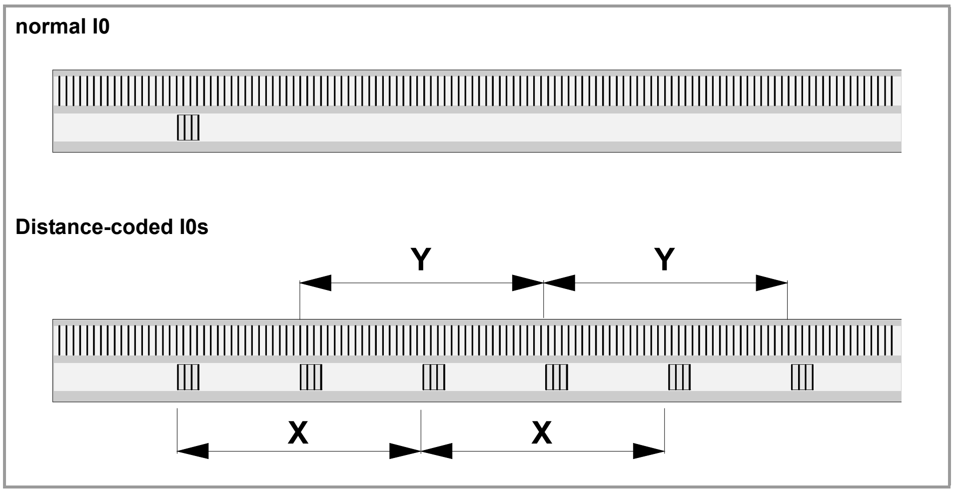


8.5 I0 signal.

Parameter.	Meaning.
I0TYPE	Reference mark (I0) type. This parameter indicates where the I0 counting (encoder reference marks) is in relation to the direction of movement during the home search. **Normal.** The encoder can have of one or more I0 marks, but there is only one selected mark (for example, with a micro (home-switch)). The encoder references the position using this I0 mark. Depending on the initial position, long movements may be necessary until the position is homed. **Increasing distance-coded.** The encoder has several I0 marks, separated at different distances, following a mathematical formula. The encoder refers to the position after exceeding two contiguous I0 marks, that is, after a few millimeters. The I0 codification will be increased when the distance between the I0 marks increases according to the direction of movement during the home search. **Decreasing distance-coded.** The encoder has several I0 marks, separated at different distances, following a mathematical formula. The encoder refers to the position after exceeding two contiguous I0 marks, that is, after a few millimeters. The I0 codification will be decreased when the distance between the I0 marks decreases according to the direction of movement during the home search. **Not evaluated.** The CNC does not use the I0 of the encoder. If the parameter DECINPUT=Yes, the CNC uses the home switch.
REFPULSE	Flank of the marker pulse (I0) used by the CNC to complete the home search. For 1 Vpp signals, always select “Positive”. **None.** The CNC does not evaluate the marker pulse (I0); it considers that the position is homed when the axis releases the home switch. This option is valid when accuracy is not an important factor; for example, the position of a magazine, feeders, etc. **Positive.** The CNC considers that the position is homed when it receives the up-flank of the marker pulse (I0). **Negative.** The CNC considers that the position is homed when it receives the down flank of the marker pulse (I0).
REFSYNC	Position of the I0 signal compared to the encoder A and B signals.
I0CODDI1	Pitch (number of waves) between 2 fixed I0 signals.
I0CODDI2	Pitch (number of waves) between 2 variable I0 signals.
EXTMULT	External multiplying factor. 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.
Distance between fixed I0s; 20 000 mm.
Distance between variable I0s; 20 020 mm.
Period of the sinusoidal signal; 20 µm.
Number of waves between fixed I0s; I0CODDI1 = 20 000 / (20 x EXTMULT) = 1000.
Number of waves between variable I0s; I0CODDI2 = 20 020 / (20 x EXTMULT) = 1001.

Glass graduation period; 100 µm.
Counting signal period; 4 µm.
External multiplying factor; EXTMULT = 100 / 4 = 25


normal I0
Distance-coded I0s
Y
Y
X
X

8.6 Feedback alarm.

Parameter.	Meaning.
FBACKAL	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(axis) mark off.

8.7 Direction of the setpoint and positive feedback.

Digital servo drives.

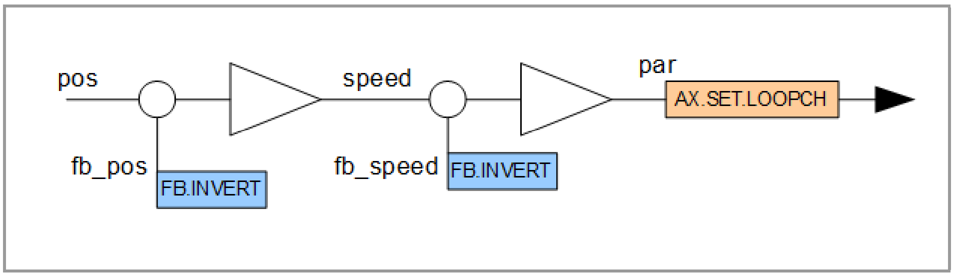
The parameters that control the direction of movement and feedback are as follows.

Parameter.		Meaning.
LOOPCH	[MPA]	Change the sign of the setpoint.
INVERT	[MPFB]	Changing the counting sign.

[MPFB] Feedback parameter.
[MPA] .... Machine parameter; axis.

Axis with a single feedback (SPEEDFBID = POSITIONFBID).

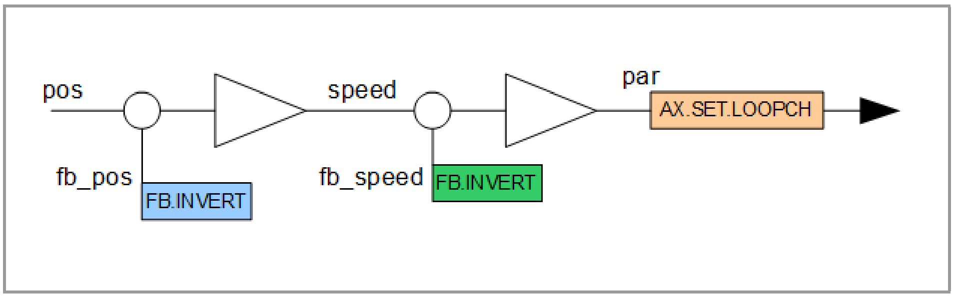
If the system has only one feedback device for the first and second feedback, both parameters can be used.



- If the axis is packed, change one of the two parameters (LOOPCH or INVERT) and the axis will close the loop without packing.
- If the axis moves in the wrong direction, change the two parameters (LOOPCH and INVERT).

Axis with two feedbacks (SPEEDFBID ≠ POSITIONFBID).

If the system has one feedback device for the first and a different one for the second feedback, then three parameters can be used (axis LOOPCH, INVERT of the first feedback and INVERT for the second feedback). First, it is recommended to close the loop with only one feedback to determine if the movement is in the intended direction and then close the loop with the second feedback.



- If the first feedback is adjusted properly and if the axis is packed, change the INVERT parameter of the feedback associated with the position loop and the axis will close the loop without packing.
- If the axis is properly controlled but moves in the wrong direction, change the three parameters (axis LOOPCH, INVERT for the first feedback and INVERT for the second feedback).

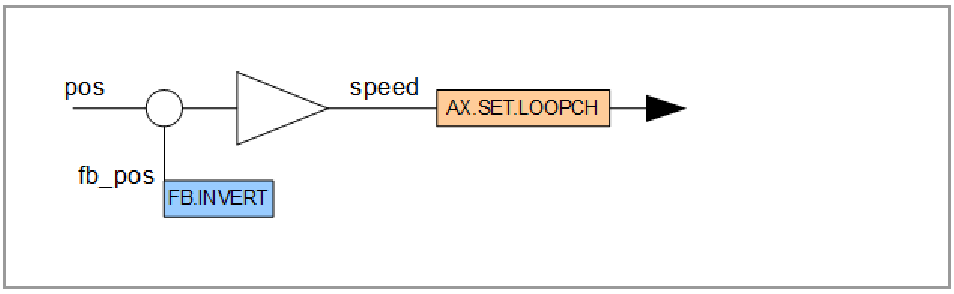
Analog drive.

The parameters that control the direction of movement and feedback are as follows.

Parameter.		Meaning.
LOOPCH	[MPA]	Change the sign of the setpoint.
INVERT	[MPFB]	Changing the counting sign.

[MPFB] Feedback parameter.
[MPA] .... Machine parameter; axis.

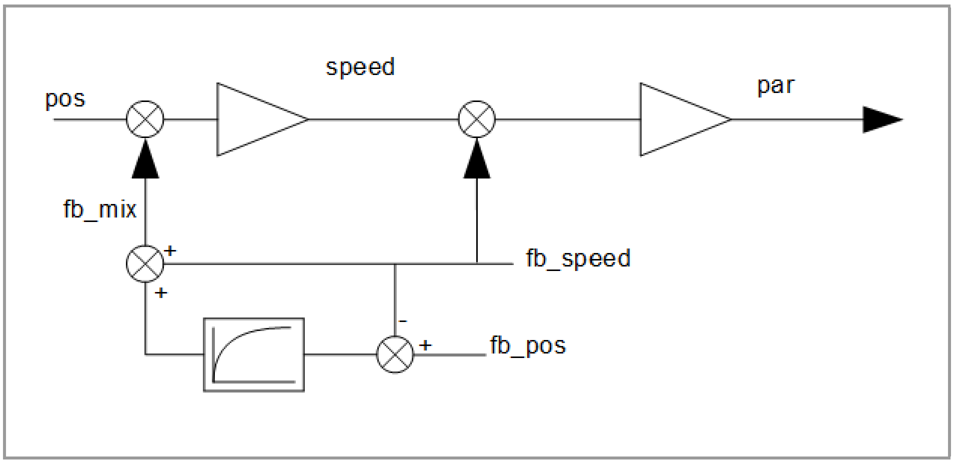
Basic control loop diagram for analog axes and spindles. In this case, the LOOPCH parameter acts on the velocity command.



- If the axis is packed, change one of the two parameters (LOOPCH or INVERT) and the axis will close the loop without packing.
- If the axis moves in the wrong direction, change the two parameters (LOOPCH and INVERT).

Feedback combination.

Conceptually, the position loop closes in the following way.



A first order filter, constant FBMIXTIME, is applied to the difference in feedback.

FBMIXTIME=0. The position loop closes with the position feedback.

By default, FBMIXTIME is set to 0, which means that the filter has a gain of 1. For the resulting equation fb_speed is canceled (subtracted and added), so the position loop closes with fb_pos.

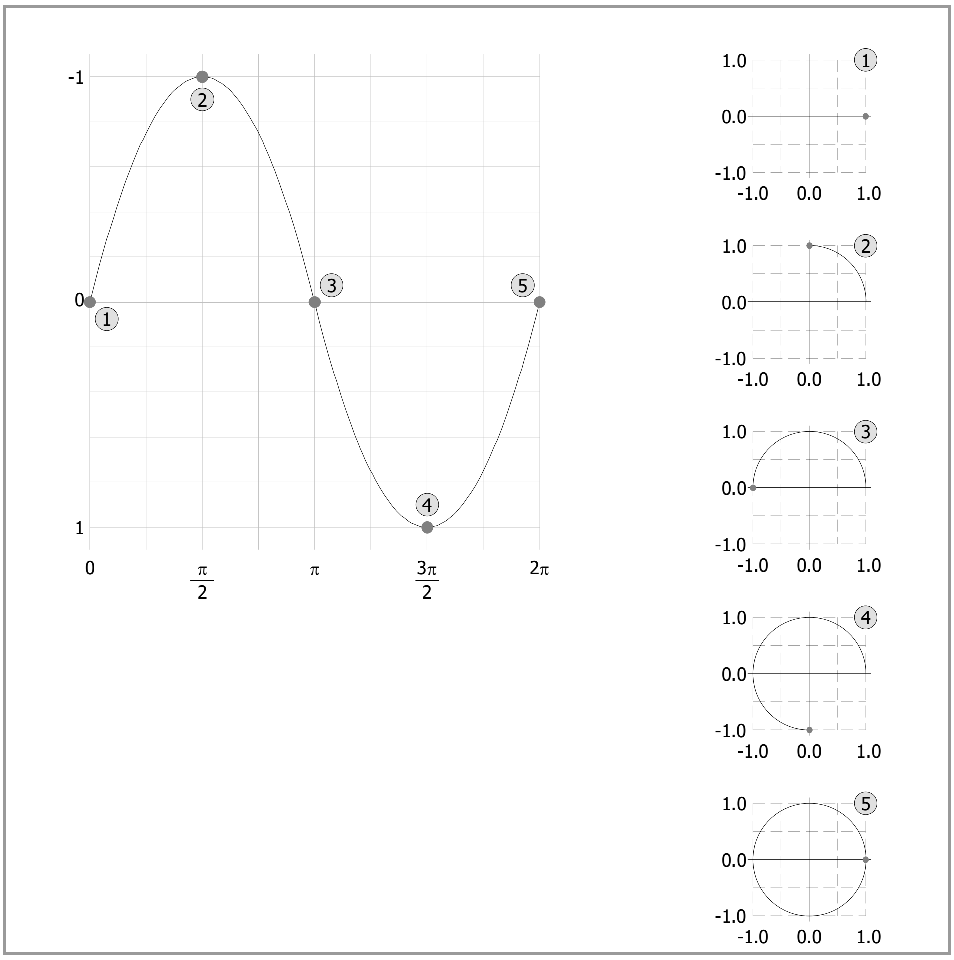
fb_mix = fb_pos-fb_speed + fb_speed

FBMIXTIME≠0. The position loop closes with a combined feedback

If FBMIXTIME has a non-zero value, the loop closes with a combined feedback. When both feedbacks are different, during the time defined in FBMIXTIME, the loop closes with the speed feedback; after which, it will close with the position feedback.

8.8 Encoder offset adjustment (circle adjustment).

After identifying and initializing the motor, it may make a high-pitch noise due to a certain misadjustment of the feedback signal generation. Although the encoder is properly factory set, its connection to the drive (cable, connector, etc.) can distort these signals. To solve this problem, and thus achieve a quieter motor operation and improve control, adjust the offset and the gains used by the drive software to process the signals provided by the feedback device. The circle setting can be done manually or automatically (parameter SC_ADJ_MODE). This procedure can only be applied to encoders, not to resolvers.



Adjusting the offset and feedback gains.

Parameter.	Meaning.
SC_SIN_GAIN [MPFB]	Compensation (proportional gain mode) of the amplitude of the feedback sine signal. In an ideal case, the gain value should be 1. This parameter is only valid when the circle of the sine and cosine signals are set manually (parameter SC_ADJ_MODE).
SC_COS_GAIN [MPFB]	Compensation (proportional gain mode) of the amplitude of the feedback cosine signal. In an ideal case, the gain value should be 1. This parameter is only valid when the circle of the sine and cosine signals are set manually (parameter SC_ADJ_MODE).
SC_SIN_OFF [MPFB]	Compensation (offset mode) of the feedback sine signal. In an ideal case, the offset value should be 0. This parameter is only valid when the circle of the sine and cosine signals are set manually (parameter SC_ADJ_MODE).
SC_COS_OFF [MPFB]	Compensation (offset mode) of the feedback cosine signal. In an ideal case, the offset value should be 0. This parameter is only valid when the circle of the sine and cosine signals are set manually (parameter SC_ADJ_MODE).
SC_ADJ_MODE [MPFB]	Circle adjustment of the encoder sine and cosine signals. **None.** There is no circle adjustment. The CNC does not compensate the offset or gains; it ignores the parameters SC_SIN_GAIN, SC_COS_GAIN, SC_SIN_OFF and SC_COS_OFF. **Automatic.** The CNC adjusts the circle automatically and continuously. The result of the adjustment is not reflected in the parameters SC_SIN_GAIN, SC_COS_GAIN, SC_SIN_OFF or SC_COS_OFF. **Manual** Manual circle adjustment using the parameters SC_SIN_GAIN, SC_COS_GAIN, SC_SIN_OFF and SC_COS_OFF.

[MPFB] Feedback parameter.

Circle adjustment.

The adjustment of the circle allows to adjust the treatment of the feedback signals (adjust their offset and gains), so that the A and B signals managed by the drive come closer to the $\sin\theta$ and $\cos\theta$ functions. This is called “circle adjustment” because the sine and cosine signals managed by the software must be mathematically correct, in other words they generate a perfect circumference. The gain and offset adjustments compensate the amplitude and offset of the A and B signals with respect to the $\sin\theta$ and $\cos\theta$ functions. The ideal values for the gain and the offset are 1 and 0 respectively.

Manual adjustment process.

1 Define the SC_ADJ_MODE parameter using the “Manual” value and validate the machine parameter table.

2 Define the parameters SC_SIN_GAIN, SC_COS_GAIN, SC_SIN_OFF and SC_COS_OFF.

Automatic adjustment process.

1 Define the SC_ADJ_MODE parameter using the “Automatic” value and validate the machine parameter table.

2 The CNC continuously carries out this adjustment.

HANDWHEELS.

9

9.1 Configuring the system handwheels.

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 9 handwheels.

General handwheel.	A general handwheel can move any machine axis. To move an axis with a general handwheel, first select the axis to be moved from the operator panel; if no axis is selected, the handwheel will not perform any action.
Individual handwheel.	An individual handwheel can only move the axis it is associated with (MPGAXIS parameter). To move an axis with an individual handwheel, no prior axis selection is needed.

Compatibility between handwheels.

- An axis may be moved indistinctly with its individual handwheel or with a general handwheel.
- If there are several axes selected while 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.

Configuring a handwheel as "feed handwheel".

The feed flywheel allows the use of one of the handwheels of the machine (general or individual) to control the feedrate of the axis, depending on how fast the handwheel is turned. Any individual or general handwheel can work as a feed handwheel. See *"9.1.2 Configure a handwheel as "feed handwheel"."* on page 530.

9.1.1 Configuring the handwheels (individual or general).

Configuring the handwheel type (individual or general).

Parameter.		Meaning.
MPGAXIS	MPMPG	Name of the axis associated with the 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.

[MPMPG]. Connections; handwheels.

Configuring the handwheel signal.

Parameter.		Meaning.
HWFBTYPE	MPMPG	Type of signal associated with the handwheel; Differential TTL / TTL.
MPGDIRECT	MPMPG	Handwheel turning direction. If the turning direction is correct, leave the parameter as it is; to reverse it, change the parameter.

[MPMPG]. Connections; handwheels.

The turning direction of the handwheel can also be defined by changing the sign of the resolution (MPGRESOL parameter).

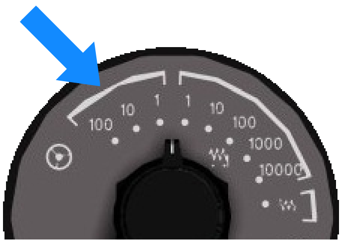
Handwheel resolution.

Parameter.		Meaning.
MPGRESOL	MPA	Table for handwheel resolution.
MPGRESOL1	MPA	Handwheel resolution in the 1 position of the switch.
MPGRESOL2	MPA	Handwheel resolution in the 10 position of the switch.
MPGRESOL3	MPA	Handwheel resolution in the 100 position of the switch.
MPGFILTER	MPA	Filter time for the handwheel. The filter softens the movements of the handwheel by averaging the pulses received in the last n CNC cycles. The parameter indicates how many CNC cycles the filter uses to average the pulses.

[MPA] ...... Machine parameters; axes and spindles.

The resolution of the handwheel sets the distance that the axis moves per handwheel pulse. The CNC can define a different resolution for each of the switch positions. A negative resolution inverts the direction of movement (opposite direction to that indicated by the A and B signals of the handwheel).

Handwheel switch.



Examples for a 100 handwheel with 100 increments (positions per turn).

- 100 pulse/turn handwheel (1 pulse for each flywheel increment).
  MPGRESOL1 = 0.0010 mm. The axis moves 0.0010 mm per increment of the handwheel.
- 25 pulse/turn handwheel (1 pulse for every four handwheel increments).
  MPGRESOL1 = 0.0010 mm. The axis moves 0.0010 mm for four handwheel increments.
  MPGRESOL1 = 0.0040 mm. The axis moves 0.0010 mm per increment of the handwheel.
- 200 pulse/turn handwheel (2 pulses for each handwheel increment).
  MPGRESOL1 = 0.0010 mm. The axis moves 0.0020 mm per increment of the handwheel.
  MPGRESOL1 = 0.0005 mm. The axis moves 0.0010 mm per increment of the handwheel.

Selecting an axis sequentially for jogging it with the handwheel.

PLC signal.		Meaning.
NEXTMPGAXIS	PLC_M (R/W)	This mark may be used to select an axis sequentially for jogging it with the handwheel. The CNC only takes into account this mark when it is in manual mode and the selector is in handwheel mode. Each time the PLC activates this mark (change from 0 to 1), the CNC behaves as follows. • If no axis has been selected, the CNC selects the first axis. • If there is an axis selected, the CNC selects the next one; if the last axis is selected, it selects the first one again. 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.

[PLC_M] PLC Mark.   (R) Consultation signal.   (R/W) Modifiable signal.

The CNC can only select the axes that are displayed in the active channel, regardless of the channel they belong to. The axes of another channel cannot be selected if they are not being displayed. An axis is de-selected when quitting the handwheel mode using the movement selector and after a reset.

Inhibiting handwheels.

Parameter.		Meaning.
INHIBITMPG1 ·· INHIBITMPG9	PLC_M (R/W)	If the PLC activates one of these marks (change from 0 to 1), it disables the corresponding handwheel. If the handwheel is disabled, the CNC ignores the pulses coming from the handwheel; the axis does not move and it does not save the pulses in the corresponding variable. See "9.1.3 Variables." on page 532. 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. The PLC has a mark for each handwheel, the INHIBITMPG1 mark disables the first handwheel, the INHIBITMPG2 mark the second one and so on.
ENABLE(axis)	PLC_M (R)	The CNC activates this mark (change from 0 to 1) to indicate to the PLC that it will move the corresponding axis or spindle in closed loop.

[PLC_M] PLC Mark.   (R) Consultation signal.   (R/W) Modifiable signal.

9.1.2 Configure a handwheel as "feed handwheel".

Usually, when machining a part for the first time, the feedrate is controlled with the switch on the operator panel. The feed handwheel can use one of the machine handwheels (general or individual) to control the feedrate depending on how fast the handwheel is turned. 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.

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.

Variables.	Meaning.
(V.)G.HANDP[hw]	Number of pulses sent by the handwheel since the system was started up.

PLC routine.

The PLC routine 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 routine. 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, the PLC
    complements the M1000 mark.
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
```

9.1.3 Variables.

The following variables may be accessed from (PRG) the part-program and from the MDI/MDA mode, PLC and (INT) from an external application. For each variable, the table indicates whether the access is reading (R) or writing (W). PLC access to the variables, for both reading and writing, will be synchronous. Access to the variables from the part program returns the value for the block preparation (it does not stop the preparation), except when indicated otherwise.

Variables.	PRG	PLC	INT
(V.)MPMPG.MPGAXIS.hw Logic number of the axis associated with the handwheel. If the variable returns a 0 value (zero), it is a general handwheel that can move any axis. Units: -.	R	R	R
(V.)MPMPG.HWFBTYPE.hw Type of signal associated with the handwheel input. Units: -.	R	R	R
(V.)MPMPG.MPGDIRECT.hw Handwheel turning direction. Units: -.	R	R	R
(V.)G.HANDP[hw] Number of pulses sent by the handwheel since the system was started up. Units: Pulses.	R(*)	R	R

(*) The CNC evaluates the variable during execution (it stops the block preparation).

Syntax of the variable.

·hw· Handwheel number.

(V.)G.HANDP[1]	Handwheel ·1·.

CONFIGURING AN AXIS.

10

10.1 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.

10.1.1 Linearlike rotary axis.

This type of rotary axis can turn in both directions. 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 POSLIMIT and NEGLIMIT set the travel limits for the axis; there are no module limits.

Parameter.		Value.
AXISTYPE	[MPA]	Rotary.
AXISMODE	[MPA]	Linearlike.
UNIDIR	[MPA]	Not being used.
SHORTESTWAY	[MPA]	Not being used.
POSLIMIT NEGLIMIT	[MPA]*	Travel limits. Limits to display the position values.
MODUPLIM MODLOWLIM	[MPA]*	Not being used.

[MPA] .... Machine parameter; axis.
[MPA] ...... Machine parameter; axis set.*

FAGOR AUTOMATION

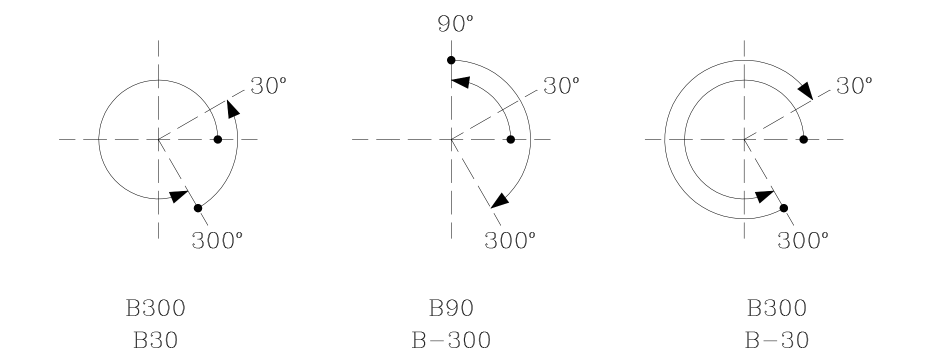
CNCelite 8058 8060 8065 8070

REF: 2210

10.1.2 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.

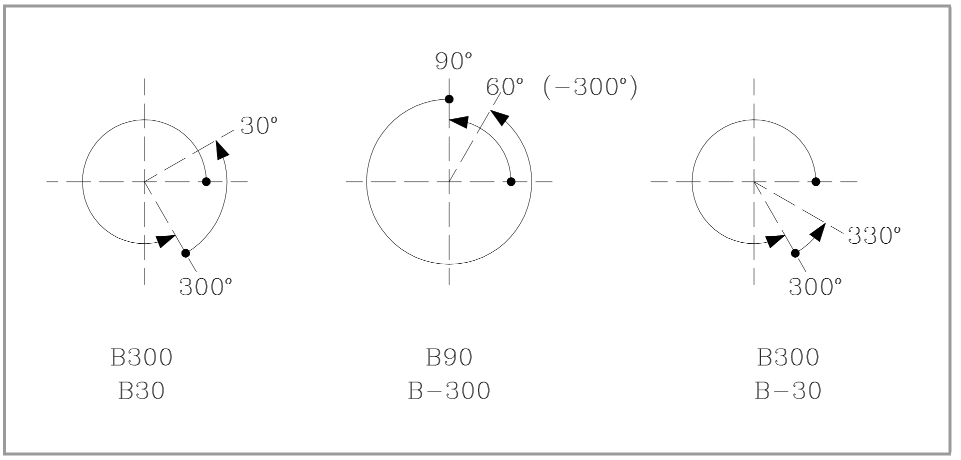
Parameter.		Value.
AXISTYPE	[MPA]	Rotary.
AXISMODE	[MPA]	Module.
UNIDIR	[MPA]	No.
SHORTESTWAY	[MPA]	No.
POSLIMIT NEGLIMIT	[MPA]*	Not being used. There are no travel limits.
MODUPLIM MODLOWLIM	[MPA]*	Limits of the module. Limits to display the position values.

[MPA]....... Machine parameter; axis.
[MPA]...... Machine parameter; axis set.*

10.1.3 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.

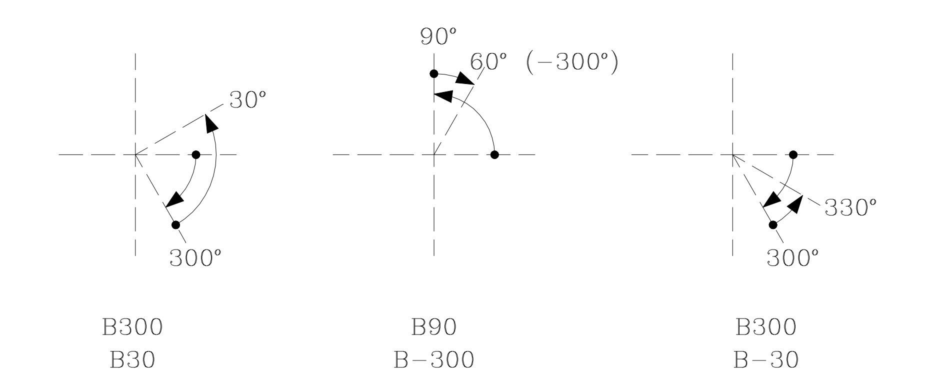
Parameter.		Value.
AXISTYPE	[MPA]	Rotary.
AXISMODE	[MPA]	Module.
UNIDIR	[MPA]	Positive / Negative.
SHORTESTWAY	[MPA]	No.
POSLIMIT NEGLIMIT	[MPA]*	Not being used. There are no travel limits.
MODUPLIM MODLOWLIM	[MPA]*	Limits of the module. Limits to display the position values.

[MPA] .... Machine parameter; axis.
[MPA] ...... Machine parameter; axis set.*

10.1.4 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	[MPA]	Rotary.
AXISMODE	[MPA]	Module.
UNIDIR	[MPA]	No.
SHORTESTWAY	[MPA]	Yes.
POSLIMIT NEGLIMIT	[MPA]*	Not being used. There are no travel limits.
MODUPLIM MODLOWLIM	[MPA]*	Limits of the module. Limits to display the position values.

[MPA]....... Machine parameter; axis.
[MPA]...... Machine parameter; axis set.*

10.2 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.

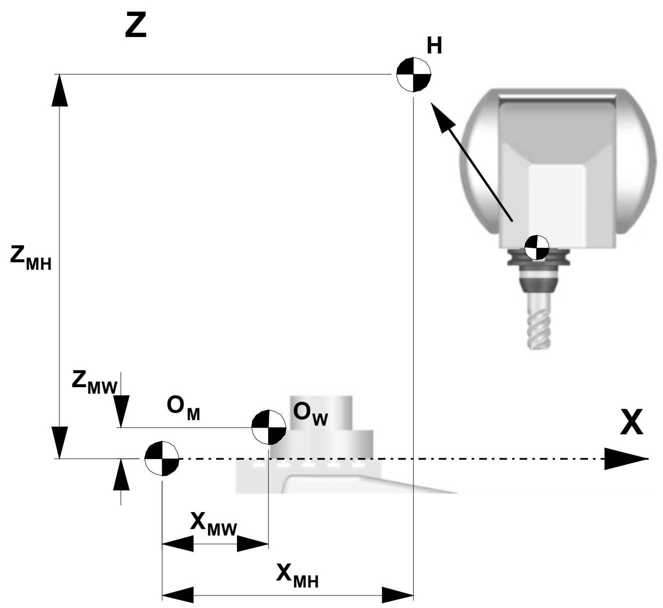
- Manual home search, in JOG mode. The axes are homed one by one. The CNC does not keep the part zero and the coordinates are displayed referred to machine reference zero.
- Automatic home search, in JOG mode. This kind is only available when the home search subroutine associated with function G74 (parameter 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.

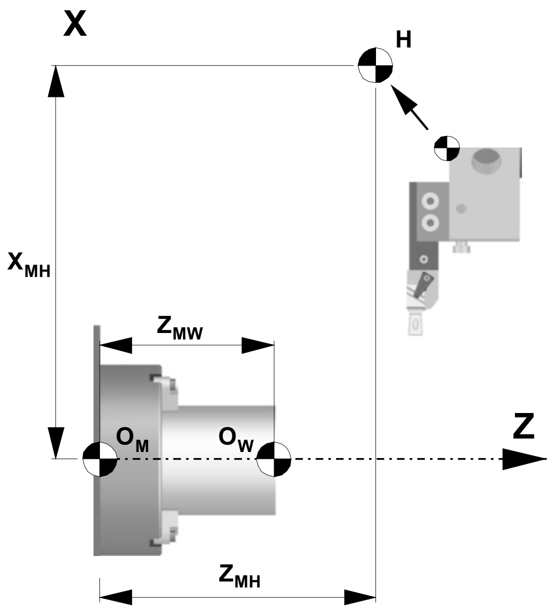
Machine reference system and machine reference point.

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 zero.
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}

10.2.1 Home search (axes and spindles).

Home search with a feedback system that does not use the I0 signal (I0TYPE=Not evaluated).

Parameter.		Meaning.
I0TYPE	[MPFD]	Reference mark (I0) type.
DECINPUT	[MPA]*	The axis has a home switch.
REFDIREC	[MPA]	Home searching direction.
REFFEED1	[MPA]*	Fast home searching feedrate.
REFFEED2	[MPA]*	Slow home searching feedrate.

[MPFB] Feedback parameter.
[MPA] .... Machine parameter; axis.
[MPA] ...... Machine parameter; axis set.*

When the feedback system does not use I0 signals, the home search is done at a specific machine point which is referred to as the 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 switch, it starts moving in the opposite direction at the feedrate set by parameter REFFEED2 until the switch is released. After releasing the switch, the CNC ends the home search.

Home search with a feedback system that does not have distance-coded reference marks (I0TYPE=Normal).

Parameter.		Meaning.
I0TYPE	[MPFD]	Reference mark (I0) type.
REFSYNC	[MPFD]	Input A B quadrant where the I0 signal is.
REFPULSE	[MPFD]	Flank of the marker pulse (I0) used by the CNC to complete the home search.
DECINPUT	[MPA]*	The axis has a home switch.
REFDIREC	[MPA]	Home searching direction.
REFFEED1	[MPA]*	Fast home searching feedrate.
REFFEED2	[MPA]*	Slow home searching feedrate.

[MPFB] Feedback parameter.
[MPA] .... Machine parameter; axis.
[MPA] ...... Machine parameter; axis set.*

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 I0s (I0TYPE=Increasing or decreasing encoding).

Parameter.		Value.
I0TYPE	[MPFD]	Reference mark (I0) type.
REFSYNC	[MPFD]	Input A B quadrant where the I0 signal is.
REFPULSE	[MPFD]	Flank of the marker pulse (I0) used by the CNC to complete the home search.
REFDIREC	[MPA]	Home searching direction.
REFFEED2	[MPA]*	Slow home searching feedrate.

[MPFB] Feedback parameter.
[MPA]....... Machine parameter; axis.
[MPA]...... Machine parameter; axis set.*

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.

Reference search for a single spindle with micro (without encoder) and change of ranges.

The spindle moves in the direction indicated by the REFDIREC parameter and at the speed indicated by the REFFEED2 parameter until the reference micro is pressed. When the CNC receives the signal from the reference micro, it considers that to be the position of I0. The REFPULSE parameter if the CNC defines the position of I0 with the positive or negative edge of the reference micro.

After receiving the signal from the reference micro, the spindle stops at a position after the one considered as zero. To stop the spindle at the I0 position, set the POSINREF parameter to "Yes".

Redefine the machine reference point after having dismounted or replaced the feedback system.

Parameter.		Value.
REFVALUE	[MPFD]	Position of the reference point.
REFSHIFT	[MPFD]	Offset of the reference point.

[MPFB] Feedback parameter.

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.

Peculiarities of home search for spindles, tandem system and analog axes.

Peculiarities of home search for spindles.

Parameter.		Value.
REFINI	[MPA]	Home search in the first movement.
NPULSES	[MPFD]	Number of encoder pulses.

[MPFB] Feedback parameter.
[MPA] .... Machine parameter; axis.

Parameter REFINI determines whether the CNC homes the spindle in its first movement or not. This parameter is only taken into account when the parameters NPULSES 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.

10.2.2 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 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	Position of the reference point. To be defined for the master and slave axes.
REFSHIFT	Offset of the reference point. 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.

Coordinate (position) difference compensation after a home search.

Parameter. PLC signal.		Meaning.
DIFFCOMP	[MPGTY]	Compensate for the coordinate (position) difference between the two axes after G74.
MAXDIFF(axis)	[PLC_M] (R)	The CNC activates (=1) 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 parameter MAXDIFF(axis).
DIFFCOMP(axis)	[PLC_M] (R/W)	This mark is used on Gantry axes to correct the position difference between the master and the slave axes.

[MPGTY] Machine parameter; gantry axis.
[PLC_M] PLC Mark. (R) Consultation signal. (R/W) Modifiable signal.

The position value correction (compensation) may be applied to any type of axis; analog, and Sercos. 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.

Correction from machine parameters.

The DIFFCOMP parameter enables the correction of the dimension difference between the master and slave axis after the machine reference search. On axes with absolute feedback, if the DIFFCOMP parameter is set to "Yes", the CNC will correct the coordinate difference when enabling the axis (rising edge of the SERVOxnON flag of the master or slave axis).

PLC correction.

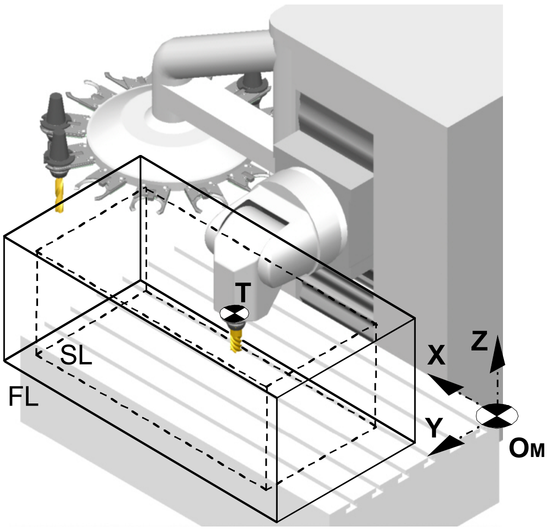
The DIFFCOMP(axis) flag corrects the dimension difference between the master and slave axis as follows.

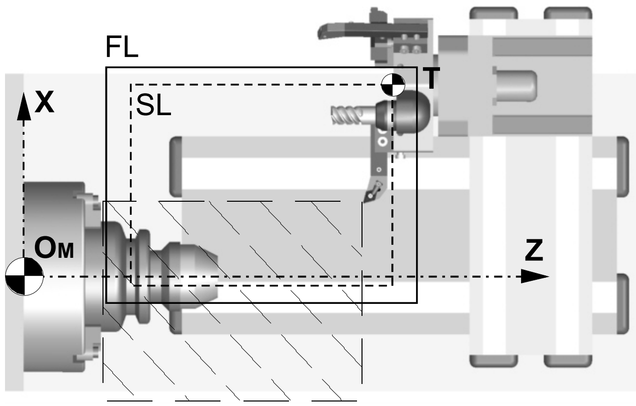
- With the up-flank of SERVO(axis)ON if DIFFCOMP(axis) is active.
- With the up-flank of DIFFCOMP(axis) if SERVO(axis)ON is active.

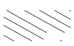
The REFPOIN(axis) mark of both axes is activated when the position value compensation is completed.

10.3 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.



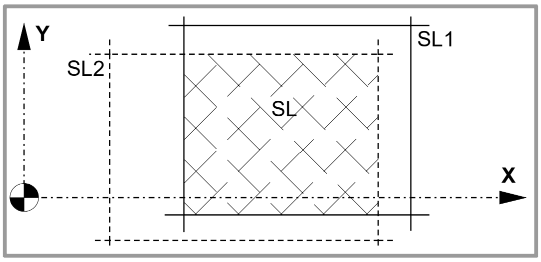


OM	Machine zero.
T	Reference point of the tool holder.
FL	Physical limits.
SL	Software limits applied by the CNC.
	Programmable positions of the axes (they depend on the active tool).

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 uses 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.

10.3.1 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.
POSLIMIT	Upper software travel limits.
NEGLIMIT	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.

10.3.2 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.

10.4 Park/unpark axes (from the PLC or CNC).

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 or spindle, the CNC will control it, because it interprets that the axis will be present again in the machine configuration.

The axes may be parked and unparked from the CNC or from the PLC. Several axes and spindles may stay parked at the same time, but they must always be parked (and unparked) one by one.

Application example.
Some machines, depending on the type of machining, may have two different configurations (axes and spindles); for example, a machine that exchanges a normal spindle (X Y Z axes) for an orthogonal one (X Y Z A B axes). When the machine is working with the normal spindle, the A B axes are not present, so there is no need to park them in order that the CNC does not take them into account and not display the errors relating to the feedback system, drives, etc.

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.
- With an active G96 or G63 and if it is the master spindle of the channel.
- With an active G33 or G95 and if it is the master spindle of the channel or the spindle that 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.

10.4.1 Parking/unparking axes from the PLC.

PLC signal.		Meaning.
PARK(axis)	PLC_M (R)	The CNC activates (=1) the mark when it is parking the axis or the spindle.
UNPARK(axis)	PLC_M (R)	The CNC activates (=1) the mark when it is unparking the axis or the spindle.
PARKED(axis)	PLC_M (R/W)	The PLC activates (=1) this mark to indicate to the CNC for it to park or unpark the axis or spindle.

[PLC_M] PLC Mark. (R) Consultation signal. (R/W) Modifiable signal.

This type of routine is good for applications that need manual parking of the axes either while the machine is off or on (with or without power).

The PLC uses the PARKED(axis) mark to commence the parking and unparking process for the axes or spindles. The CNC uses signals PARK(axis) and UNPARK(axis) to inform the PLC that the parking or unparking processes are in execution. 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.

To park an axis, the axis enable signals must be deactivated. Likewise, after unparking the axis, the axis enabled signals must be activated. For safety reasons, after parking and unparking an axis, the CNC deactivates the REFPOIN(axis) signal.

Example for parking and unparking an axis:

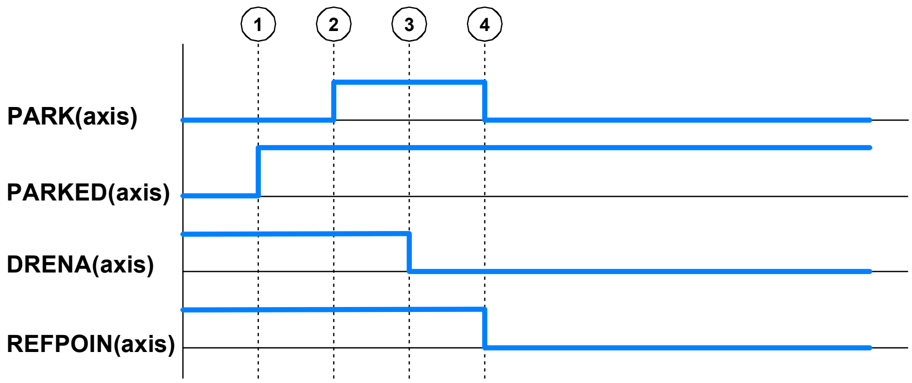
```
I10 = PARKEDV
```

Axis present. "V" axis presence sensor.

```
NOT (PARKV OR UNPARKV OR PARKEDV) AND··· = DRENAV = SPENAV = SERVOVON
```

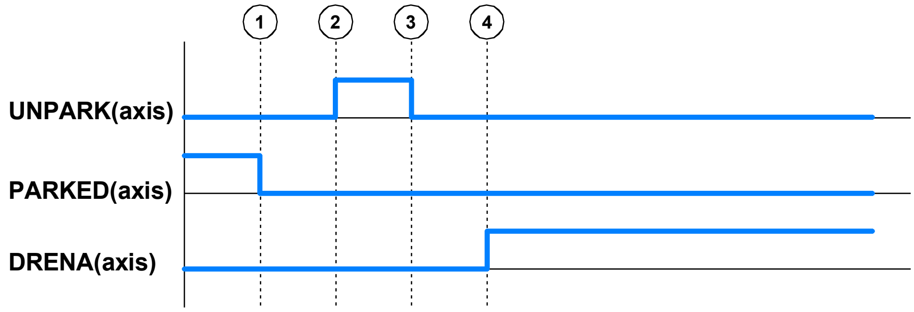
If the axis is neither parked nor being parked and the enabling conditions are met, the axis gets enabled.

Routine to park an axis or spindle from the PLC.



1 The PLC activates the PARKED(axis) mark and parks the corresponding axis.
2 The CNC activates the PARK(axis) mark and begins parking the axis.
3 For digital axes and spindles, the PLC disables the drive (DRENA (axis)).
4 The CNC completes the operation and deactivates the PARK(axis) and REFPOIN (axis) marks.

Routine to unpark an axis or spindle from the PLC.



1 The PLC deactivates the PARKED(axis) mark to unpark the corresponding axis.
2 The CNC activates the UNPARK (axis) mark and starts unparking the axis.
3 The CNC completes the operation and deactivates the UNPARK(axis) signal.
4 For digital axes and spindles, the PLC enables the drive (DRENA (axis)).

10.4.2 Parking/unparking axes from the CNC.

PLC signal.		Meaning.
PARK(axis)	PLC_M (R)	The CNC activates (=1) the mark when it is parking the axis or the spindle.
UNPARK(axis)	PLC_M (R)	The CNC activates (=1) the mark when it is unparking the axis or the spindle.
PARKED(axis)	PLC_M (R/W)	The PLC activates (= 1) this mark to indicate to the CNC that the axis or spindle is parked or unparked.

[PLC_M] PLC Mark. (R) Consultation signal. (R/W) Modifiable signal.

This type of routine 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 maneuver for parking/unparking axes from a part program or MDI/MDA mode is controlled using the programming instructions #PARK and #UNPARK.

The CNC uses the PARKED(axis) and UNPARK(axis) marks to commence the parking and unparking process for the axes or spindles. The PLC uses the PARKED(axis) signal to indicate to the CNC that the axis or spindle is parked or unparked. 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.

To park an axis, the axis enable signals must be deactivated. Likewise, after unparking the axis, the axis enabled signals must be activated. For safety reasons, after parking and unparking an axis, the CNC deactivates the REFPOIN(axis) signal.

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).

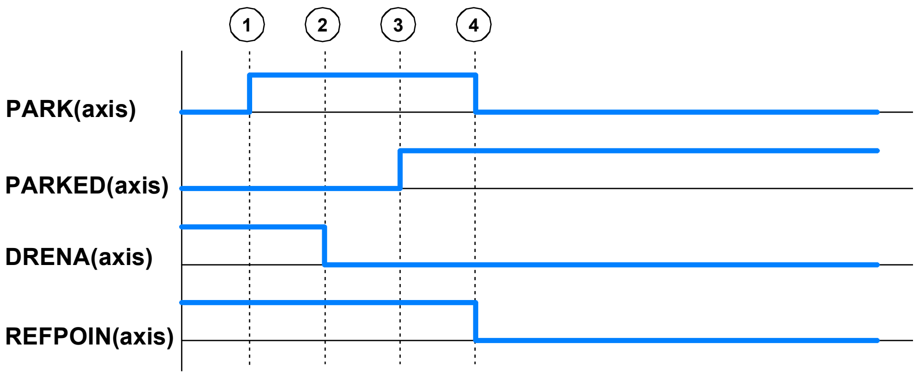
```
UNPARKB AND I15 = RES PARKEDB
```

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.

Routine to park an axis or spindle from the CNC.



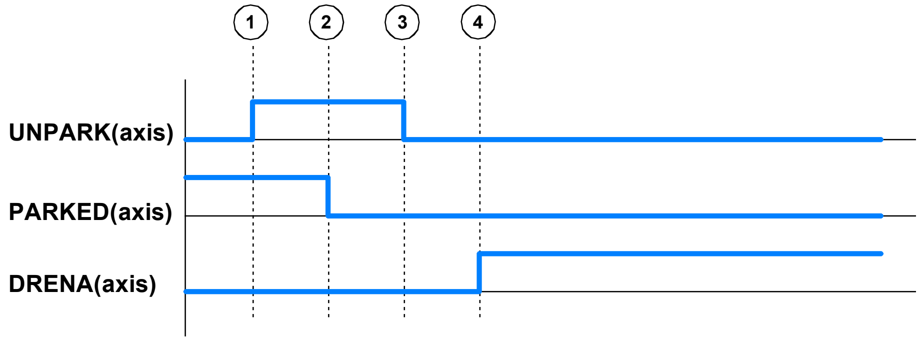
1 When the CNC executes the instruction #PARK, it checks whether the requested axis can be parked or not. If it can, the CNC activates the PARK(axis) mark to indicate to the PLC that it must park the corresponding axis.

2 For digital axes and spindles, the PLC disables the drive (DRENA (axis)).

3 The PLC parks the required axis. After checking that the axis is parked (presence sensors), the PLC activates the PARKED(axis) mark.

4 The CNC will determine that the axis has been parked when it detects that the PARKED(axis) signal has been activated and the PARK(axis) and REFPOIN(axis) marks have been deactivated, ending the process.

Routine to unpark an axis or spindle from the CNC.



1 When the CNC executes an #UNPARK instruction, it activates the UNPARK(axis) mark to indicate to the PLC that the corresponding axis must be unparked.

2 The PLC unparks the required axis. After checking that the axis is unparked (presence sensors), the PLC deactivates the PARKED(axis) mark.

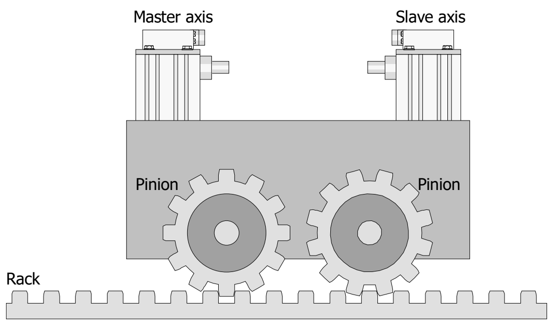
3 The CNC will determine that the axis has been unparked when it detects that the PARKED(axis) signal has been deactivated and the UNPARK(axis) signal has been deactivated, ending the process.

4 For digital axes and spindles, the PLC enables the drive (DRENA (axis)).

TANDEM AXIS (SPINDLE).

11

A tandem axis consists of two motors mechanically coupled to each other forming a single transmission system (axis or spindle). The CNC only displays the master axis, which is the programable axis; the other axis (slave axis) is not displayed and is not programmable; it is controlled by the CNC.



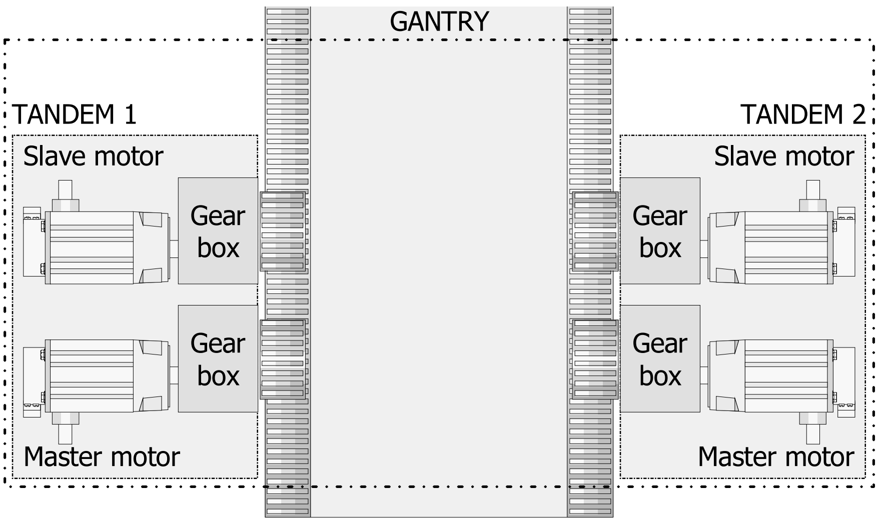
Example of a tandem axis made up of two pinions and a rack.

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 rack-and-pinion.
- The rigidity of the rack-and-pinion system is greater than that of long leadscrews.

Tandem axes on gantry machines.

One of the many applications for the control of a tandem axis has to do with gantry machines. The following example shows a gantry axis with two tandem axes.



11.1 Configuring a tandem (spindle) axis.

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).
- The CNC can apply a preload between both motors (PRELOAD parameter).
- Each motor may have a different rated torque.
- The torque distribution between both motors may be different from 1:1 ratio (parameter TORQDIST); for example, on motors whose rated torque is different.
- Relative to each other, motors can have an opposite turning direction.
- Both axes (parameter AXISTYPE) and drives must be of the same type.
- All the axes that will be interpolated with the tandem axis must operate in Sercos, position or speed mode.
- The channel changing permission (parameter AXISEXCH) must be the same in both axes.
- Both axes and drives must have the same software limits (same POSLIMIT and NEGLIMIT 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.
- The CNC cannot park an axis (spindle) if it is part of a tandem pair, whether it is the master or the slave (PARK(axis)/UNPARK (axis) marks), even if both axes are decoupled (TANDEMOFF(axis) mark)).
- Both spindles must be in the same gear (range).

Axes or spindles of the tandem system.

Parameter.		Meaning.
MASTERAXIS	MPTDM	Master axis/spindle. The master motor of the tandem, besides generating torque, it is in charge of positioning.
SLAVEAXIS	MPTDM	Slave axis/spindle. The slave motor only provides torque. The CNC closes the position loop only with the position of the master axis of the tandem.

[MPTDM] . Machine parameters; tandem.

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. The axes and spindles must be previously defined in the “Axes” table.

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.

Torque load distribution in the tandem.

Parameter.		Meaning.
TORQDIST	MPTDM	Torque distribution. 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 %.

[MPTDM].. Machine parameters; tandem.

Apply a preload value in the tandem.

Parameter.		Meaning.
PRELOAD	MPTDM	Preload between both motors. The parameter indicates the percentage of the rated torque of the master motor that will be applied as preload. If the preload is set to 0, the CNC disables the preload.
PRELFITI	MPTDM	Time to apply the preload. 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. If the time is set to 0, the CNC disables the filter.

[MPTDM].. Machine parameters; 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. See "11.4 Effect of the preload." on page 562.

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.

Proportional gain (Kp) for the tandem.

Parameter.		Meaning.
TPROGAIN	MPTDM	Proportional gain (Kp) for the tandem. The proportional controller generates an output proportional to the torque error between the two motors.

[MPTDM].. Machine parameters; tandem.

Proportional gain.		
S	Speed.	$S = Kp \times T.err$
S.max	Maximum speed.	
Kp	Proportional gain.	$Kp = \left(\frac{S.max}{T.nom}\right) \times TPROGAIN$
T.nom	Rated torque.	
T.err	Torque error between motors.	$T.err = (-T.mst + T.slv + PRELOAD)$
T.mst	Torque of the master motor.	
T.slv	Torque of the slave motor.	

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) x 0.1= 10 rpm/Nm.

Integral gain (Ki) for the tandem.

Parameter.		Meaning.
TINTIME	MPTDM	Integral gain (Ki) for the tandem. The integral controller generates an output proportional to the integral of the torque error between the two motors.

[MPTDM] . Machine parameters; tandem.

Integral gain.		
S	Speed.	$$S = Ki \times \sum T.err$$
S.max	Maximum speed.	
Kp	Proportional gain.	$$Ki = \left(\frac{ControlTime}{IntegralTime}\right) \times Kp$$
Ki	Integral gain.	
T.nom	Rated torque.	
T.err	Torque error between motors.	$$T.err = (-T.mst + T.slv + PRELOAD)$$
T.mst	Torque of the master motor.	
T.slv	Torque of the slave motor.	

Torque compensation limit.

Parameter.		Meaning.
TCOMPLIM	MPTDM	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.

[MPTDM] . Machine parameters; tandem.

Type of feedback; internal (motor) or external (direct).

Parameter.		Meaning.
FBACKSRC	MPA	Type of feedback used to close the position loop; internal (motor feedback) or external (direct 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. Tandem axes do not admit the internal+extenrnal type of feedback. When external feedback is used, it is recommended that the FBACKDIFF parameter is defined.
FBACKDIFF	MPA	Maximum difference between feedbacks. If the difference exceeds the set value, the CNC will display the corresponding error message. If it is set to 0 (zero), there will be no monitoring. The CNC takes this parameter into account only if the tandem axis uses external feedback to close the position loop.
FBMIXTIME	MPA	This parameter may be used to set the time constant to be used when combining feedbacks; this means that it sets the delay between the coordinates entered to the loop from the internal (motor) and external feedback (direct). The CNC only takes this parameter into account when the feedback is external. If the parameter is set to a value other than 0, the combined feedback is enabled; if is set to 0 (zero), only the external feedback is used.

[MPA] .... Machine parameters; general.

On machines that have a lot of backlash and use external feedback to obtain greater accuracy may suffer some instability. This kind of machines run smoothly with internal feedback, but it may lose precision; with external feedback, however, the accuracy improves but the machine movements may be jerky. Combining both feedbacks both precision and smoothness may be achieved.

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.

FBACKSRC	FBMIXTIME	Type of feedback.
Internal.	(Not being used)	Internal feedback.
External.	0	External feedback.
External.	Value greater than 0 (zero).	The combined (mixed) feedback is active.
Internal+External.	(Not being used)	Invalid type for tandem axes (spindles).

Dwell for dead axis.

Parameter.		Meaning.
DWELL	MPA	Dwell for dead axis. 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 SERVO(axis)ON of the master axis; if the enabling of the slave axis is slower, the PLC routine must verify that both axes are enabled before generating the movement.

[MPA] .... Machine parameters; general.

Decouple the axes of the tandem.

PLC signal.		Meaning.
TANDEMOFF(axis)	PLC_M (R/W)	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.

[PLC_M] PLC Mark. (R) Consultation signal. (R/W) Modifiable signal.

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 can be moved in the usual way from the operator panel, in MDI/MDA mode, etc. When moving the master axis, the CNC will not generate a setpoint for the slave axis or 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).

11.2 Considerations for the PLC routine.

Some general points to consider.

PLC signal.		Meaning.
PARK(axis)	PLC_M (R)	The CNC cannot park an axis (spindle) if it is part of a tandem pair, whether it is the master or the slave.
UNPARK(axis)	PLC_M (R)	
PARKED(axis)	PLC_M (R/W)	
TANDEMOFF(axis)	PLC_M (R/W)	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. See *"Decouple the axes of the tandem."* on page 559.

[PLC_M] PLC Mark. (R) Consultation signal. (R/W) Modifiable signal.

Considerations for tandem spindles.

PLC signal.		Meaning.
SPENA(axis)	PLC_M (R/W)	The SPENA(axis) mark corresponds to the "speed enable" signal and the DRENA(axis) to the "drive enable" signal. The PLC must enable and disable the SPENA(axis) and DRENA(axis) signals of the master and slave spindles at the same time.
DRENA(axis)	PLC_M (R/W)	
PLCCNTL	PLC_M (R/W)	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.
SANALOG	PLC_M (R/W)	
SPDLEREV	PLC_M (R/W)	
GEAR1	PLC_M (R/W)	Both spindles must be in the same gear (range).
GEAR2	PLC_M (R/W)	
GEAR3	PLC_M (R/W)	
GEAR4	PLC_M (R/W)	

[PLC_M] PLC Mark. (R) Consultation signal. (R/W) Modifiable signal.

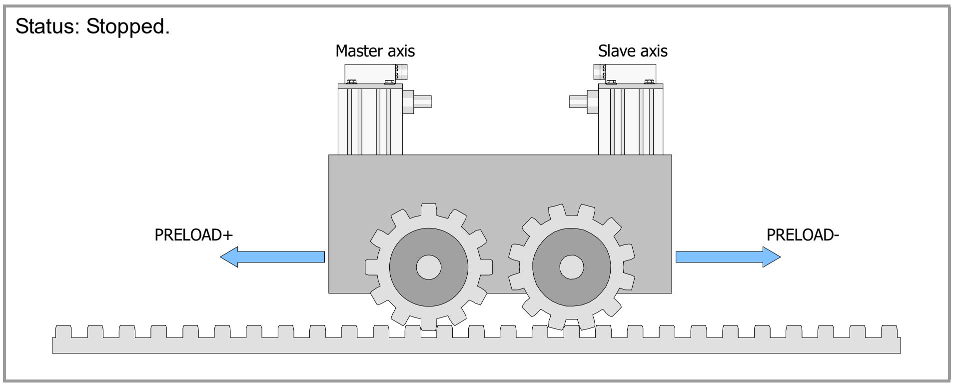
11.3 Special considerations during a home search.

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.

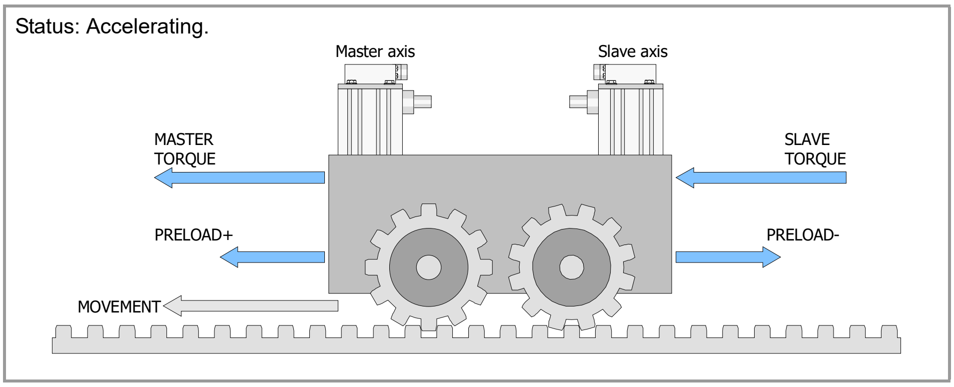
11.4 Effect of the preload.

The following diagrams show the effect of pre-load in different situations.

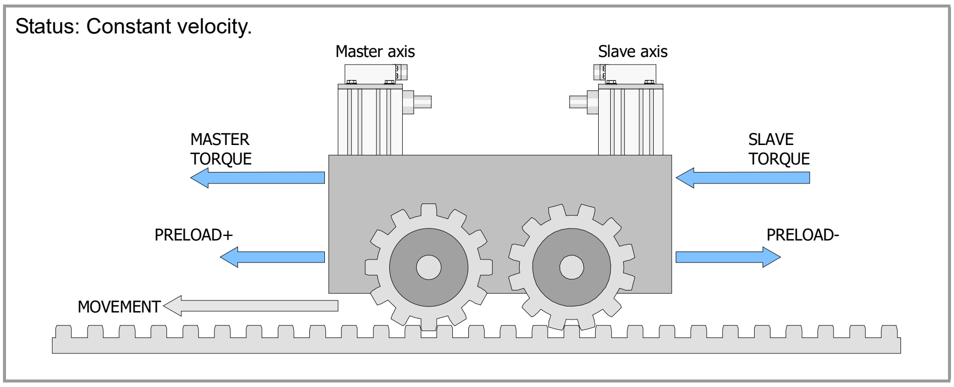
Preload at rest.



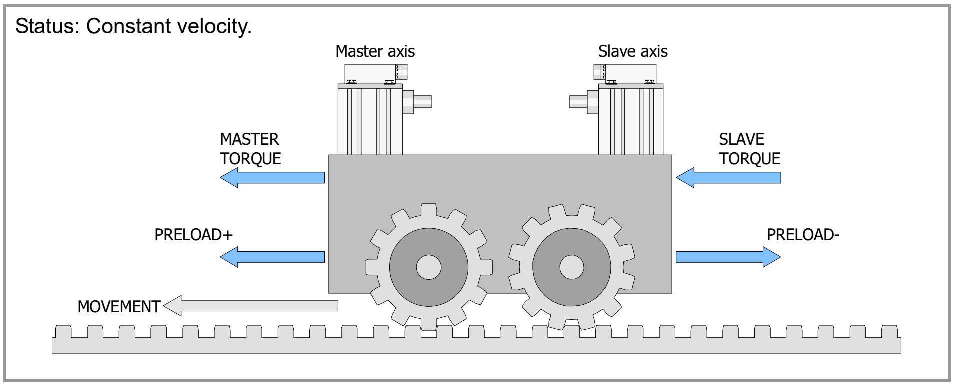
Preload with acceleration.



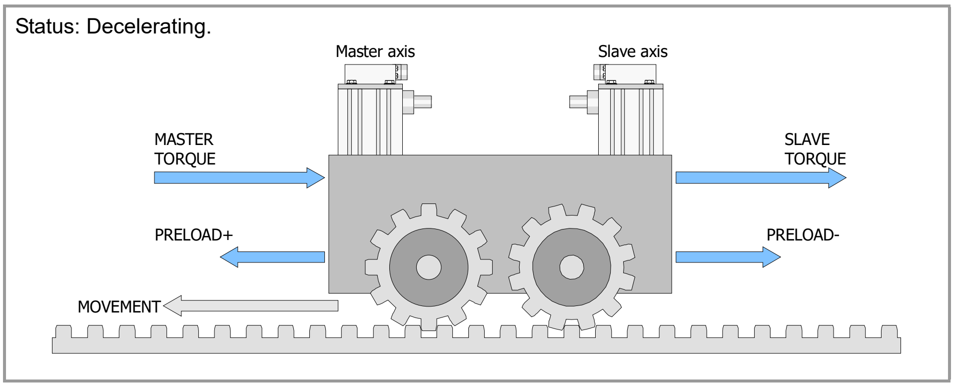
Preload at constant speed. Friction torque > Preload.



Preload at constant speed. Friction torque < Preload.

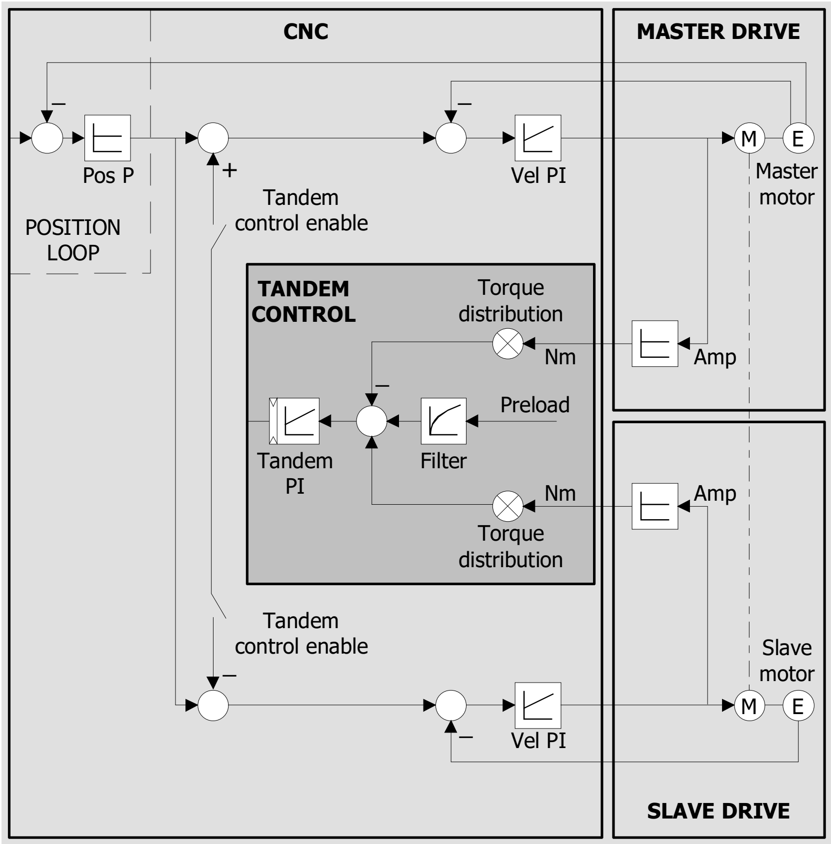


Preload with deceleration.



11.5 Tandem axis configuration. Block diagram.

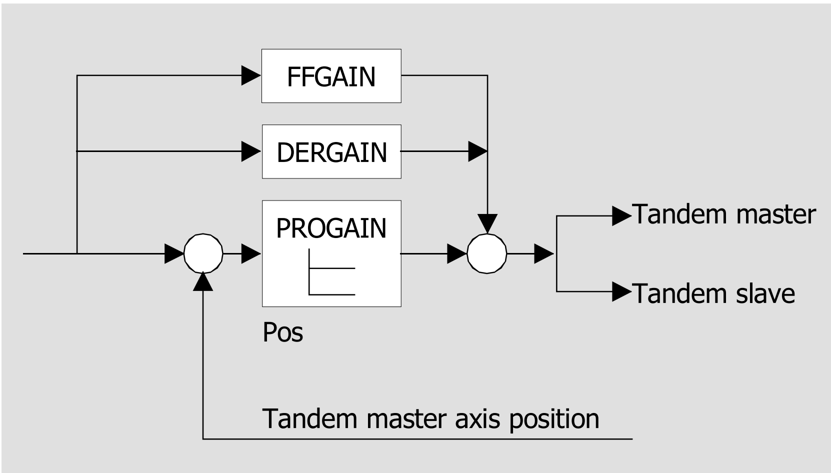
The block diagram of the tandem control system shows the master tandem axis with its slave tandem axis. The block diagram for a gantry machine consists of two identical diagrams to the one shown in the figure.



The block diagram has an area for each drive and another one for the CNC that comprises the position loop, velocity loop and the tandem control.

Position and velocity loop.

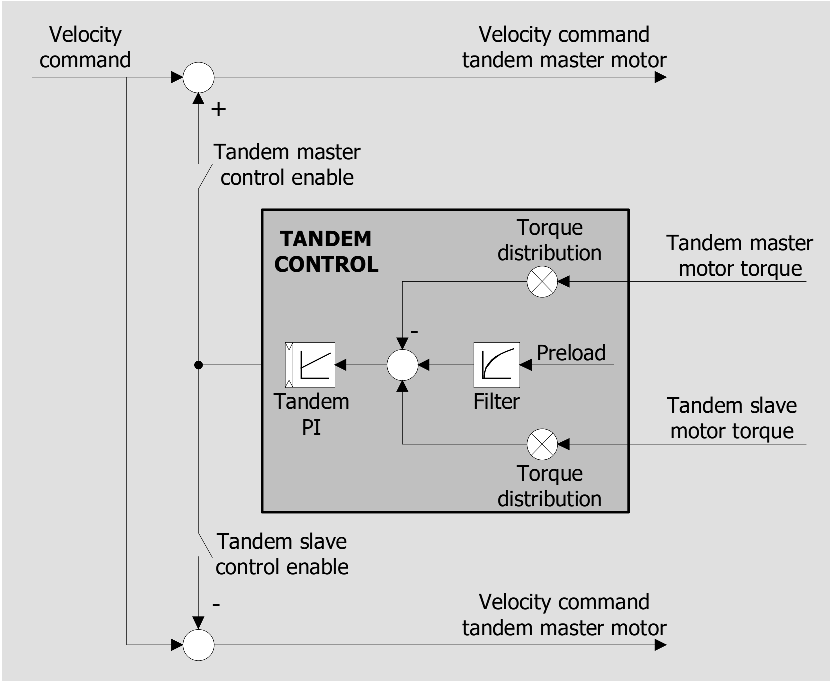
The position loop is closed only with the position of the master axis of the tandem. The velocity command of the master axis/spindle of the tandem is also sent to the slave axis/spindle of the tandem closing the velocity loop. The tandem control changes the velocity command of the master axis and that 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.

Percentage of the rated torque of the master axis of the tandem.

Torque of the slave motor of the tandem.

Percentage of the rated torque of the slave axis of the tandem.

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. The preload sets a traction between both motors in order to eliminate the rack-and-pinion backlash in the resting position. The preload 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; otherwise, the motors will move even without control 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.

11.6 Tandem adjustment procedure.

This procedure contemplate the type of machine. In general, a tandem machine has a low resonance frequency; therefore, the CNC must generate position setpoints without frequency components higher than the resonance frequency.

It is recommended to begin the process with low jerk values (lower than 10 m/sec³) and low Kv; it is always possible to increase these in a subsequent readjustment.

Steps for the adjustment.

1 Move both axes independently.

The first step is to ensure the proper 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 abrupt movements since the second motor is dragging the first motor. In this situation, any acceleration or deceleration forces the second motor 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.

Inverting 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 the parameters INVERT and LOOPCH.

- 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 (the system has sufficient 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 (parameter ACFGAIN), 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" (for example, half the torque for 50 % distribution).

5 Define the preload.

To define the pre-load, monitor the torque of each motor. 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.

6 Readjusting the velocity loop.

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.

11.7 Variables.

The following variables may be accessed from (PRG) the part-program and from the MDI/MDA mode, PLC and (INT) from an external application. For each variable, the table indicates whether the access is reading (R) or writing (W). PLC access to the variables, for both reading and writing, will be synchronous. Access to the variables from the part program returns the value for the block preparation (it does not stop the preparation), except when indicated otherwise.

Variables.	PRG	PLC	INT
(V.)MPTDM.MASTERAXIS[nb] Logic number of the master axis (spindle). Units: -.	R	R	R
(V.)MPTDM.SLAVEAXIS[nb] Logic number of the slave axis (spindle). Units: -.	R	R	R
(V.)MPTDM.TORQDIST[nb] Torque distribution (percentage to be provided by the master motor). Units: Percentage.	R	R	R
(V.)MPTDM.PRELOAD[nb] Preload between both motors (percentage of the rated torque of the master motor). Units: Percentage.	R(*)	R	R
(V.)MPTDM.PRELFITI[nb] Time to apply the preload. Units: Milliseconds.	R(*)	R	R
(V.)MPTDM.TPROGAIN[nb] Proportional gain (Kp) for the tandem. Units: Percentage.	R(*)	R	R
(V.)MPTDM.TINTTIME[nb] Integral gain (Ki) for the tandem. Units: Milliseconds.	R(*)	R	R
(V.)MPTDM.TCOMPLIM[nb] Torque compensation limit. Units: Percentage.	R(*)	R	R
(V.)[ch].A.TPIIN.xn **(V.)[ch].A.TPIIN.sn** **(V.)[ch].SP.TPIIN.sn** PI input of the master (spindle) axis of the tandem. The axis must be a valid tandem master (spindle) axis, otherwise, the variable will return a zero value. Units: rpm.	R(*)	R	R
(V.)[ch].A.TPIOUT.xn **(V.)[ch].A.TPIOUT.sn** **(V.)[ch].SP.TPIOUT.sn** Output of the PI of the master axis of the tandem. The axis must be a valid tandem master (spindle) axis, otherwise, the variable will return a zero value. Units: rpm.	R(*)	R	R
(V.)[ch].A.TFILTOUT.xn **(V.)[ch].A.TFILTOUT.sn** **(V.)[ch].SP.TFILTOUT.sn** Output of the pre-load filter of the tandem. Units: -.	R(*)	R	R

(*) The CNC evaluates the variable during execution (it stops the block preparation).

Variables.	PRG	PLC	INT
(V.)[ch].A.PRELOAD.xn **(V.)[ch].A.PRELOAD.sn** **(V.)[ch].SP.PRELOAD.sn** Preload in the tandem. Units: Newton × meter.	R/W(*)	R/W	R/W
(V.)[ch].A.FTEO.xn **(V.)[ch].A.FTEO.sn** **(V.)[ch].SP.FTEO.sn** Velocity command for Sercos. Units: rpm.	R(*)	R	R
(V.)[ch].A.TORQUE.xn **(V.)[ch].A.TORQUE.sn** **(V.)[ch].SP.TORQUE.sn** Current torque in Sercos. The PLC reading of this variable comes in tenths (x10). Units: -.	R(*)	R	R

(*) The CNC evaluates the variable during execution (it stops the block preparation).

Syntax of the variables.

·nb· tandem axis number.

·ch· Channel number.

·xn· Name, logic number or index of the axis.

·sn· Name, logic number or index of the spindle.

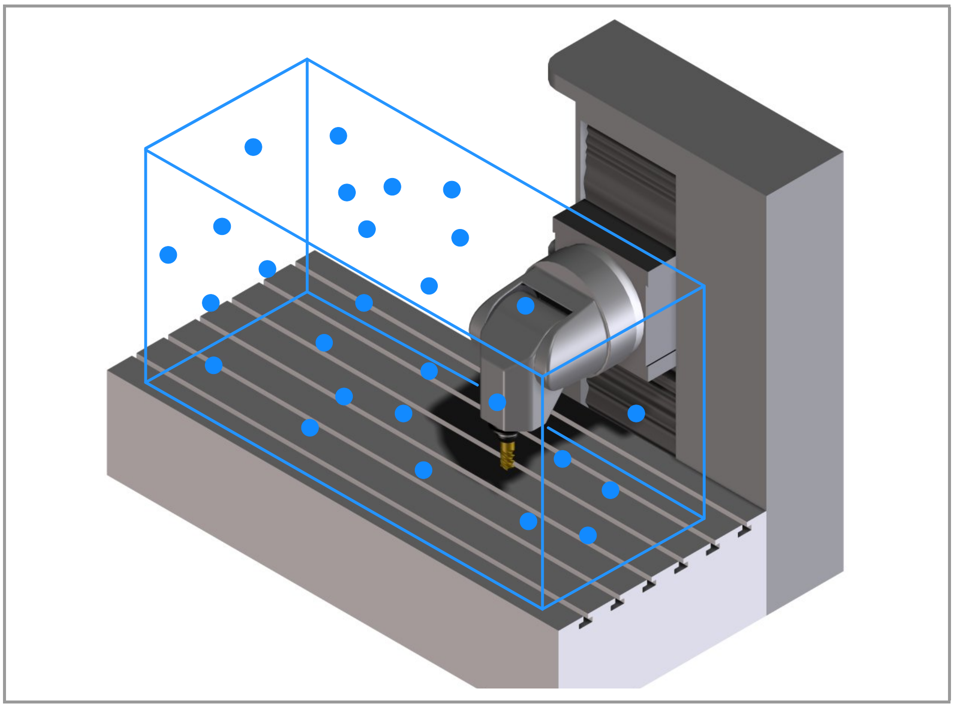
V.MPTDM.MASTERAXIS[2]	Tandem 2 axis.
V.A.TPIIN.Z	Z axis.
V.A.TPIIN.S	Spindle S.
V.SP.TPIIN.S	Spindle S.
V.SP.TPIIN	Master spindle.
V.A.TPIIN.4	Axis or spindle with logic number ·4·.
V.[2].A.TPIIN.1	Axis with index ·1· in the channel ·2·.
V.SP.TPIIN.2	Spindle with index ·2· in the system.
V.[2].SP.TPIIN.1	Spindle with index ·1· in the channel ·2·.
V.MPTDM.MASTERAXIS[2]	Tandem 2 axis.

VOLUMETRIC COMPENSATION.

12

One of the causes of the lack of precision in the positioning stems from geometric machine errors that arise during manufacturing and assembly, wear, elastic deformations, etc. Volumetric compensation corrects these geometric errors to a large extent, thus improving the precision of the positioning.

The volume to be compensated is defined by a cloud of points, in each of which the error to be corrected is measured. This error is recorded in a file that is then uploaded to the CNC. At the intermediate points, the CNC interpolates the error based on the eight adjacent known points. For points that are outside the volume, the CNC applies the compensation of the point closest to the volume.



Activating and configuring volumetric offsets.

Volumetric compensations are configured in the machine parameters and are activated from the PLC (marks VOLCOMP1 to VOLCOMP4). The tables that define the compensation are defined in an external file.

The CNC can have four volumetric compensations configured, formed by linear and rotary axes, where the same axis can be included in several compensations. All compensations can be simultaneously active, except those that share axes; two compensations that share axes cannot be active at the same time.

If the CNC has activated either a medium or large volumetric compensation (software option), it can also allow for a basic volumetric compensation to be defined if the file has an csv extension. In this case, the CNC will provide the necessary parameters for its definition.

Volumetric compensation types.

The CNC has three types of volumetric compensation, which are selectable via its corresponding software option: basic, medium or large. The “Large volumetric compensation” option also allows the other two types of compensation to be used: medium and basic. The “Medium volumetric compensation” also allows for the basic option to be used.

Type.	Description.
Basic.	• The total number of points for the volume remain unchanged at 15625. • The volume to be compensated is defined by up to 1000 points on each axis. • This compensation corrects translation errors. • The compensation tables are generated by the calibration application; they are not editable from the CNC. • The OEM must define all the compensation data (axes that move, axes to be compensated, position and size of the volume to be compensated, etc.) in the machine parameters and select the file containing the compensation data. • The basic compensation is the fastest to calibrate, but it can be less precise than the others (it corrects fewer error components).
Medium. Large.	• Volume compensation of up to 10 m³ (medium) or more than 10 m³ (large). • This compensation corrects the 21 geometric error components (translation, rotation and squaring). • The compensation tables are generated by the calibration application; they are not editable from the CNC. • The OEM must only define the axes to be compensated for the machine parameters and select the file containing the calibration data. The rest of the data (axes that move, error to be compensated, etc.) are implicit in the file. • Medium and large compensations are more precise than basic compensation, but they are slower to calibrate.

The volumetric compensation includes cross compensation and ballscrew compensation, and therefore it is not necessary to define the latter two; however, if they are defined, the CNC takes them into account. The CNC applies the cross compensation and ballscrew compensation first, followed by the volumetric compensation.

On the gantry axes, compensation is defined for the master axis and the CNC also applies it to the slave axis.

12.1 Basic volumetric compensation.

For basic volumetric compensation, it is necessary to define both axes: the one that moves and the one that is compensated. This makes it possible to define another type of compensation that does not involve the same three axes; for example, a common cross compensation or 2D compensation for the falling ram.

12.1.1 Setup. General; machine parameters.

The CNC displays these parameters when loading a file with the extension csv in the parameter VCOMPFILE.

Parameter.		Meaning.
VOLCOMP	MPG	This parameter shows the volumetric compensation table.
VMOVAXIS1 VMOVAXIS2 VMOVAXIS3	MPG	Axis whose movement affects another axis. At least one axis must be defined. These linear or rotary axes define the volume to be compensated. If it is a gantry axis, it is only necessary to define the master axis.
NPOINTSAX1 NPOINTSAX2 NPOINTSAX3	MPG	Number of points that define the volume to be compensated for each axis. The total number of points for the volume is limited to 15,625, and is calculated in the following manner: NPOINTSAX1 × NPOINTSAX2 × NPOINTSAX3. The total number of points for the volume can be distributed among all the axes, up to a maximum of 1000 points per axis.
INIPOSAX1 INIPOSAX2 INIPOSAX3	MPG	Initial position of the basic volumetric compensation for the MOVAXIS axis.
INCREAX1 INCREAX2 INCREAX3	MPG	Interval between points for the MOVAXIS axis in basic volumetric compensation.
VCOMPAXIS1 VCOMPAXIS2 VCOMPAXIS3	MPG	These parameters set the axes to be compensated with volumetric compensation. The axes may be linear or rotary, and at least one axis must be defined. 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. An axis can be included in several different compensations, but compensations that share axes cannot be active at the same time. If it is a gantry axis, it is only necessary to define the master axis; the CNC also applies the compensation to the slave axis.
VCOMPFILE	MPG	The data to be compensated must be a text file (in csv format). The OEM or the company responsible for the calibration must generate this file and define it in the machine parameter VCOMPFILE. The units in the file data (millimeters or inches) must be those defined by the CNC (parameter INCHES).

[MPG] General machine parameter.

12.1.2 Setup. PLC. General modifiable signals.

PLC signal.		Meaning.
VOLCOMP1 VOLCOMP2 VOLCOMP3 VOLCOMP4	PLC_M (R/W)	If the PLC activates one of these marks (changing it from 0 to 1), the CNC activates the corresponding volumetric compensation (parameter VOLCOMP). If the PLC deactivates one of these marks (changing it from 1 to 0), the CNC deactivates the corresponding volumetric compensation. The mark VOLCOMP1 corresponds to the first volumetric compensation (parameter VOLCOMP 1); the mark VOLCOMP2 is for the second and so forth.

[PLC_M] PLC Mark.(R/W) Modifiable signal.

All volumetric compensations can be simultaneously active, as long as there are no common axes among them. The CNC applies the volumetric compensation after applying the leadscrew compensation and the cross compensation. Volumetric compensation remains active after a reset, error or end of program (M30).

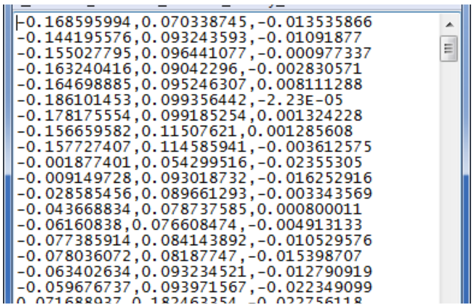
12.1.3 File containing basic volumetric compensation data.

The data to be compensated must be a text file (in csv format). The OEM or the company responsible for the calibration must generate this file and define it in the machine parameter VCOMPFILE. The units in the file data (millimeters or inches) must be those defined by the CNC (parameter INCHES).

Format of the data file.

The text file consists of three data columns and as many rows are there are points to be compensated, in which the data for each row are separated by the character "," (a comma).

Number of rows = NPOINTSAX1 × NPOINTSAX2 × NPOINTSAX3



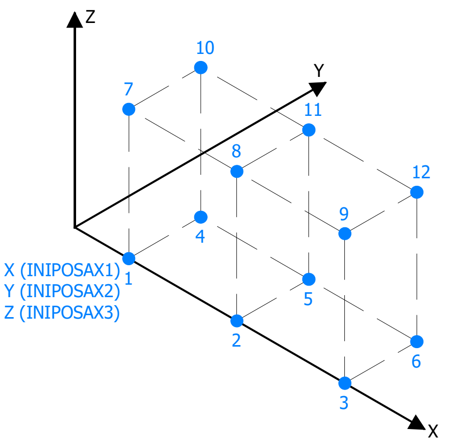
```
-0.168595994,0.070338745,-0.013535866
-0.144195576,0.093243593,-0.01091877
-0.155027795,0.096441077,-0.000977337
-0.163240416,0.09042296,-0.002830571
-0.164698885,0.095246307,0.008111288
-0.186101453,0.099356442,-2.23E-05
-0.178175554,0.099185254,0.001324228
-0.156659582,0.11507621,0.001285608
-0.157727407,0.114585941,-0.003612575
-0.001877401,0.054299516,-0.02355305
-0.009149728,0.093018732,-0.016252916
-0.028585456,0.089661293,-0.003343569
-0.043668834,0.078737585,0.000800011
-0.06160838,0.076608474,-0.004913133
-0.077385914,0.084143892,-0.010529576
-0.078036072,0.08187747,-0.015398707
-0.063402634,0.093234521,-0.012790919
-0.059676737,0.093971567,-0.022349099
```

Example.
NPOINTSAX1=3
NPOINTSAX2=2
NPOINTSAX3=2

Number of rows on the table or points to be compensated.
NPOINTSAX1 × NPOINTSAX2 × NPOINTSAX3 = 3 × 2 × 2 = 12

- The columns correspond to the axes defined in the parameters VCOMPAXIS1 (first column), VCOMPAXIS2 (second column) and VCOMPAXIS3 (third column).
- The rows are created with the consecutive error measurements, according to the order of the axes defined in MOVAXIS; first all the measurements on the first axis; then for each position on the second axis, all the measurements on the first axis once again; and for each position on the third axis, all the measurements of the first axis on each position of the second axis.

VMOVAXIS1=X	NPOINTSAX1=3	INCREAX1=30	VCOMPAXIS1=X
VMOVAXIS2=Y	NPOINTSAX2=2	INCREAX2=30	VCOMPAXIS2=Y
VMOVAXIS3=Z	NPOINTSAX3=2	INCREAX3=36	VCOMPAXIS3=Z



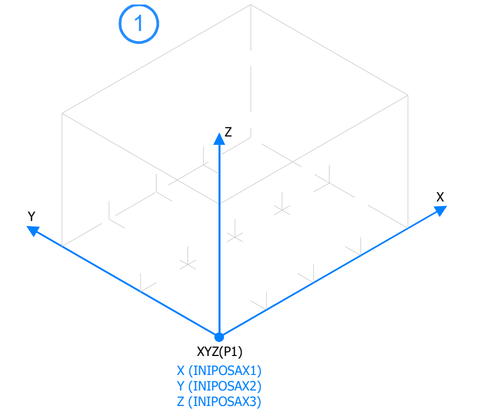
	Error X,	Error Y,	Error Z
1	-0.1685,	0.0703,	-0.0135
2	-0.1441,	0.0932,	-0.0109
3	-0.1550,	0.0964,	-0.0009
4	0.1632,	0.0904,	-0.0028
5	0.1646,	0.0952,	0.0081
6	-0.1861,	0.0993,	-2,230
7	-0.1781,	0.0991,	0.0013
8	-0.1566,	0.1150,	0.0012
9	0.1577,	0.1145,	-0.0036
10	0.0018,	0.0542,	-0.0235
11	-0.0091,	0.0930,	-0.0162
12	-0.0285,	0.0896,	-0.0033

12.1.4 Sequence to define the values in the file.

Assuming a trihedron on the following axes.

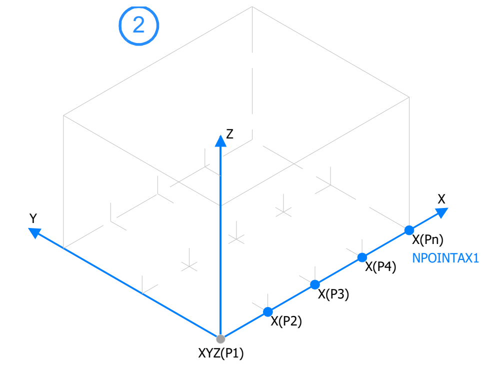
VMOVAXIS1=X VMOVAXIS2=Y VMOVAXIS3=Z.

1 The first row corresponds to the point "X(P1), Y(P1), Z(P1)". This point corresponds to the initial point of the volume to be compensated (parameters INIPOSAX1, INIPOSAX2 and INIPOSAX3).



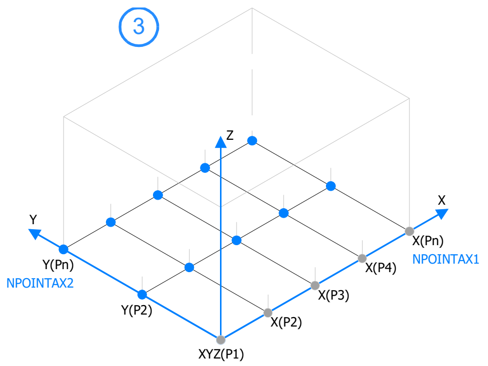
X(P1) Y(P1) Z(P1)

2 The second row corresponds to the point "X(P2), Y(P1), Z(P1)", the third to point "X(P3), Y(P1), Z(P1)", and so on until completing all the points of X (parameter NPOINTSAX1). The distance between the points on the X axis is defined in the parameter INCREAX1.



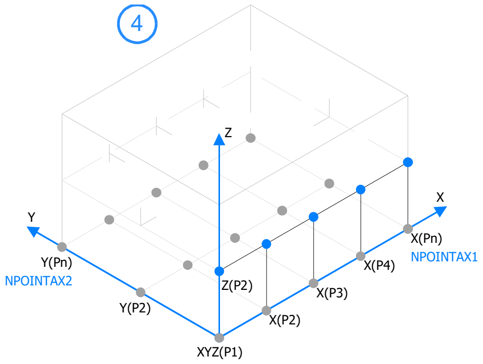
X(P2) Y(P1) Z(P1)
...
X(Pn) Y(P1) Z(P1)

3 Repeat the same operation to define all the points on the X axis, from the point Y(P2) to the last point on the Y axis (parameter NPOINTSAX2). The distance between points on the Y axis is defined in the parameter INCREAX2.

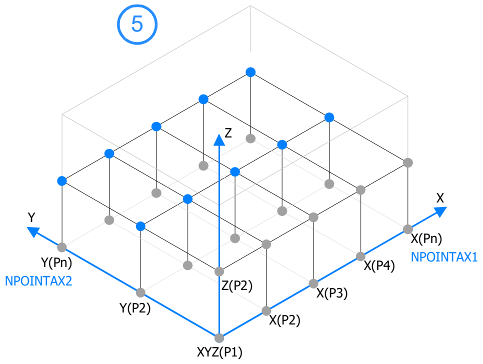


X(P1) Y(P2) Z(P1)
X(P2) Y(P2) Z(P1)
...
X(Pn) Y(P2) Z(P1)
...
X(P1) Y(Pn) Z(P1)
X(P2) Y(Pn) Z(P1)
...
X(Pn) Y(Pn) Z(P1)

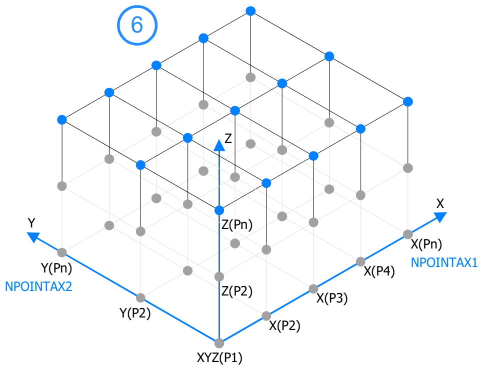
4 Once all the points on the plane have been defined (coordinate Z(P1)), repeat the same operation from point "X(P1), Y(P1), Z(P2)" until the last point on the Z axis (parameter NPOINTSAX3).



X(P1) Y(P1) Z(P2)
X(P2) Y(P1) Z(P2)
...
X(Pn) Y(P1) Z(P2)



X(P1) Y(P2) Z(P2)
X(P2) Y(P2) Z(P2)
...
X(Pn) Y(P2) Z(P2)
...
X(P1) Y(Pn) Z(P2)
X(P2) Y(Pn) Z(P2)
...
X(Pn) Y(Pn) Z(P2)



X(P1) Y(P1) Z(Pn)
X(P2) Y(P1) Z(Pn)
...
X(Pn) Y(P1) Z(Pn)
...
X(P1) Y(Pn) Z(Pn)
X(P2) Y(Pn) Z(Pn)
...
X(Pn) Y(Pn) Z(Pn)
...
X(P1) Y(Pn) Z(Pn)
X(P2) Y(Pn) Z(Pn)
...
X(Pn) Y(Pn) Z(Pn)

12.1.5 Variables.

The following variables may be accessed from (PRG) the part-program and from the MDI/MDA mode, PLC and (INT) from an external application. For each variable, the table indicates whether the access is reading (R) or writing (W). PLC access to the variables, for both reading and writing, will be synchronous. Access to the variables from the part program returns the value for the block preparation, except when indicated otherwise.

General; machine parameters.

Variables.	PRG	PLC	INT
(V.)MPG.VMOVAXIS1[tbl] **(V.)MPG.VMOVAXIS2[tbl]** **(V.)MPG.VMOVAXIS3[tbl]** Name of the axis that generates changes when moved. Units: -.	R(*)	R	R
(V.)MPG.NPOINTSAX1[tbl] **(V.)MPG.NPOINTSAX2[tbl]** **(V.)MPG.NPOINTSAX3[tbl]** Number of points of basic volumetric compensation on each axis. Units: -.	R	R	R
(V.)MPG.INIPOSAX1[tbl] **(V.)MPG.INIPOSAX2[tbl]** **(V.)MPG.INIPOSAX3[tbl]** Initial position of the basic volumetric compensation for the MOVAXIS axis. Units: -.	R	R	R
(V.)MPG.INCREAX1[tbl] **(V.)MPG.INCREAX2[tbl]** **(V.)MPG.INCREAX3[tbl]** Interval between points for the MOVAXIS axis in basic volumetric compensation. Units: -.	R	R	R
(V.)MPG.VCOMPAXIS1[tbl] **(V.)MPG.VCOMPAXIS2[tbl]** **(V.)MPG.VCOMPAXIS3[tbl]** Name of the axis to be compensated. Units: -.	R(*)	R	R
(V.)MPG.VCOMPFILE[tbl] File containing volumetric compensation data. Units: -.	R	R	R

(*) The CNC evaluates the variable during execution (it stops the block preparation).

Syntax of the variable.

·tbl· Table number.

V.MPG.VMOVAXIS2[1]	Name of the second axis on the first volumetric compensation table.
V.MPG.NPOINTAX2[2]	Number of points for the second axis on the second volumetric compensation table.

General modifiable signals.

Variables.	PRG	PLC	INT
(V.)PLC.VOLCOMP1 **(V.)PLC.VOLCOMP2** **(V.)PLC.VOLCOMP3** **(V.)PLC.VOLCOMP4** These variables reflect the value of the marks VOLCOMP1 to VOLCOMP4 of the PLC. If the PLC activates one of these marks (changing it from 0 to 1), the CNC activates the corresponding volumetric compensation (parameter VOLCOMP). Units: -.	R(*)	-	R/W

(*) The CNC evaluates the variable during execution (it stops the block preparation).

Syntax of the variable.

V.PLC.VOLCOMP1	Volumetric compensation ·1·.

Volumetric compensation.

Variables.	PRG	PLC	INT
(V.)[ch].A.VOLCOMP.xn Value that the volumetric compensation is adding to the axis. If the volumetric compensation is not active, the variable returns the value to 0 (zero). Units: Millimeters, inches or degrees.	R(*)	R	R

(*) The CNC evaluates the variable during execution (it stops the block preparation).

Syntax of the variable.

·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·.

12.2 Medium or large volumetric compensation.

12.2.1 Setup. General; machine parameters.

Parameter.		Meaning.
VOLCOMP	MPG	This parameter shows the volumetric compensation table.
VCOMPAXIS1 VCOMPAXIS2 VCOMPAXIS3	MPG	These parameters set the axes to be compensated with volumetric compensation. The axes may be linear or rotary, and at least one axis must be defined. 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. An axis can be included in several different compensations, but compensations that share axes cannot be active at the same time. If it is a gantry axis, it is only necessary to define the master axis; the CNC also applies the compensation to the slave axis.
VCOMPFILE	MPG	The compensation tables are generated by the calibration application; they are not editable from the CNC. The units in the file data (millimeters or inches) must be those defined by the CNC (parameter INCHES).

[MPG] General machine parameter.

12.2.2 Setup. PLC. General modifiable signals.

PLC signal.		Meaning.
VOLCOMP1 VOLCOMP2 VOLCOMP3 VOLCOMP4	PLC_M (R/W)	If the PLC activates one of these marks (changing it from 0 to 1), the CNC activates the corresponding volumetric compensation (parameter VOLCOMP). If the PLC deactivates one of these marks (changing it from 1 to 0), the CNC deactivates the corresponding volumetric compensation. The mark VOLCOMP1 corresponds to the first volumetric compensation (parameter VOLCOMP 1); the mark VOLCOMP2 is for the second and so forth.

[PLC_M] PLC Mark.(R/W) Modifiable signal.

All volumetric compensations can be simultaneously active, as long as there are no common axes among them. The CNC applies the volumetric compensation after applying the leadscrew compensation and the cross compensation. Volumetric compensation remains active after a reset, error or end of program (M30).

12.2.3 Variables.

The following variables may be accessed from (PRG) the part-program and from the MDI/MDA mode, PLC and (INT) from an external application. For each variable, the table indicates whether the access is reading (R) or writing (W). PLC access to the variables, for both reading and writing, will be synchronous. Access to the variables from the part program returns the value for the block preparation, except when indicated otherwise.

General; machine parameters.

Variables.	PRG	PLC	INT
(V.)MPG.VCOMPAXIS1[tbl] **(V.)MPG.VCOMPAXIS2[tbl]** **(V.)MPG.VCOMPAXIS3[tbl]** Name of the axis to be compensated. Units: -.	R(*)	R	R
(V.)MPG.VCOMPFILE[tbl] File containing volumetric compensation data. Units: -.	R	R	R

(*) The CNC evaluates the variable during execution (it stops the block preparation).

Syntax of the variable.

·tbl· Table number.

V.MPG.VMOVAXIS2[1]	Name of the second axis on the first volumetric compensation table.
V.MPG.NPOINTAX2[2]	Number of points for the second axis on the second volumetric compensation table.

General modifiable signals.

Variables.	PRG	PLC	INT
(V.)PLC.VOLCOMP1 **(V.)PLC.VOLCOMP2** **(V.)PLC.VOLCOMP3** **(V.)PLC.VOLCOMP4** These variables reflect the value of the marks VOLCOMP1 to VOLCOMP4 of the PLC. If the PLC activates one of these marks (changing it from 0 to 1), the CNC activates the corresponding volumetric compensation (parameter VOLCOMP). Units: -.	R(*)	-	R/W

(*) The CNC evaluates the variable during execution (it stops the block preparation).

Syntax of the variable.

V.PLC.VOLCOMP1	Volumetric compensation ·1·.

Volumetric compensation.

Variables.	PRG	PLC	INT
(V.)[ch].A.VOLCOMP.xn Value that the volumetric compensation is adding to the axis. If the volumetric compensation is not active, the variable returns the value to 0 (zero). Units: Millimeters, inches or degrees.	R(*)	R	R
(V.)[ch].A.PIVOT.xn 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.	R(*)	R	R

(*) The CNC evaluates the variable during execution (it stops the block preparation).

Syntax of the variable.

·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·.

12.3 Error messages (cause and solution).

0046 'Nonexistent axis'

DETECTION During the validation of the machine parameters.

CAUSE The possible causes are:

- The user has assigned the name of an axis that does not exist to a machine parameter.
- Either the VMOVAXIS or VCOMPAXIS axis is undefined in the basic volumetric compensation.

SOLUTION Correct the machine parameters. The valid axis names are the ones defined in parameter AXISNAME.

1445 'Wrong parameter value'

DETECTION During execution.

CAUSE A parameter has an invalid value in an instruction or a fixed cycle.

SOLUTION Check the program.

1447 'Software option not allowed'

DETECTION During execution.

CAUSE The CNC does not have the software option required to execute the programmed command.

SOLUTION In diagnosis mode, it is possible to check the software options offered by the CNC.

9347 'Axis repeated in VOLCOMP'

DETECTION On CNC power-up or when validating the machine parameters.

CAUSE The same axis is defined twice in the VMOVAXIS or VCOMPAXIS volumetric compensation parameters.

SOLUTION Correct the machine parameters.

23101 'Error when analyzing the volumetric compensation definition file'

DETECTION On CNC power-up or when validating the machine parameters.

CAUSE The volumetric definition file does not contain the proper data.

Basic volumetric compensation:

- The file is not in csv format.
- The format of the file defined in VCOMPFILE is incorrect.
- The file defined in VCOMPFILE does not have all the points defined. The number of points is defined by the expression:
  NPOINTSAX1 × NPOINTSAX2 × NPOINTSAX3

Medium and large volumetric compensation:

- The format of the file defined in VCOMPFILE is incorrect.

SOLUTION Select a proper file. The file must be in csv format and have all the points defined in the basic volumetric compensation.

23102 'Volumetric compensation system error; Compensation off'

DETECTION During execution.

CAUSE The CNC has detected an internal error when calculating the volumetric compensation and has canceled the compensation values.

SOLUTION Contact Fagor Automation.

23103 'The VOLCOMP mark cannot be activated if a table is not validated'

DETECTION During execution.

CAUSE The PLC has tried to activate a volumetric compensation that is not defined properly and therefore the machine cannot be compensated.

SOLUTION Define a proper volumetric compensation in the machine parameters (VOLCOMP parameters).

23104 'The VOLCOMP mark cannot be activated while there is another table active on common axes'

DETECTION	During execution.
CAUSE	The PLC has tried to activate a volumetric compensation while another one is active and one of the axes used is common to both compensations.
SOLUTION	There cannot be two active volumetric compensations having an axis in common. Before activating a compensation, cancel the active one using the corresponding PLC mark.

CONFIGURING THE HSC MODE.

13

Nowadays, lots of parts are designed using CAD-CAM systems. This type of information is later post-processed to generate a CNC program, usually made up of a large number of very short blocks of several mm or just 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.

Default HSC mode.

The command to execute programs made up of lots of small blocks, typical of high speed machining, is carried out with a single instruction #HSC. This function offers several ways to work; optimizing part surface finish (SURFACE mode) optimizing the contour error (CONTERROR mode) or the machining speed (FAST mode).

The default machining mode is defined by parameter HSCDEFAULTMODE, where Fagor offers the SURFACE mode as default. The more sophisticated algorithms of the SURFACE mode obtain more accurate machining. Tests run at Fagor have obtained an average accuracy improvement of 25% to 30%. In parallel, the CNC controls machine movements more smoothly reducing considerably the vibrations due to part geometry or machine dynamics. Reducing machine vibrations results in higher surface quality of the machined parts.

13.1 Influence of the type of acceleration and of the filters in HSC mode.

HSC optimizing the chordal error (CONTERROR mode).

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.

13.2 Configuration of the HSC mode.

13.2.1 Setup. General; machine parameters.

Parameter.		Meaning.
HSCDEFAULMODE	MPG	Default mode when programming #HSC ON.
FEEDAVRG	MPG	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	MPG	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	MPG	This parameter indicates the maximum angle between two paths, under which the machining operation is carried out in square corner mode.
HSCFILTFREQ	MPG	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	MPG	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	MPG	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	MPG	Minimum feedrate at the corners. This value should not be modified because it is possible to exceed the dynamics of the axes.
FSMOOTHFREQ	MPG	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.

[MPG] General machine parameter.

Parameter.		Meaning.
FASTFILTFREQ	MPG	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	MPG	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	MPG	Path filter frequency for profiles of linear acceleration. 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	MPG	Default value of the maximum path error in HSC.
SURFFILFREQ	MPG	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.

[MPG] General machine parameter.

13.2.2 Setup. Machine parameters; axis set.

Parameter.		Meaning.
CORNERACC	MPA	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	MPA	Maximum contouring acceleration permitted. If this parameter is set with a ·0· value, the maximum acceleration of the axis is respected.
CORNERJERK	MPA	Maximum Jerk permitted at the corners. If this parameter is set with a ·0· value, the maximum jerk of the axis is respected.
CURVJERK	MPA	Maximum contouring Jerk permitted. If this parameter is set with a ·0· value, the maximum jerk of the axis is respected.
FASTACC	MPA	Maximum acceleration permitted (FAST mode). If this parameter is set with a ·0· value, the maximum acceleration of the axis is respected.
MAXERROR	MPA	Position error. Maximum axis position error in HSC when working outside the plane/trihedron.

[MPA] machine parameter; axes.

13.3 Variables.

The following variables may be accessed from (PRG) the part-program and from the MDI/MDA mode, PLC and (INT) from an external application. For each variable, the table indicates whether the access is reading (R) or writing (W). PLC access to the variables, for both reading and writing, will be synchronous. Access to the variables from the part program returns the value for the block preparation, except when indicated otherwise.

13.3.1 Analysis of the loop time (cycle time) at the CNC.

Variables.	PRG	PLC	INT
(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 CNCTIME, group the subroutines in a the same file or reduce the number of external applications Units: %.	R(*)	R	R
(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 CNCTIME parameter. Units: %.	R(*)	R	R

(*) The CNC evaluates the variable during execution (it stops the block preparation).

Syntax of the variable.

```
V.G.NCTIMERATE
V.G.LOOPTIMERATE
```

13.3.2 Analysis of the loop time (cycle time) in the channel.

Variables.	PRG	PLC	INT
(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. Units: %.	R(*)	R	R
(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. Units: %.	R(*)	R	R
(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. Units: %.	R(*)	R	R

(*) The CNC evaluates the variable during execution (it stops the block preparation).

Syntax of the variable.

·ch· Channel number.

V.[2].G.CHTIMERATE Channel ·2·.

13.3.3 Variables associated with the feedrate on the tool path.

Variables.	PRG	PLC	INT
(V.)[ch].G.PATHFEED Theoretical feedrate on the tool path. Units: In millimeters/minute or inches/minute.	R(*)	R	R
(V.)[ch].G.FREAL Real filtered feedrate on the tool path. This variable takes into account the accelerations and decelerations of the machine. Units: In millimeters/minute or inches/minute.	R(*)	R	R

(*) The CNC evaluates the variable during execution (it stops the block preparation).

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 of the variable.

·ch· Channel number.

V.[2].G.FREAL Channel ·2·.

13.3.4 Variables associated with feedrate limitation.

Variables.	PRG	PLC	INT
(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. 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. Units: %.	R(*)	R	R
(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. 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. Units: %.	R(*)	R	R
(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. The variable returns one of the following values. [-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. Units: mm or inches.	R(*)	R	R
(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. Units: -.	R(*)	R	R
(V.)[ch].G.PARLIMF Cause that limits the feedrate at the block being executed. The variable returns one of the following values. [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. Units: -.	R(*)	R	R
(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. Units: -.	R(*)	R	R

(*) The CNC evaluates the variable during execution (it stops the block preparation).

Variables.	PRG	PLC	INT
(V.)[ch].G.PARLIMC Cause that limits the feedrate at the corner in the block being executed. The variable returns one of the following values. [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). Units: -.	R(*)	R	R

(*) The CNC evaluates the variable during execution (it stops the block preparation).

Syntax of the variable.

·ch· Channel number.

V.[2].G.PERFRATE	Channel ·2·.

13.3.5 Variables associated with the block being executed.

Variables.	PRG	PLC	INT
(V.)[ch].G.BLKN Last block executed (number). If no label has been executed, the variable will return a -1 value. Units: -.	R	R	R
(V.)[ch].G.LINEN Number of program blocks executed or prepared. 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. Units: -.	R	R	R

Syntax of the variable.

·ch· Channel number.

V.[2].G.BLKN	Channel ·2·.

13.3.6 Variables associated with the coordinates in the loop.

Variables.	PRG	PLC	INT
(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. Units: Millimeters, inches or degrees.	R(*)	R	R
(V.)[ch].A.ADDMANOF.xn Distance moved with G201. Units: Millimeters, inches or degrees.	R(*)	R	R
(V.)[ch].A.INDPOS.xn **(V.)[ch].A.INDPOS.sn** **(V.)[ch].SP.INDPOS.sn** Theoretical coordinate of the independent interpolator. Units: Millimeters, inches or degrees.	R(*)	R	R
(V.)[ch].A.FILTERIN.xn **(V.)[ch].A.FILTERIN.sn** **(V.)[ch].SP.FILTERIN.sn** Theoretical coordinate of the interpolator before the filter. Units: Millimeters, inches or degrees.	R(*)	R	R
(V.)[ch].A.FILTEROUT.xn **(V.)[ch].A.FILTEROUT.sn** **(V.)[ch].SP.FILTEROUT.sn** Theoretical coordinate of the interpolator after the filter. Units: Millimeters, inches or degrees.	R(*)	R	R
(V.)[ch].A.LOOPTPOS.xn **(V.)[ch].A.LOOPTPOS.sn** **(V.)[ch].SP.LOOPTPOS.sn** Theoretical coordinate at the input of the position loop. Units: Millimeters, inches or degrees.	R(*)	R	R
(V.)[ch].A.LOOPPOS.xn **(V.)[ch].A.LOOPPOS.sn** **(V.)[ch].SP.LOOPPOS.sn** Real coordinate at the input of the position loop. Units: Millimeters, inches or degrees.	R(*)	R	R

(*) The CNC evaluates the variable during execution (it stops the block preparation).

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·.

13.3.7 Variables associated with the velocity in the loop.

Variables.	PRG	PLC	INT
(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. Units: -.	R(*)	R	R
(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. Units: -.	R(*)	R	R

(*) The CNC evaluates the variable during execution (it stops the block preparation).

Syntax of the variable.

·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·.

13.3.8 Variables associated with the velocity command and the feedback.

Variables.	PRG	PLC	INT
(V.)[ch].A.POSCMD.xn **(V.)[ch].A.POSCMD.sn** **(V.)[ch].SP.POSCMD.sn** Position command for Sercos. Units: -.	R(*)	R	R
(V.)[ch].A.POSNC.xn **(V.)[ch].A.POSNC.sn** **(V.)[ch].SP.POSNC.sn** Position feedback. Units: -.	R(*)	R	R
(V.)[ch].A.FTEO.xn **(V.)[ch].A.FTEO.sn** **(V.)[ch].SP.FTEO.sn** Velocity command for Sercos (in rpm). Units: rpm.	R(*)	R	R

(*) The CNC evaluates the variable during execution (it stops the block preparation).

Syntax of the variable.

·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·.

13.4 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.G.PERFRATE	Percentage of blocks managed by the CNC in block preparation, with respect to the best possible.
V.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.G.CHTIMERATE	Percentage of loop time (cycle time) used by the channel.
V.G.PREPTIMERATE	Percentage of loop time (cycle time) used by the channel for block preparation.
V.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.G.PERFRATE — A value lower than 100 means that the CNC can process more blocks.

V.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.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.G.PARLIMF	Cause that limits the feedrate at the block being executed.
V.G.AXLIMF	Logic number of the axis that limits the feedrate at the block being executed.
V.G.PARLIMC	Cause that limits the feedrate at the corner in the current block.
V.G.AXLIMC	Logic number of the axis that limits the feedrate at the corner, in the block being executed.
V.G.LIMERROR	Value of the error that cancels the splines (CONTERROR mode).
V.G.PATHFEED	Theoretical feedrate on the tool path.
V.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.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.G.AXLIMF variable to know which axis is limiting the feedrate and the V.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.G.AXLIMC variable to know which axis is limiting the feedrate and the V.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 V.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.G.BLKN	Last block executed (number).
V.G.LINEN	Number of program blocks executed or prepared.
V.G.PATHFEED	Theoretical feedrate on the tool path.

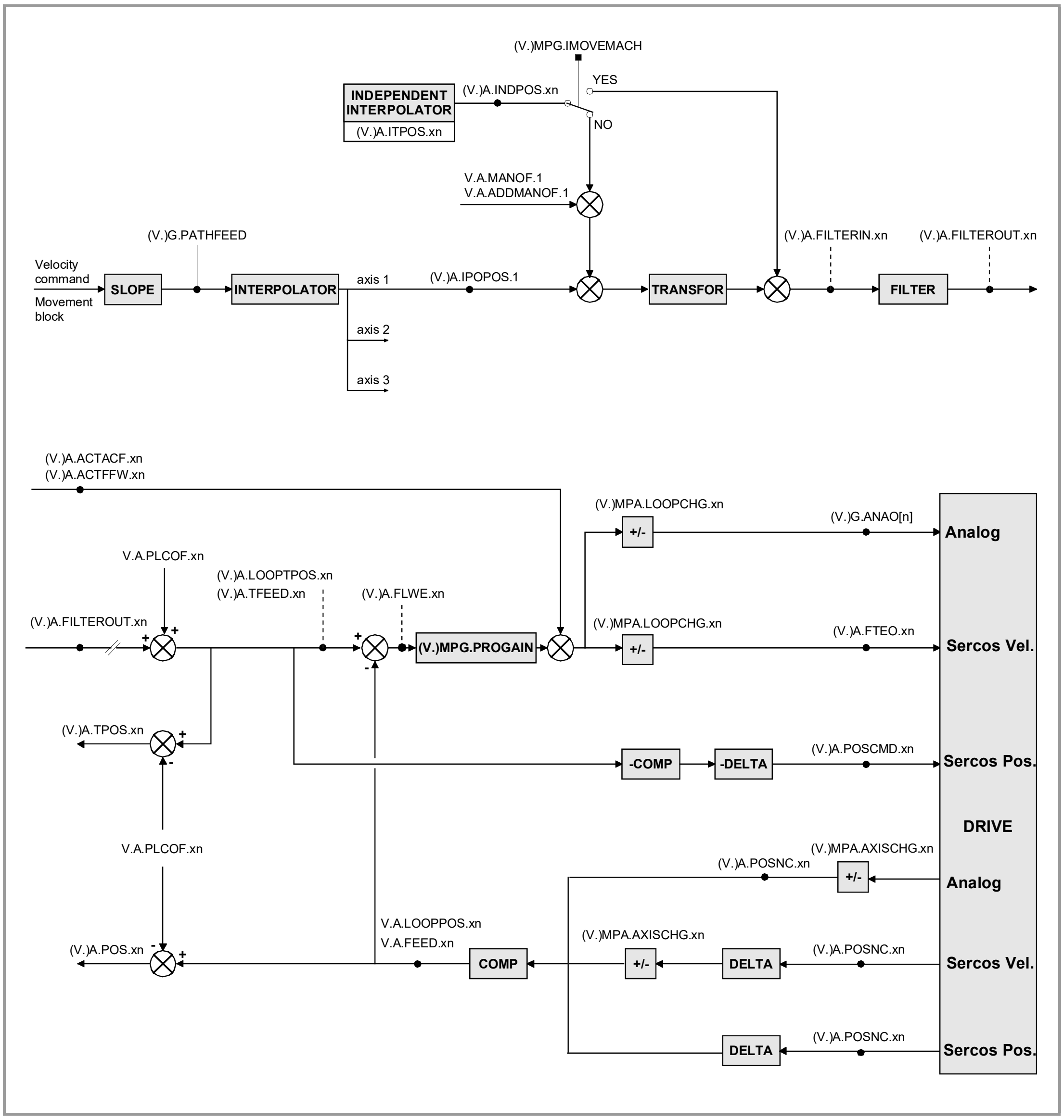
Then, use the oscilloscope to analyze the following variables, sequentially because it only admits four, to find the cause of the feedrate problem.

Variable.	Meaning.
V.G.PARLIMF	Cause that limits the feedrate at the block being executed.
V.G.AXLIMF	Logic number of the axis that limits the feedrate at the block being executed.
V.G.PARLIMC	Cause that limits the feedrate at the corner in the current block.
V.G.AXLIMC	Logic number of the axis that limits the feedrate at the corner, in the block being executed.
V.G.LIMERROR	Value of the error that cancels the splines (CONTERROR mode).
V.G.PATHFEED	Theoretical feedrate on the tool path.
V.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.A.IPOPOS.xn	Theoretical position value (coordinate) at the output of the interpolator, before the transformation; i.e. in part coordinates.
V.A.ADDMANOF.xn	Distance moved with G201.
V.A.INDPOS.xn	Theoretical coordinate of the independent interpolator.
V.A.FILTERIN.xn	Theoretical coordinate of the interpolator before the filter.
V.A.FILTEROUT.xn	Theoretical coordinate of the interpolator after the filter.
V.A.LOOPTPOS.xn	Theoretical coordinate at the input of the position loop.
V.A.LOOPPOS.xn	Real coordinate at the input of the position loop.
V.A.TFEED.xn	Instantaneous theoretical speed value at the input of the position loop.
V.A.FEED.xn	Instantaneous real speed value at the input of the position loop.
V.A.POSCMD.xn	Position command for Sercos.
V.A.FTEO.xn	Velocity command for Sercos (in rpm).
V.A.POSNC.xn	Position feedback.

13.5 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.

TOOL AND MAGAZINE MANAGEMENT.

14

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 *"14.1 Types of tool magazine."* on page 601.

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 *"14.3 Communication between manager and PLC."* on page 604.

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.

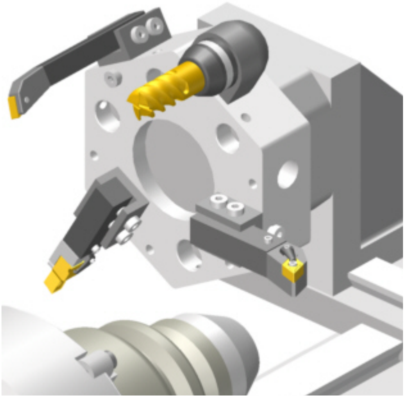
This subroutine is common to the whole system.

i

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.

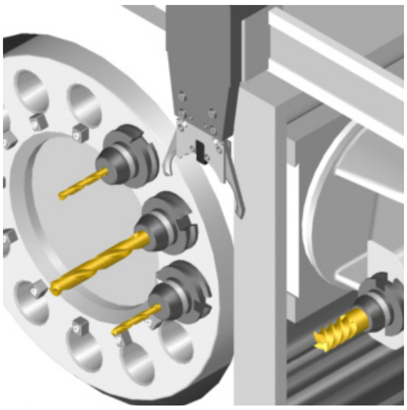
14.1 Types of tool magazine.

Tool magazines may be divided into 4 large groups:

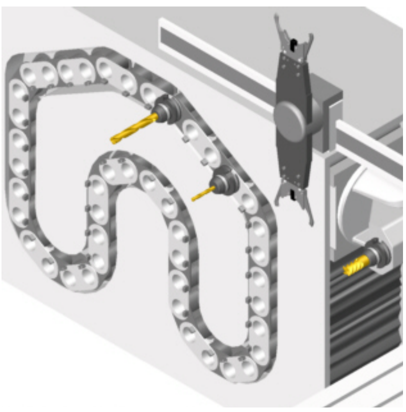


Turret type.

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.

14.2 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.

14.3 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 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.

14.3.1 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

The meaning of the marks and registers is as follows.

Marks / Registers.	Meaning,
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.

14.3.2 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	SPDLTOMZ SPDLTOMZ1	SPDLTOMZ2	SPDLTOMZ3	SPDLTOMZ4
M	MZROT MZROTMZ1	MZROTMZ2	MZROTMZ3	MZROTMZ4
M	TCHANGEOK TCHANGEOKMZ1	TCHANGEOKMZ2	TCHANGEOKMZ3	TCHANGEOKMZ4
R	MZPOS MZPOSMZ1	MZPOSMZ2	MZPOSMZ3	MZPOSMZ4

The meaning of the marks and registers is as follows.

Marks / Registers.	Meaning,
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.

Marks / Registers.	Meaning,
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.

14.3.3 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

The meaning of the marks and registers is as follows.

Marks / Registers.	Meaning,
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).

14.3.4 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 meaning of the marks and registers is as follows.

Marks / Registers.	Meaning,
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).

14.4 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 [ch] character with the channel number, maintaining the brackets.

V.[ch].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.[ch].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.[ch].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.[ch].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.

14.5 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.

14.6 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.

14.6.1 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

14.6.2 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.

14.6.3 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
```

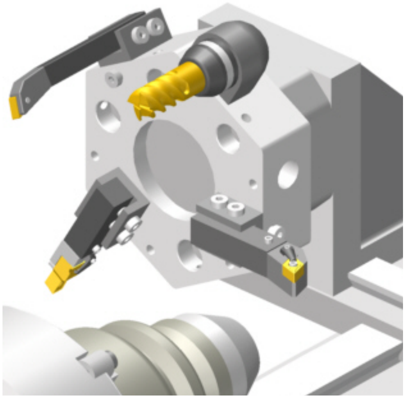
A tool change has been requested and TMOPERATION=4. The PLC has taken the tool from the spindle to ground.

```
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.

14.7 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.

It is recommended that the routine associated with T include the M06 function. When the CNC selects a tool, the routine associated with the T executes the M06 function (tool change), the machining stops and the tool manager sends the PLC the code for the operation to be carried out. If the subroutine associated with T does not include the function M06, when the CNC selects a tool, it internally executes a T# M6 block to make the change.

Subroutine associated with T.	Behavior when the T function is executed.
Exists.	The CNC executes the T function and the subroutine executes the M06 block.
It does not exist.	The CNC internally generates a T# M06 block.

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.

14.7.1 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

14.7.2 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.

14.7.3 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

Marks activated by the M06 subroutine so the PLC activates the relevant mark of the tool manager.

Mark.	Meaning.
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:

Function.	Meaning.
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.

14.7.4 Program of the M06 subroutine.

```
%L SUB_MZ_ROT
   V.PLC.M[1111]=1
   #WAIT FOR [V.PLC.M[1111]==0]; MZROT mark to the tool manager.
#RET

%L SUB_SPD_TO_GR   ;  Remove the tool from the spindle (take it to ground).
   #MSG ["Extract tool T%D and press START", V.TM.TOOL]
                             ; Message for the operator to extract the tool.
   M0                        ; Wait for the operation to be completed.
   #MSG [""]                 ; Remove message.
   V.PLC.M[1107]=1           ; SPDLTOGR mark to the tool manager.
   #WAIT FOR [V.PLC.M[1107]==0]
#RET

%L SUB_GR_TO_SPD   ;  Insert the ground tool in the spindle.
   #MSG ["Insert tool T%D and press START", V.TM.NXTOOL]
                             ; Message for the operator to insert the tool.
   M0                        ; Wait for the operation to be completed.
   #MSG [""]                 ; Remove message.
   V.PLC.M[1108]=1           ; GRTOSPDL mark to the tool manager.
   #WAIT FOR [V.PLC.M[1108]==0]
#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.
   #WAIT FOR [V.PLC.M[1109]==0]
#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.
   #WAIT FOR [V.PLC.M[1110]==0]
#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
   #ESBLK                    ; Begin of single block activation
   #DSTOP                    ; STOP key disable
   M6                        ; Order the tool manager to start the tool change.

   $IF [[[V.G.FULLSTATUS & 255]<9] | [[V.G.FULLSTATUS & 255]>13]]
     $IF [[V.G.CNCAUTSTATUS & 4096] | [V.G.CNCAUTSTATUS & 8192]]
       V.S.EXECUTION = 0     ; Simulated Theorical or G
     $ELSE
       V.S.EXECUTION = 1     ; Execution
     $ENDIF
   $ELSE
     V.S.EXECUTION = 0       ; Simulation
   $ENDIF

   $IF [V.S.EXECUTION == 1]  ; CNC in execution

   $SWITCH V.PLC.TMOPERATION ; 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.

    $ENDIF
    #DSBLK                    ; End of single block activation.
    #ESTOP                    ; STOP key enable
#RET
```

14.7.5 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

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.

14.8 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.

14.8.1 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

14.8.2 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.

14.8.3 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

Marks activated by the M06 subroutine so the PLC activates the relevant mark of the tool manager.

Mark.	Meaning.
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:

Function.	Meaning.
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.

14.8.4 Program of the M06 subroutine.

%L SUB_SPD_TO_GR ; Remove the tool from the spindle (take it to ground).

```
    #MSG ["Extract tool T%D and press START", V.TM.TOOL]
                                 ; Message for the operator to extract the tool.
    M0                           ; Wait for the operation to be completed.
    #MSG [""]                    ; Remove message.
    V.PLC.M[1107]=1              ; SPDLTOGR mark to the tool manager.
    #WAIT FOR [V.PLC.M[1107]==0]
#RET
```

%L SUB_GR_TO_SPD ; Insert the ground tool in the spindle.

```
    #MSG ["Insert tool T%D and press START", V.TM.NXTOOL]
                                 ; Message for the operator to insert the tool.
    M0                           ; Wait for the operation to be completed.
    #MSG [""]                    ; Remove message.
    V.PLC.M[1108]=1              ; GRTOSPDL mark to the tool manager.
    #WAIT FOR [V.PLC.M[1108]==0]
#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.
    #WAIT FOR [V.PLC.M[1109]==0]
#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.
    #WAIT FOR [V.PLC.M[1110]==0]
#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

```
    #ESBLK                       ; Begin of single block activation
    #DSTOP                       ; STOP key disable
    M6                           ; Order the tool manager to start the tool change.

    $IF [[[V.G.FULLSTATUS & 255]<9] | [[V.G.FULLSTATUS & 255]>13]]
      $IF [[V.G.CNCAUTSTATUS & 4096] | [V.G.CNCAUTSTATUS & 8192]]
        V.S.EXECUTION = 0        ; Simulated Theorical or G
      $ELSE
        V.S.EXECUTION = 1        ; Execution
      $ENDIF
    $ELSE
      V.S.EXECUTION = 0          ; Simulation
    $ENDIF

    $IF [V.S.EXECUTION == 1]     ; CNC in execution

    $SWITCH V.PLC.TMOPERATION    ; 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           ; 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.
$ENDIF
#DSBLK                        ; End of single block activation.
#ESTOP                        ; STOP key enable
#RET
```

14.8.5 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

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
```

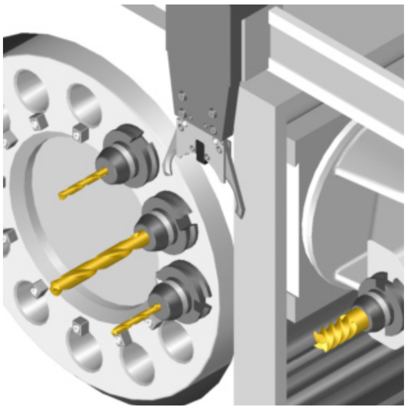
The lamp of the USER12 key turns on when there is an emergency.

```
TMINEM AND DFU B12KEYBD1 = SET RESTMEM
```

Pressing the USER13 key removes the emergency.

14.9 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.

14.9.1 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.
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
8	0	0	SPDLTOGR + GRTOSPDL

TM => PLC			PLC => TM
TMOPERATION	TAKEPOS	LEAVEPOS	
9	-4	#	GRTOSPDL + SPDLTOCH1 + CH1TOMZ
10	#	-4	MZTOCH1 + CH1TOSPDL + SPDLTOGR
11	0	0	MZROT
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

14.9.2 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 = 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.

14.9.3 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

Marks activated by the M06 subroutine so the PLC activates the relevant mark of the tool manager.

Mark.	Meaning.
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:

Function.	Meaning.
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.

14.9.4 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.
    #WAIT FOR [V.PLC.M[1101]==0]
#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.
    #WAIT FOR [V.PLC.M[1102]==0]
#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.
    #WAIT FOR [V.PLC.M[1103]==0]
#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.
    #WAIT FOR [V.PLC.M[1105]==0]
#RET
```

%L SUB_SPD_TO_GR ; Remove the tool from the spindle (take it to ground).

```
    #MSG ["Extract tool T%D and press START", V.TM.TOOL]
                                ; Message for the operator to extract the tool.
    M0                          ; Wait for the operation to be completed.
    #MSG [""]                   ; Remove message.
    V.PLC.M[1107]=1             ; SPDLTOGR mark to the tool manager.
    #WAIT FOR [V.PLC.M[1107]==0]
#RET
```

%L SUB_GR_TO_SPD ; Insert the ground tool in the spindle.

```
    #MSG ["Insert tool T%D and press START", V.TM.NXTOOL]
                                ; Message for the operator to insert the tool.
    M0                          ; Wait for the operation to be completed.
    #MSG [""]                   ; Remove message.
    V.PLC.M[1108]=1             ; GRTOSPDL mark to the tool manager.
    #WAIT FOR [V.PLC.M[1108]==0]
#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

```
    #ESBLK                      ; Begin of single block activation
    #DSTOP                      ; STOP key disable
    M6                          ; Order the tool manager to start the tool change.

    $IF [[[V.G.FULLSTATUS & 255]<9] | [[V.G.FULLSTATUS & 255]>13]]
      $IF [[V.G.CNCAUTSTATUS & 4096] | [V.G.CNCAUTSTATUS & 8192]]
        V.S.EXECUTION = 0       ; Simulated Theorical or G
      $ELSE
        V.S.EXECUTION = 1       ; Execution
      $ENDIF
    $ELSE
      V.S.EXECUTION = 0         ; Simulation
    $ENDIF
```

```
$IF [V.S.EXECUTION == 1]       ; CNC in execution

$SWITCH V.PLC.TMOPERATION ; 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.

    $ENDIF
    #DSBLK                     ; End of single block activation.
    #ESTOP                     ; STOP key enable
#RET
```

14.9.5 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

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
```

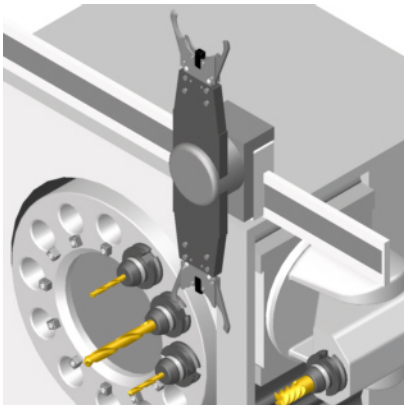
The lamp of the USER12 key turns on when there is an emergency.

```
TMINEM AND DFU B12KEYBD1 = SET RESTMEM
```

Pressing the USER13 key removes the emergency.

14.10 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.

14.10.1 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

14.10.2 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.

14.10.3 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

Marks activated by the M06 subroutine so the PLC activates the relevant mark of the tool manager.

Mark.	Meaning.
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:

Function.	Meaning.
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.

14.10.4 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.
   #WAIT FOR [V.PLC.M[1101]==0]
#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.
   #WAIT FOR [V.PLC.M[1102]==0]
#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.
   #WAIT FOR [V.PLC.M[1104]==0]
#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.
   #WAIT FOR [V.PLC.M[1106]==0]
#RET

%L SUB_SPD_TO_GR   ;  Remove the tool from the spindle (take it to ground).
   #MSG ["Extract tool T%D and press START", V.TM.TOOL]
                                 ; Message for the operator to extract the tool.
   M0                            ; Wait for the operation to be completed.
   #MSG [""]                     ; Remove message.
   V.PLC.M[1107]=1               ; SPDLTOGR mark to the tool manager.
   #WAIT FOR [V.PLC.M[1107]==0]
#RET

%L SUB_GR_TO_SPD   ;  Insert the ground tool in the spindle.
   #MSG ["Insert tool T%D and press START", V.TM.NXTOOL]
                                 ; Message for the operator to insert the tool.
   M0                            ; Wait for the operation to be completed.
   #MSG [""]                     ; Remove message.
   V.PLC.M[1108]=1               ; GRTOSPDL mark to the tool manager.
   #WAIT FOR [V.PLC.M[1108]==0]
#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
   #ESBLK                        ; Begin of single block activation
   #DSTOP                        ; STOP key disable
   M6                            ; Order the tool manager to start the tool change.

   $IF [[[V.G.FULLSTATUS & 255]<9] | [[V.G.FULLSTATUS & 255]>13]]
     $IF [[V.G.CNCAUTSTATUS & 4096] | [V.G.CNCAUTSTATUS & 8192]]
       V.S.EXECUTION = 0         ; Simulated Theorical or G
     $ELSE
       V.S.EXECUTION = 1         ; Execution
     $ENDIF
   $ELSE
     V.S.EXECUTION = 0           ; Simulation
   $ENDIF
```

```
$IF [V.S.EXECUTION == 1]       ; CNC in execution

$SWITCH V.PLC.TMOPERATION ; 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.

    $ENDIF
    #DSBLK                      ; End of single block activation.
    #ESTOP                      ; STOP key enable
#RET
```

14.10.5 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

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
```

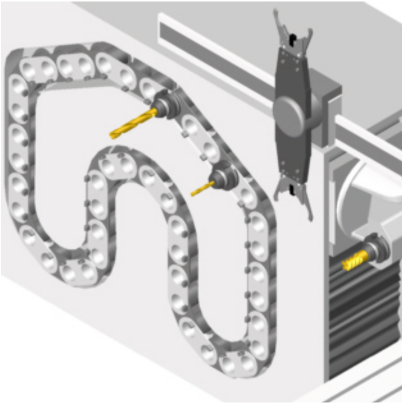
The lamp of the USER12 key turns on when there is an emergency.

```
TMINEM AND DFU B12KEYBD1 = SET RESTMEM
```

Pressing the USER13 key removes the emergency.

14.11 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.

14.11.1 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.
12	Leave the spindle tool in the magazine and take a tool from the same magazine (as TMOPERATION=5). It is an optimised operation, valid when both tools are special and can collide in the arm or when it is not desirable for them to coincide in the changer 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.
16	Leave the tool of the arm (holder 2) in the magazine. Take a tool from the magazine and insert it in the arm (holder 1).

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			PLC => TM
TMOPERATION	TAKEPOS	LEAVEPOS	
1	-1	0	CH1TOSPDL
2	0	#	SPDLTOCH2 + TCHANGEOK + CH2TOMZ
3	0	0	GRTOSPDL
4	0	0	SPDLTOGR

TM => PLC TMOPERATION	TAKEPOS	LEAVEPOS	PLC => TM
5	-1	#	(a) SPDLTOCH2 + CH2TOMZ + CH1TOSPDL (b) SPDLTOCH2 + CH1TOSPDL + TCHANGEOK + CH2TOMZ
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
16	#	#	CH2TOMZ + MZTOCH1

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 + (TCHANGEOK) + CH2TOMZ
T7 M6	0 6	0 -4	0 2	- - - SPDLTOCH2 + CH2TOMZ + GRTOSPDL
T8 M6	0 8	0 -4	0 -4	- - - SPDLTOGR + GRTOSPDL
T3 T4 M6	11 11 7	3 4 -1	0 3 -4	MZTOCH1 CH1TOMZ + MZTOCH1 SDPLTOGR + CH1TOSPDL
T0 M6	0 2	0 0	0 4	- - - SPDLTOCH2 + (TCHANGEOK) + CH2TOMZ
T9 M6	0 3	0 -4	0 0	- - - GRTOSPDL
T0 M6	0 4	0 0	0 -4	- - - SPDLTOGR
T3 T3 M6	11 0 1	3 0 -1	0 0 0	MZTOCH1 - - - CH1TOSPDL
T5 T8 M6	11 11 6	5 0 -4	0 5 3	MZTOCH1 CH1TOSPDL SPDLTOCH2 + CH2TOMZ + GRTOSPDL

14.11.2 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 = 12

Leave the spindle tool in the magazine and take a tool from the same magazine (as TMOPERATION=5). It is an optimised operation, valid when both tools are special and can collide in the arm or when it is not desirable for them to coincide in the changer arm.

TAKEPOS=#	Magazine position to pick up the tool.
LEAVEPOS=#	Magazine position to leave the tool.

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.

TMOPERATION = 16

Leave the tool of the arm (holder ·2·) in the magazine. Take a tool from the magazine and insert it in the arm (holder ·1·). This operation is an optimization of the tool change that permits placing the tool of claw ·2· of the tool changer arm in the magazine.

The sequence of this operation is the following.

TAKEPOS=#	Position occupied by the tool.
LEAVEPOS=#	Position to leave the tool.

1 Leave the tool of holder ·2· in the magazine and activate the markCH2TOMZ.
2 Take the tool from the magazine with holder 1 and activate the mark MZTOCH1.

14.11.3 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

Marks activated by the M06 subroutine so the PLC activates the relevant mark of the tool manager.

Mark.	Meaning.
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:

Function.	Meaning.
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.

14.11.4 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.
    #WAIT FOR [V.PLC.M[1101]==0]
#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.
    #WAIT FOR [V.PLC.M[1102]==0]
#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.
    #WAIT FOR [V.PLC.M[1104]==0]
#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).

```
    #MSG ["Extract tool T%D and press START", V.TM.TOOL]
                                  ; Message for the operator to extract the tool.
    M0                            ; Wait for the operation to be completed.
    #MSG [""]                     ; Remove message.
    V.PLC.M[1107]=1               ; SPDLTOGR mark to the tool manager.
    #WAIT FOR [V.PLC.M[1107]==0]
#RET
```

%L SUB_GR_TO_SPD ; Insert the ground tool in the spindle.

```
    #MSG ["Insert tool T%D and press START", V.TM.NXTOOL]
                                  ; Message for the operator to insert the tool.
    M0                            ; Wait for the operation to be completed.
    #MSG [""]                     ; Remove message.
    V.PLC.M[1108]=1               ; GRTOSPDL mark to the tool manager.
    #WAIT FOR [V.PLC.M[1108]==0]
#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

```
#ESBLK                        ; Begin of single block activation
#DSTOP                        ; STOP key disable
M6                            ; Order the tool manager to start the tool change.

$IF [[[V.G.FULLSTATUS & 255]<9] | [[V.G.FULLSTATUS & 255]>13]]
  $IF [[V.G.CNCAUTSTATUS & 4096] | [V.G.CNCAUTSTATUS & 8192]]
    V.S.EXECUTION = 0         ; Simulated Theorical or G
  $ELSE
    V.S.EXECUTION = 1         ; Execution
  $ENDIF
$ELSE
  V.S.EXECUTION = 0           ; Simulation
$ENDIF

$IF [V.S.EXECUTION == 1]      ; CNC in execution

$SWITCH V.PLC.TMOPERATION ; 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.

V.PLC.TMOPERATION = 0
$WHILE V.TM.MZWAIT == 1
$ENDWHILE                      ; Wait for the tool manager.

$ENDIF
#DSBLK                         ; End of single block activation.
#ESTOP                         ; STOP key enable
#RET
```

14.11.5 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

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.

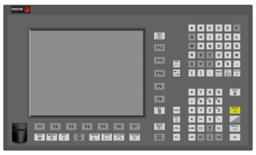
KEYBOARD SIMULATION VIA THE PLC. KEY CODES.

15

15.1 Selecting the language and the keyboard distribution.

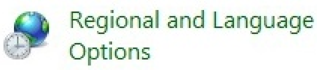
The keyboard distribution controls the characters that appear on screen when the keys are pressed. So that the characters on the screen to match the keys, the appropriate keyboard layout must be selected in the operating system. The key codes (scan codes) also depend on the selected keyboard layout.

In order to change the keyboard distribution, the desired input language and keyboard distribution must be added to Windows.

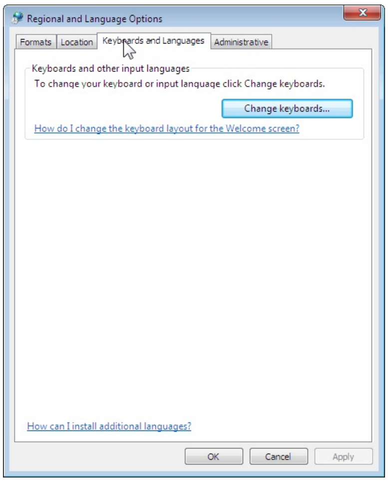
Keyboards.		
 Spanish (Spain).	 Spanish (Spain).	 Spanish (Spain).
 English (United States).	 Spanish (Spain).	 Spanish (Spain).

Adding an input language and a keyboard distribution.

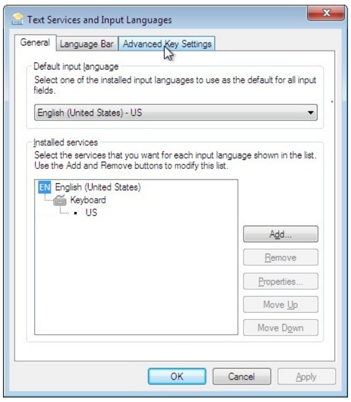
1 Click on the Start menu > Control panel > Regional and language configuration. Depending on how Windows is configured, it may be necessary to first select "Clock, language and region".



2 Click on the "Keyboards and languages" tab and then click on "Change keyboards".



3 In the "Installed services” section, click on the button "Add". Add the desired input language and the keyboard distribution (in this case, English (United States)). Click on "OK" to finish.



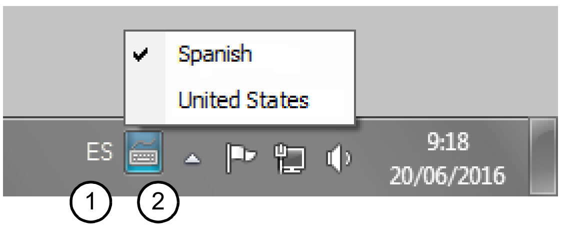
Changing the input language.

On the login screen, click on the language button (top left corner of the screen) and select the language "English (United States)".

Change the input language (only for the active window).

1 In the language bar, click on the button "Input language" and select the language "English (United States)".

2 Next, click on the button "Keyboard distribution" and select the distribution "United States".



15.2 Key codes.

The key codes (scan codes) may be used to determine which is the last key accepted by the CNC and also to simulate the keyboard from the PLC. Each key is assigned two codes; one for pressing it and another one for releasing it. The code does not depend on the key (printed character), but rather its position on the keyboard (keyboard layout). For example, the code $27 on a "Spanish (Spain)" keyboard layout corresponds to the character "Ñ" and on an "English (United States)" keyboard layout, it corresponds to the character ";". See *"15.5.1 Scan codes. "Spanish (Spain)" keyboard layout."* on page 677. See *"15.5.2 Scan codes. Keyboard distribution "English (United States)"."* on page 678.

CNC's own keys.

The keys of the CNC have no associated key code.

- The [START], [STOP] and [RESET] keys may be actuated from the PLC using the CYSTART, _STOP and RESETIN keys respectively.
- The keys that have an associated hotkey, such as those for accessing work modes, may be simulated through their hotkeys.
- Keys that do not have an associated hotkey cannot be simulated.
- The Fagor logo key cannot be simulated. This key is not found on all keyboards.

The task window ([CTRL] + [A]) shows the list of available hotkeys for the CNC.

15.2.1 Keyboard shortcuts. CNC's own keys.

Work mode.	Key.	Hotkey.
Task window.		[CTRL]+[A]
Main menu.	MAIN MENU	[CTRL]+[SHIFT]+[F1]
Automatic mode.	AUTO	[CTRL]+[F6]
Manual mode.	MANUAL	[CTRL]+[F7]
MDI/MDA mode.	MDI	[CTRL]+[F8]
EDISIMU mode.	EDIT	[CTRL]+[F9]
User tables.	TABLES	[CTRL]+[F10]
Tool and magazine table.	TOOLS	[CTRL]+[F11]
Utilities mode.	UTILITIES	[CTRL]+[F12]
User mode.		[SHIFT]+[F1]
Calculator.	CALC	[CTRL]+[K]
Help.	HELP	[CTRL]+[F4]
Toolbar to modify the machine dynamics.		[CTRL]+[H]

Operations at the CNC.		Hotkey.
Previous horizontal menu.	BACK	[CTRL]+[F1]
Window change.	FOCUS	[CTRL]+[F2]
Changing the screen.	NEXT	[CTRL]+[F3]
Two-color key.		[ALT]+[B]
Restore a value.	RECALL	[CTRL]+[F5]
"Single block" mode.	SINGLE	[CTRL]+[B]
Show/Hide virtual operator panel.		[CTRL]+[J]
Show/Hide PLC messages.		[CTRL]+[M]
Show/Hide CNC messages.		[CTRL]+[O]
Minimize/Restore the CNC.		[CTRL]+[W]
Turn the CNC off.		[ALT]+[F4]
Channel synchronization window.		[ALT]+[S]
Show / hide the window for errors and warnings.		[ALT]+[W]

15.3 Consulting the last key accepted by the CNC.

The (V.)G.KEY variable keeps the code of the last key accepted by the CNC (press key code). This variable can be read and written from the PLC and read from the program and interface. To read the variable from the PLC, use the command CNCRD.

Reading variables from the PLC.
The command CNCRD saves the code of the key in the register R100. The PLC activates the M100 mark when the operation begins and keeps it active until the operation ends.

```
{condition} = CNCRD(G.KEY,R100,M100)
```

15.4 Keyboard simulation via the PLC.

To simulate the keyboard from the PLC, write the code of the key in the variable (V.)G.KEY. Each key is assigned two codes; one for pressing it and another one for releasing it. For each key that the PLC sends, both codes must be written; it is recommended to use a delay of at least 200 ms between both codes (for safety). To write the variable from the PLC, use the command CNCRD.

Writing variables from the PLC.
The command CNCWR writes the value of the R101 register in the variable. The PLC activates the M101 mark when the operation begins and keeps it active until the operation ends.

```
{condition} = CNCWR(R101,G.KEY,M101)
```

Uppercase and lowercase letters.

One key can have more than one character. To send the first character (lowercase), use the indicated codes; for the second character (uppercase), add the code corresponding to the [SHIFT] key, in the following way. When sending the two codes, it is recommended to use a delay of at least 200 ms between each (for safety).

1 Send the code associated for pressing the [SHIFT] key.
2 Send the code associated with pressing the letter key.
3 Send the code associated for releasing the [SHIFT] key.
4 Send the code associated with releasing the letter key.

Initialize the records.

```
() = MOV $1E R200   ; Press the [A] key
() = MOV $9E R201   ; Release the [A] key
() = MOV $2A R202   ; Press the [SHIFT] key
() = MOV $AA R203   ; Release the [SHIFT] key
```

The first user key (B0KEYBD1) sends the character "a".

```
DFU B0KEYBD1 = CNCWR(R200,G.KEY,M200) = TG1 200 200
T200 = M300
DFD M300 = CNCWR(R201,G.KEY,M201) = TG1 201 200
T201 = M301
```

The second user key (B1KEYBD1) sends the character "A".

```
DFU B1KEYBD1 = CNCWR(R202,G.KEY,M202)
     = CNCWR(R200,G.KEY,M203)
     = TG1 202 200
T202 = M302
DFD M302 = CNCWR(R201,G.KEY,M204)
     = CNCWR(R203,G.KEY,M205)
     = TG1 203 200
T203 =M303
```

Keystroke combination.

To simulate from a keystroke combination from the PLC (for example [CTRL]+[F1]), proceed as follows. When sending the two codes, it is recommended to use a delay of at least 200 ms between each (for safety).

1 Send the code associated for pressing the first key (for example, the [CTRL] key).
2 Send the code associated for pressing the second key (for example, the [F1] key).
3 Send the code associated for releasing the first key (for example, the [CTRL] key).
4 Send the code associated for releasing the second key (for example, the [F1] key).

Initialize the records.

```
() = MOV $1D R204   ; Press the [CTRL] key
() = MOV $9D R205   ; Release the [CTRL] key
() = MOV $3B R206   ; Press the [F1] key
() = MOV $BB R207   ; Release the [F1] key
```

The first user key (B0KEYBD1) sends the combination [CTRL]+[F1].

```
DFU B0KEYBD1 = CNCWR(R204,G.KEY,M204)
      = CNCWR(R206,G.KEY,M206)
      = TG1 202 200
T202 = M302
DFD M302 = CNCWR(R207,G.KEY,M207)
      = CNCWR(R205,G.KEY,M205)
      = TG1 203 200
T203 =M303
```

15.5 Key codes depending on keyboard layout.

15.5.1 Scan codes. "Spanish (Spain)" keyboard layout.

On keys with several characters, to send the first character (lowercase), use the indicated codes; for the second character (uppercase), add the code corresponding to the [SHIFT] key, as follows; for the third character, use the code corresponding to the [ALT GR] key.

Character.	Códigos de tecla.	
	Press.	Release.
a A	$1E	$9E
b B	$30	$B0
c C	$2E	$AE
d D	$20	$A0
e E	$12	$92
f F	$21	$A1
g G	$22	$A2
h H	$23	$A3
i I	$17	$97
j J	$24	$A4
k K	$25	$A5
l L	$26	$A6
m M	$32	$B2
n N	$31	$B1
ñ Ñ	$27	$A7
o O	$18	$98
p P	$19	$99
q Q	$10	$90
r R	$13	$93
s S	$1F	$9F
t T	$14	$94
u U	$16	$96
v V	$2F	$AF
w W	$11	$91

Character.	Códigos de tecla.	
	Press.	Release.
x X	$2D	$AD
y Y	$15	$95
z Z	$2C	$AC
ç Ç }	$2B	$AB
º ª \	$29	$A9
' ?	$0C	$8C
¡ ¿	$0D	$8D
` ^ [	$1A	$9A
+ *]	$1B	$9B
´ ¨ {	$28	$A8
, ;	$33	$B3
. :	$34	$B4
- _	$35	$B5
< >	$56	$D6
0 = |	$0B	$8B
1 ! @	$02	$82
2 " #	$03	$83
3 ·	$04	$84
4 $ ~	$05	$85
5 %	$06	$86
6 &	$07	$87
7 /	$08	$88
8 (	$09	$89
9)	$0A	$8A

Key.	Códigos de tecla.	
	Press.	Release.
F1	$3B	$BB
F2	$3C	$BC
F3	$3D	$BD
F4	$3E	$BE
F5	$3F	$BF
F6	$40	$C0
F7	$41	$C1
F8	$42	$C2
F9	$43	$C3
F10	$44	$C4
F11	$57	$D7
F12	$58	$D8
Escape.	$01	$81
Tab.	$0F	$8F
Enter.	$1C	$9C
Insert.	$E0$52	$E0$D2
Delete.	$E0$53	$E0$D3

Key.	Códigos de tecla.	
	Press.	Release.
Shift lock.	$3A	$BA
Uppercase (SHIFT).	$2A/ $36	$AA / $B6
Control (CTRL).	$1D	$9D
Alternative (ALT).	$38	$B8
Alternative graphic (ALT GR).	$E0$38	$E0$B8
Space bar.	$39	$B9
Back (Backspace).	$0E	$8E
Print screen.	$E0$37	$E0$B7
Page up.	$E0$51	$E0$D1
Page down.	$E0$49	$E0$C9
Begin.	$E0$47	$E0$C7
End.	$E0$4F	$E0$CF
Up arrow.	$E0$48	$E0$C8
Down arrow.	$E0$50	$E0$D0
Right arrow.	$E0$4D	$E0$CD
Left arrow.	$E0$4B	$E0$CB

15.5.2 Scan codes. Keyboard distribution "English (United States)".

On keys with several characters, to send the first character (lowercase), use the indicated codes; for the second character (uppercase), add the code corresponding to the [SHIFT] key.

Character.	Códigos de tecla.	
	Press.	Release.
a A	$1E	$9E
b B	$30	$B0
c C	$2E	$AE
d D	$20	$A0
e E	$12	$92
f F	$21	$A1
g G	$22	$A2
h H	$23	$A3
i I	$17	$97
j J	$24	$A4
k K	$25	$A5
l L	$26	$A6
m M	$32	$B2
n N	$31	$B1
o O	$18	$98
p P	$19	$99
q Q	$10	$90
r R	$13	$93
s S	$1F	$9F
t T	$14	$94
u U	$16	$96
v V	$2F	$AF
w W	$11	$91

Character.	Códigos de tecla.	
	Press.	Release.
x X	$2D	$AD
y Y	$15	$95
z Z	$2C	$AC
‘ ~	$29	$A9
- _	$0C	$8C
= +	$0D	$8D
[{	$1A	$9A
] }	$1B	$9B
; :	$27	$A7
’ "	$28	$A8
\ |	$2B	$AB
, <	$33	$B3
. >	$34	$B4
/ ?	$35	$B5
1 !	$02	$82
2 @	$03	$83
3 #	$04	$84
4 $	$05	$85
5 %	$06	$86
6 ^	$07	$87
7 &	$08	$88
8 *	$09	$89
9 (	$0A	$8A
0)	$0B	$8B

Key.	Códigos de tecla.	
	Press.	Release.
F1	$3B	$BB
F2	$3C	$BC
F3	$3D	$BD
F4	$3E	$BE
F5	$3F	$BF
F6	$40	$C0
F7	$41	$C1
F8	$42	$C2
F9	$43	$C3
F10	$44	$C4
F11	$57	$D7
F12	$58	$D8
Escape.	$01	$81
Tab.	$0F	$8F
Enter.	$1C	$9C
Insert.	$E0$52	$E0$D2
Delete.	$E0$53	$E0$D3

Key.	Códigos de tecla.	
	Press.	Release.
Shift lock.	$3A	$BA
Uppercase (SHIFT).	$2A/ $36	$AA / $B6
Control (CTRL).	$1D	$9D
Alternative (ALT).	$38	$B8
Alternative graphic (ALT GR).	$E0$38	$E0$B8
Space bar.	$39	$B9
Back (Backspace).	$0E	$8E
Print screen.	$E0$37	$E0$B7
Page up.	$E0$51	$E0$D1
Page down.	$E0$49	$E0$C9
Begin.	$E0$47	$E0$C7
End.	$E0$4F	$E0$CF
Up arrow.	$E0$48	$E0$C8
Down arrow.	$E0$50	$E0$D0
Right arrow.	$E0$4D	$E0$CD
Left arrow.	$E0$4B	$E0$CB

15.6 Example. Keyboard simulation via the PLC.

The first user key (B0KEYBD1) performs the following operations.

1. The CNC goes into jog mode.
2. The CNC goes into MDI mode.
3. The CNC performs a machine home search on the X axis.
4. After the home search, the CNC exits the MDI mode.

For each key to be sent from the PLC, the corresponding 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).

```
; Initialize the records.
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 ; [START] (CYSTART=1)
DFD ZERO = SET M501
; The CNC comes out of 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
```

16 SUBROUTINES.

The CNC has a series of subroutines that must be defined depending on the configuration of the machine (subroutine associated with a tool change, home search, etc.).

Subroutine.	Meaning.
Subroutine associated with the start.	Name: PROGRAM_START PROGRAM_START_C1 PROGRAM_START_C2 PROGRAM_START_C3 PROGRAM_START_C4 Channels: Subroutine per channel. Path: ..\Users\Sub.
Subroutine associated with the reset.	Name: PROGRAM_RESET PROGRAM_RESET_C1 PROGRAM_RESET_C2 PROGRAM_RESET_C3 PROGRAM_RESET_C4 Channels: Subroutine per channel. Path: ..\Users\Sub.
Subroutines associated with the kinematics calibration cycle.	Name: KinCal_Begin.nc KinCal_End.nc Channels: Subroutine common to all the channels. Path: ..\Mtb\Sub.
Subroutines associated with the tool calibration cycle.	Name: Sub_Probe_Tool_Begin.fst Sub_Probe_Tool_End.fst Channels: Subroutine common to all the channels. Path: ..\Mtb\Sub.
Subroutines associated with the part measuring cycle.	Name: Sub_Probe_Piece_Begin.fst Sub_Probe_Piece_End.fst Channels: Subroutine common to all the channels. Path: ..\Mtb\Sub.
Subroutine associated with the displacement of an axis at a dimension.	Name: Sub_ManMove.nc. Channels: Subroutine common to all the channels. Path: ...\Fagor Sub (supplied by Fagor) ..\Mtb\Sub (defined by the OEM).
Subroutine associated with program execution for the 8055 MC/TC.	Name: 9998 / 9999 Channels: Subroutine common to all the channels. Path: General parameter SUBPATH.
Subroutine associated with the tool change (function T).	Name: General parameter TOOLSUB. Channels: Subroutine per channel. Path: ..\Mtb\Sub.
Subroutine associated with the tool change (function M6).	Name: MPROGNAME parameters of the M function table. Channels: Subroutine common to all the channels. Path: ..\Mtb\Sub.
Subroutine associated with function G74.	Name: General parameter REFPSUB. Channels: Subroutine per channel. Path: ..\Mtb\Sub
Subroutines associated with M functions.	Name: MPROGNAME parameters of the M function table. Channels: Subroutine common to all the channels. Path: ..\Mtb\Sub.

Subroutine.	Meaning.
OEM subroutines associated with functions G180 through G189.	Name: General parameter OEMSUB(G180) to OEMSUB(G189). Channels: Subroutine per channel. Path: ..\Mtb\Sub.
OEM subroutines associated with functions G380 through G399.	Name: General parameter OEMSUB(G380) to OEMSUB(G399). Channels: Subroutine per channel. Path: ..\Mtb\Sub.
Interruption subroutines.	Name: General parameter INIT1SUB to INIT4SUB. Channels: Subroutine per channel. Path: ..\Mtb\Sub.
Subroutine associated with the command #INITIALSUB.	Name: General parameter INITIALSUB. Channels: Subroutine per channel. Path: ..\Mtb\Sub.
Subroutine associated with the command #PIERCING.	Name: General parameter PIERCING. Channels: Subroutine per channel. Path: ..\Mtb\Sub.
Subroutine associated with the command #CUTTING ON.	Name: General parameter CUTTINGON. Channels: Subroutine per channel. Path: ..\Mtb\Sub.
Subroutine associated with the command #CUTTING OFF.	Name: General parameter CUTTINGOFF. Channels: Subroutine per channel. Path: ..\Mtb\Sub.
Subroutine associated with the command #FINALSUB.	Name: General parameter FINALSUB. Channels: Subroutine per channel. Path: ..\Mtb\Sub.
User subroutines associated with functions G500 to G599.	Name: Name of the function (G500.nc, G501.nc, etc.) Channels: Subroutine common to all the channels. Path: ..\Users\Sub.
User subroutines associated with functions G8000 to G8999.	Name: Name of the function (G8000.nc, G8001.nc, etc.) Channels: Subroutine common to all the channels. Path: ..\Users\Sub.
Subroutine associated with the command #CAX.	Name: General parameter CAXSUB. Channels: Subroutine per channel. Path: ..\Mtb\Sub.

16.1 Run subroutines from RAM (extension *.fst).

If the same subroutines are executed repeatedly during execution, it is more efficient to load them into the RAM memory of the CNC because this way, they may be accessed faster and execution time may be consequently optimized. This option is valid for OEM as well as for user subroutines. To load a subroutine into RAM, it must have the extension fst. The reserved RAM space for subroutines is 5 Mb.

fst user subroutines.

Subroutines with the fst extension that are not saved in the folder..\MTB\Sub. User routines having a fst extension are loaded into RAM memory during block preparation. The CNC checks whether or not it is loaded into RAM memory; if it is not and there is room for it, it loads it.

At the end of the program (M02/M30) or after a reset, if no other channel is executing subroutines, the CNC deletes them from its RAM memory. This way, if a user subroutine having a fst extension is edited or modified, the CNC assumes the changes the next time it executes it.

OEM (manufacturer's) subroutines.

Subroutines with the fst extension saved in the folder..\MTB\Sub.

- Being the CNC in USER mode, the OEM routines having a fst extension are loaded into RAM memory when starting up the CNC application.

  When the OEM is debugging his subroutines, they must have another extension so the changes are assumed without having to restart the application. Once they are debugged, the OEM must change the extension of the subroutines to fst so they are loaded into RAM memory.
- Being the CNC in SETUP mode, the OEM subroutines with an fst extension will be loaded into RAM memory during the preparation of the blocks. The CNC checks whether or not it is loaded into RAM memory; if it is not and there is room for it, it loads it. At the end of the program (M02/M30) or after a reset, if no other channel is executing subroutines, the CNC deletes them from its RAM memory. This way, the changes made in the subroutine will be taken into account the next time the program is executed.

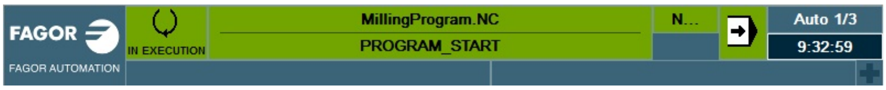
16.2 Subroutine associated with the start.

For each channel, a subroutine can be associated with the execution start, which can be executed by pressing the [START] key, in automatic mode, to start the execution of the entire program; that is if no program start point has been selected. The CNC does not call upon the subroutine either when executing a cycle from the conversational mode. This subroutine may be used, for example, to have certain machining conditions defined to determine the execution of user programs.

If the subroutine exists, the CNC will execute it immediately after pressing the [START] key, before starting the program execution. If that subroutine is missing, the CNC executes the program directly.

Execution of subroutine

During execution, the CNC displays the name of the subroutine on the general status bar. The CNC does not display the blocks under execution, however, it executes the subroutine as a single block, this means that the block-by-block execution is not affected.



16.2.1 Configure the subroutines.

Name and location of the subroutine.

The name of the subroutine must be PROGRAM_START (without an extension) and it will be saved in the folder ..\Users\Sub. If there are several channels, there may be a different subroutine for each channel, whose name must be PROGRAM_START_Cn, where n is the channel number (between 1 and 4).

Name.	Channel.
PROGRAM_START PROGRAM_START_C1	Channel 1. The CNC accepts both names for the subroutine associated with the first channel and where both subroutines exist, the CNC will execute the PROGRAM_START.
PROGRAM_START_C2	Channel 2.
PROGRAM_START_C3	Channel 3.
PROGRAM_START_C4	Channel 4.

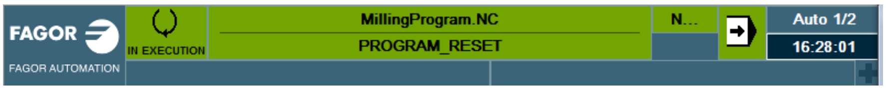
16.3 Subroutine associated with the reset.

For each channel, the reset may have an associated subroutine, which will be executed after pressing the [RESET] key on the operator panel or when the PLC activates the RESETIN mark. This subroutine, for example, establishes any initial conditions that are different from those set by the reset or conditioned to the configuration of the machine, activates operations/modes that disable the reset, etc.

If this subroutine exists, the CNC will execute it immediately after the reset. If this subroutine is missing, the CNC executes the reset directly.

Execution of subroutine

During execution, the CNC displays the name of the subroutine on the general status bar. The CNC does not display the blocks under execution, however, it executes the subroutine as a single block, this means that the block-by-block execution is not affected.



The CNC executes the subroutine with the "Functions G" option of the simulated execution mode.

- The execution takes into account tool radius compensation (functions G41 and G42), hence drawing the tool center path.
- The execution does not send M H S T functions to the PLC.
- The execution does not move the axes of the machine nor starts the spindle.
- The execution takes into account the dwells programmed with G4.
- The execution takes into account the program stops programmed with M00 and M01.

16.3.1 Configure the subroutines.

Name and location of the subroutine.

The name of the subroutine must be PROGRAM_RESET (without an extension) and it will be saved in the folder ..\Users\Sub. If there are several channels, there may be a different subroutine for each channel, whose name must be PROGRAM_RESET_Cn, where n is the channel number (between 1 and 4).

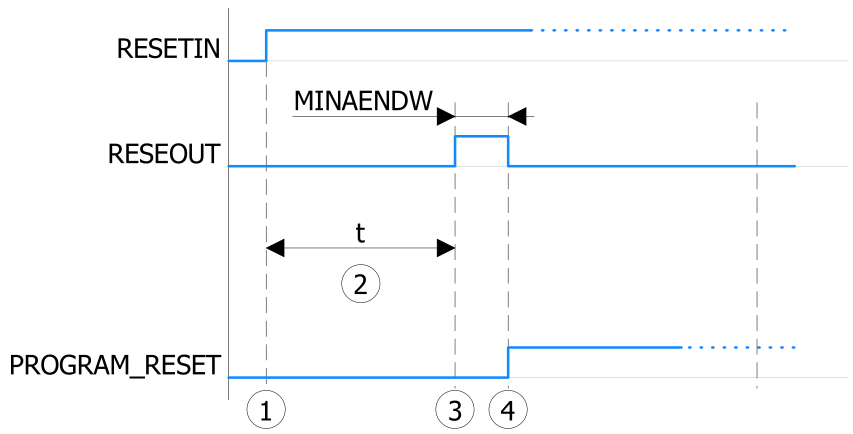
Name.	Channel.
PROGRAM_RESET PROGRAM_RESET_C1	Channel 1. The CNC accepts both names for the subroutine associated with the first channel and where both subroutines exist, the CNC will execute the PROGRAM_RESET.
PROGRAM_RESET_C2	Channel 2.
PROGRAM_RESET_C3	Channel 3.
PROGRAM_RESET_C4	Channel 4.

Executing the reset and the associated subroutine.

Each channel can have a subroutine associated with the reset. The reset process is as follows.

1. The user presses the [RESET] key on the operator panel or the PLC activates the RESETIN (RESETIN=1) mark.
2. The channel assumes the initial conditions (those defined by the machine parameters).
3. The channel activates the RESETOUT (RESETOUT=1) mark to indicate that the reset has finalized. This mark stays active for the time period set in general parameter MINAENDW.
4. The channel executes the subroutine associated with the reset (PROGRAM_RESET), should there be one.

Reset process using RESETIN.



(1) The PLC activates the RESETIN mark.
(2) Time to assume the initial conditions.
(3) The channel activates the RESETOUT mark.
This mark stays active for the time period set in general parameter MINAENDW.
(4) End of reset.
The channel deactivates the RESETOUT mark.
The channel executes the subroutine associated with the reset (PROGRAM_RESET).

The RESETIN mark can be deactivated at any time.

16.4 Subroutine associated with 8055-MC and 8055-TC programs (subroutines 9998 and 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. Each program in the language of the 8055 MC/TC has a call to the corresponding subroutine at the beginning and at the end.

Subroutine.	Meaning.
9998	Routine that the CNC will execute at the beginning of each program of the 8055-MC and 8055-TC.
9999	Routine that the CNC will execute at the end of each program of the 8055-MC and 8055-TC.

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 are missing, the CNC will issue an error message every time when trying to execute a 8055-MC or 8055-TC part-program.

16.4.1 Configure the subroutines.

Name and location of the subroutine.

Parameter.		Meaning.
SUBPATH	CHN	This parameter indicates the directory, by default, for the user subroutines and the necessary subroutines to execute 8055-MC and 8055-TC programs.

[CHN]....... General machine parameter of the channel.

The name of the subroutines is 9998 and 9999. Both subroutines must be saved in the folder path indicated by the general parameter SUBPATH. All channels use the same subroutines.

16.5 Subroutines associated with the kinematics calibration cycle.

The kinematics calibration cycle has two associated subroutines (KinCal_Begin.nc and KinCal_End.nc), which the CNC executes before and after the cycle.

Name.	Meaning.
KinCal_Begin.nc	Subroutine associated with the beginning of the kinematics calibration cycle.
KinCal_End.nc	Subroutine associated with the end of the kinematics calibration cycle.

Fagor supplies both subroutines as incomplete and it is the manufacturer's responsibility to define both subroutines. Software updates do not modify any existing subroutines.

16.5.1 Configure the subroutines.

Name and location of the subroutine.

The names of the subroutines are KinCal_Begin.nc and KinCal_End.nc. Both subroutines must be saved in the folder ..\Mtb\Sub. All channels use the same subroutines.

16.6 Subroutines associated with the tool calibration cycle.

The subroutines supplied by Fagor provide basic probe management, so they should be adjusted and configured appropriately by the OEM. Fagor provides the subroutine Sub_Probe_Tool_Begin.fst associated with probe input 1 and the subroutine Sub_Probe_Piece_Begin.fst associated with probe input 2.

The tool calibration cycle, for both the ISO and the editor, has two associated subroutines that the CNC executes before and after the cycle. These subroutines can be used to enable the probe and indicate if it is in safe mode.

Name.	Meaning.
Sub_Probe_Tool_Begin.fst	Subroutine associated with the beginning of the tool calibration cycle.
Sub_Probe_Tool_End.fst	Subroutine associated with the end of the tool calibration cycle.

Subroutines and M functions defined in the cycle.

Up to 4 M functions may be defined in each probing cycle to be executed before the cycle and another 4 M functions to be executed afterwards. All these functions may have a subroutine associated with them. When starting the execution, the cycle first executes the subroutine Sub_Probe_Tool_Begin.fst and then the M-before functions with their associated subroutines. At the end of the execution, the cycle executes first the M-after functions with their corresponding associated subroutines and then the subroutine Sub_Probe_Tool_End.fst.

Management of the PROBE1ENA/PROBE2ENA marks in the subroutine.

These mark indicate that the probe has is enabled. It is recommended that the OEM manages both marks from the subroutine Sub_Probe_Tool_Begin.fst, so that execution waits until the mark is active.

Management of the PROBE1MONIT/PROBE2MONIT marks in the subroutine.

These marks are associated with the safe mode of the probe. It is recommended that the OEM manages both marks from the subroutine Sub_Probe_Tool_Begin.fst to warn, should they be deactivated, that the probe is in non-safe mode.

16.6.1 Configure the subroutines.

Name and location of the subroutine.

The name of the subroutine is Sub_Probe_Tool_Begin.fst and Sub_Probe_Tool_End.fst. Both subroutines must be saved in the folder ..\Mtb\Sub. All channels use the same subroutines.

Since it is an OEM subroutine with an .fst extension, the CNC loads this subroutine into the RAM of the CNC. See *"16.1 Run subroutines from RAM (extension *.fst)."* on page 683.

16.6.2 Example of a subroutine.

Example of a subroutine Sub_Probe_Tool_Begin.fst supplied by Fagor (to be adjusted by the OEM).

```
#ESBLK
; Activate PROBE 1 Hardware by PLC output.

; Check PROBE is READY with PLC Input from Probe Hardware.
#MSG["WAIT FOR ENABLING PROBE1"]
#WAIT FOR[V.PLC.PROBE1ENA==1]
#MSG[""]

; Select PROBE 1 input for using in Probing.
#SELECT PROBE[1]
#FLUSH
$IF [V.G.PRBST1==1]
  #WARNING["PROBE1 SIGNAL TOUCHING"]
  M0
$ENDIF

; Check probe in safe mode feature is actived.
$IF [V.PLC.PROBE1MONIT==0]
  #MSG["PROBE NOT IN SAFE MODE"]
$ENDIF
#RETDSBLK

#COMMENT BEGIN
PLC signals to add in the logic:

PROBE1ENA; PROBE 1 is enabled confirmation signal to CNC.
If not used in PLC, it is actived by default.

PROBE1MONIT; It activates PROBE1 in SAFE mode.
If probe is actived in no G100/G103 motion, CNC will stop motion and shows an error.
If not used in PLC, it is actived by default.

#COMMENT END
```

Example of a subroutine Sub_Probe_Tool_End.fst supplied by Fagor (to be adjusted by the OEM).

```
#ESBLK
;Deactivate PROBE1 Hardware by PLC output

#RETDSBLK
```

16.7 Subroutines associated with the part measuring cycle.

The subroutines supplied by Fagor provide basic probe management, so they should be adjusted and configured appropriately by the OEM. Fagor provides the subroutine Sub_Probe_Tool_Begin.fst associated with probe input 1 and the subroutine Sub_Probe_Piece_Begin.fst associated with probe input 2.

The part measuring cycle, for both the ISO and the editor, has two associated subroutines that the CNC executes before and after the cycle. These subroutines can be used to enable the probe and indicate if it is in safe mode.

Name.	Meaning.
Sub_Probe_Piece_Begin.fst	Subroutines associated with the start of the part measuring cycle.
Sub_Probe_Piece_End.fst	Subroutines associated with the end of the part measuring cycle.

Subroutines and M functions defined in the cycle.

Up to 4 M functions may be defined in each probing cycle to be executed before the cycle and another 4 M functions to be executed afterwards. All these functions may have a subroutine associated with them. When starting the execution, the cycle executes first the subroutine Sub_Probe_Piece_Begin.fst and then the M-before functions with their associated subroutines. At the end of the execution, the cycle executes first the M-after functions with their corresponding associated subroutines and then the subroutine Sub_Probe_Piece_End.fst.

Management of the PROBE1ENA/PROBE2ENA marks in the subroutine.

These mark indicate that the probe has is enabled. It is recommended that the OEM manages both marks from the subroutine Sub_Probe_Piece_Begin.fst, so that the execution waits until the mark is active.

Management of the PROBE1MONIT/PROBE2MONIT marks in the subroutine.

These marks are associated with the safe mode of the probe. It is recommended that the OEM manages both marks from the subroutine Sub_Probe_Piece_Begin.fst to warn, should they be deactivated, that the probe is in non-safe mode.

16.7.1 Configure the subroutines.

Name and location of the subroutine.

The name of the subroutine is Sub_Probe_Piece_Begin.fst and Sub_Probe_Piece_End.fst. Both subroutines must be saved in the folder ..\Mtb\Sub. All channels use the same subroutines.

Since it is an OEM subroutine with an .fst extension, the CNC loads this subroutine into the RAM of the CNC. See *"16.1 Run subroutines from RAM (extension *.fst)."* on page 683.

16.7.2 Example of a subroutine.

Example of a subroutine Sub_Probe_Piece_Begin.fst supplied by Fagor (to be adjusted by the OEM).

```
#ESBLK
; Activate PROBE 2 Hardware by PLC output.

; Check PROBE is READY with PLC Input from Probe Hardware.
#MSG["WAIT FOR ENABLING PROBE2"]
#WAIT FOR[V.PLC.PROBE2ENA==1]
#MSG[""]

; Select PROBE 2 input for using in Probing.
#SELECT PROBE[2]
#FLUSH
$IF [V.G.PRBST2==1]
  #WARNING["PROBE2 SIGNAL TOUCHING"]
  M0
$ENDIF

; Check probe in safe mode feature is actived.
$IF [V.PLC.PROBE2MONIT==0]
  #MSG["PROBE NOT IN SAFE MODE"]
$ENDIF
#RETDSBLK

#COMMENT BEGIN
PLC signals to add in the logic:

PROBE2ENA; PROBE 2 is enabled confirmation signal to CNC.
If not used in PLC, it is actived by default.

PROBE2MONIT; It activates PROBE2 in SAFE mode.
If probe is actived in no G100/G103 motion, CNC will stop motion and shows an error.
If not used in PLC, it is actived by default.

#COMMENT END
```

Example of a subroutine Sub_Probe_Piece_End.fst supplied by Fagor (to be adjusted by the OEM).

```
#ESBLK
;Deactivate PROBE2 Hardware by PLC output

#RETDSBLK
```

16.8 Subroutine associated with the displacement of an axis at a dimension.

The movement of an axis to a dimension is executed with the subroutine Sub_ManMove.nc. This subroutine, for example, defines how the movements are in G95. When executing a movement from an axis to a dimension, the CNC executes the subroutine defined by the OEM; if it does not exist, it executes the one supplied by Fagor.

16.8.1 Configure the subroutines.

Name and location of the subroutine.

The name of the subroutine is Sub_ManMove.nc. The subroutine can be stored in ..\Fagor\Sub (supplied by Fagor) or ..\Mtb\Sub (defined by the OEM). All channels use the same subroutines.

16.9 Subroutine associated with the tool change (function T).

The subroutine associated with the tool is executed automatically every time a T function (tool selection) is executed. If the M06 function is included in this subroutine, the tool loading process will commence when the "T" code has been executed.

Characteristics of a turret type magazine.

For a turret magazine, it is recommended that the routine associated with T include the M06 function. When the CNC selects a tool, the routine associated with T executes the function M06 (tool change). If the subroutine associated with T does not include the function M06, when the CNC selects a tool, it internally executes a T# M6 block to make the change.

16.9.1 Configure the subroutines.

Assign the subroutine to the command.

Parameter.		Meaning.
TOOLSUB	CHN	Subroutine associated with "T".

[CHN]....... General machine parameter of the channel.

Name and location of the subroutine.

This subroutine must be defined in the general parameter TOOLSUB and saved in the folder ..\Mtb\Sub. If there are several channels, there may be a different subroutine for each channel.

16.10 Subroutine associated with the tool change (function M6).

To properly configure the magazines and the tool change, the routine associated with the tool and the function M06 must be programmed. The M06 function executes the tool change.

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. The M06 function must also be programmed within the subroutine so the CNC prompts the tool manager to start making the tool change.

- When there is no tool magazine, the function M6 behaves like a normal function, as it does not execute anything related to the tool change, therefore, it is not necessary to assign this subroutine.
- For a turret magazine, it is recommended that the routine associated with T include the M06 function. When the CNC selects a tool, the routine associated with T executes the function M06 (tool change). If the subroutine associated with T does not include the function M06, when the CNC selects a tool, it internally executes a T# M6 block to make the change.
- For other magazines, it is recommended that this function in the "M" function table be customized, so it executes the subroutine for the tool changer installed on the machine. The M06 function must also be programmed within the subroutine so the CNC prompts the tool manager to start making the tool change.

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.

16.10.1 Configure the subroutines.

This subroutine must be defined in the “M” function table and saved in the folder ..\Mtb\Sub. See *"16.12 Subroutines associated with M functions."* on page 697.

16.11 Subroutine associated with function G74.

Function G74 (machine home search) can be programmed in 2 ways, either by indicating the axes and the order in which the axes are referenced or by programming only G74 (without axes). When executing a block that only contains the G74 function (without axes), it calls the subroutine associated with this function. This subroutine must contain the axes and the order in which the axes and the spindle must be referenced.

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

16.11.1 Configure the subroutines.

Assign the subroutine to the function.

Parameter.		Meaning.
REFPSUB	CHN	Subroutine associated with function G74.

[CHN]....... General machine parameter of the channel.

Name and location of the subroutine.

This subroutine must be defined in the general parameter REFPSUB (by default, sub_ref.nc) and saved in the folder ..\Mtb\Sub. If there are several channels, there may be a different subroutine for each channel.

16.12 Subroutines associated with M functions.

The "M" functions may have an associated subroutine that the CNC will execute instead of the function. If, within a subroutine associated with an "M" function, the same "M" function is programmed, the CNC will executed this, but not its associated subroutine.

16.12.1 Configure the subroutines.

M FUNCTION TABLE : M VALUES				
MNUM	SYNCHTYPE	MPROGNAME	MTIME	MPLC
12	Without Syncl	sub_M12.nc	500	Yes
15	Before-Before		0	Yes
33	Before-After	sub_M33.nc	100	No
50	After-After		0	No

Assign the subroutine to the function.

Parameter.		Meaning.
MPROGNAME	FUNM	Name of the subroutine associated with the M function.

[FUNM] .... Machine parameters; M function table.

Name and location of the subroutine.

These subroutines must be defined in the “M” function table, in the MPROGNAME field and saved in the folder ..\Mtb\Sub.

16.12.2 Variables.

The following variables may be accessed from (PRG) the part-program and from the MDI/MDA mode, PLC and (INT) from an external application. For each variable, the table indicates whether the access is reading (R) or writing (W).

Variable.	PRG	PLC	INT
(V.)MPM.MPROGNAME[pos] Name of the subroutine associated with the M function. Units: Text.	-	-	R

Syntax.

·pos· Position inside the "M" function table.

`MPM.MPROGNAME[12]`	Position ·12· of the "M" function table.

16.13 OEM subroutines associated with functions G180 to G189 / G380 to G399.

The CNC allows the machine manufacturer to define up to 30 subroutines per channel and associate them with functions G180 through G189 and G380 through G399 in such a way that when a channel executes one of these functions, it will execute the associated subroutine. These OEM subroutines may be executed either in a non-modal or modal way and it also allows resetting the local parameters of the subroutine.

16.13.1 Configure the subroutines.

Assign the subroutines to the functions.

Parameter.		Meaning.
OEMSUB (G180) .. OEMSUB (G189)	CHN	Subroutines associated with functions G180 through G189.
OEMSUB (G380) .. OEMSUB (G399)	CHN	Subroutines associated with functions G380 through G399.

[CHN]....... General machine parameter of the channel.

Name and location of the subroutine.

These subroutines must be defined in the general parameters OEMSUB and saved in the folder ..\Mtb\Sub. These subroutines can be loaded into RAM (extension fst).

16.13.2 Variables.

The following variables may be accessed from (PRG) the part-program and from the MDI/MDA mode, PLC and (INT) from an external application. For each variable, the table indicates whether the access is reading (R) or writing (W).

Variable.	PRG	PLC	INT
(V.)[ch].MPG.OEMSUB1 **..** **(V.)[ch].MPG.OEMSUB10** OEM subroutines associated with functions G180 through G189. Units: -.	-	-	R
(V.)[ch].MPG.OEMSUB11 **..** **(V.)[ch].MPG.OEMSUB30** OEM subroutines associated with functions G380 through G399. Units: -.	-	-	R

16.14 Interruption subroutines.

The interruption subroutines are defined by the machine manufacturer and will be executed from the PLC. When the PLC commands the execution of one of these subroutines, the channel interrupts the execution of the program and executes the corresponding interruption subroutine.

16.14.1 Configure the subroutines.

Assign the subroutines to the functions.

Parameter.		Meaning.
INT1SUB ·· INT4SUB	CHN	Interruption subroutines.

[CHN]....... General machine parameter of the channel.

Name and location of the subroutine.

These subroutines must be defined in the general parameters INIT1SUB to INIT4SU and saved in the folder ..\Mtb\Sub. If there are several channels, there may be a different subroutine for each channel.

16.14.2 Variables.

The following variables may be accessed from (PRG) the part-program and from the MDI/MDA mode, PLC and (INT) from an external application. For each variable, the table indicates whether the access is reading (R) or writing (W).

Variable.	PRG	PLC	INT
(V.)[ch].MPG.INT1SUB **(V.)[ch].MPG.INT2SUB** **(V.)[ch].MPG.INT3SUB** **(V.)[ch].MPG.INT4SUB** Interruption subroutines. Units: Text.	-	-	R

Syntax.

·ch· Channel number.

```
[2].MPG.INT1SUB                    Channel 2.
```

16.15 Subroutine associated with the command #INITIALSUB.

Subroutine specifically for the laser model. The subroutines supplied by Fagor provide basic laser management, so they should be configured appropriately by the OEM. The OEM is responsible for ensuring that the subroutine complies with all safety aspects relating to handling the laser.

The command #INITIALSUB has an associated subroutine, which is executed by the CNC along with the command. This command specifically for laser machines, where this subroutine must perform the initial laser operations.

16.15.1 Configure the subroutines.

If this version is updated, the CNC behaves as follows.

- *For the two first two installation options, the CNC will only copy the subroutines in the folder ..\Mtb\Sub if they do not already exist.*
- *In the third installation option, "rename previous version and install it completely", the CNC always updates the subroutines.*

Assign the subroutine to the command.

Parameter.		Meaning.
INITIALSUB	CHN	Subroutine associated with the command #INITIALSUB.

[CHN]....... General machine parameter of the channel.

Name and location of the subroutine.

This subroutine must be defined in the general parameter INITIALSUB (by default, Initialsub.fst) and saved in the folder ..\Mtb\Sub. If there are several channels, there may be a different subroutine for each channel.

Since it is an OEM subroutine with an .fst extension, the CNC loads this subroutine into the RAM of the CNC. See *"16.1 Run subroutines from RAM (extension *.fst)."* on page 683.

16.15.2 Variables.

The following variables may be accessed from (PRG) the part-program and from the MDI/MDA mode, PLC and (INT) from an external application. For each variable, the table indicates whether the access is reading (R) or writing (W).

Variable.	PRG	PLC	INT
(V.)[ch].MPG.INITIALSUB Subroutine associated with the command #INITIALSUB. Units: Text.	-	-	R

Syntax.

·ch· Channel number.

```
[2].MPG.INITIALSUB                Channel ·2·.
```

16.16 Subroutine associated with the command #PIERCING.

Subroutine specifically for the laser model. The subroutines supplied by Fagor provide basic laser management, so they should be configured appropriately by the OEM. The OEM is responsible for ensuring that the subroutine complies with all safety aspects relating to handling the laser.

The command #PIERCING executes the associated subroutine after updating the piercing variables ((V.) TTPIR.name). This command is specifically for laser machines, where this subroutine must perform the initial piercing operations.

Management of the PIERCING mark in the subroutine.

The OEM must manage the PIERCING mark from this subroutine. At the start of the subroutine (or when the OEM deems it necessary), the OEM must activate the PIERCING mark (assigning it the value 1) to indicate to the PLC that it must start the piercing operation. At the end of the subroutine (or when the OEM deems it necessary), the OEM must deactivate this mark (assigning it the value 0) to indicate to the CNC that it has finished the piercing operation.

- The instruction #PLC makes it possible to manage a PLC mark from the part or subroutine program, without interrupting the block preparation.
- This mark shows the state of the variable V.PLC.PIERCING. The OEM can also use this variable in the subroutine associated with the piercing (by default, Piercing.fst) to modify the status of this mark. Entering this variable interrupts the block preparation.

16.16.1 Configure the subroutines.

If this version is updated, the CNC behaves as follows.

- *For the two first two installation options, the CNC will only copy the subroutines in the folder ..\Mtb\Sub if they do not already exist.*
- *In the third installation option, "rename previous version and install it completely", the CNC always updates the subroutines.*

Assign the subroutine to the command.

Parameter.		Meaning.
PIERCING	CHN	Subroutine associated with the command #PIERCING.

[CHN]....... General machine parameter of the channel.

Name and location of the subroutine.

This subroutine must be defined in the general parameter PIERCING (by default, Piercing.fst) and saved in the folder ..\Mtb\Sub. If there are several channels, there may be a different subroutine for each channel.

Since it is an OEM subroutine with an .fst extension, the CNC loads this subroutine into the RAM of the CNC. See "16.1 Run subroutines from RAM (extension *.fst)." on page 683.

16.16.2 Variables.

The following variables may be accessed from (PRG) the part-program and from the MDI/MDA mode, PLC and (INT) from an external application. For each variable, the table indicates whether the access is reading (R) or writing (W).

Variable.	PRG	PLC	INT
(V.)[ch].MPG.PIERCING Subroutine associated with the command #PIERCING. Units: Text.	-	-	R

Syntax.

·ch· Channel number.

[2].MPG.PIERCING	Channel ·2·.

16.17 Subroutine associated with the command #CUTTING ON and #CUTTING OFF.

Subroutines specifically for the laser model. The subroutines supplied by Fagor provide basic laser management, so they should be configured appropriately by the OEM. The OEM is responsible for ensuring that the subroutine complies with all safety aspects relating to handling the laser.

The command #CUTTINGON executes the associated subroutine after updating the cutting variables ((V.)TTCUT.name). The command #CUTTINGON executes the associated subroutine. Both commands are specifically for laser machines, where these subroutines must perform the initial associated cutting operations.

Management of the CUTTING mark in the subroutine.

The OEM must manage the CUTTING mark from these subroutines. At the start of the associated subroutine #CUTTING ON (or when the OEM deems it necessary), the OEM must activate the CUTTING mark (assigning it a value of 1) to indicate to the PLC that it should start the cutting operation. At the end of the associated subroutine #CUTTING OFF (or when the OEM deems it necessary), the OEM must deactivate this mark (assigning it the value 0) to indicate to the CNC that it has finished the piercing operation.

- The instruction #PLC makes it possible to manage a PLC mark from the part or subroutine program, without interrupting the block preparation.
- This mark shows the state of the variable V.PLC.CUTTING. The OEM can also use this variable in the subroutines associated with cutting (by default, Cuttingon.fst / Cuttingoff.fst) to modify the status of this mark. Entering this variable interrupts the block preparation.

16.17.1 Configure the subroutines.

If this version is updated, the CNC behaves as follows.

- *For the two first two installation options, the CNC will only copy the subroutines in the folder ..\Mtb\Sub if they do not already exist.*
- *In the third installation option, "rename previous version and install it completely", the CNC always updates the subroutines.*

Assign the subroutine to the command.

Parameter.		Meaning.
CUTTINGON	CHN	Subroutine associated with the command #CUTTING ON.
CUTTINGOFF	CHN	Subroutine associated with the command #CUTTING OFF.

[CHN]....... General machine parameter of the channel.

Name and location of the subroutine.

These subroutines associated must be defined for the general parameters CUTTINGON (by default, Cuttingon.fst) and CUTTINGOFF (by default, Cuttingoff.fst). Both subroutines must be saved in the folder ..\Mtb\Sub. If there are several channels, there may be a different subroutine for each channel.

Since it is an OEM subroutine with an .fst extension, the CNC loads this subroutine into the RAM of the CNC. See *"16.1 Run subroutines from RAM (extension *.fst)."* on page 683.

16.17.2 Variables.

The following variables may be accessed from (PRG) the part-program and from the MDI/MDA mode, PLC and (INT) from an external application. For each variable, the table indicates whether the access is reading (R) or writing (W).

Variable.	PRG	PLC	INT
(V.)[ch].MPG.CUTTINGON Subroutine associated with the command #CUTTING ON. Units: Text.	-	-	R
(V.)[ch].MPG.CUTTINGON Subroutine associated with the command #CUTTING OFF. Units: Text.	-	-	R

Syntax.

·ch· Channel number.

[2].MPG.CUTTINGON	Channel ·2·.
[2].MPG.CUTTINGOFF	Channel ·2·.

16.18 Subroutine associated with the command #FINALSUB.

Subroutine specifically for the laser model. The subroutines supplied by Fagor provide basic laser management, so they should be configured appropriately by the OEM. The OEM is responsible for ensuring that the subroutine complies with all safety aspects relating to handling the laser.

The command #FINALSUB has an associated subroutine, which is executed by the CNC along with the command. This command is specifically for laser machines, where this subroutine must perform the final laser operations.

16.18.1 Configure the subroutines.

If this version is updated, the CNC behaves as follows.

- *For the two first two installation options, the CNC will only copy the subroutines in the folder ..\Mtb\Sub if they do not already exist.*
- *In the third installation option, "rename previous version and install it completely", the CNC always updates the subroutines.*

Assign the subroutine to the command.

Parameter.		Meaning.
FINALSUB	CHN	Subroutine associated with the command #FINALSUB.

[CHN]....... General machine parameter of the channel.

Name and location of the subroutine.

This subroutine must be defined in the general parameter FINALSUB (by default, Finalsub.fst) and saved in the folder ..\Mtb\Sub. If there are several channels, there may be a different subroutine for each channel.

Since it is an OEM subroutine with an .fst extension, the CNC loads this subroutine into the RAM of the CNC. See *"16.1 Run subroutines from RAM (extension *.fst)."* on page 683.

16.18.2 Variables.

The following variables may be accessed from (PRG) the part-program and from the MDI/MDA mode, PLC and (INT) from an external application. For each variable, the table indicates whether the access is reading (R) or writing (W).

Variable.	PRG	PLC	INT
(V.)[ch].MPG.FINALSUB Subroutine associated with the command #FINALSUB. Units: Text.	-	-	R

Syntax.

·ch· Channel number.

```
[2].MPG.FINALSUB                 Channel ·2·.
```

16.19 User subroutines associated with functions G500 to G599.

The CNC lets the uses define up to 100 subroutines, common to all channels and that will be associated with functions G500 through G599, so when the CNC executes one of these functions, it will execute its associated subroutine. These OEM subroutines may be executed either in a non-modal or modal way and it also allows resetting the local parameters of the subroutine.

These subroutines are loaded into RAM memory when they are executed for the first time. If there is no room in RAM, the CNC will issue a warning and it will execute the subroutine from the disk. At the end of the program (M30), if no other channel is executing subroutines, the CNC deletes them from its RAM memory. This way, if a user subroutine is edited or modified, the CNC assumes the changes the next time it executes it.

Subroutines supplied by Fagor.

Subroutine.	Meaning.
G500	HSC cancellation.
G501	HSC activation for roughing operations.

16.19.1 Configure the subroutines.

If this version is updated, the CNC behaves as follows.

- *For the two first two installation options, the CNC will only copy the subroutines in the folder ..\Mtb\Sub if they do not already exist.*
- *In the third installation option, "rename previous version and install it completely", the CNC always updates the subroutines.*

Name and location of the subroutine.

The subroutines associated with functions G500 to G599 will have the same name as the function (without an extension) and they will be saved in the folder ..\Users\Sub.

G500	will have subroutine G500 associated with it.
G501	will have subroutine G501 associated with it.
. . .	
G599	will have subroutine G599 associated with it.

16.20 User subroutines associated with functions G8000 to G8999.

The CNC lets the user define up to 1000 subroutines, common to all channels and associated with functions G8000 through G8999, so when the CNC executes one of these functions, it will execute its associated subroutine. These subroutines may be executed either in a non-modal or modal way and it also allows resetting the local parameters of the subroutine.

These subroutines are loaded into RAM memory when they are executed for the first time. If there is no room in RAM, the CNC will issue a warning and it will execute the subroutine from the disk. At the end of the program (M30), if no other channel is executing subroutines, the CNC deletes them from its RAM memory. This way, if a user subroutine is edited or modified, the CNC assumes the changes the next time it executes it.

Some of these subroutines have a function pre-assigned by Fagor, being ProGTL3 language macros. If one of these G functions is programmed, the CNC executes the corresponding macro and not the subroutine.

16.20.1 Configure the subroutines.

If this version is updated, the CNC behaves as follows.

- *For the two first two installation options, the CNC will only copy the subroutines in the folder ..\Mtb\Sub if they do not already exist.*
- *In the third installation option, "rename previous version and install it completely", the CNC always updates the subroutines.*

Name and location of the subroutine.

The subroutines associated with functions G8000 to G8999 will have the same name as the function (without an extension) and they will be saved in the folder ..\Users\Sub.

G8000	The subroutine G8000 will be associated.
G8001	The subroutine G8001 will be associated.
. . .	
G8999	The subroutine G8999 will be associated.

16.21 Subroutine associated with the #CAX statement (spindle as C axis).

The OEM may have associated a subroutine to the #CAX instruction, which the CNC executes together with the command. The #CAX OFF command does not execute the subroutine. Within the subroutine, the #CAX instruction is programmed to activate the spindle as C-axis.

16.21.1 Configure the subroutine.

Assign the subroutines to the functions.

Parameter.		Meaning.
CAXSUB	CHN	Subroutine associated with the #CAX instruction.

[CHN]....... General machine parameter of the channel.

Name and location of the subroutine.

The name of the subroutine shall be defined in the CAXSUB parameter and shall be stored in the ..\Mtb\Sub. folder.

16.21.2 Considerations.

- The subroutine is executed whenever the #CAX statement is programmed, with or without arguments. The #CAX OFF command does not execute the subroutine.

```
#CAX
#CAX [{SpindleName}]
#CAX [{SpindleName}, {CAxisName}]
```

- Within the subroutine, #CAX must be programmed to activate the spindle as the C axis. This instruction is programmed alone in the block, without arguments. The instruction recalls the arguments of the programme's #CAX instruction.
- Within the subroutine is the wildcard @S, which contains the argument {headerName} of the #CAX instruction of the programme. This wildcard is only functional within the subroutine.
- The subroutine allows programming #CAX OFF.
- The following variables allow you to customise the operation of the subroutines for different heads. These variables are only functional within the subroutine.

(V.)G.NLOGPRGCAX	Logical number of the spindle to be activated as C axis, programmed in the #CAX instruction.
(V.)G.NAMEPRGCAX	Name taken by the C axis, programmed in the #CAX instruction; if none is defined, name defined in the CAXNAME parameter.
(V.)G.NLOGCAX	Logical number of the active spindle as C-axis.

Example.

Main program.
%main
#CAX[S1, C3]
(The wildcard @S saves S1)
(The CNC executes the CAXSUB subroutine without executing the actions associated with #CAX)
M30

Subroutine associated with the #CAX instruction.
%cax_sub
(V.G.NLOGPRGCAX -> Logical number of spindle S1).
(V.G.NAMEPRGCAX -> Name assigned to the C-axis in the #CAX instruction; if none defined, name defined in the CAXNAME parameter)
M41.@S
(M41.S1)
(If S1 is not available on the channel, the CNC will give the corresponding error)
#CAX
(The instruction assumes S1 and C3 as call parameters)
(The CNC activates the spindle as C-axis and executes the actions associated with #CAX)
#RET

16.22 Assistance for subroutines.

16.22.1 Subroutine help files.

Help files may be associated with each OEM subroutine (G180, G380, etc), user subroutine (G500, G800, etc) and each global subroutine called upon using #MCALL or #PCALL and they will be displayed while editing.

The help window is displayed while editing, after a blank space or tab following a G function or following the name of the subroutine. The help window is only informative, it cannot be accessed with the cursor nor browse through it. The help window closes with [ESC], deleting the key word or going on to another line of the program.

The help window of the subroutines is only available when the editor uses the CNC language; when the editor is enabled for the 8055 CNC, these helps are not available. The help window of the subroutines is available even when the contextual helps of the editor are disabled.

When the help file is displayed, its text may be inserted into the part-program using the [INS] key.

Edit the help files.

Each subroutine may have two help files; a text file (txt) and an image file (bmp). There's no need to define both files; either one may be defined alone.

When the help window is only informative, it cannot be accessed with the cursor nor browse through it with the page-up-down keys. This is why it is recommended to use short help files; for example, that only contain the description of the parameters of the subroutine. Also, since the text of the help file can be inserted into the program ([INS] key), the following is recommended.

- That the help file contains the line calling the subroutine. Since the user must have written part of the call to display the help window, the editor deletes the call before inserting the help text.
- That all the lines of the help file follow the format of a CNC comment, except the line containing the call to the subroutine.

Help file example for a subroutine.

```
G180 P0= P1= P2= P3= P4= P5=
#COMMENT BEGIN
---------------- G180 ----------------
P1 = Movement in X
P2 = Movement in Y
P3 = Movement in Z
P4 = Feedrate F
P5 = Speed S
--------------------------------------
#COMMENT END
```

Name and location of files.

Name of the help files.

The name of the files must follow the following rule:

Subroutine.	Name of the help files.
G180-G189 G380-G399 G500-G599 G800-G899 G8000-G8999	The name of the files must be the function it is associated with. For example *G180.txt* y *G180.bmp*.
#MCALL #PCALL	The name of the files must be the name of the subroutine. For example *subroutine.txt* and *subroutine.bmp*.

Where to save the help files.

The machine manufacturer can save the help files in the folders ..\Mtb\Sub\Help and ..\Mtb\Sub\Help\{language}. Since the modifications to the MTB directory in the "User" work mode disappear when turning the unit off, the user must save his help files in the folder ..\Users\Sub\Help and ..\Users\Sub\Help\{language}.

The CNC looks for the files in the following order and displays the first that it finds, this is the reason why it is recommended that the user does not define subroutines and/or help files with the same name as those of the OEM. If there are no help files, the CNC will not show any help and it will not display an error.

..\Users\Sub\Help\{idioma}

..\Users\Sub\Help

..\Mtb\Sub\Help\{idioma}

..\Mtb\Sub\Help\

For version V2.60.01, the CNC no longer searches for help files in the following folders.
..\Users\Session\Help\{idioma}
..\Mtb\Sub\Help\{idioma}.
..\Users\Help\{idioma}.

For versions prior to V2.00.01, the CNC first searched for the help files in the manufacturer folders and then in the user folders. For these and later versions, it does so in the opposite order.

16.22.2 List of available subroutines.

The editor allows having in a text file (txt) a list that is displayed while editing the part-program, every time a #PCALL or an #MCALL instruction is edited.

The editor shows the list of subroutines while editing, after a blank space or tab following a G180-G189 or following a #PCALL or #MCALL instruction. This list works the same way as the lists of variables, it is possible to use the arrow keys to scroll through the various elements. Pressing [ENTER], the editor inserts the selected line in the current position of the cursor. The list of subroutines disappears with [ESC], deleting the key word or going on to another line of the program.

This help is always active, regardless of the editing help softkey, "Prog. help".

List of subroutines.

The list of subroutines must be in a text (txt) file. The file must be edited so each line is the name of a possible subroutine to be called.

Example of a file with a list of subroutines.

```
C:\CNC8070\USERS\SUB\FAGOR.NC
SUBROUTINE.NC
EXAMPLE.NC
POSITIONING.NC
```

Name and location of files.

The name of the file should be *pcall.txt*.

Where to save the list of subroutines.

The machine manufacturer must save the *pcall.txt* file in the folder ..\Mtb\Sub\Help. Since the modifications to the MTB directory in the "User" work mode disappear when turning the unit off, the user must save his *pcall.txt* file in the folder ..\Users\Sub\Help.

The CNC looks for the help files in both files; if the files are not there, the CNC will not show any help. If there is a *pcall.txt*file in both directories, the list will show the names of the subroutines contained in both of them.

For versions V2.00.01, the CNC no longer searches for help files in the following folders.
..\Users\Session\Help\{idioma}
..\Mtb\Sub\Help\{idioma}.
..\Users\Help\{idioma}.

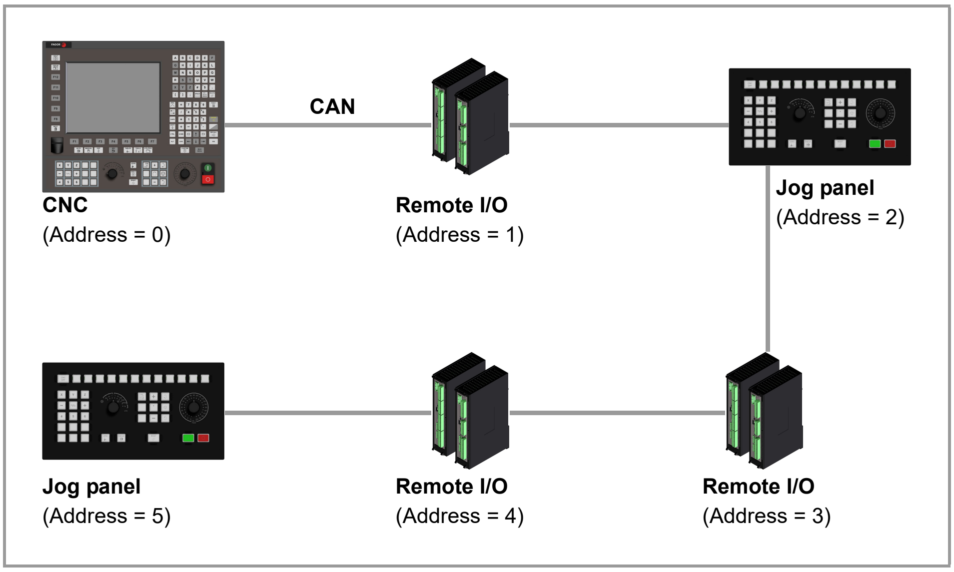
HARDWARE.

17

17.1 Management of several operator panels.

17.1.1 Identify the order of the jog operator panels on the CAN bus.

The CNC numbers the operator panels following the order (sequence) that they occupy in the CAN bus (Address switch). The first operator 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.

17.1.2 Operation of the operator panels.

When enabling a keyboard 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 routine 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 of each operator 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 (parameter GROUPID).
- The keys to select the work mode (automatic, jog, etc.) only affect the active channel.

Behavior of override switches.

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.

17.1.3 Define the number of operator panels and their relationship with the channels.

Since the CNC may have several operator panels connected to it, the CNC and PLC allows configuring the behavior of each one of them with respect to the channels. An operator panel can be disabled, associated with a particular channel (for example, on a TT lathe) or always associated with the same active channel (for example, a machine with two work stations where both operator panels work the same way). When an operator panel is assigned to a channel, it is always operative even when the channel is not the active one. When there are several operator panels assigned to the same channel, operations may be carried out from any of them indistinctively.

Default association of the operator panels with regard to the channels.

Parameter.		Meaning.
NKEYBD	[MPG]	Number of operator panels.
KEYBD1CH ·· KEYBD8CH	[MPG]	Channel to which the operator panel is assigned. Each operator panel may be disabled or assigned to a specific channel. The assignment of operator panels can be modified from the PLC using the marks KEYBD1CH to KEYBD8CH.

[MPG] General machine parameter.

Change the default behavior of the jog keyboards with respect to the channels.

Parameter.		Meaning.
KEYBD1CH ·· KEYBD8CH	[PLC_R] (R/W)	These registers may be used to change the default behavior of the keyboards with respect to the channels, set by machine parameters. KEYBD1CH refers to the first operator panel, KEYBD1CH to the second, and so on. The register can take the following values. 0 : Configuration defined in the machine parameters. 1 : Operator panel assigned to channel 1. 2 : Operator panel assigned to channel 2. 3 : Operator panel assigned to channel 3. 4 : Operator panel assigned to channel 4.
FOCUS1CH ·· FOCUS8CH	[PLC_M] (R)	For systems with several HMIs, these marks indicate which is the active channel for each one. FOCUS1CH corresponds to the first HMI, FOCUS2CH to the second and so on.

[PLC_R] ... PLC register.
[PLC_M] PLC Mark.
(R)............ Consultation signal.
(R/W) Modifiable signal.

- The KEYBDnCH registers can associate an operator panel to a specific channel or restore the configuration defined in the machine parameters.
- An operator panel is disabled if the machine parameter KEYBDnCH = "Keyboard disabled" and the register KEYBDnCH = 0; it is also disabled if the PANELOFFn mark is activated.
- The FOCUSnCH marks, with the KEYBDnCH registers, always assign a keyboard to the active channel of the HMI.

```
() = MOV FOCUSCH1 KEYBD1CH
```
When HMI 1 changes channel, keyboard 1 does the same.

17.1.4 Disable the operator panels.

Parameter.	Meaning.
PANELOFF [PLC_M] PANELOFF1 (R/W) ·· PANELOFF8	If the PLC activates one of these marks, it disables the corresponding CAN operator panel. There is one mark for each operator 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 operator panel. The "address" switch determines the order of the elements in the CAN bus. The first operator panel will be the one with the lowest number and so on.

[PLC_M] PLC Mark.
(R/W) Modifiable signal.

17.1.5 Disable the keyboards.

Parameter.	Meaning.
QWERTYOFF1 [PLC_M] ·· (R/W) QWERTYOFF8	If the PLC activates one of these marks, it disables the corresponding CAN keyboard, including the soft keys. There is a mark for each CAN keyboard. The QWERTYOFF1 mark disables the first one, the QWERTYOFF2 mark the second one and so on. The "address" switch determines the order of the elements in the CAN bus. The first keyboard will be the one with the lowest number and so on The keyboards are not enabled and disabled immediately, especially for the central unit and it may take several seconds.

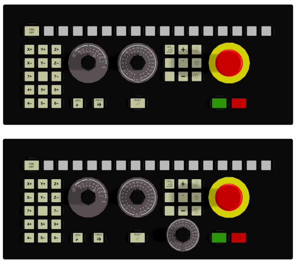
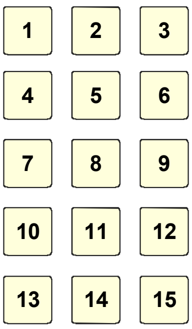
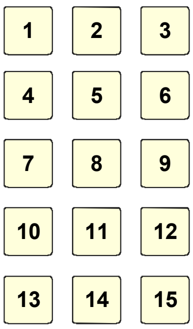
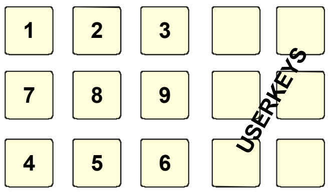
[PLC_M] PLC Mark.
(R/W) Modifiable signal.

17.1.6 Configure the jog keys.

Parameter.	Meaning.
JOGKEYDEF [MPMAN] JOGKEYBD2DEF … JOGKEYBD8DEF	These parameters may be used to configure the jog keypad of each panel. Parameter JOGKEYDEF corresponds to the first operator panel, JOGKEYBD2DEF to the second one and so on.
JOGKEYDEF1 [MPMAN] … JOGKEYDEF15	Axis and moving direction of each jog key. 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.
JOGTYPE [MPMAN]	Behavior of the jog keys, when there are different keys to select the axis and the moving direction. In this case, moving an axis requires both keys to be active. • With the “axis pressed” option, the axis will move while both keys are pressed. • With the “axis maintained” option, the axis key only selects it. The axis will move while the direction key is kept pressed. To de-select the axis, press [ESC] or [STOP].

[MPMAN] . Machine parameter; jog mode.

The relationship between these parameters and the jog keys is the following.

OP-PANEL OP-PANEL+SPDL RATE	OP-PANEL-329	QC-C65□-10K
		
		

The jog keypad may consist of the following types of keys. Both types of keys may be defined at the same keyboard. To define the behavior of each key, assign to them one of the following values.

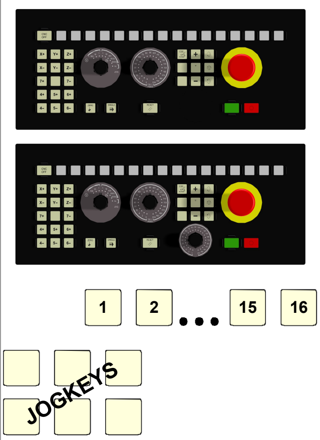
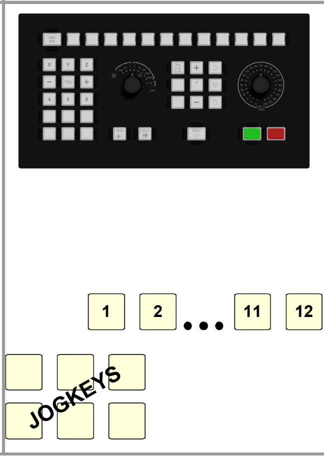
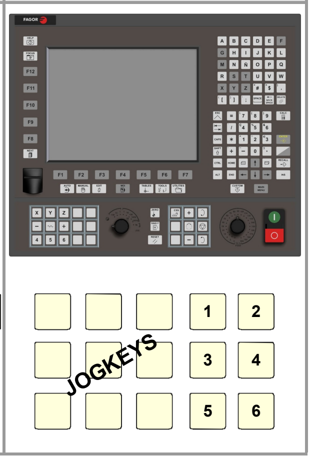
Value.	Meaning.
1-, 2-, 3-, .. 10- 1+, 2+, 3+, .. 10+	Keys to define the axis and the jogging direction. The parameter is set with a value between 1- and 10+ (signed). The sign indicates the positive direction (+) or the negative direction (-) of movement and the number corresponds to the logic axis, according to parameter AXISNAME.
1 .. 10	Keys to define the axis to be jogged. The parameter is set with a value between 1 and 10 (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.
R	Rapid key. The parameter is set with the "R" value.

17.1.7 Configure the user keys as jog keys.

Parameter.	Meaning.
USERKEYDEF [MPMAN] USERKEYBD2DEF ··· USERKEYBD8DEF	These parameters may be used to configure the user keys of each panel as jog keys. Parameter USERKEYDEF corresponds to the first operator panel, USERKEYBD2DEF to the second one and so on.
USERKEYDEF1 [MPMAN] ··· USERKEYDEF15	Axis and moving direction of each user key. 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.
JOGTYPE [MPMAN]	Behavior of the jog keys, when there are different keys to select the axis and the moving direction. In this case, moving an axis requires both keys to be active. • With the "axis pressed" option, the axis will move while both keys are pressed. • With the "axis maintained" option, the axis key only selects it. The axis will move while the direction key is kept pressed. To de-select the axis, press [ESC] or [STOP].

[MPMAN]. Machine parameter; jog mode.

The relationship between these parameters and the jog keys is the following.

OP-PANEL OP-PANEL+SPDL RATE	OP-PANEL-329	QC-C65☐-10K
 	 	

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:

Value.	Meaning.
1-, 2-, 3-, .. 10- 1+, 2+, 3+, .. 10+	Keys to define the axis and the jogging direction. The parameter is set with a value between 1- and 10+ (signed). The sign indicates the positive direction (+) or the negative direction (-) of movement and the number corresponds to the logic axis, according to parameter AXISNAME.
1 .. 10	Keys to define the axis to be jogged. The parameter is set with a value between 1 and 10 (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.
R	Rapid key. The parameter is set with the "R" value.

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.

17.1.8 Check the status of the keys.

Parameter.		Meaning.
KEYBD1 / KEYBD3	[PLC_R] (R)	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 ·· KEYBD1_8 / KEYBD2_8	[PLC_R] (R)	These registers indicate (bit=1) which key has been pressed on each operator panel. Registers KEYBD1_1 to KEYBD3_1 correspond to the first operator panel, KEYBD1_2 to KEYBD3_2 to the second one and so on.

[PLC_R] ... PLC register. *(R) .... Consultation signal.*

17.1.9 Disable the keys.

Parameter.		Meaning.
KEYDIS1 / KEYDIS3	[PLC_R] (R/W)	These registers inhibit (bit=1) the keys and the switches on all operator panels at the same time.
KEYDIS1_1 / KEYDIS3_1 ·· KEYDIS1_8 / KEYDIS3_8	[PLC_R] (R/W)	These registers inhibit (bit=1) the keys and the switches on each operator panel. Registers KEYDIS1_1 to KEYDIS4_1 correspond to the first operator panel, KEYDIS1_2 to KEYDIS4_2 to the second one and so on.

[PLC_R] ... PLC register. *(R) .... Modifiable signal.*

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
```

17.1.10 Manage the LED's (lamps) of the keys.

Parameter.		Meaning.
KEYLED1 / KEYLED2	[PLC_R] (R/W)	These registers turn on all the operator panel key LEDs (bit=1) at the same time.
KEYLED1_1 / KEYLED2_1 ·· KEYLED1_8 / KEYLED2_8	[PLC_R] (R/W)	These registers turn on all the control panel key LEDs (bit=1). Registers KEYLED1_1 and KEYLED2_1 correspond to the first operator panel, KEYLED1_2 and KEYLED2_2 to the second one and so on.

[PLC_R] ... PLC register. *(R) .... Modifiable signal.*

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
```

17.2 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=
```

17.3 Hardware. Managing local digital I/Os.

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.

17.3.1 Configuring local digital outputs (XTX platform).

Parameter.		Meaning.
NLOCOUT	MPG	Number of local digital outputs. For laser applications (PWM and synchronized switching), the value of this parameter must set to 8.

[MPG]....... Machine parameters; general.

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 the 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.

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

Parameter.		Meaning.
EXPSCHK	MPG	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.

[MPG]....... Machine parameters; general.

17.3.3 Variables.

The following variables may be accessed from (PRG) the part-program and from the MDI/MDA mode, PLC and (INT) from an external application. For each variable, the table indicates whether the access is reading (R) or writing (W). PLC access to the variables, for both reading and writing, will be synchronous. Access to the variables from the part program returns the value for the block preparation (it does not stop the preparation).

Variable.	PRG	PLC	INT
(V.)MPG.NLOCOUT Number of local digital outputs. Units: -.	R	R	R
(V.)MPG.EXPSCHK Check the 24 V power supply at the local digital outputs. Units: -.	R	R	R

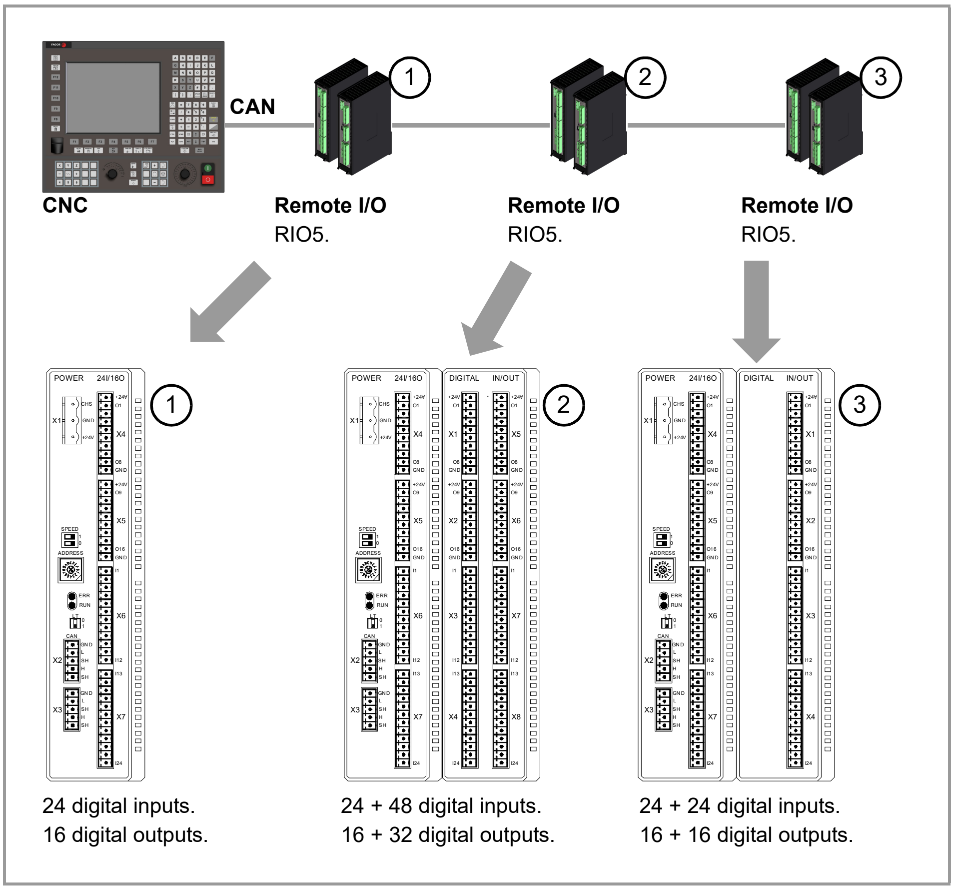
Syntax of the variables (from the part-program or MDI/MDA mode).

```
V.MPG.NLOCOUT
V.MPG.EXPSCHK
```

17.4 Numbering the digital inputs and outputs of the CANopen bus. Default numbering (according to the order of the remote groups).

By default, the CNC numbers the digital inputs and outputs according to the logical order of the remote groups (Address switch). The first module will be the one with the lowest number, and so on. Within each group, the order of the inputs and outputs is from top to bottom and from left to right. If the bus configuration is changed (adding or removing modules), the CNC will renumber all I/Os from the new module.

17.4.1 Example. RIO5 modules.



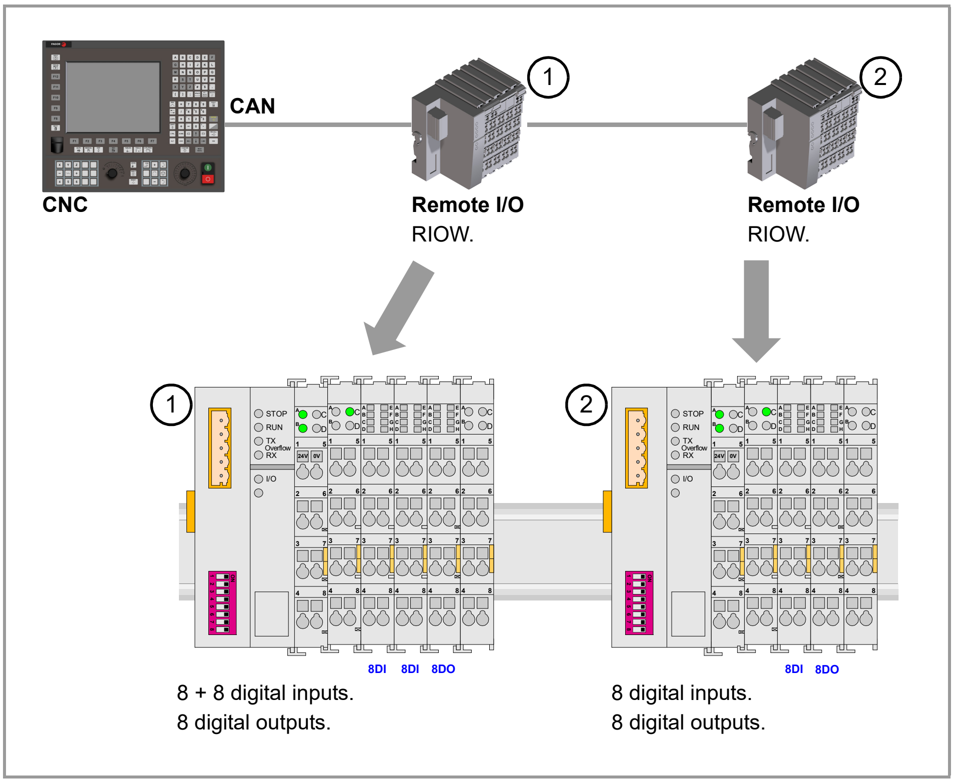
Example 1.

	Group ·1· (address = 1)	**Group ·2·** (address = 2)	**Group ·3·** (address = 3)
Digital inputs.	1 ·· 24	25 ·· 48 49 ·· 96	97 ·· 120 121 ·· 144
Digital outputs.	1 ·· 16	17 ·· 32 33 ·· 64	65 ·· 80 81 ·· 96

Example 2.

	Group ·1· (address = 1)	**Group ·2·** (address = 3)	**Group ·3·** (address = 2)
Digital inputs.	1 ·· 24	73 ·· 96 97 ·· 144	25 ·· 48 49 ·· 72
Digital outputs.	1 ·· 16	49 ·· 64 65 ·· 96	17 ·· 32 33 ·· 48

17.4.2 Example. RIOW modules.



Example 1.

	Group ·1· (address = 1)	**Group ·2·** (address = 2)
Digital inputs.	1 ·· 8 9 ·· 16	17 ·· 24
Digital outputs.	1 ·· 8	9 ·· 16

Example 2.

	Group ·1· (address = 2)	**Group ·2·** (address = 1)
Digital inputs.	9 ·· 16 17 ·· 24	1 ·· 8
Digital outputs.	9 ·· 16	1 ·· 8

17.5 Numbering the digital inputs and outputs of the CANopen bus. Base index.

Available up to version v2.00.

The CNC can customize the numbering of the I/Os for each module, assigning a base index to the first one. The CNC sequentially numbers the other I/Os of the module, from top to bottom and from left to right. This method allows the bus configuration to be changed (adding or removing modules), maintaining the numbering of the other I/Os.

Configuration requirements.

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.

When inserting a new module, the CNC will assign the numbering of the table to the first modules, and the last will be assigned the next valid base index to the highest one assigned until then.

- In the RIO5 series, each single module (including the leader module) counts as one.
- In the RIO5 series, each double module (module with two boards) it counts as two. This module must be assigned two base indexes for the inputs and two base indexes for the outputs; one for each board.
- In the RIOR, RIOW series and third-party modules, each node counts as a single module.

Configure the digital inputs.

The digital inputs and outputs are configured via machine parameters.

Parameter.		Meaning.
NDIMOD	MPG	Number of digital input modules. If this parameter is set to 0 (zero), the CNC sequentially numbers the digital inputs according to the order in which the modules are located in the bus.
DIMODADDR	MPG	Base address table for the digital input modules.
DIMOD n	MPG	Base address of the digital input module.

[MPG]....... Machine parameters; general.

Configure the digital outputs.

The digital inputs and outputs are configured via machine parameters.

Parameter.		Meaning.
NDOMOD	MPG	Number of digital output modules. If this parameter is set to 0 (zero), the CNC sequentially numbers the digital outputs according to the order in which the modules are located in the bus.
DOMODADDR	MPG	Base address table for the digital output modules
DOMOD n	MPG	Base address of the digital output module.

[MPG]....... Machine parameters; general.

Variables.

The following variables may be accessed from (PRG) the part-program and from the MDI/MDA mode, PLC and (INT) from an external application. For each variable, the table indicates whether the access is reading (R) or writing (W). PLC access to the variables, for both reading and writing, will be synchronous. Access to the variables from the part program returns the value for the block preparation (it does not stop the preparation).

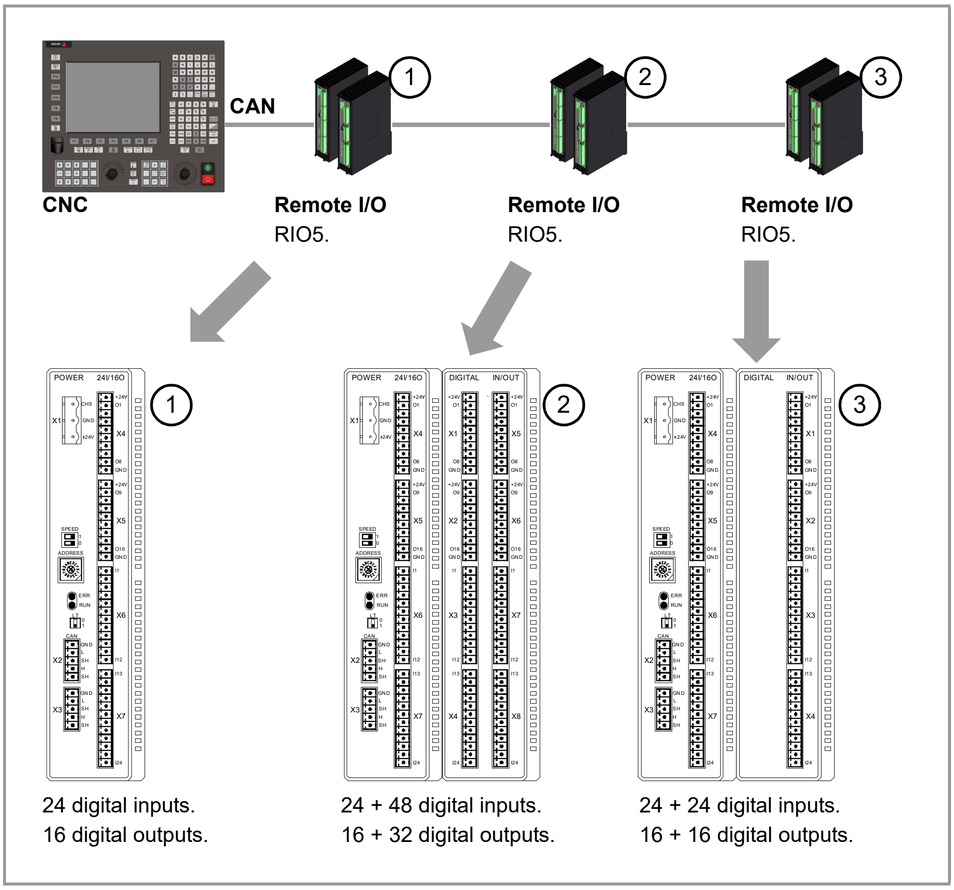
Variable.	PRG	PLC	INT
(V.)MPG.NDIMOD Number of digital input modules. Units: -.	R	R	R
(V.)MPG.DIMODADDR[nb] Base address of the digital input module. Units: -.	R	R	R
(V.)MPG.NDOMOD Number of digital output modules. Units: -.	R	R	R
(V.)MPG.DOMODADDR[nb] Base address of the digital output module. Units: -.	R	R	R

Syntax of the variables (from the part-program or MDI/MDA mode).

·nb· Module number.

V.MPG.NDIMOD	
V.MPG.DIMODADDR[4]	Fourth digital input module.
V.MPG.NDOMOD	
V.MPG.DOMODADDR[4]	Fourth digital output module.

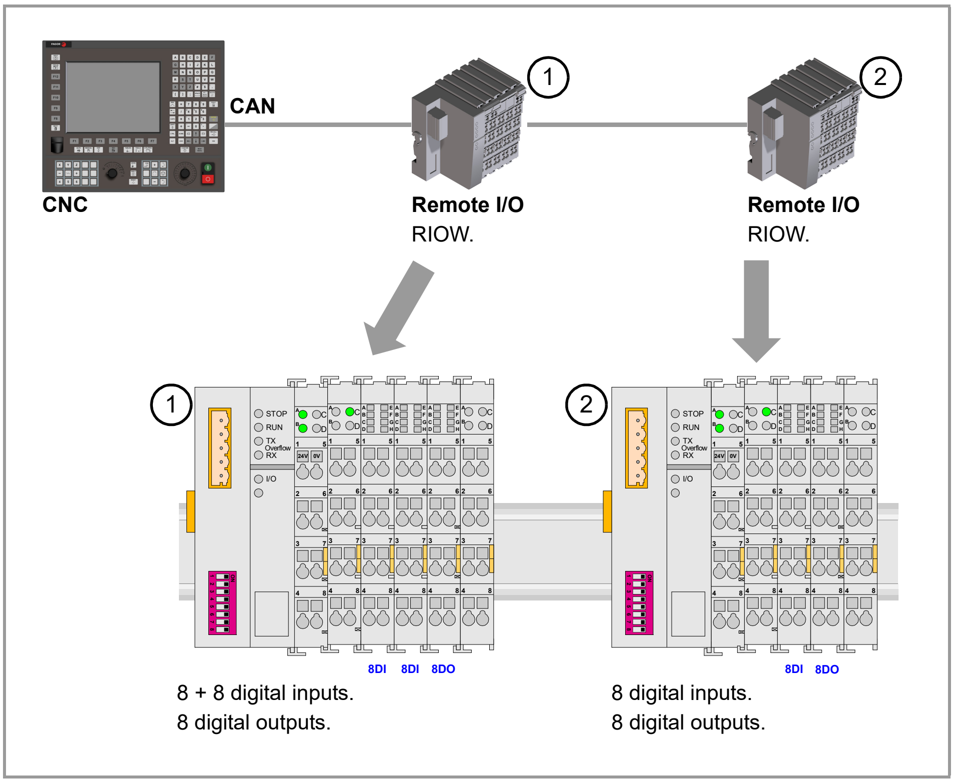
17.5.1 Example. RIO5 modules.



	Group ·1· (address = 1)	**Group ·2·** (address = 2)	**Group ·3·** (address = 3)
Digital inputs.	**Base index*** ·1· (I1 of X6) **Numbering.** 1 ·· 24	**Base index*** ·25· (I1 of X6) ·49· (I1 of X3) ·73· (I1 of X7) **Numbering.** 25 ·· 48 49 ·· 72 73 ·· 96	**Base index*** ·97· (I1 of X6) ·121· (I1 of X3) **Numbering.** 97 ·· 120 121 ·· 144
Digital outputs.	**Base index*** ·1· (I1 of X6) **Numbering.** 1 ·· 16	**Base index*** ·17· (I1 of X6) ·33· (I1 of X6) ·49· (I1 of X6) **Numbering.** 17 ·· 32 33 ·· 48 49 ·· 64	**Base index*** ·65· (I1 of X6) ·97· (I1 of X6) **Numbering.** 80 ·· 96 97 ·· 112

(*) For each base index, indicate the connector and the input or output it corresponds to.

17.5.2 Example. RIOW modules.



	Group ·1· (address = 1)	**Group ·2·** (address = 2)
Digital inputs.	**Base index*** ·1· (I1) ·9· (I1) **Numbering.** 1 ·· 8 9 ·· 16	**Base index*** ·17· (I1) **Numbering.** 17 ·· 24
Digital outputs.	**Base index*** ·1· (O1) **Numbering.** 1 ·· 8	**Base index*** ·17· (O1) **Numbering.** 9 ·· 16

(*) For each base index, the corresponding input or output is indicated.

17.6 Numbering the digital inputs and outputs of the CANopen bus. Logic blocks.

The CNC allows the digital I/O nodes in logic blocks (virtual modules) to be divided, and then to establish the numbering of the I/Os of each logic block, assigning a base index to the first one.

- In RIO5 and RIOR nodes, each module is a logic block, several modules cannot be grouped in a block or divide a module into blocks. The logic blocks must have the same number and size as the modules physically detected in the bus.
- In the RIO5 nodes, each double module (module with two boards) it counts as two, in other words, two logic blocks.
- In RIOW nodes and third-party nodes, it is possible to display as many logic blocks as desired, both in number and in size.

Configuration requirements.

The configuration must meet the following requirements. If the parameters are not set according to these requirements, the CNC will ignore it and it will number the digital inputs sequentially according to the order of the modules in the bus.

- The parameters of all I/O nodes of the bus must be set; no node can be left undefined.
- The parameters of all the resources of a node must be set; no I/O can be left without a number.
- The maximum number of logic blocks in the node will be 64, both for digital inputs and digital outputs.
- The number of I/Os in a logic block must be other than zero and a multiple of 8 (8, 16, 24, 32, ...1024).
- The values of the base index must be comply with the formula "8n + 1" (i.e. 1, 9, 17, 25...).
- The blocks may be distributed at will within the range of resources managed by PLC (1...1024). The PLC admits blanks, but the blocks cannot overlap; in other words, a PLC digital input or output can only belong to a single node-module of the bus.

Configure the digital inputs.

The digital inputs and outputs are configured via machine parameters.

Parameter.		Meaning.
NDIMOD	MPG	Total number of digital input logic blocks. Possible values: From 0 to 42 (by default; 0).
DIMODADDR	MPG	Base addresses of the digital input logic blocks.
MNEMONIC	MPG	Mnemonic of the logic block.
NODE	MPG	Node number of the header to which the logic block belongs. Possible values: From 1 to 126 (by default; 0).
BLOCK	MPG	Number of the logic block. Possible values: From 1 to 64 (by default; 0).
ADDRESS	MPG	Base address of the logic block. Possible values: From 1 to 1017 (by default, the first valid value).
NDI	MPG	Number of digital inputs of the module. Possible values: From 8 to 1024 (by default; 0).

[MPG]....... Machine parameters; general.

Configure the digital outputs.

The digital inputs and outputs are configured via machine parameters.

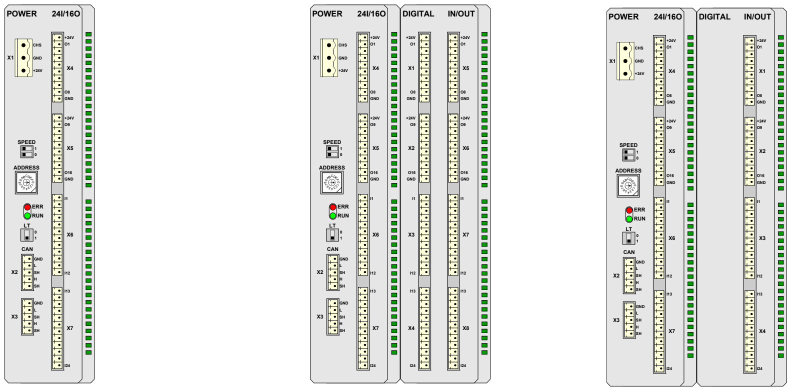
Parameter.		Meaning.
NDOMOD	MPG	Total number of digital output logic blocks. Possible values: From 0 to 42 (by default; 0).
DOMODADDR	MPG	Base addresses of the digital output logic blocks.
MNEMONIC	MPG	Mnemonic of the logic block.
NODE	MPG	Node number of the header to which the logic block belongs. Possible values: From 1 to 125 (by default; 0).
BLOCK	MPG	Number of the logic block. Possible values: From 1 to 64 (by default; 0).
ADDRESS	MPG	Base address of the logic block. Possible values: From 1 to 1017 (by default, the first valid value).
NDO	MPG	Number of digital outputs of the module. Possible values: From 8 to 1024 (by default; 0).

[MPG] ...... Machine parameters; general.

17.6.1 Example. RIO5 modules.

A CANopen bus offers the following groups.

- Node 1: Remote RIO5 group with 1 module of 24 digital inputs.
- Node 2: Remote RIO5 group with 3 module of 24 digital inputs each.
- Node 3: Remote RIO5 group with 2 module of 24 digital inputs each.



Group ·1·	Group ·2·	Group ·3·
24 digital inputs.	24 + 48 digital inputs.	24 + 24 digital inputs.
16 digital outputs.	16 + 32 digital outputs.	16 + 16 digital outputs.

In the RIO5 module must be configured as one logic block. Each RIO5 node is to be configured as follows.

- The first RIO5 node as a block of 24 inputs from I1 on.
- The second RIO5 node as three blocks of 24 inputs from I49, I73 and I97 on.
- The third RIO5 node as two blocks of 24 inputs from I257 and I129 on.

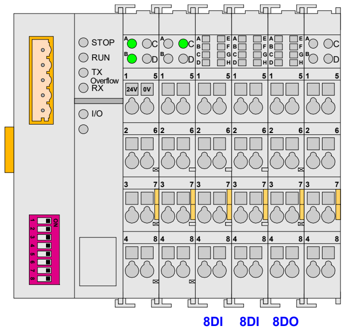
Parameter.	Value.	Comment.
NDIMOD	6	In the RIO5 series, each module must be configured as one logic block. Each double module (module with two boards) it counts as two blocks.
DIMODADDR	- - -	See the table below.

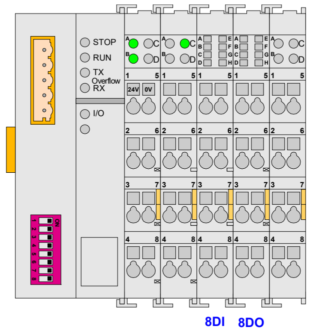
DIMODADDR				
MNEMONIC	NODE	BLOCK	ADDRESS	NDI
DIMOD 1	1 (RIO5)	1	1	24
DIMOD 2	2 (RIO5)	1	49	24
DIMOD 3	2 (RIO5)	2	73	24
DIMOD 4	2 (RIO5)	3	97	24
DIMOD 5	3 (RIO5)	1	257	24
DIMOD 6	3 (RIO5)	2	129	24

17.6.2 Example. RIOW modules.

A CANopen bus offers the following groups.

- Node 1: Remote RIOW group with 16 digital inputs and 8 digital outputs.
- Node 2: Remote RIOW group with 8 digital inputs and 8 digital outputs.





Node 1 (RIOW)
16 digital inputs.
8 digital outputs

Node 2 (RIOW)
8 digital inputs.
8 digital outputs

Each RIOW node is to be configured as follows.

- The first RIOW node as 2 blocks, one 8 inputs from I1 on and the other from I33 on. Also a block of 8 digital outputs from O81 on.
- The second RIOW node as a block of 8 inputs from I113 on. Also a block of 8 digital outputs from O49 on.

Parameter.	Value.	Comment.
NDIMOD	3	3 blocks; 2 for the first node and 1 for the second one.
DIMODADDR	- - -	See the table below.

DIMODADDR				
MNEMONIC	NODE	BLOCK	ADDRESS	NDI
DIMOD 1	1 (RIOW)	1	1	8
DIMOD 2	1 (RIOW)	2	33	8
DIMOD 3	2 (RIOW)	1	113	8

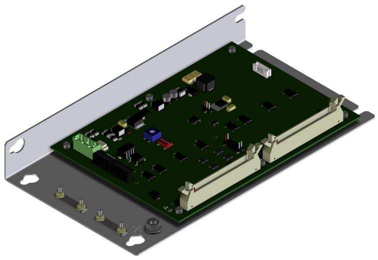
Parameter.	Value.	Comment.
NDOMOD	2	2 blocks, one per each RIOW node.
DOMODADDR	- - -	See the table below.

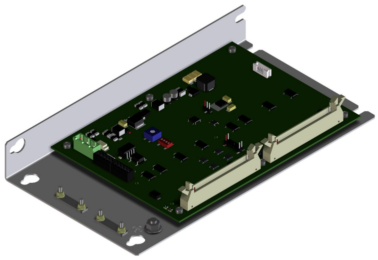
DOMODADDR				
MNEMONIC	NODE	BLOCK	ADDRESS	NDO
DOMOD 1	1 (RIOW)	1	81	8
DOMOD 2	2 (RIOW)	1	49	8

17.6.3 Example. RIOR modules.

A CANopen bus offers the following groups.

- Node 1: Remote RIOR group with 48 digital inputs and 32 digital outputs.
- Node 2: Remote RIOR group with 48 digital inputs and 32 digital outputs.





Node 1 (RIOR)
48 digital inputs.
32 digital outputs

Node 2 (RIOR)
48 digital inputs.
32 digital outputs

In the RIOR module must be configured as one logic block. Each RIOR node is to be configured as follows.

- The first RIOR node as a block of 48 inputs from I1 on. Also a block of 32 digital outputs from O1 on.
- The second RIOR node as three blocks of 48 inputs from I97 on. Also a block of 32 digital outputs from O81 on.

Parameter.	Value.	Comment.
NDIMOD	2	2 blocks, one per each RIOR node. In the RIOR series, each module must be configured as one logic block.
DIMODADDR	- - -	See the table below.

DIMODADDR				
MNEMONIC	**NODE**	**BLOCK**	**ADDRESS**	**NDI**
DIMOD 1	1 (RIOR)	1	1	48
DIMOD 2	2 (RIOR)	1	97	48

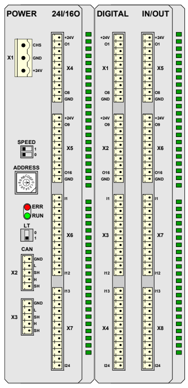
Parameter.	Value.	Comment.
NDOMOD	2	2 blocks, one per each RIOR node. In the RIOR series, each module must be configured as one logic block.
DOMODADDR	- - -	See the table below.

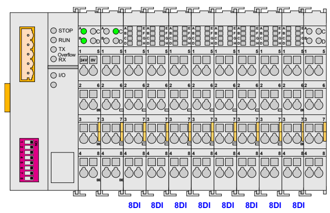
DOMODADDR				
MNEMONIC	**NODE**	**BLOCK**	**ADDRESS**	**NDO**
DOMOD 1	1 (RIOR)	1	1	32
DOMOD 2	2 (RIOR)	1	81	32

17.6.4 Example. RIO5 and RIOW modules.

A CANopen bus offers the following groups.

- Node 1: Remote RIO5 group with 3 module of 24 digital inputs each.
- Node 2: Remote RIOW group with 64 digital inputs.





Node 1 (RIO5)
24 + 48 digital inputs.

Node 2 (RIOW)
64 digital inputs.

The first RIOW node is to be configured in 2 blocks, one 24 inputs from I104 on and the other from I200 on.

Parameter.	Value.	Comment.
NDIMOD	5	2 blocks for the RIOW and 3 blocks for the RIO5. In the RIO5 series, each module must be configured as one logic block. Each double module (module with two boards) it counts as two blocks.
DIMODADDR	- - -	See the table below.

DIMODADDR				
MNEMONIC	NODE	BLOCK	ADDRESS	NDI
DIMOD 1	1 (RIO5)	1	1	24
DIMOD 2	1 (RIO5)	2	25	24
DIMOD 3	1 (RIO5)	3	49	24
DIMOD 4	2 (RIOW)	1	104	24
DIMOD 5	2 (RIOW)	2	200	40

17.7 Configuring and numbering the PT100 inputs.

To use the PT100 sensors, the CNC must activate the input to which it is connected. These inputs are deactivated by default and must only be activated if they have a sensor connected to them. Likewise, if the input is active in the machine parameter and there is no sensor connected, the CNC returns the corresponding error.

Configure the PT100 inputs.

Parameter.		Meaning.
NPT100	MPG	Total number of active PT100 inputs.
PT100	MPG	Analog input table associated with the PT100 inputs.
PT100 n	MPG	Analog input associated with the PT100 probe. Each parameter can have any valid value (from 0-32); it is not necessary to follow any particular order. If set to 0 (zero), PT100 input is not activated.

[MPG]....... Machine parameters; general.

Customizing the numbering of analog inputs for PT100 probes.

When numbering the analog inputs, the CNC considers the PT100 inputs as analog inputs. This way, for numbering, the CNC considers the following:

- The RIO5 modules have 6 analog inputs; four analog inputs (AI1..AI4) plus two PT100 inputs (AI5..AI6).
- The RIOR modules have 4 analog inputs; the two analog inputs (AI1..AI2) and the two PT100 inputs (AI3..AI4).

Example.
In a system with an RIO5 module (node 1) and an RIOR (node 2), the CNC identifies the analog inputs as follows.

- (address = 0) CNC
  (0) Analog inputs. none
  (0) PT100 inputs none
- (address = 1) RIO5
  (1) Analog inputs. . . . . 1..4
  (1) PT100 inputs.. . . . . 5..6
- (address = 2) RIOR
  (2) Analog inputs. . . . . 7..8
  (2) PT100 inputs.. . . . . 9..10

To have 3 PT100 inputs active (for example, the two of the first module and the first one of the second module), the PT100 parameters may be set as follows.

NPT100 = 3
PT100 1 = 5
PT100 2 = 6
PT100 3 = 9

Enable/disable a PT100 sensor from the PLC.

The PT100 inputs are activated via the machine parameters. In order to be able to temporarily disable the sensor (for example, during a spindle change), the PLC has the following marks.

PLC signal.		Meaning.
PT100OFF1 .. PT100OFF20	PLC_M	Disable the PT100 probe temporarily. If the PLC activates one of these marks, the CNC disables the corresponding sensor. The PT100OFF1 mark corresponds to the probe indicated in parameter PT100 1 and PT100OFF2 to the probe indicated in parameter PT100, 2 and so on.

[PLC_M] PLC Mark.

17.7.1 CNC variables.

The following variables may be accessed from (PRG) the part-program and from the MDI/MDA mode, PLC and (INT) from an external application. For each variable, the table indicates whether the access is reading (R) or writing (W). PLC access to the variables, for both reading and writing, will be synchronous. Access to the variables from the part program returns the value for the block preparation (it does not stop the preparation), except when indicated otherwise.

Variable.	PRG	PLC	INT
(V.)MPG.NPT100 Total number of active PT100 inputs. Units: -.	R	R	R
(V.)MPG.PT100[nb_pt] Analog input associated with the PT100 probe [nb_pt]. Units: -.	R	R	R
(V.)G.ANAI[nb_ai] Voltage or temperature of the analog input [nb_ai]. Units: Volts / tenths of a degree.	R(*)	R	R

(*) The CNC evaluates the variable during execution (it stops the block preparation).

Syntax of the variables (from the part-program or MDI/MDA mode).

·nb_pt· Number of PT100 probe.

·nb_ai· Number of the analog input.

```
V.MPG.NPT100
V.MPG.PT100[4]        Analog input associated with the fourth PT100 probe.
V.G.ANAI[3]           PT100 probe temperature connected to analog input 3.
```

17.8 Configuring the CANopen bus.

Configuring the CANopen bus.

Parameter.	Meaning.
CANOPENFREQ MPG	CANopen bus communication frequency.

[MPG]....... Machine parameters; general.

CANopen bus communication frequency.

When using the CANopen protocol, the transmission speed at the bus is defined in each node and they all must run at the same speed. The transmission speed depends on the total length of the bus. Using the following illustrative values; assigning other values may cause communication errors due to signal distortion.

Speed.	Length of the CAN bus.
Autoscan.	The CNC adjusts the bus frequency at every startup depending on the speed of the rest of the modules. Depending on the configuration of the bus, this option can make the CNC startup slower than if there is a set frequency.
1000 kbps.	Up to 20 meters.
800 kbps.	From 20 to 40 meters.
500 kbps.	From 40 to 100 meters.
250 kbps.	From 100 to 500 meters.

i

The speed of 250 kHz is only available to communicate with the keyboards and RIOW and RIOR series remote modules; this speed is not available for the RIO5 series remote modules.

17.8.1 Variables.

The following variables may be accessed from (PRG) the part-program and from the MDI/MDA mode, PLC and (INT) from an external application. For each variable, the table indicates whether the access is reading (R) or writing (W). PLC access to the variables, for both reading and writing, will be synchronous. Access to the variables from the part program returns the value for the block preparation (it does not stop the preparation).

Variable.	PRG	PLC	INT
(V.)MPG.CANOPENFREQ CANopen bus communication frequency. Units: -.	R	R	R

Syntax of the variables (from the part-program or MDI/MDA mode).

```
V.MPG.CANOPENFREQ
```

17.9 Configuring the serial line.

Configuring the serial line.

Parameter.		Meaning.
RSTYPE	MPG	Type of serial line. Standard configuration for RS232 and full-duplex configuration for RS422.

[MPG] ...... Machine parameters; general.

i

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.

17.9.1 Variables.

The following variables may be accessed from (PRG) the part-program and from the MDI/MDA mode, PLC and (INT) from an external application. For each variable, the table indicates whether the access is reading (R) or writing (W). PLC access to the variables, for both reading and writing, will be synchronous. Access to the variables from the part program returns the value for the block preparation (it does not stop the preparation).

Variable.	PRG	PLC	INT
(V.)MPG.RSTYPE Type of serial line. Units: -.	R	R	R

Syntax of the variables (from the part-program or MDI/MDA mode).

```
V.MPG.RSTYPE
```

LASER.

18

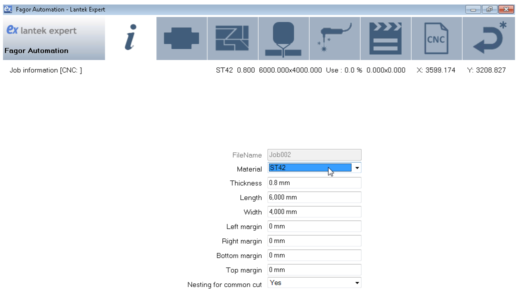
18.1 “Lantek Expert Inside” application. Manual and automatic nesting.

18.1.1 About the application.

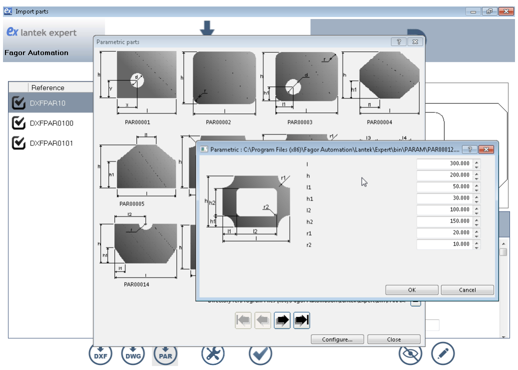
Fagor Automation and Lantek have collaborated on the Lantek Expert Inside software integration for the CNC 8060 and 8070 models. From figures drawn in dxf, dwg and parametric files, this software allows these to be distributed on a sheet to optimize their use, define the input/output strategy (lead-in/lead-out), etc., to ultimately create a part-program in Fagor language. The software can be installed either on the CNC or on a PanelPC.

The simple interface, suitable for touch screens, provides a review of the cut in real time and any machine-level modifications. It also helps with tasks such as individual piece repetitions or the reuse of sheet material without the need for intermediate procedures.

Fagor provides two types of nesting, which are available via software options; manual nesting (the user places the pieces on the sheet as required) and automatic nesting (the software places the pieces on the sheet based on their configuration parameters). For both, their simple operation can be summarized in the following steps.

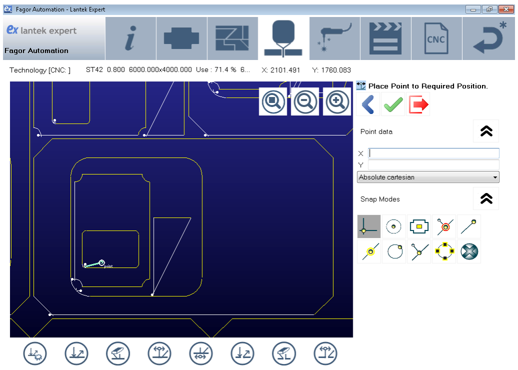


Defining general information for the nesting jobs. This data is the minimum information required to create a job: material, thickness, length and width of the sheet, left/right and top/bottom margins.



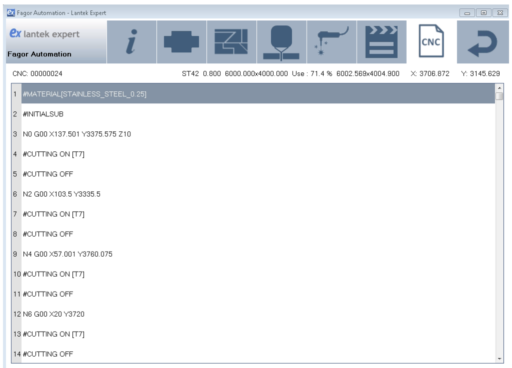
Use default parts or import pieces from external files (dxf, dwg or parametric parts). It is also possible to open an existing nesting job made in Lantek Expert.

Part nesting; Manual or automatic nesting.



Manage technology; manual or automatic selection of the entry and exit points (lead-in/lead-out), move in/out points, select micro joints, move micro joints, etc.

Manual or automatic selection of the order and machining movement.



Simulate the part and generate the CNC program. The integrated Fagor postprocessor will create the part-program for the CNC.

18.1.2 Downloading the application manual.

The nesting manual can be downloaded from:

https://campaign.lanteksms.com/lantek-expert-inside-manual

18.1.3 Configuring the application startup.

There are three different ways to startup the nesting.

- The CNC starts up with a .bat file (for example CNC8070L.bat). In this .bat file, add the line, "start c:\Lantek\Expert\Expert.exe". The nesting software starts up after the CNC.
- For the HMI machine parameters, configure the USERKEY key as "Application" and assign it to the path "c:\Lantek\Expert\Expert.exe". The nesting software starts up when the user presses the [CUSTOM] key.
- In the file ..\MTB\MMC\Config\Apps.ini add the line "Lantek=c:\Lantek\Expert\Expert.exe". To commence nesting, go into Diagnosis mode and from the horizontal menu select Apps > Lantek.

18.1.4 Accessing the application.

To go from the CNC to the nesting application:

- If only the nesting application and the CNC are open, the [CTRL]+[W] keys alternately change from one to the other.
- The [CUSTOM] key accesses the nesting software (if configured by the OEM).
- In Diagnosis mode, from the horizontal menu, select Apps > Lantek (if configured by the OEM).

18.1.5 Configuring the application language.

The application has two languages installed, usually English (recommended) and a local language. List of available languages.

Code.	Language.
Cs	Czech.
from	German.
en	English (United States).
en-US	English (United States).
es	Spanish (Spain).
es-MX	Spanish (México).
fr	French.
it	Italian.
ja	Japanese.
ko	Korean
nl	Dutch (Netherlands).
pl	Polish.
pt	Portuguese (Brazil).
pt-PT	Portuguese (Portugal).
ru	Russian.
sv	Swedish.
th	Thai.
tr	Turkish.
zh-CHS	Chinese (Simplified).
zh-CHT	Chinese (Traditional).

To change the language, follow these steps.

1 Editi the file ..\LantekInside\Expert\Expert.exe.config with a text editor.

2 Change the parameter for the first language (it is recommended to configure it as "en" (English UK) or "en-US" (English US).

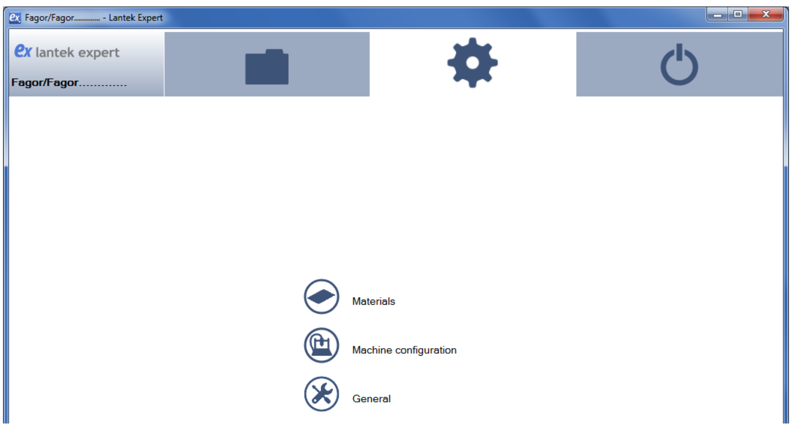
```
<setting name="Language" serializeAs="String">
<value>en</value>
```

3 Change the parameter for the local language; select the desired language.

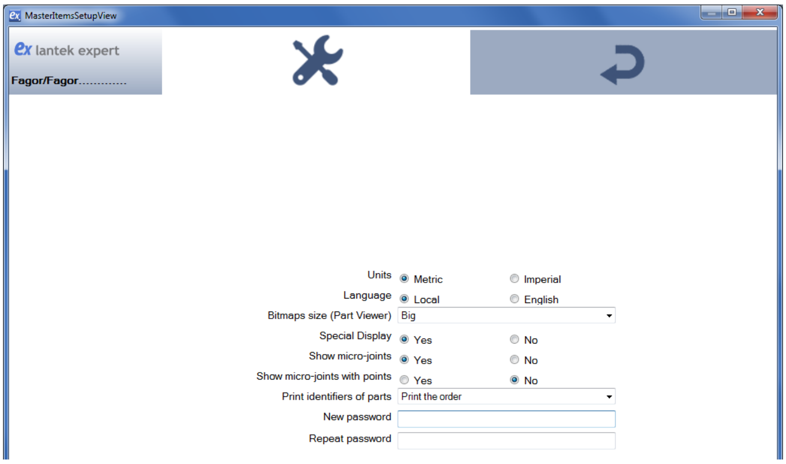
```
<setting name="LocalLanguage" serializeAs="String">
<value>zh-CHS</value>
```

4 Save the changes.

5 Open the application and go to the main menu.

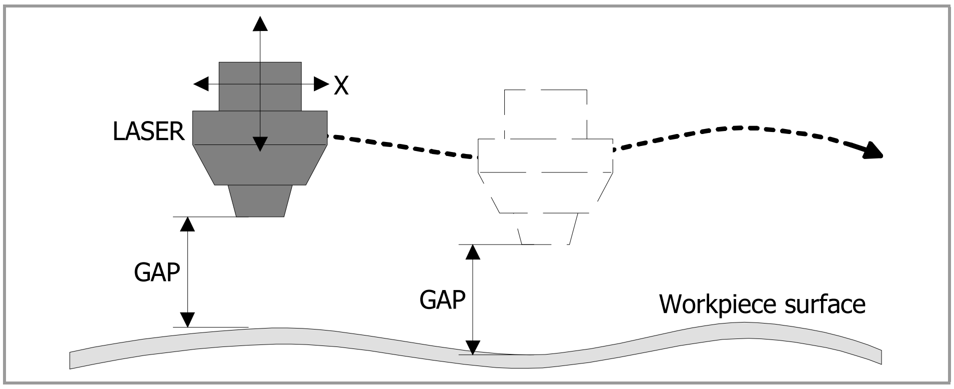


6 From the general button, select one of the languages.



18.2 Gap control.

Gap control makes it possible to maintain a set distance between the laser nozzle and the surface of the sheet. This distance is calculated by a sensor connected to the CNC, so that it offsets the sensor variations on the distance programmed with additional movements in the longitudinal axis.



The CNC allows only one gap control. When trying to activate the gap control from a channel when it is already active in another channel, the CNC will issue the corresponding error message.

For the gap control, the software option "GAP CONTROL" should be available.

The CNC does not allow work when the three functionalities are active simultaneously; leapfrog (#LEAP), gap control (#GAPCTRL), and synchronous switching (#SWTOUT).

The PLC can activate the gap control by executing the corresponding CNC block with the CNCEX command. The instruction executed from the PLC updates the instruction programmed from the CNC and vice-versa.

Sensor calibration.

For correct following by the sensor, it is recommended that the sensor signal be calibrated with the maximum number of points possible.

Programming from the CNC.

The #GAPCTRL ON instruction may be used to activate the gap control. The #GAPCTRL OFF instruction disables the gap control. The programming format is the following; the arguments appear between curly brackets and the optional ones, between angle brackets. Refer to the operating manual for more information.

Activating gap control.

```
#GAPCTRL ON [<X..C> / <X..C{pos}> <, GAP{gap}>]
```

<X..C>	Optional (by default , the last programmed value). Axis associated with the sensor.
<X..C{pos}>	Optional (by default , the last programmed value). Axis associated with the sensor and approach coordinate. Units: mm or inches.
<GAP{gap}>	Optional (by default , the last programmed value). Distance (gap) to be kept from the surface. Units: mm or inches.

```
#GAPCTRL ON [Z, GAP 1]
#GAPCTRL ON [Z-11, GAP 2]
#GAPCTRL ON [GAP 5]
#GAPCTRL ON [Z-10]
#GAPCTRL ON
```

Canceling gap control.

#GAPCTRL OFF

```
#GAPCTRL OFF
```

Programming from the PLC.

The PLC can activate the gap control by executing the corresponding CNC block with the CNCEX command.

Programming format.

The programming format is the following; the arguments appear between curly brackets and the optional ones, between angle brackets.

{condition} = CNCEX [{block}, {plc_mark}<,{channel}>]

{condition}	Condition to execute the CNCEX command.
{block}	Block to be executed.
{plc_mark}	Control mark. The PLC activates the mark when the operation begins and keeps it active until the operation ends.
{channel}	Optional (by default, the first channel). Channel where the PLC must execute the block.

```
{condition] = CNCEX [#GAPCTRL ON [Z, GAP 2], M123]
```

```
{condition] = CNCEX [#GAPCTRL OFF], M123]
```

Execution.

After activating gap control, the axis associated with the sensor moves in programmed feed (if there is no programmed feed, the feed defined in MAXFEED), until detecting the part with the sensor or the programmed coordinate (POS). If the axis reaches the programmed coordinate, without receiving the sensor signal, the CNC will show the corresponding error or not, depending on the GAPTOLCANCEL machine parameter.

The CNC considers that the approach block is complete when the sensor reaches the gap with the tolerance defined in the GAPTOL machine parameter. Once the part is detected, the CNC will keep the defined gap, including if it detects the part when executing #GAPCTRL, without lowering.

During the approach movement, the CNC is able to stop the movement (signals _STOP o _FEEDHOL). In this case, the axis will remain stationary and without gap control until the movement is resumed.

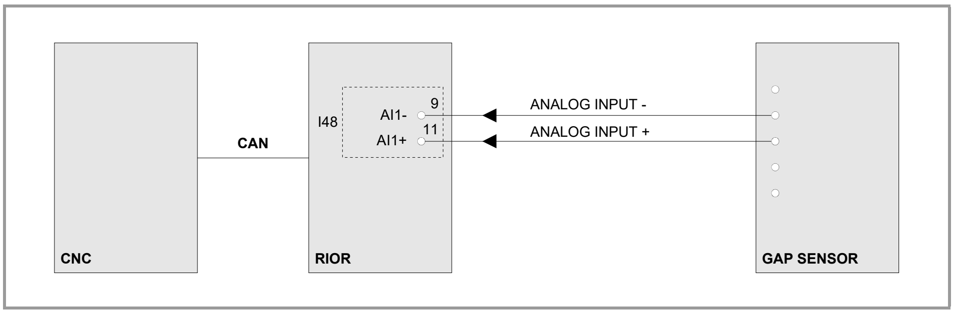
18.2.1 Connecting the gap sensor.

The analog input for the sensor can be installed in remote CAN modules. Use the machine parameters GAPANAINTYPE and GAPANAINID to define the CNC where the sensor input is located. See *"18.2.4 Configuring the gap sensor."* on page 749.

Electrical diagram.

Connecting a sensor to an analog input on remote modules.

The analog inputs are available on modules RIO5 and RIOR.

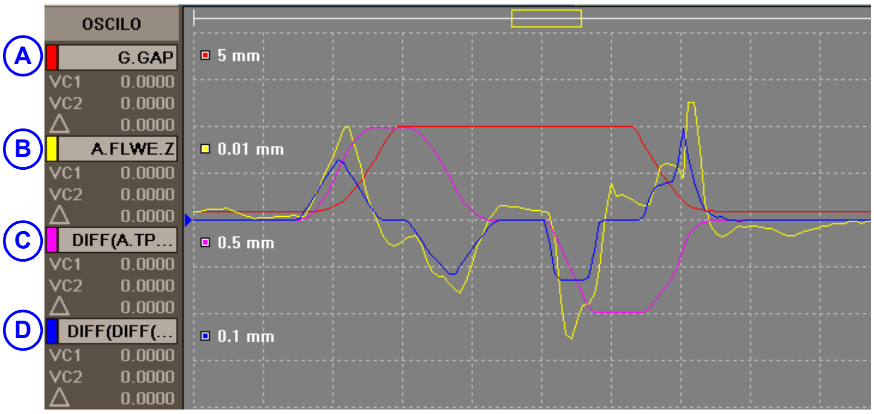


18.2.2 Adjusting the axis that controls the gap.

The axis controlling the gap must not be adjusted in the standard way, rather it must be done with approach movements with the gap control active. The parameters GAPGAIN, PROGAIN, FFGAIN and ACFGAIN will be obtained from these movements. As a reference to prevent overshooting the gap, it is recommended to work with the following values.

Sercos speed **Analog.**	**Sercos position.**
• GAPGAIN = 0.20. • FFGAIN = 90 %. • ACFGAIN = 30 %.	• GAPGAIN = 0.20. • It is recommended to reduce the FFGAIN values by between 0.5 and 1% of the standard adjustment values. • The ACFGAIN values can match those of a standard adjustment.

Speed and acceleration profile without overshooting the gap.



(A) The G.GAP variable reflects the real gap value, as detected by the sensor.
(B) The A.FLWE.Z variable reflects the following error of the Z axis (sensor axis).
(C) The DIFF (A.TPOS.Z) signal indicates the speed on Z.
(D) The DIFF(DIFF(A.TPOS.Z)) signal indicates the acceleration on Z.

The axis that controls the gap should be adjusted so that it can reach zero follow-up errors (parameters FFGAIN and ACFGAIN). When the axes are Sercos, it is possible to make a prior adjustment using the FineTune application.

Parameter.	Meaning.
FFGAIN	Percentage of Feed-Forward in automatic.
ACFGAIN	Percentage of AC-Forward in automatic.

Loop time.

It is advisable to use a cycle time of 2 milliseconds (LOOPTIME=2).

Parameter.	Meaning.
LOOPTIME	Cycle time of the CNC.

18.2.3 Configuring the gap.

Configuring the gap limits.

Parameter.	Meaning.
GAPMIN	Minimum permitted value for the gap; below this value, the CNC will show an error or not as defined in the parameter GAPERRORCANCEL.
GAPMAX	Maximum permitted value for the gap; above this value, the CNC will show an error or not as defined in the parameter GAPERRORCANCEL. This parameter should be lower than GAPDISTLIMIT.
GAPTOL	This parameter defines a positioning window on the programmed gap, to finish the approach block.

If the gap exceeds the range defined in the parameters GAPMIN and GAPMAX, the PLC deactivates the INPOSGAP mark. The CNC will display an error or not, depending on how the parameter GAPERRORCANCEL has been set. In the event of an error, the CNC stops the movement of the axes, according to the braking ramp and controlling the gap during the ramp.

If the axis exceeds the tolerance set in the GAPTOL parameter, the PLC deactivates the INTOL mark. The CNC will display an error or not, depending on how the parameter GAPTOLCANCEL has been set. In the event of an error, the CNC stops the movement of the axes, according to the braking ramp and controlling the gap during the ramp.

Configure the CNC response when exceeding the gap limits.

Parameter.	Meaning.
GAPERRORCANCEL	This parameter indicates if the CNC should cancel the error for gap outside the range, when the gap exceeds the GAPMIN - GAPMAX range. Regardless of the value of this parameter, the CNC keeps monitoring the INPOSGAP mark.
GAPTOLCANCEL	This parameter indicates if the CNC should cancel the error for gap outside the tolerance range, when the gap exceeds the margin defined in the parameter GAPTOL.

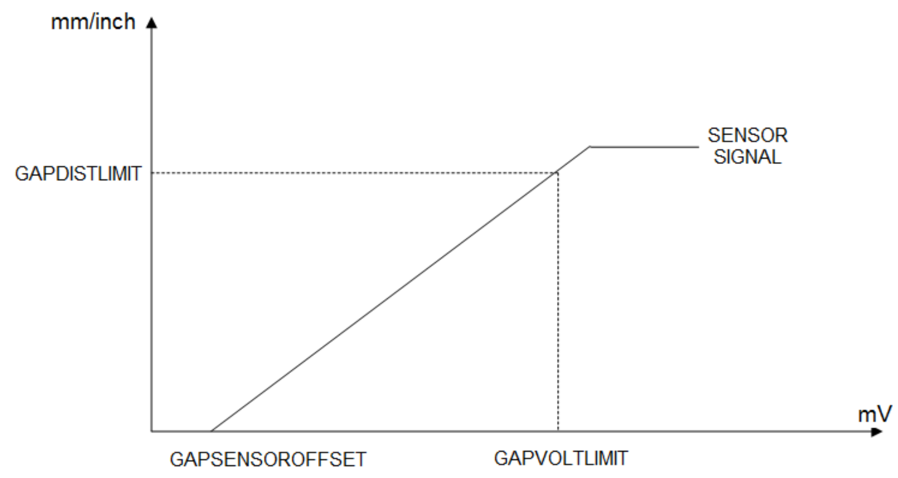
18.2.4 Configuring the gap sensor.

Configuring the analog input to which the sensor is connected.

Parameter.	Meaning.
GAPANAINTYPE	This parameter indicates the location of the analog input connected to the sensor.
GAPANAINID	This parameter identifies the analog input where the sensor is connected.

Configure the gap signal (relationship between distance and voltage).

Parameter.	Meaning.
GAPSENSOROFFSET	Offset (in millivolts) to apply to the sensor from the CNC. If a Fagor analog input is used, this parameter is not required, as each has its own offset.
GAPSENSORCH	Change the sensor signal sign.
GAPDISTLIMIT	Distance corresponding to the limit analog signal of the sensor (parameter GAPVOLTLIMIT).
GAPVOLTLIMIT	Voltage (in millivolts) corresponding to the limit path of the sensor (parameter GAPDISTLIMIT).



Adjusting the sensor collision signal.

PLC signals.	Meaning.
GAPCOLLISIONMODE	The gap control sensor may have a collusion signal connected to the CNC, which should be managed by the PLC via the GAPCOLLISION mark. If the PLC activates the GAPCOLLISION mark, the CNC will display an error or not, depending on how this parameter has been set. The error does not activate the emergency.

18.2.5 Fast approach to the gap ("far position" signal from the sensor).

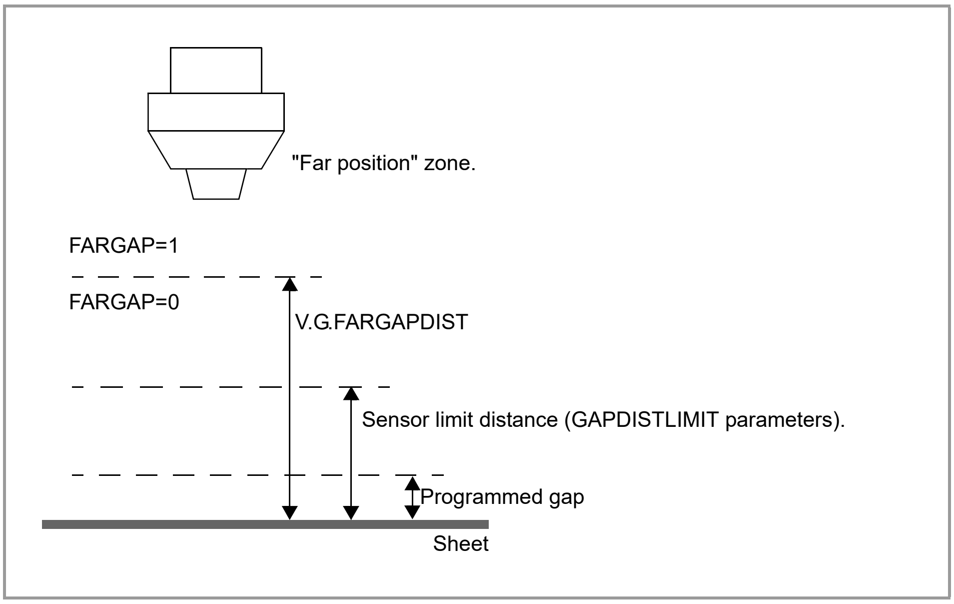
Modifiable logic signals.

PLC signals.	Meaning.
FARGAP	"Far position" signal provided by the gap sensor. The PLC must activate (=1) this marker when the sensor is in the far position and deactivate it (=0) when it leaves the far position. The distance from the plate to the far position point must be defined in the variable (V.)G.FARGAPDIST

Variables.

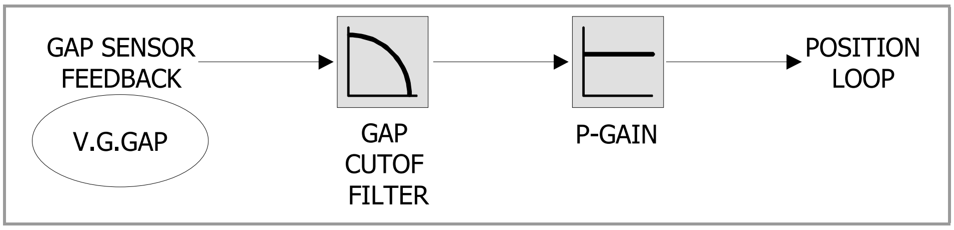
PLC signals.	Meaning.
(V.)G.FARGAPDIST	Distance from the sheet metal to the far position point of the sensor.

When the gap sensor provides a "far position" signal, the CNC can use this information in the approach to the gap, in the first positioning (#GAPCTRL) and in jumps over the gap. (#LEAP). Once the sensor leaves the far position (the PLC deactivates the FARGAP signal), the CNC assumes that the sensor is at a distance (V.)G.FARGAPDIST from the plate, and calculates the braking ramp in order to position itself in the gap as fast as possible.



18.2.6 Adjusting the sensor position loop.

GAPCONTROL	
GAPGAIN	Proportional gain applied to the signal sensor, in position. To use values greater than 0.2 (close to 1), it is advisable to use the parameter GAPSENSORFILTER.
GAPAPPROACHDYN	Dynamic response during the final part of the movement approaching the plate.
GAPSENSORFILTER	Low step filter; it is advisable to define a filter of order 2 and cutoff frequency of 30 Hz.
ORDER	Filter order.
FREQUENCY	Break frequency.



Adjust with the aid of the oscilloscope.

1 Executing a program when gap control is active.

2 Selecting the variables (V.)G.GAP y (V.)A.GAPCOMP.Z (sensor axis).

(V.)G.GAP	Actual gap value, detected by the sensor.
(V.)A.GAPCOMP.Z	Compensation introduced by gap control into the axis associated with the sensor to maintain the gap.

3 Attempting to reduce the gap (variable G.GAP) while incrementing the proportional gain (parameter GAPGAIN). Increasing the proportional gain with small increments (e.g. 0.05). Using the GAPAPPROACHDYN parameter to adjust the GAPGAIN value during the final part of the plate approach, when the CNC has detected the sensor signal.

4 Reducing the filter frequency (machine parameter FREQUENCY) to obtain a softer response. Since the gap value increases, increase the proportional gain (machine parameter GAPGAIN).

18.2.7 PLC signals.

Logic consultation signals.

PLC signals.	Meaning.
INPOSGAP	The CNC activates this mark if the gap is within the range defined by the parameters GAPMIN-GAPMAX. If the gap exceeds the range defined in the parameters GAPMIN-GAPMAX, the CNC disables the INPOSGAP mark and stops the movement of the axes. The CNC will display an error or not, depending on how the parameter GAPERRORCANCEL has been set. In the event of an error, the CNC stops the movement of the axes, according to the braking ramp and controlling the gap during the ramp.
INTOL	The CNC activates this mark if the gap is within the tolerance limit defined by the parameter GAPTOL regarding the gap value programmed. If the axis exceeds the tolerance set in the GAPTOL parameter, the CNC deactivates the INTOL mark. The CNC will display an error or not, depending on how the parameter GAPTOLCANCEL has been set. In the event of an error, the CNC stops the movement of the axes, according to the braking ramp and controlling the gap during the ramp.

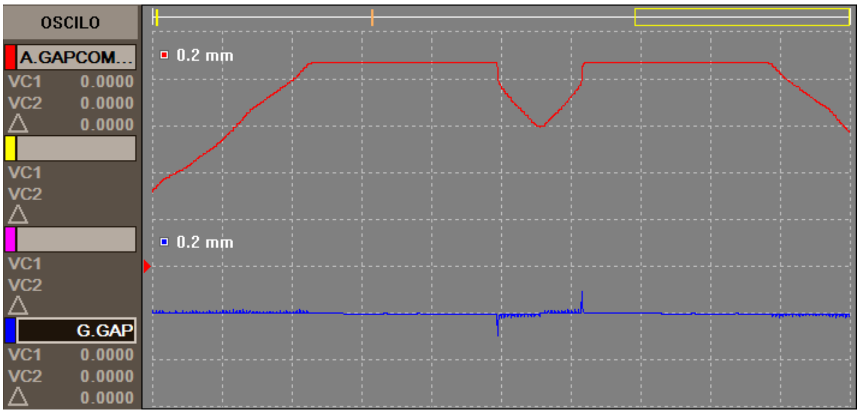
Modifiable logic signals.

PLC signals.	Meaning.
_FEEDHOL	If the PLC sets this mark at a high logic level, the CNC channel momentarily interrupts the movement of the axes. When the mark returns to the high logic level, the movement of the axes continues. All the stops and starts of the axes are carried out with the corresponding acceleration and deceleration. • This mark does not affect the independent movement of an axis (independent interpolator). • On hirth axes, if the axis does not stop in a particular position, the CNC does not activate the MATCH(axis) mark. • This mark does not turn off the laser. If the power control is active, the laser removes the minimum defined power. If the PLC sets the _FEEDHOL at a low logic level in a motionless block, the CNC continues executing the program until it detects a motion block. 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.
_STOP	If the PLC sets this mark at a low logic level, the CNC channel interrupts the execution of the part program. To resume the execution of the program, besides setting this mark at a high logical level, the PLC should activate the CYSTART mark. • This mark does not affect the independent movement of an axis (independent interpolator), which also does not affect the key [STOP]. • This mark does not turn off the laser. • This mark does not disable gap control, except in approach movement. In this case, the axis will remain stationary and without gap control until the movement is resumed. The treatment received by the signal _STOP is similar to that given to the [STOP] key in the CNC. With this signal at a low logic level, all keys remain enabled.
ENABLEGAP	This mark is active by default. The PLC deactivates this mark to disable the active gap control. The PLC activates this mark to enable the active gap control in the CNC; if no gap control is active, this mark does nothing.
GAPCOLLISION	The PLC activates this mark to indicate that the collision sensor has been activated. This mark is not kept in memory. When this mark is active, the CNC will display an error or not, depending on how the parameter GAPCOLLISIONMODE has been set.

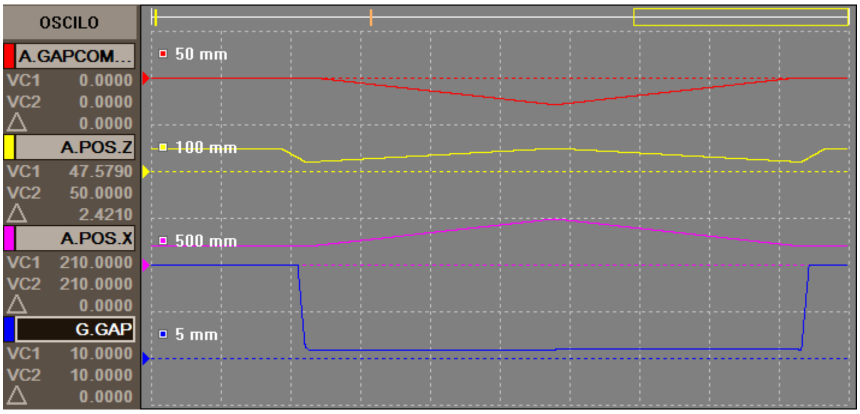
18.2.8 Gap compensation examples.

(V.)G.GAP	Actual gap value, detected by the sensor.
(V.)A.POS.Z	Z axis. Actual machine coordinators of the sensor.
(V.)A.POS.X	Actual machine coordinates for X-axis.
(V.)A.GAPCOMP.Z	Compensation introduced by gap control into the axis associated with the sensor to maintain the gap.

Part with a profile irregularity.



Part with a rising profile.



18.2.9 Operating the PLC to remove the axis associated with the gap control.

```
DFU {conditions} AND NOT M666 AND NOT INCYCLE
   = RES ENABLEGAP
   = RES ENABLELEAP
   = SET IRESETZ
   = TG1 333 12 ; Dwell, 12 milliseconds.

T333 = M666

DFD M666 = RES IRESETZ
   = MOVE ABS (Z, 513000,50000000,NULL)

DFDB B10KEYLED1 = SET ENABLEGAP = SET ENABLELEAP
```

CALCULATION OF THE KINEMATICS DIMENSIONS.

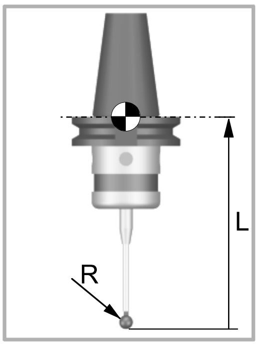
19

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.

19.1 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.

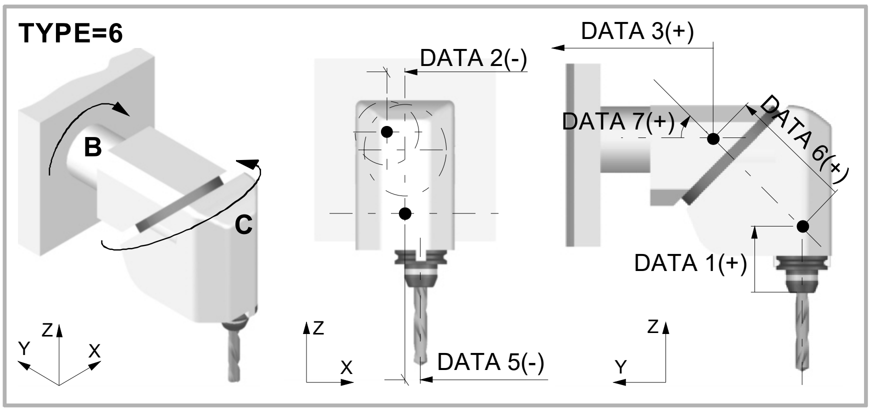
The CNC can have configured two probes. Before any probing move, the CNC must know which is the active probe, or, which is the same, which of the two probes it must attend to. It is selected via part-program or MDI using the instruction #SELECT PROBE.

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.

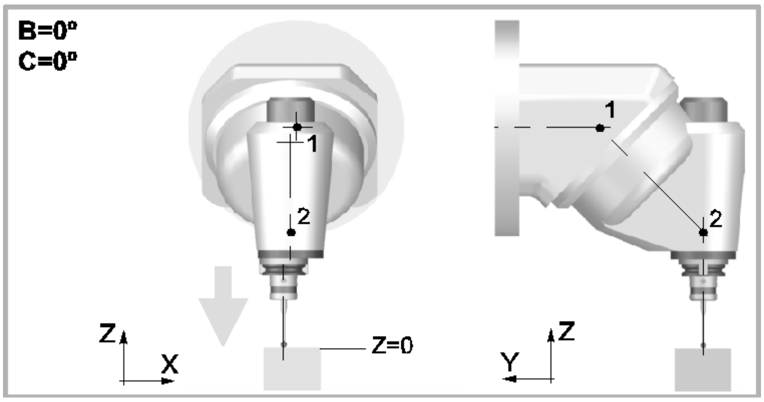
TDATA1	It indicates the distance between the tip of the quill and the secondary rotary axis along the Z axis.
TDATA2	It indicates the distance between the secondary rotary axis and the main axis along the X axis.
TDATA3	It indicates the distance between the secondary rotary axis and the main axis along 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.

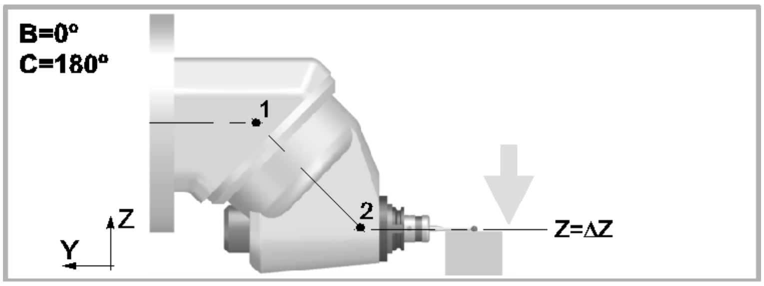
First probing move.

With rotary axes B and C positioned at 0º, make a probing movement on the Z side and preset the position Z = 0.



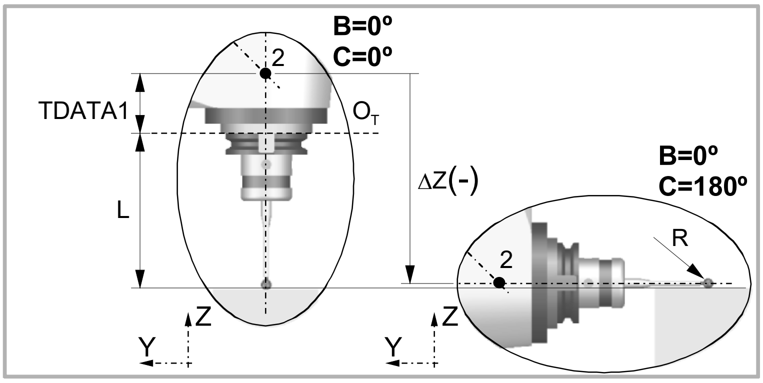
Second probe.

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.

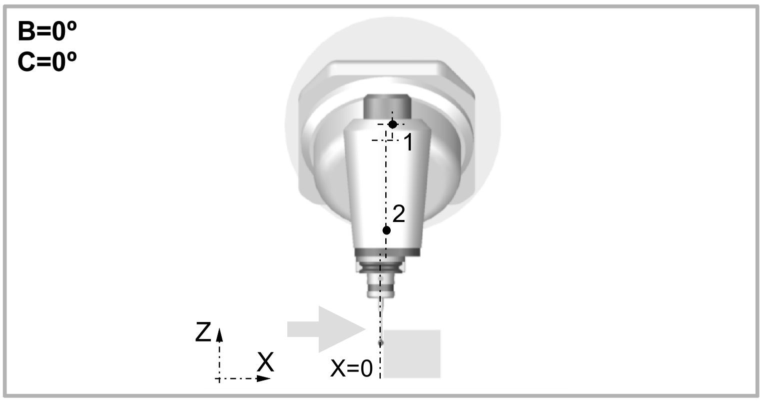


Solution.	
$$\lvert TDATA1\rvert = \lvert \Delta Z\rvert + R - L$$	
TDATA1	Dimensions to calculate.
OT	Reference point of the tools.
ΔZ	Coordinate shown by the CNC.
L	Probe length.
R	Probe ball radius.

Calculation of parameter TDATA5.

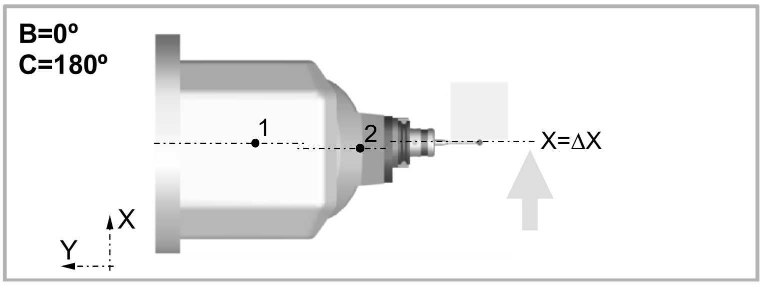
First probing move.

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.



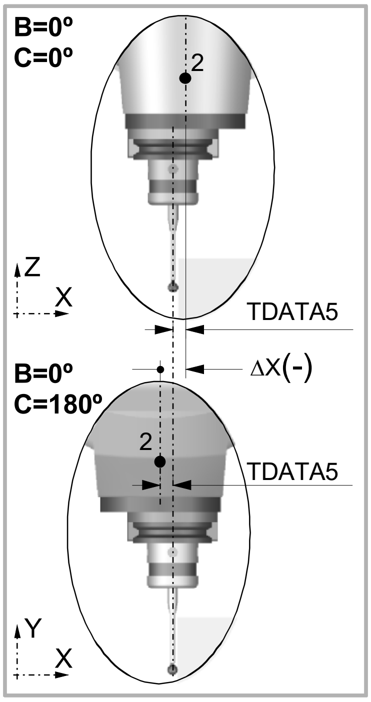
Second probe.

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 ∆X.



Mathematical calculations.

After changing the position with respect to the center of rotation ·2·, the only unknown data after both probing movements is TDATA5.

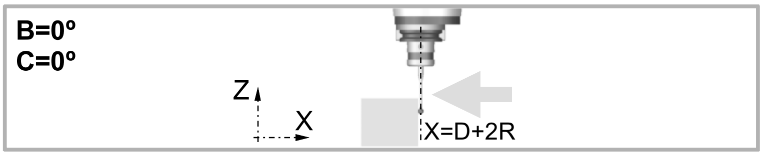


Solution.	
$$\lvert TDATA5\rvert = 0,5 \times \lvert \Delta Z\rvert$$	
TDATA5	Dimensions to calculate.
∆X	Coordinate shown by the CNC.

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.

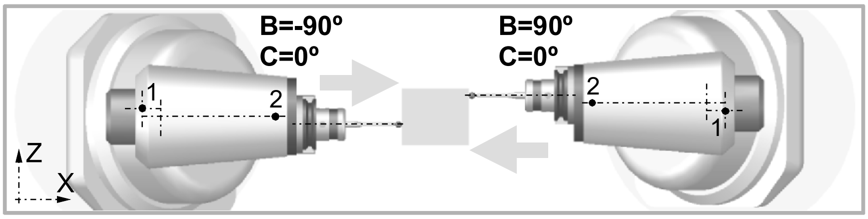


Second probe.

With the C rotary axis positioned at 0º and B at -90º, make a probing movement on the first X side. Take note of the coordinate displayed on the CNC, depicted in the diagram as ΔX'.

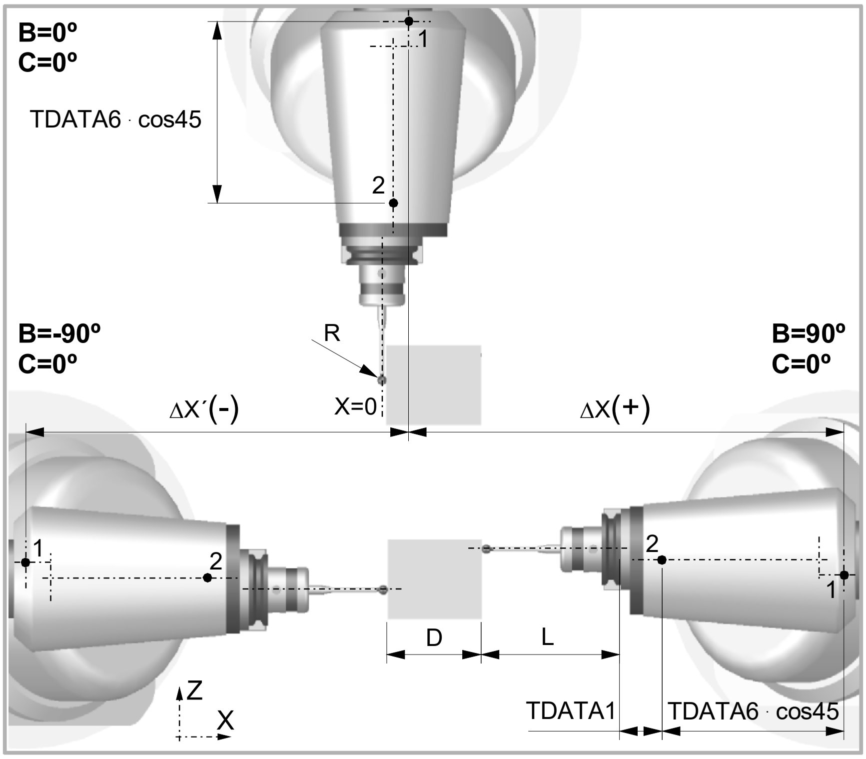
Third probing move.

With the C rotary axis positioned at 0º and B at 90º, make a probing movement on the second X side. Take note of the coordinate displayed on the CNC, depicted in the diagram 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.

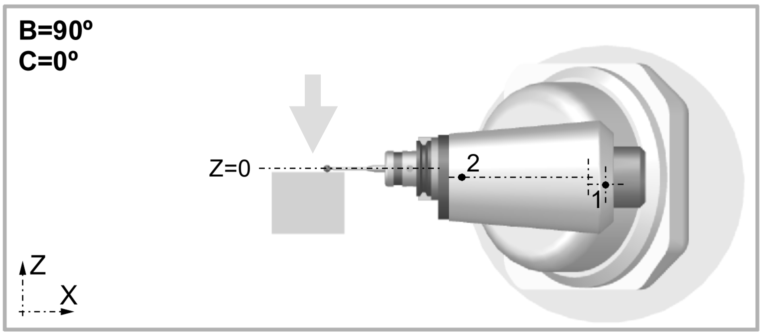


Solution.	
$$\lvert TDATA6\rvert = \frac{\lvert\Delta X\rvert + \lvert\Delta X'\rvert - D - 2L - 2\lvert TDATA1\rvert}{2 \bullet \cos 45}$$	
TDATA1	Known dimension.
TDATA6	Dimensions to calculate.
D	Length of the part.
ΔX' ΔX	Coordinates shown by the CNC.
L	Probe length.
R	Probe ball radius.

Calculation of parameter TDATA2.

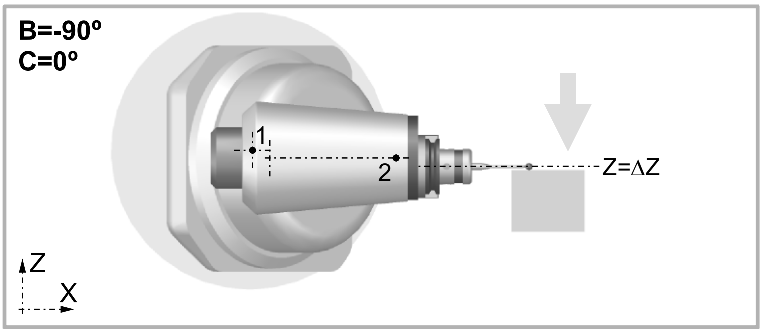
First probing move.

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.



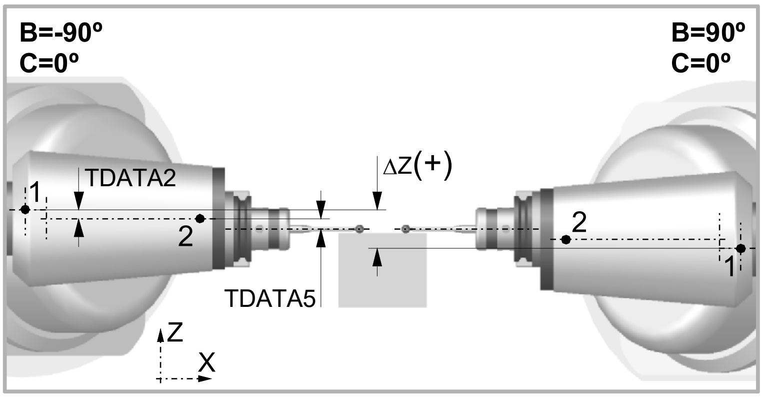
Second probe.

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.

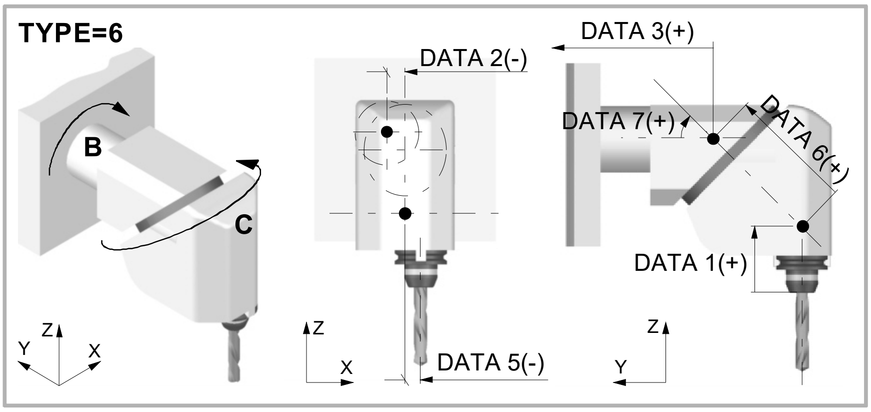
After changing the position with respect to the center of rotation ·1·, the only unknown data after both probing movements is TDATA2.



Solution.	
$$\lvert TDATA2\rvert = 0,5 \cdot \lvert \Delta Z\rvert - \lvert TDATA5\rvert$$	
TDATA2	Dimensions to calculate.
TDATA5	Known dimension.
ΔZ	Coordinate shown by the CNC.

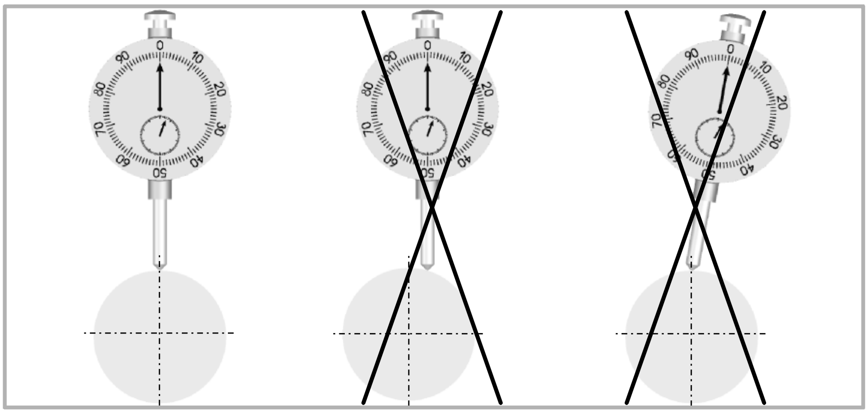
19.2 Angular spindle. Calculation of the dimensions using a dial indicator.

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.

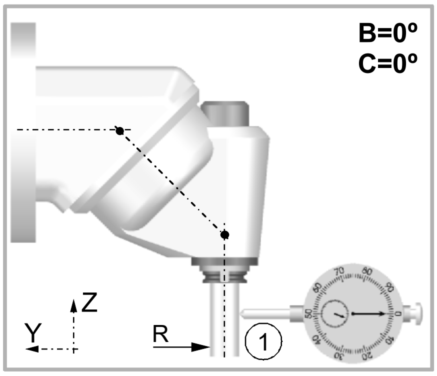
When touching a cylinder with a dial indicator, contact must be made on the outside line of the cylinder. Otherwise, there will be a measuring error when presetting the value of the cylinder value. The dial indicator should be as perpendicular as possible to the tangent of the surface being measured.



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.

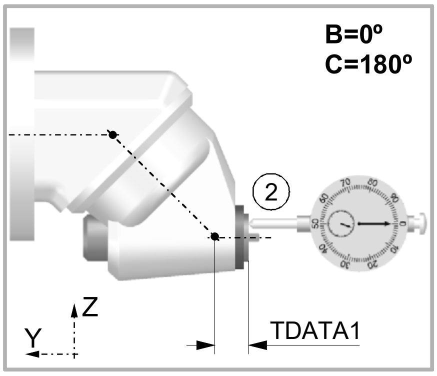
Calculation of parameter TDATA11.

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.

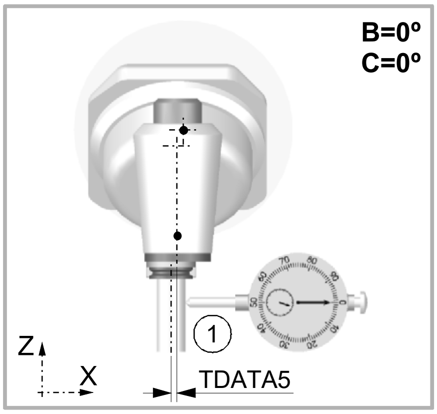


1. Remove the dial indicator, without loosening it and position the rotary C axis at 180º.
2. Touch with the dial indicator at point 2.
3. Jot down the position value (coordinate) shown at the CNC ΔY).

Solution.	
$\lvert TDATA1 \rvert = \lvert \Delta Y \rvert$	
TDATA1	Dimensions to calculate.
ΔY	Coordinate shown by the CNC.

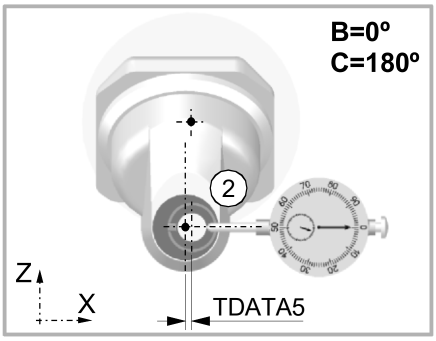
Calculation of parameter TDATA5.

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.

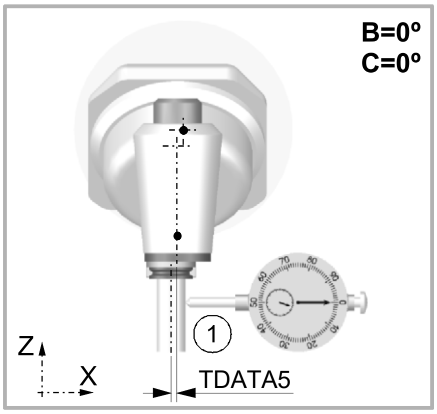


1. Remove the dial indicator, without loosening it and position the rotary C axis at 180º.
2. Touch with the dial indicator at point 2.
3. Jot down the position value (coordinate) shown at the CNC ΔX).

Solution.	
$$\lvert \mathrm{TDATA5} \rvert = 0,5 \cdot \lvert \Delta X \rvert$$	
TDATA5	Dimensions to calculate.
ΔX	Coordinate shown by the CNC.

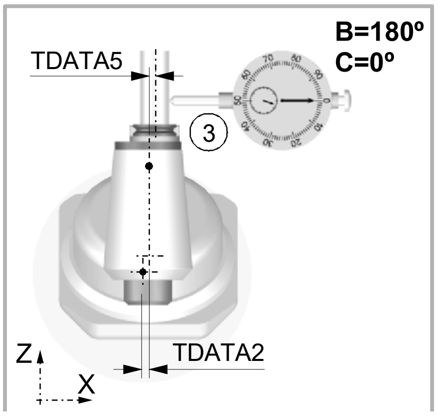
Calculation of parameter TDATA2.

First contact (made when calculating TDATA5).



1. Keep previous preset.

Second contact.

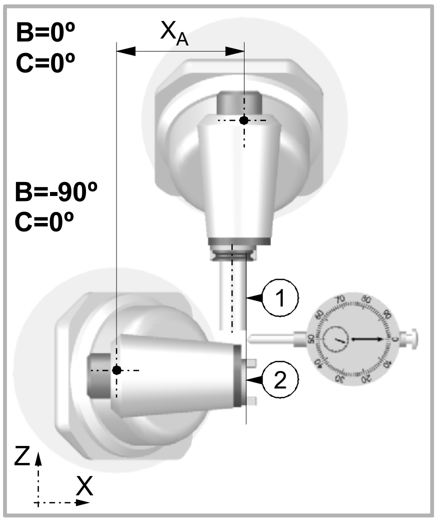


1. Remove the dial indicator, without loosening it and position the rotary C axis at 0º.
2. Position B rotary axis at 0º.
3. Touch with the dial indicator at point 2.
4. Jot down the position value (coordinate) shown at the CNC ΔX).

Solution.	
$$\lvert \mathrm{TDATA2} \rvert = 0,5 \cdot \lvert \Delta X \rvert - \lvert \mathrm{TDATA5} \rvert$$	
TDATA1	Dimensions to calculate.
TDATA5	Dimensions to calculate.
ΔX	Coordinate shown by the CNC.

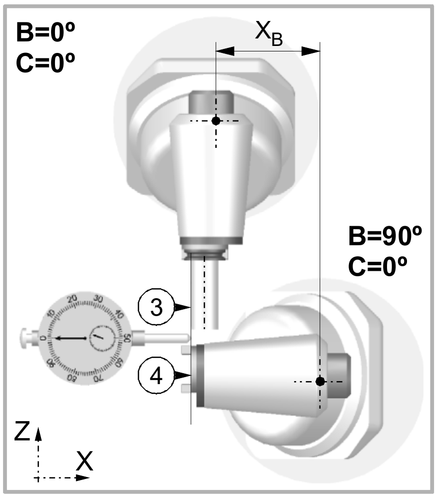
Calculation of parameter TDATA6.

First and second contacts.

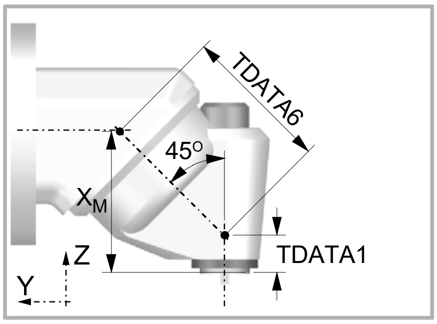


1 Position rotary axes B and C at 0º.
2 Move the table in the X axis until the dial indicator and the cylinder touch each other at point 1.
3 Preset X=R so X=0 coincides with the shaft of the cylinder placed in the spindle.
4 Remove the dial indicator, without loosening it and position the B rotary axis 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.

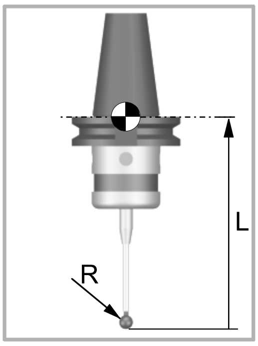
$$|X_M| = 0,5 \cdot (|X_B| + |X_A|)$$

$$|TDATA6| = \sqrt{2} \cdot (|X_M| - |TDATA1|)$$

TDATA5	Dimensions to calculate.
ΔX	Coordinate shown by the CNC.

19.3 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.

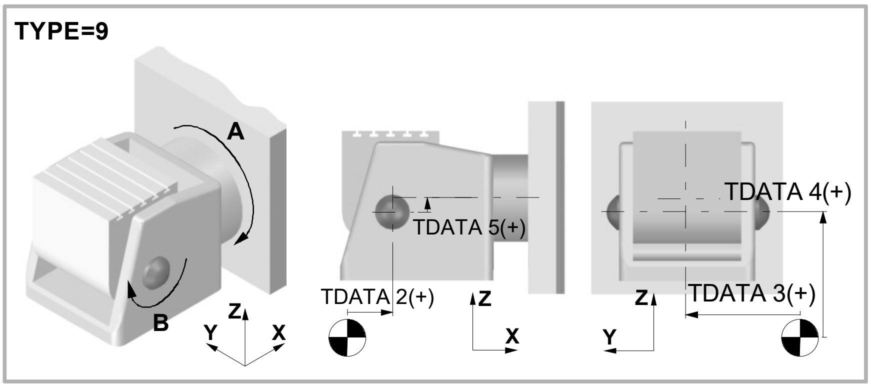
The CNC can have configured two probes. Before any probing move, the CNC must know which is the active probe, or, which is the same, which of the two probes it must attend to. It is selected via part-program or MDI using the instruction #SELECT PROBE.

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).

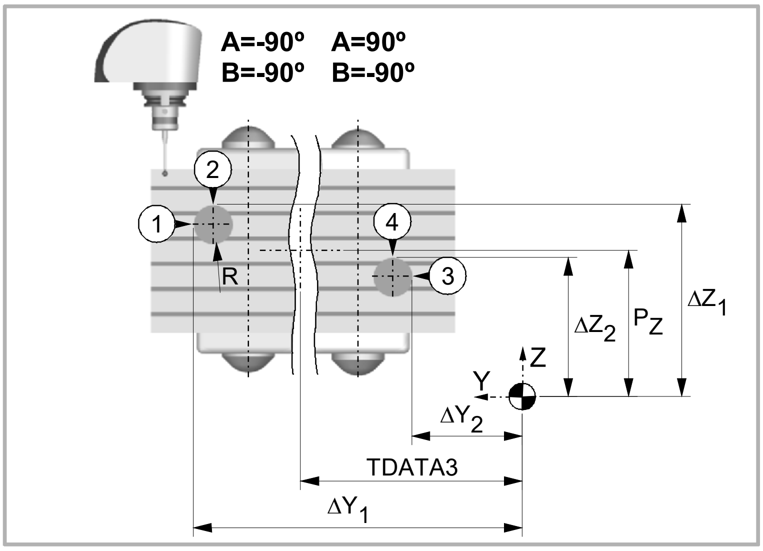
In this type of kinematics the following dimensions must be defined. All the parameters may be set with a positive or negative value. The (+) sign in the illustrations indicates that the direction is assumed as positive.

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 diagram shows two different positions that have the center of axis A in common. The coordinate for the Y axis is the parameter TDATA3 and the coordinate for the Z axis is 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.

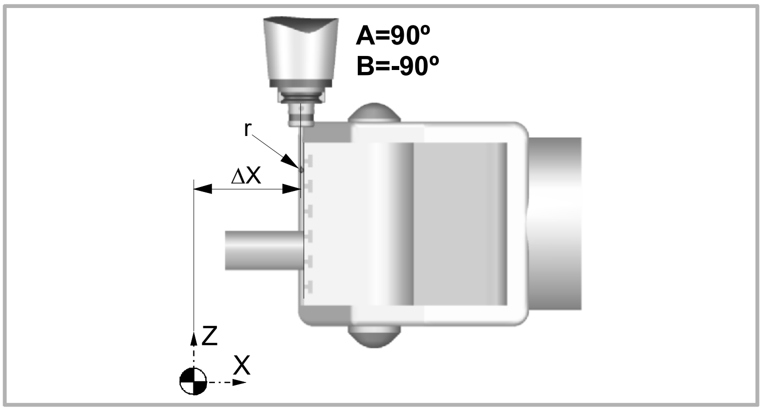
2 Make a probing movement along the Y axis at point 3. Jot down the position value (coordinate) shown at the CNC, appearing in the drawing as ΔY2.

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.	
$$\lvert TDATA3\rvert = 0,5 \cdot (\lvert\Delta Y_1\rvert + \lvert\Delta Y_2\rvert)$$ $$\lvert Pz\rvert = 0,5 \cdot (\lvert\Delta Z_1\rvert + \lvert\Delta Z_2\rvert - R)$$	
TDATA3 Pz	Coordinates of the A axis.
ΔY1, ΔZ1 ΔY2, ΔZ2	Coordinates shown by the CNC.
R	Cylinder radius.

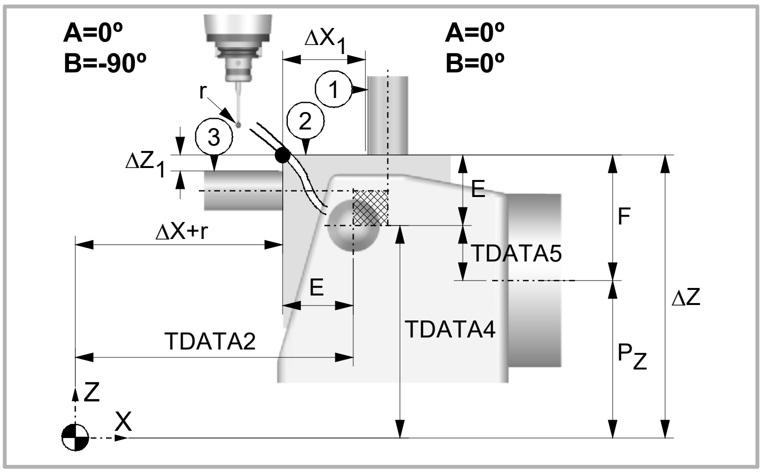
Origin preset on the X axis.



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. $$\lvert E\rvert = 0,5\cdot(\lvert\Delta X_1\rvert + \lvert\Delta Z_1\rvert + 2R + r)$$	
Calculation of parameters. $$\lvert F\rvert = \lvert\Delta Z\rvert - \lvert Pz\rvert$$ $$\lvert TDATA5\rvert = \lvert F\rvert - \lvert E\rvert$$ $$\lvert TDATA4\rvert = \lvert TDATA4\rvert - \lvert Pz\rvert$$ $$\lvert TDATA2\rvert = \lvert\Delta X\rvert + r + \lvert E\rvert$$	
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.

DMC (DYNAMIC MACHINING CONTROL).

20

The DMC is only available for the master spindle. The spindle must be enabled for the DMC in the machine parameter DMCSPDL.

The choice of machining feedrate depends on the material to be machined, the tool (material, number of teeth, etc.) and the depth of pass. The feedrate for machining is set at the start and remains constant until the end. If the machining conditions change (for example, due to tool wear), the programmed feedrate may no longer be appropriate, which will have a negative effect on the tool life, machining time, etc.

DMC (dynamic machining control) adapts the feedrate during machining to maintain the cutting power as close as possible to ideal machining conditions. DMC adapts the feedrate by changing the override.

DMC optimizes the use of the machine and the tool, which makes it possible to increase the material removal rate (MRR) without impairing the life of the tool, since it is operating according to its rated conditions. Optimization is based on the cutting power to be reached, which is also referred to as target power. This power not only depends on the tool, but also on the material and the cutting conditions (feedrate, spindle turning speed, depth of pass and sideways step (pass)), and therefore its value must be linked to the entire combination of the tool+material+cutting conditions.

DMC is only available for milling operations with "Milling" and "Surface milling” operations. This function can be applied to roughing and finishing operations, but this function provides the most benefits in terms of machining time and tool life in roughing operations.

20.1 Setup.

20.1.1 Configuring the override limits for the DMC.

Parameter.	Meaning.
MINDMCOVR	Minimum DMC override of all channel axes.
MAXDMCOVR	Maximum DMC override of all channel axes.

These parameters indicate the minimum and maximum percentage of feedrate (feedrate override) that DMC (dynamic machining control) can apply on the channel. Both percentages can be changed in the instruction #DMC ON. The feedrate percentage applied by DMC is added to that selected using the operator panel switch.

$$\mathrm{Override}(\%) = \frac{\mathrm{OverrideDMC}(\%) \times \mathrm{OverrideJOG}(\%)}{100}$$

The resulting override (overrideDMC + overrideJOG) may exceed the set limits for both parameters. The CNC always abides by the maximum limit set in the machine parameter MAXOVR.

If the user selects an override with the switch on the operator panel that is lower than MINDMCOVR, the CNC inhibits DMC (but does not deactivate it); when the override once again exceeds MINDMCOVR, DMC will once again operate as normal.

The feedrate percentage set by the program (variable V.G.PRGFRO) or by the PLC (variable V.PLC.FRO) inhibit DMC, but they do not deactivate it; if canceled, DMC resumes control over the feedrate percentage.

20.1.2 Enable a spindle to allow the DMC.

Parameter.	Meaning.
DMCSPDL	Spindle with enabled power control. If this parameter is set to ·No·, the CNC cannot enable the DMC for the spindle.

20.1.3 Displaying the DMC behavior on the oscilloscope.

The following variables enable the DMC behavior to be displayed on the oscilloscope. See *"20.4.4 Machining analysis."* on page 783.

Variables.	Description.
(V.)G.LEARNEDPWRSP	Target power calculated for DMC in the learning phase (percentage of rated power). This variable does not include the no-load power. Units: Percentage.
(V.)G.DMCACTPWR	Active power on the spindle, measured by DMC (percentage of the rated power). Units: Percentage.
(V.)G.DMCOVR	Feedrate percentage (feed override) calculated by DMC. Units: Percentage.

To access the variables from the oscilloscope, omit the prefix "V." from the mnemonic.

The variables and data required for representation on the oscilloscope are found in the ConfigDMC.osc file (in the ../Fagor/Tuning file). This file can be loaded in the oscilloscope from the softkeys menu, using the normal method.

20.2 Programming DMC.

20.2.1 Activating the DMC.

The instruction #DMC ON activates DMC, always on the master spindle. DMC is only available for milling operations with "Milling" and "Surface milling" operations. DMC can be applied to roughing and finishing operations, but this function provides the most benefits in terms of machining time and tool life in roughing operations.

Programming.

This instruction must be programmed alone in the block. All commands are optional when defining this instruction.

Programming format.

The programming format is the following; the arguments appear between curly brackets and the optional ones, between angle brackets.

```
#DMC ON [<PWRSP {power}> <, OVRMIN{%}> <, OVRMAX{%}> <, FZMIN{feed}>
<, FZMAX{feed}>]
```

PWRSP{power}	Target power or ideal cutting power is defined as the percentage of the rated power of the spindle. • Optional; if not programmed, it is calculated by the CNC. • Values: 0 — 100 %.
OVRMIN{%}	Minimum override permitted for DMC. • Optional (by default, the machine parameter value MINDMCOVR). • Values: 10 — 100 %.
OVRMAX{%}	Maximum override permitted for DMC. • Optional (by default, the machine parameter value MAXDMCOVR). • Values: 100 — 255 %.
FZMIN{feed}	Minimum feedrate per tooth permitted during DMC. • Optional; if not programmed or FZMIN > FZMAX, the CNC does not monitor the minimum feedrate per tooth. • Values: 0 — 99999.9999 mm/tooth 0 — 3937.00787 inches/tooth.
FZMAX{feed}	Maximum feedrate per tooth permitted during DMC. • Optional; if not programmed or FZMIN > FZMAX, the CNC does not monitor the maximum feedrate per tooth. • Values: 0 — 99999.9999 mm/tooth 0 — 3937.00787 inches/tooth.

```
#DMC ON
    (The CNC activates DMC with the default values.)
    (DMC initiates the learning phase to calculate the target power).
#DMC ON [PWRSP 80, OVRMIN 90, OVRMAX 110, FZMIN 0.8, FZMAX 1.3]
    (The CNC activates DMC with the programmed values).
#DMC ON [OVRMIN 90, OVRMAX 110, FZMIN 0.8, FZMAX 1.3]
    (The CNC activates DMC with the programmed values).
    (DMC initiates the learning phase to calculate the target power).
```

Target power or ideal cutting power.

The target power is programmed as a percentage of the rated power of the spindle. The programming of the target power is optional; if it is not programmed, the CNC conducts a learning phase to determine it. See *"20.4.1 DMC operation."* on page 778.

Feed per tooth.

The feedrate modified with the override obeys the minimum and maximum feedrate per tooth set for the tool. So that the CNC can oversee the feedrate per tooth, the number of teeth on the tool must be determined from the tool table. See *"20.3 Define the tool data."* on page 777.

Considerations and limitations.

Work modes.

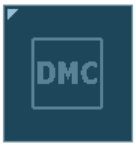
- DMC can only be active when running; in a simulation, the CNC will analyze the activation and deactivation instructions, but will not start up DMC.
- DMC can only be active in automatic mode; it is not activated in manual mode. When entering in tool inspection, it is deactivated and resumes operation after the axes are repositioned.
- DMC is incompatible with some types of operations, such as threading, drilling, etc. In addition, DMC will be automatically deactivated if the active tool is not a "Milling" or "Surface milling" type.

Tool change or offset.

The following tool-related actions deactivate DMC. The user is responsible for reprogramming DMC with the appropriate values for the new tool.

- Changing the tool.
- Changing the offset.
- Executing an M6.
- Changing any of the offset characteristics that affect the power consumption of the spindle (by writing the corresponding variable).

Tool inspection.



When entering the tool inspection mode, DMC is deactivated. At the end of the tool inspection, the DMC is automatically activated and allows learning phase may be repeated by pressing the "DMC learning" softkey.

DMC behavior after being deactivated.

The tool paths in G0, axis stops using [STOP] and the softkey “Set DMC off” temporally deactivate the DMC. In these cases, DMC acts as follows:

- If DMC has not begun to measure the no-load power, the process will not begin.
- If DMC is waiting for the spindle to reach the programmed speed or is measuring the no-load power, DMC is deactivated when it finishes measuring the no-load power.
- During the learning phase, DMC does not consider the time that the cause of the deactivation persists as learning time.
- During the execution without a load, DMC is deactivated, but it will continue to detect entries into the part.
- If the tool is starting to enter the part, DMC will be deactivated once it has entered.
- If the tool is inside the part, DMC is deactivated, but it will continue to detect the part exits and excessive power consumption.
- If the tool is exiting from the part, DMC is deactivated once the exit is completed.

20.2.2 Deactivating the DMC.

The #DMCºOFF instruction deactivates DMC. The M02 or M30 (end of program) functions and the reset also deactivate DMC. A spindle stop, function M5, deactivates DMC.

Programming.

This instruction must be programmed alone in the block. This instruction has no commands.

Programming format.

The programming format is:

#DMC OFF

```
#DMC OFF
    (The CNC deactivates DMC).
```

20.2.3 Summary of the variables.

The following variables may be accessed from (PRG) the part-program and from the MDI/MDA mode, PLC and (INT) from an external application. For each variable, the table indicates whether the access is reading (R) or writing (W). PLC access to the variables, for both reading and writing, will be synchronous. Access to the variables from the part program returns the value for the block preparation (it does not stop the preparation), except when indicated otherwise.

Variables.	PRG	PLC	INT
(V.)[ch].MPG.MINDMCOVR Minimum DMC override of all channel axes. Units: Percentage.	R	R	R
(V.)[ch].MPG.MAXDMCOVR Maximum DMC override of all channel axes. Units: Percentage.	R	R	R
(V.)[ch].MPA.DMCSPDL.sn Spindle with enabled power control. Units: -.	R	R	R
(V.)[ch].G.FRO Active feedrate (feed override) percentage. Units: Percentage.	R(*)	R	R
(V.)[ch].G.PRGFRO Feedrate (feed override) percentage defined by the program. Units: Percentage.	R/W	R	R
(V.)[ch].PLC.FRO Feedrate (feed override) percentage defined by the PLC. Units: Percentage.	R(*)	R/W	R
(V.)[ch].G.CNCFRO Feedrate percentage (feed override) selected on the switch of the operator panel. Units: Percentage.	R(*)	R	R/W
(V.)[ch].G.NCUTTERS Tool being prepared. Number of teeth. Units: Teeth.	R/W	R	R
(V.)[ch].TM.NCUTTERS[offset] [Offset] of the active tool. Number of teeth. Units: Teeth.	R/W(*)	R/W	R/W
(V.)TM.NCUTTERST[tool][offset] [Offset] of the [tool]. Number of teeth. Units: Teeth.	R/W(*)	R/W	R/W

(*) The CNC evaluates the variable during execution (it stops the block preparation).
(**) The CNC evaluates the variable during execution, but it stops the block preparation.

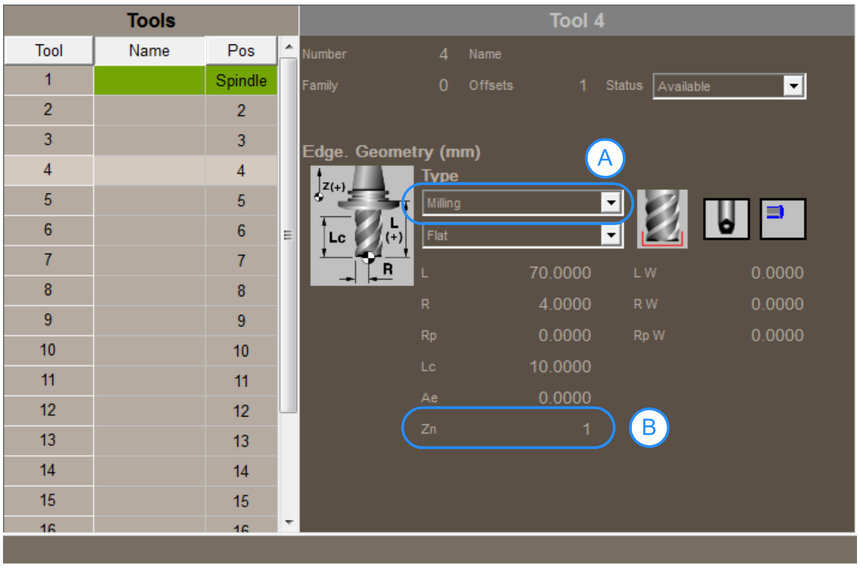
Variables.	PRG	PLC	INT
(V.)[ch].G.DMCPWRSP Target power, defined as a percentage of the rated power of the spindle. Value programmed in the PWRSP command of the #DMC ON instruction. Units: Percentage.	R	R	R
(V.)[ch].G.DMCOVRMIN Minimum feedrate (feed override) percentage allowed for DMC. Value programmed in the OVRMIN command of the #DMC ON instruction. Units: Percentage.	R	R	R
(V.)[ch].G.DMCOVRMAX Maximum feedrate (feed override) percentage allowed for DMC. Value programmed in the OVRMAX command of the #DMC ON instruction. Units: Percentage.	R	R	R
(V.)[ch].G.DMCFZMIN Minimum feedrate per tooth permitted during DMC. Value programmed in the FZMIN command of the #DMC ON instruction. Units: Millimeters/tooth or inches/tooth.	R	R	R
(V.)[ch].G.DMCFZMAX Maximum feedrate per tooth permitted during DMC. Value programmed in the FZMAX command of the #DMC ON instruction. Units: Millimeters/tooth or inches/tooth.	R	R	R
(V.)[ch].G.DMCON DMC status. Units: -.	R(*)	R	R
(V.)[ch].G.LEARNEDPWRSP Target power calculated for DMC in the learning phase (percentage of rated power). Units: Percentage.	R(*)	R	R
(V.)[ch].G.DMCACTPWR Active power on the spindle, measured by DMC (percentage of the rated power). Units: Percentage.	R(*)	R	R
(V.)[ch].G.DMCOVR Feedrate percentage (feed override) calculated by DMC. Units: Percentage.	R(*)	R	R
(V.)[ch].G.DMCFZ Feedrate per tooth, calculated by DMC. Units: Millimeters/tooth or inches/tooth.	R(*)	R	R
(V.)[ch].G.DMCNOLOADPWR Spindle power without a load, measured by DMC. Units: Kilowatts.	R(*)	R	R
(V.)[ch].G.DMCSAVEDTIME Time saved by the DMC action. Units: Seconds.	R(*)	R	R

(*) The CNC evaluates the variable during execution (it stops the block preparation).
(**) The CNC evaluates the variable during execution, but it stops the block preparation.

20.3 Define the tool data.

The number of teeth on the tool must be defined, either from the tool table or by using variables, so that the CNC can limit the feedrate per tooth between the minimum and maximum defined by the instruction #DMC ON. This information is only available for "Milling", "Drilling", "Surface milling", "Reaming" and "Other" type tools.

Tool table.



(A) Tool type.
(B) Number of teeth (between 1 and 100).

Variables.	Description.
(V.)[ch].G.NCUTTERS	Tool being prepared. Number of teeth. Units: Teeth.
(V.)[ch].TM.NCUTTERS[offset]	[Offset] of the active tool. Number of teeth. Units: Teeth.
(V.)TM.NCUTTERST[tool][offset]	[Offset] of the [tool]. Number of teeth. Units: Teeth.

20.4 Operating with DMC.

20.4.1 DMC operation.

After executing the instruction #DMC ON, the CNC will activate DMC as long as the spindle has a digital drive system, is turning in M3 or M4 and has reached the programmed turning speed (REVOK mark).

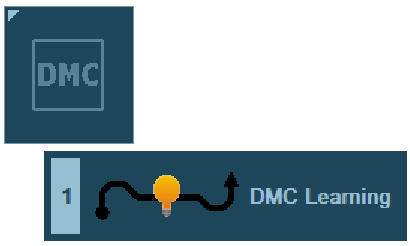
Calculating the no-load power.

The first time that this function is executed, the CNC stops the feedrate of the axes until the spindle reaches the programmed turning speed and has stabilized. Next, and with the axes stopped, the CNC measure the power consumed by the spindle without a load (without machining). The entire process to measure the no-load power can be completed in just a few seconds, during which time the CNC stops the feedrate of the axes.

Knowing the no-load power enables the CNC to detect the part entries and exits during the machining.

Learning phase.

Every time #DMC ON is programmed without target power (PWRSP command), DMC determines it through a learning phase that it initiates automatically. Once said value is obtained, the normal DMC operation will begin.



The learning phase can be repeated at any time, while the DMC is active, by pressing the "DMC learning" softkey in the automatic mode. After pressing the softkey, while the DMC is active, the next entry to the part will begin a learning phase, whether or not the PWRSP had been programmed while activating the DMC.

With the axes in movement, the learning phase begins once DMC detects the entry into the part. DMC waits for the feedrate to reach the programmed value, and during the movement of the axes, it calculates the target power ("power consumed" – "no-load power"). The learning phase has a one minute duration, starting from when the tool enters the part at a distance equal to the radius. If the tool exits the part, it ceases counting the time until the tool re-enters the piece.

DMC will write the target power value obtained in the learning phase in the variable (V.)G.LEARNEDPWRSP, so that it can be used on the following parts that are machined with the same part program, avoiding the repetition of the learning phase.

It is recommended to perform the learning phase with a depth of pass as close as possible to that which will be used during the machining; otherwise, if the depth of pass is shallower, the target power calculated may not be appropriate.

DMC operation.

Once the target power is known, and after detecting the entry into the part, DMC will begin to function as normal. During the machining, DMC adapts the feedrate so that the cutting power ("power consumed" – "no-load power") is as close as possible to the target power. DMC adapts the feedrate by changing the override. See *"20.4.3 Percentage of feedrate (feedrate override)."* on page 782.

Processing of part entries and exits.

DMC detects the part entries and exits and carries out a special feedrate override process so that said transitions are smooth and do not damage the tool. For entries into the part, DMC use a 75% feedrate until the tool enters the part along a length equal to the tool radius. The CNC also attempts to optimize the time on the paths without a load.

The target power and the part entries and exits.

To ensure the proper operation of the DMC, the target power (programmed or obtained by learning) must be at least 20% of the no-load power. If the DMC detects this type of situation, it will display a 3103 warning.

- If the actual values of the no-load power and the target power during machining are similar, it is likely that the DMC cannot accurately distinguish the part inputs and outputs or it may detect false inputs or outputs. In this case, it is recommended to check the value of the target power.
- Programming a higher target power, which is not actually going to be achieved, can mean that the DMC never detects part inputs and performs all the machining as if the tool were working in a vacuum.

For both these reasons, if the actual target power is less than 20% of the no-load power, it is recommended to deactivate the DMC function during this machining process.

Monitoring of consumed power.

During operation, DMC continuously monitors the power consumed by the spindle to detect any problems with the tool or in the machining.

Collision detection.

If the instantaneous power exceeds the target power in a predefined range, the CNC considers that a collision has occurred. In this case, the CNC displays warning 3101, stops the feedrate of the axes, while keeping the spindle turning (this behavior is the equivalent of having pressed the [STOP] button). After verifying the cause of the warning, the user can continue the machining (pressing [START]), enter in tool inspection or terminate the execution to replace the damaged tool.

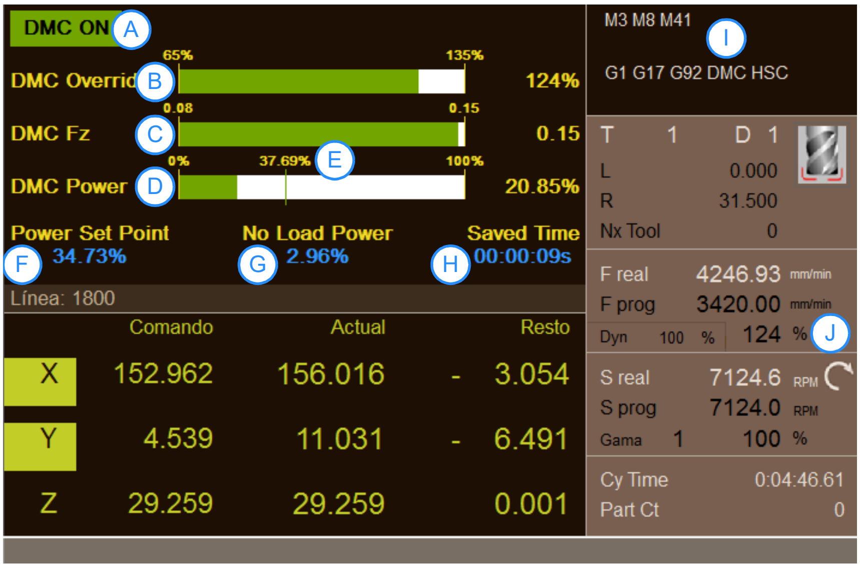
Worn tool. Excessive continuous power consumption.

If DMC detects excessive power consumption during a certain amount of time, it understands that the tool is worn or deteriorated and displays the warning 3100, without stopping the execution. The user decides if it is necessary to stop the execution.

20.4.2 Automatic mode. DMC status and progress.

During the execution of a program while DMC is active, the automatic mode can display the status and progress of this function; to do this, on the “Display” softkey of the horizontal menu, select the option "DMC". To return to the standard screen in automatic mode, select the “Standard” option on the same softkey. The data displayed on the DMC page are the same as on the standard page, except for those for DMC itself (in the upper left corner).

Automatic mode. DMC status and progress.

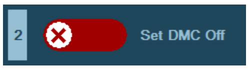


- **A** DMC status; ON/OFF.
- **B** DMC override. The consumption bar limits will be those defined in the machine parameters. The override applied by DMC is shown on the right.
- **C** Feed per tooth. The consumption bar limits are the values programmed in the instruction #DMC ON (FZMIN and FZMAX commands); if they have not been programmed, the CNC does not display this bar.
- **D** Power consumed by the spindle (defined as the percentage of rated power).
- **E** Power consumed by the spindle under ideal conditions;
  "target power" + "no-load power".
- **F** Target power or ideal cutting power (defined as a percentage of rated power); learned power or power programmed in the instruction #DMC ON (PWRSP command).
- **G** No-load power (defined as a percentage of the rated power).
- **H** Time saved by the DMC action.
- **I** When DMC is active, the program history will display "DMC" and will not change, even though the CNC performs incompatible functions that temporarily deactivate DMC.
- **J** Final override (DMC override + switch override).

Vertical softkey menu.

Softkey.	Meaning.
	**DMC options (dropdown menu).** • Repeat the DMC learning phase. • Deactivating the DMC.

DMC options.

Softkey.	Meaning.
	This softkey allows the DMC learning phase to be repeated at any time, as long as the DMC is active.
	This softkey allows the DMC to be deactivated.
	This softkey allows the DMC to be reactivated.

20.4.3 Percentage of feedrate (feedrate override).

The feedrate (feed override) percentage can be set, from highest to lowest priority, by the program (variable V.G.PRGFRO), by the PLC (variable V.PLC.FRO) or from the switch on the operator panel. The value set by the program has the highest priority, while the values set from the switch on the operator panel have the lowest priority.

- The percentage set by the program or by the PLC has a higher priority than that set by DMC; both percentages inhibit DMC. To cancel the feedrate percentage set by program or PLC, define its variables with a value of 0 (zero). See *"20.4.3 Percentage of feedrate (feedrate override)."* on page 782.
- The percentage set by DMC affects the percentage set from the switch.

$$\text{Override}(\%) = \frac{\text{OverrideDMC}(\%) \times \text{OverrideJOG}(\%)}{100}$$

- DMC can change its override within the limits set by the instruction #DMC ON (commands OVRMIN and OVRMAX); if they are not defined, the limits will be set by the machine parameters MINDMCOVR and MAXDMCOVR. The final override (overrideDMC + overrideJOG) may exceed these limits.
- The CNC always abides by the maximum limit set in the machine parameter MAXOVR.
- If the user selects an override with the switch on the operator panel that is lower than MINDMCOVR, the CNC inhibits DMC (but does not deactivate it); when the override once again exceeds MINDMCOVR, DMC will once again operate as normal.
- The CNC abides by the minimum and maximum feedrate set by the tool in the instruction #DMC ON (commands FZMIN and FZMAX).

20.4.4 Machining analysis.

This section shows an image of the automatic mode and a screenshot of the oscilloscope taken during a machining with DMC.

Data for the machining.

Data.	Description.
Material.	Material: Aluminum. Kc: 350 N/mm².
Tool.	Diameter: Ø 63 mm. Number of teeth: Nz = 4. Feed per tooth: Fz = 0.12 mm/tooth. Depth of radial pass: Ae = 35 mm. Depth of axial pass: Ap= 1.5 mm.
Cutting conditions.	Cutting (surface) speed: Vc = 1410 m/min. Turning speed: S = 7124 RPM. Feedrate: F = 3420 mm/min.

Machine parameters.

Machine parameter.	Description.
MINDMCOVR	Minimum DMC override of all channel axes. Value: 65 %.
MAXDMCOVR	Maximum DMC override of all channel axes. Value: 135 %.

Programming.

Instruction to activate DMC.

```
#DMC ON [FZMAX 0.15, FZMIN 0.08]
```

- The target power is not programmed; DMC executes the learning phase.
- The minimum feedrate is limited to 0.08 mm/tooth; with the F and S cutting conditions programmed, this is equal to 67% minimum override.
- The maximum feedrate is limited to 0.15 mm/tooth; with the F and S cutting conditions programmed, this is equal to 125% maximum override.

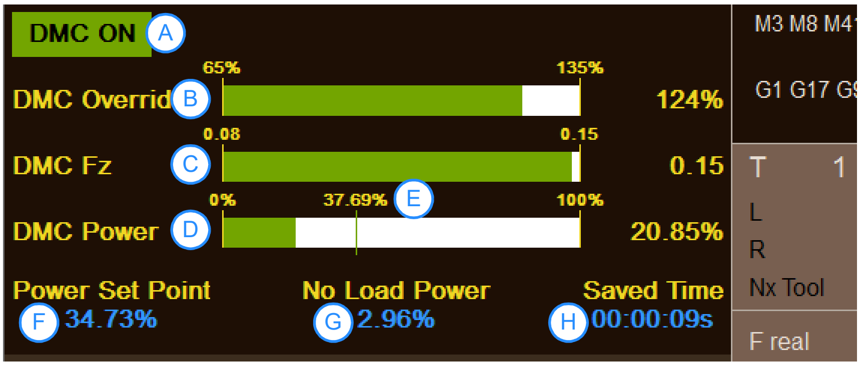
Execution.

Measurement of no-load power and learning phase.

- No-load power. 2.96 %
- PowerSetPoint (learned target power). 34.73 %

Automatic mode information.

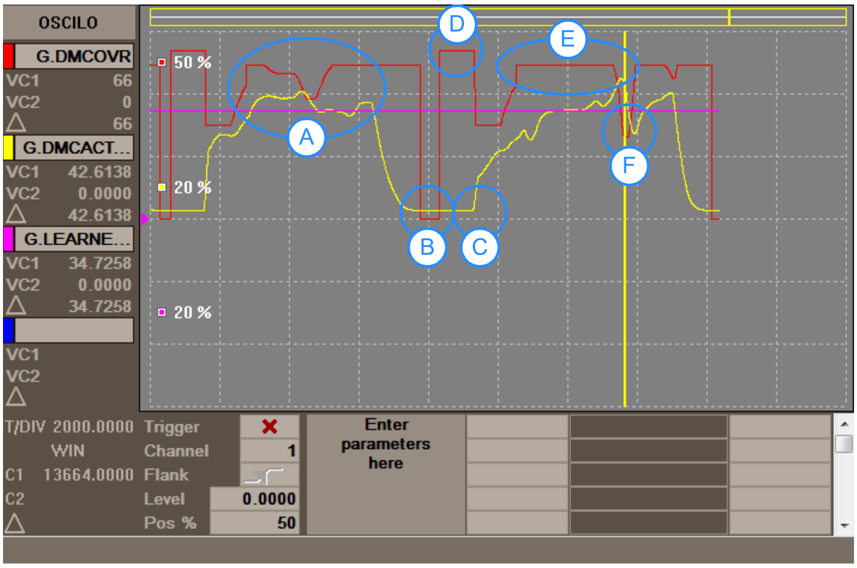
Automatic mode. DMC status and progress.



- A DMC status; ON/OFF.
- B DMC override; minimum = 65 %, maximum = 135, active override = 124 %.
- C Feedrate per tooth; minimum 0.08, maximum 0.15, active = 0.15 mm/tooth.
- D Power consumption; instantaneous consumption = 20.85%
- E Power to reach ideal conditions; 37.69% "target power" + "no-load power".
- F Learned target power; 34.73%.
- G No-load power; 2.96%.
- H Time saved by the DMC action.

Oscilloscope screenshot.

Oscilloscope.



- A Override control for power optimization.
- B Machining in G00.
- C Entry into the part.
- D No-load movement; maximum override (machine parameter MAXDMCOVR).
- E Machining at the maximum feedrate per tooth.
- F Machining at the minimum feedrate per tooth.

20.5 Error messages (cause and solution)

2427 #DMC ON [<PWRSP value><,OVRMIN value><,OVRMAX value><,FZMIN value><,FZMAX value>]

DETECTION	During editing and execution.
CAUSE	The syntax of the instruction is wrong.
SOLUTION	Check the syntax of the instruction in the programming manual.

2428 #DMC OFF

DETECTION	During editing and execution.
CAUSE	The syntax of the instruction is wrong.
SOLUTION	Check the syntax of the instruction in the programming manual.

2429 Parameter DMC out of range

DETECTION	During editing and execution.
CAUSE	A command in the instruction #DMC has an invalid value.
SOLUTION	Check the syntax of the instruction in the programming manual.

2430 #DMC ON: FZMIN must be less than FZMAX.

DETECTION	During execution.
CAUSE	For the function #DMC, the minimum feedrate is greater than the maximum feedrate.
SOLUTION	The minimum feedrate (FZMIN) must be less than the maximum feedrate (FZMAX).

2431 #DMC ON: OVRMIN must be less than OVRMAX.

DETECTION	During execution.
CAUSE	For the function #DMC, the minimum override is greater than the maximum override.
SOLUTION	The minimum override (OVRMIN) must be less than the maximum override (OVRMAX).

2432 #DMC ON: Programming not allowed with analog spindle.

DETECTION	During execution.
CAUSE	The CNC has attempted to activate DMC on an analog spindle.
SOLUTION	DMC can only be activated on digital spindles.

2433 ON/OFF must be programmed.

DETECTION	During execution.
CAUSE	The command ON/OFF is not programmed to activate or deactivate the function.
SOLUTION	Check the syntax of the instruction in the programming manual. Include the command ON to activate the function (for example, #DMC ON [...]) or the OFF command to cancel it (for example, #DMC OFF).

2458 #DMC ON: The master spindle must use the parameter DMCSPDL = Yes.

DETECTION	During execution.
CAUSE	An attempt was made to activate the DMC for a spindle that does not have permission to do so.
SOLUTION	To activate the DMC on a spindle, then it must have the machine parameter set to DMCSPDL=Yes.

3100 Excessive power on the spindle detected during DMC.

DETECTION	During execution.
CAUSE	DMC has detected excessive power consumption (drive variable TV51) during a period of time. This can indicate that the tool is worn or deteriorated.
SOLUTION	Check the tool status.

3101 Excessive power on the spindle detected during DMC. Press START to continue.

DETECTION	During execution.
CAUSE	DMC has detected that the instantaneous power (drive variable TV51) has exceeded the target power in a preset range. This can indicate that the tool has collided with the part, and so the CNC stops the axes, keeping the spindle turning.
SOLUTION	Check the status of the tool and the machining. The CNC allows the execution to be resumed by pressing [START], to enter in the tool inspection mode or to finish the execution.

3103 Target power insufficient for the proper operation using DMC.

DETECTION	During execution.
CAUSE	The values of the no-load power and the target power during machining are similar. It is possible that the DMC cannot accurately distinguish the part inputs and outputs.
SOLUTION	The target power (programmed or obtained by learning) must be at least 20% of the no-load power. In this case, it is recommended to deactivate the DMC function for this machining process. Programming a higher target power, which is not actually going to be achieved, can mean that the DMC never detects part inputs and performs all the machining as if the tool were working in a vacuum.

HD GRAPHICS.

21

21.1 Configure the graphics for the rotary axes cycles (·M· model).

In the F3D graphics, the machine configuration (xca file) must be loaded that corresponds to the defined axis configuration. The xca files may be loaded either from the softkey menu or from the program using the instruction #DEFGRAPH.

Defining the machine.

The configuration files for machines with an indexer are In the folder../Fagor/Grafdata/Machines/Mill. All the configurations are designed for a channel; besides the rotary axis of the table (the indexer), they have two other rotary axes on the spindle.

File.	Machine type.
Mill A@table B,C@spindle.xca	Machine with the indexer in X.
Mill B@table A,C@spindle.xca	Machine with the indexer in Y.
Mill C@table A,B@spindle.xca	Machine with the indexer in Z.

The default configuration of the axes in the xca files is as follows.

Axis 1:	X	Axis 4:	Rotary axis of the indexer.
Axis 2:	Y	Axis 5:	First rotary axis of the spindle.
Axis 3:	Z	Axis 6:	Second rotary axis of the spindle.

If the default configuration and the real configuration of the machine axes are different, a second file must be edited (with the same name and def extension), with the correct association between the two CNC axes and those defined in the xca file. In the folder of each machine is a sample def file. If they are going to be copied to another folder, both files must be copied (xca and def).

Defining the part.

Each one of these xca files can define and use four parts on the table. To use these parts, the xca file must be modified to add the center of the indexer as an offset with regard to the machine zero. To modify these files, copy both files (xca and def) in the folder ../Mtb/Grafdata o ../Users/Session/Grafdata and change it indicated below.

For cylindrical parts, their axis of rotation must be the same as for the indexer.

Example.

For a rotary axis, A is the file "Mill A@table B, C@spindle.xca", in which the axis offset is defined as {translation = "0 0 0"}.

```
<Connection plug="stdPlug@SYS-Tool1" socket="stdSocket@S1"
rotation="0 1 0 -90"/>
<!-- Position of the center of A axis referred to zero machine -->
<Connection plug="stdPlug@A1" socket="stdSocket@home"
translation="0 0 0"/>
```

The default value "0 0 0" must be replaced with the correct values, for example "-300 -100 -250".

```
<Connection plug="stdPlug@A1" socket="stdSocket@home"
translation="-300 -100 -250"/>
```

Positioning the tool.

The graphics position the tool according to the front or longitudinal axis defined in the machine parameters of the linear axes (FACEAXIS and LONGAXIS), and according to the option selected in the tool table.

If the machine parameters LONGAXIS or FACEAXIS change, the CNC must be restarted to that the graphics take on the new configuration.

If the parameters have not been set for the axes, the default position of the tool in the different machine configurations will be as follows:

File.	Tool default position.
Mill A@table B,C@spindle.xca	FACEAXIS = Z LONGAXIS =X
Mill B@table A,C@spindle.xca	FACEAXIS = Z LONGAXIS =X
Mill C@table A,B@spindle.xca	FACEAXIS = Z LONGAXIS =X

ETHERCAT BUS.

22

For a remote I/Os connectors or third-party drives, first acquire the corresponding software option.

EtherCAT bus based on ENI (EtherCAT Network Information) files allowing the CNC to communicate with remote I/Os and with CiA 402 compliant drives. EtherCAT is configured as an inline topology, where the CNC is the master and the other elements (remote I/Os, drives, etc.) are slaves, which are distributed by nodes (up to 32). Fagor supplies a configurator and mapper to create the files needed to configure and manage the bus resources. Files created by both programs must be copied to the CNC's ../Mtb/Data folder.

The "KPA EtherCAT Studio" configurator defines the bus topology and creates the ENI file with the bus slave information. The bus master interprets this file and initializes the slaves connected to the bus.

From the ENI file generated by the configurator, the "Fagor EtherCAT Mapper" application maps the bus resources to the PLC resources (inputs, outputs and registers) and creates the file with the mapping information. The bus master uses this file to exchange information between the bus resources and the PLC.

Resources available in the bus.

Resources of the bus.	Total in the bus.
Digital inputs.	1024
Digital outputs.	1024
General purpose analog inputs.	40
General purpose analog outputs.	40
Analog inputs for Pt100 temperature sensors.	10

CAN bus compatibility

This bus is compatible with the CAN bus, responsible for the management of keyboards and operator panels. The CNC can have remote I/Os on both buses simultaneously (CAN/EtherCAT), provided that both buses do not use the same PLC resources.

Cycle time of the bus.

The cycle time of the EtherCAT bus is defined in the ETHERCATTIME parameter of the CNC.

- This parameter must be greater than or equal to the speed loop time (SYSTEMTIME).
- This parameter may be less than the interpolator cycle time (CNCTIME), e.g., for laser devices connected to the bus.
- The position loop time for EtherCAT axes (POS_LOOPTIME) must be a multiple of this parameter.
- The cycle time for the EtherCAT bus, defined in the configurator, must be equal to this parameter.

22.1 FagorECLite configurator (Fagor EtherCAT Configurator Lite).

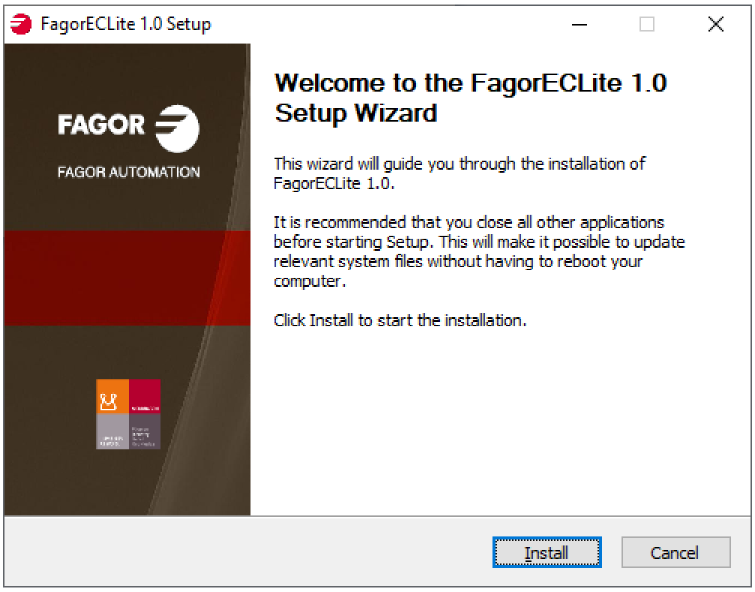
FagorECLite (Fagor EtherCAT Configurator Lite) is a simple licence-free application for generating configuration and resource mapping files for Fagor EtherCAT devices; controllers and I/O modules. FagorECLite generates the configuration file and the resource mapping file. For configurations with third-party EtherCAT modules, it is necessary to use KPA EtherCAT Studio (licensed) and FagorEtherCATMapper.

FagorECLite is compatible with the Quercus and Optima systems (v02.01 or higher).

22.1.1 Installation.

To install the application, double click on the FagorECLite_x.x.exe file (where x.x is the version) and follow the instructions. The application is installed in the folder

C:\Program Files (x86)\Fagor Automation\FagorECLite



22.1.2 Generation of configuration files.

Generation of configuration and mapping files.

Start the FagorECLite.bat file (double click or via another application) and describe the configuration.

1. Type of CNC.
2. Cycle time, which must be the same as the ETHERCATTIME parameter of the CNCelite (only available for Quercus and Optima system).
3. Number of BCSD controllers (only available for Optima system).
4. Number of digital RIOR modules (RIOR-E-48I32O).
5. Number of RIOR analogue modules (RIOR-E-48I32O-ANALOG).

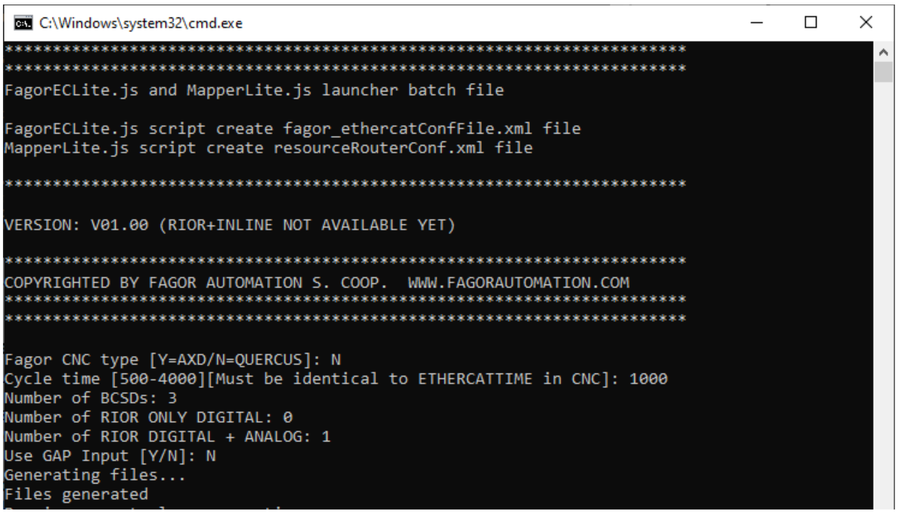
6 Number of RIOW Inline nodes (only available if there are no RIOR modules). For each node, indicate the number of modules and the type, in this order.

Inline module.	Type.
IB IL 24 DO 8, /HD, 2A, /HD-ECO, FLM DO 8 M12	8
IB IL 24 DO 32	10
IB IL 24 DI 8, /HD, /T2, /HD-ECO, FLM DI 8 M12	3
IB IL 24 DI 32	5
IB IL AO 2/UI	22
IB IL AI 4/I, /U	16
IB IL AI 4/I/4-20-, U/0-10-, RTD 4/PT100-, /PT1000-, UTH 4/J-, /K-, /L-ECO	15

7 Input for laser gap control. The RIOR module number and the analogue input used must be indicated.

8 The application creates the files in its main folder. Each file has a suffix describing the configuration.

Example for a configuration of three BCSD controllers, one RIOR-E-48I32O-ANALOG (without gap control) and 1000µs cycle time.



The generated files are called.

- fagor_ethercatConfFile_QC_CYCLE_TIME_1000_BCSD_3_RIOR_ANA_1.xml
- resourceRouterConf_BCSD_3_RIOR_ANA_1.xml

Copying the files to the CNC.

1 Close the CNC.

2 Copy the files to the CNC, in the ..MTB\Data folder, and delete the suffixes describing the configuration.

- fagor_ethercatConfFile.xml

- resourceRouterConf.xml

3 Start the CNC. The CNC loads the new configuration automatically.

22.1.3 Resource mapping.

The application maps module resources (inputs, outputs, etc.) to PLC resources in the following order.

- BCSD drives.

  The mapping starts at register R800 (Drive Torque). Each regulator uses one register.

- RIOR-E modules.

  Mapping starts at register R832 (AI, PT100, AO), DI1, DO1 and PT100OFF1. Each module uses 48 DI, 32DO, 8 registers and two Pt100 probe disable signals.

- RIOR-E-Inline modules.

  Mapping starts at the first available register, from R800 (AI, PT100, AO), DI1, DO1 and PT100OFF1. The resources used depend on the modules included.

Example of mapping; 2 BCSD + 2 RIOR-E-48I32O-ANALOG.

Product.	Module resources.		PLC resource.	Description.
BCSD-1	IN0	(1)	R800	Drive Torque.
BCSD-2	IN0	(1)	R801	Drive Torque.
RIOR-1	DI DO AI PT100 AO DIS_PT100	(48) (32) (2) (2) (4) (2)	I1..I48 O1..O32 R832..R833 R834..R835 R836..R839 PT100OFF1..PT100OFF2	Digital inputs. Digital outputs. Analog inputs. PT100 inputs. Analog outputs. Disable PT100.
RIOR-2	DI DO AI PT100 AO DIS_PT100	(48) (32) (2) (2) (4) (2)	I49..I96 O33..O64 R840..R841 R842..R843 R844..R847 PT100OFF3..PT100OFF4	Digital inputs. Digital outputs. Analog inputs. PT100 inputs. Analog outputs. Disable PT100.

Example of mapping; 2 BCSD + 1 RIOW-E-Inline.

Product.	Module resources.		PLC resource.	Description.
BCSD-1	IN0	(1)	R800	Drive Torque.
BCSD-2	IN0	(1)	R801	Drive Torque.
RIOW-3	DI DO AO AI	(32) (32) (4) (4)	I1..I32 O1..O32 R802..R805 R806..R809	Digital inputs. Digital outputs. Analog outputs. Analog inputs.

22.2 "KPA EtherCAT Studio" configurator.

Fagor supplies the configurator "KPA EtherCAT Studio" (by Koening-pa GmbH) under license to define the bus topology and create the ENI file containing the bus slave information. This file must be copied to the folder ../Mtb/Data of the CNC with the name "fagor_ethercatConfFile.xml". This file also uses the "Fagor EtherCAT Mapper" program for determining the resources available in the bus.

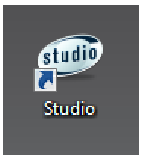
The configurator can work both in online and offline modes; for either case a license is required.

- In online mode, the configurator is connected to the EtherCAT bus and scans it in order to detect the modules present in the bus. Connect the PC to the first slave using an Ethernet cable.
- In offline mode, the configurator does not access the EtherCAT bus. The user should manually build the bus topology from the modules stored in the program library.

Fagor delivers a pendrive with the installer and the configurator license "KPA EtherCAT Studio". This license is linked to the pendrive; the user will be able to install the configurator on various computers, but will need to connect the pendrive to use it.

22.2.1 Installing the EtherCAT configurator.

Install the configurator in a PC with Windows 7 32/64 bits or Windows 10 64 bits. Insert the pendrive into the USB port and double click on the file "SetupKPA_xx_xx.exe" (where xx_xx indicates the version) to begin the installation. Follow the instructions on the screen to complete the process. Once installed, the following icons will appear on the desktop.

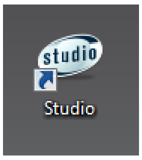
Icon.	Function.
	**Studio.** EtherCAT Configurator.
	**Master (Win64).** Access to the EtherCAT modules through the Ethernet network.
	**KPA Licensing utilities.** Access to the user license.

Help.

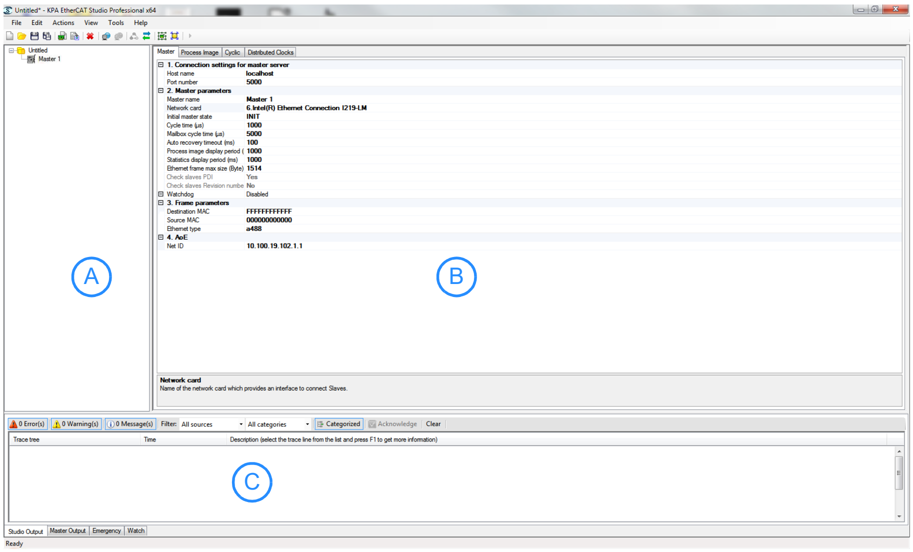
For detailed information on the program operation, refer to the help for the program.

22.2.2 Run the configurator.

1 For an online configuration of the bus, run the "Master (Win64)" program". This program can be used to access the bus through the LAN Ethernet connector. When the program is running, an icon will appear in the task bar notification area (next to the clock). For an offline configuration of the bus, this program is not necessary.



2 Execute the "Studio" program to run the "KPA EtherCAT Studio" configurator.



A EtherCAT Topology.

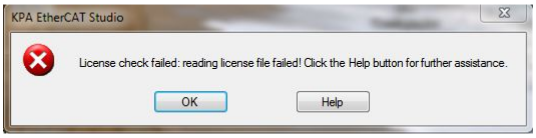
B Help information.

C Program errors, warnings and messages.

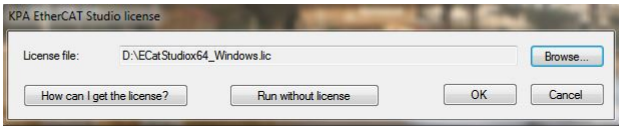
22.2.3 Problems with the license.

License error on starting up the program.

The pendrive with the license must be connected to the PC. If the following error occurs on start up of the program, it means that the program cannot find the license. Insert the pendrive into the USB port and click “OK”.

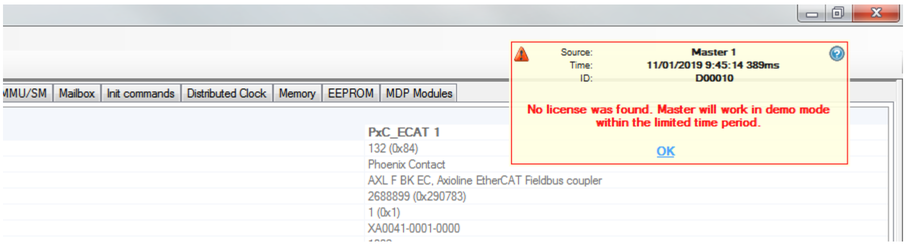


Click the "Browse" button to select the unit where the pendrive is connected and click the "OK" button again.



License error while detecting the modules.

Ignore this warning. Fagor supplies the "Basic" license, which may be used to configure the bus. Some options in the configurator are not available with this license.

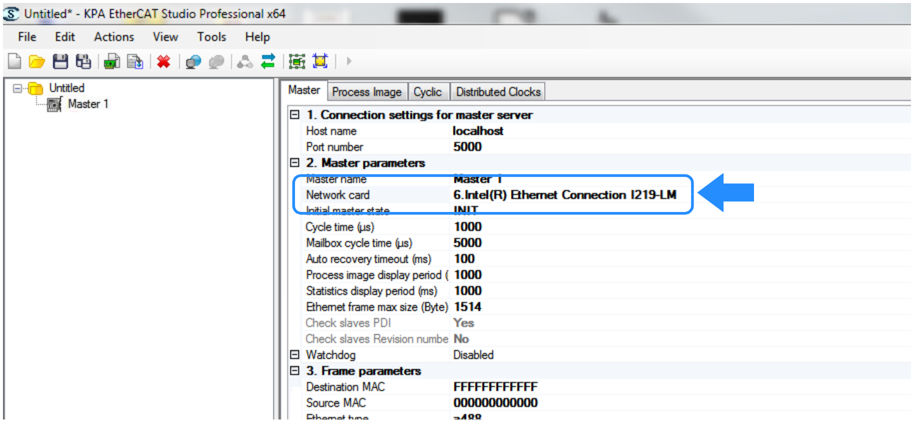


22.3 "KPA EtherCAT Studio" configurator. Configuring a topology with remote I/Os.

22.3.1 Configuring the EtherCAT bus.

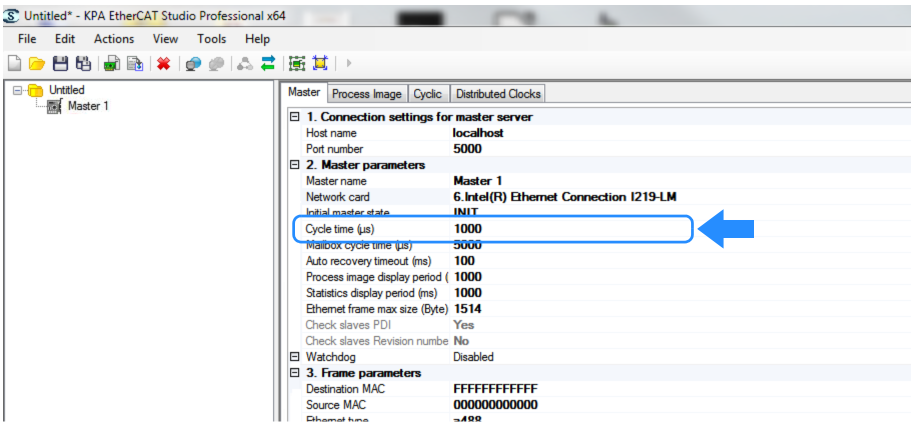
Selecting the Ethernet network adapter for online mode.

Select the network adapter where the EtherCAT bus is connected.



Configuring the bus cycle time.

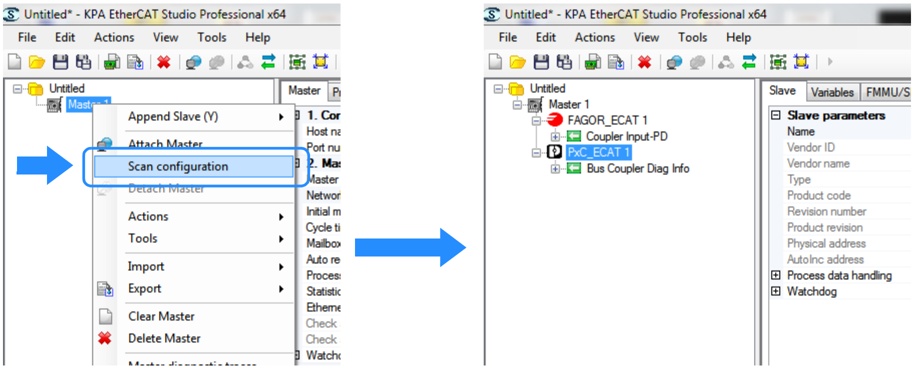
The bus cycle times must coincide with the CNC position loop time (CNCTIME parameter).



22.3.2 Online mode configuration. Scanning the bus typology.

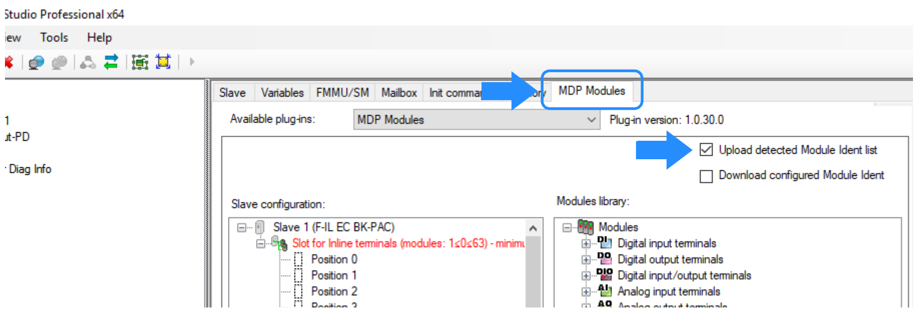
Scanning the bus to detect the headers.

To detect the headers, place the mouse pointer over the topology master (left window), click the right mouse button and select the "Scan Configuration" option. The configurator shows the detected headers.



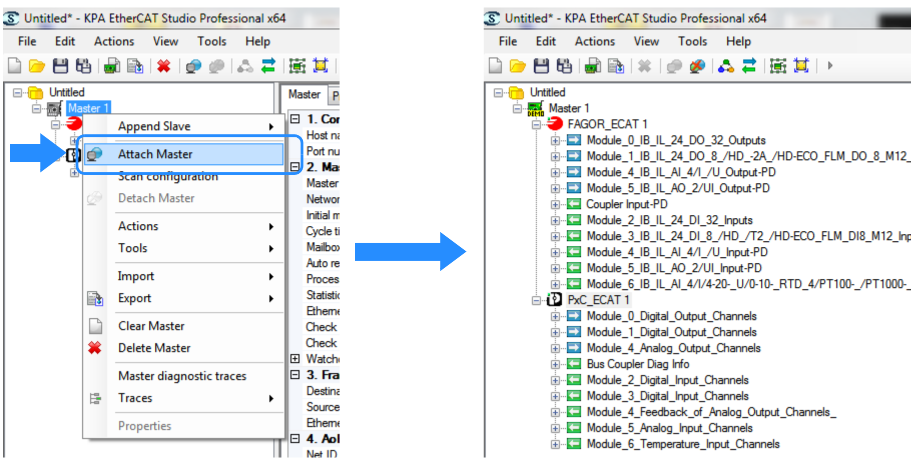
Scanning the bus to detect the resources of each node.

In the "MDP modules" tab, select the option "Upload detected Module Ident list". This option allows the headers to read the list of identifiers for the other modules.



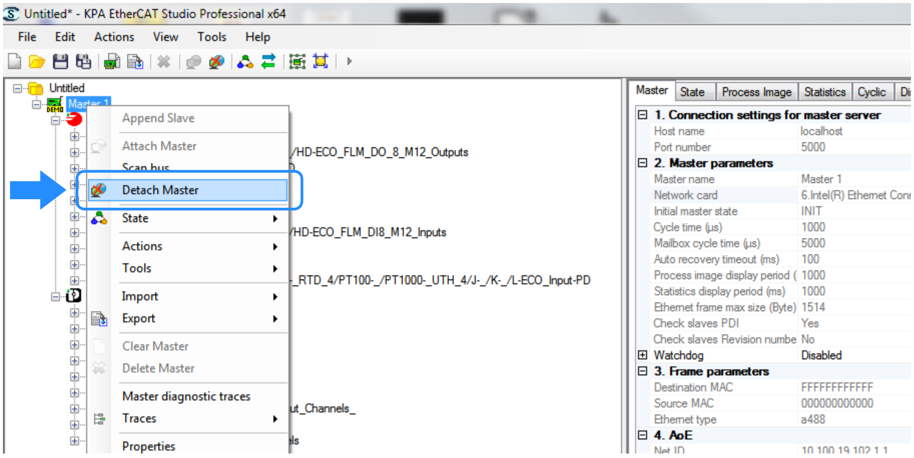
To detect the resources of each node, place the mouse pointer over the topology master (left window), click the right mouse button and select the "Attach master" option. The configurator shows the detected modules.

NOTE: It is recommended to display the information of the headers to facilitate the detection of the modules.



Decoupling the bus master.

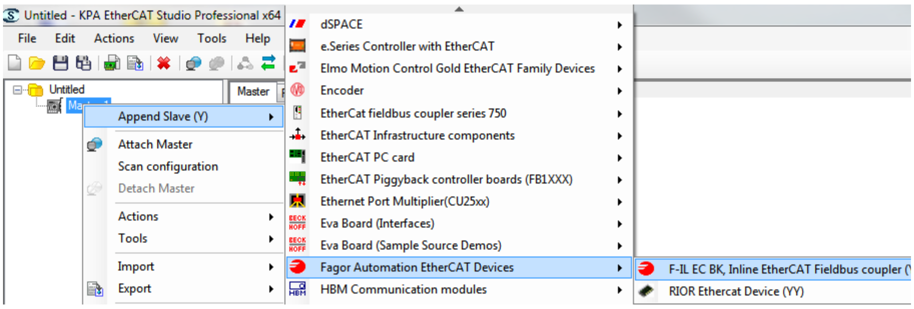
A preliminary step for exporting the configuration is to decouple the bus master. To do this, place the mouse pointer over the topology master (left window), click the right mouse button and select the "Detach master" option.



22.3.3 Offline mode configuration. Build the bus topology manually.

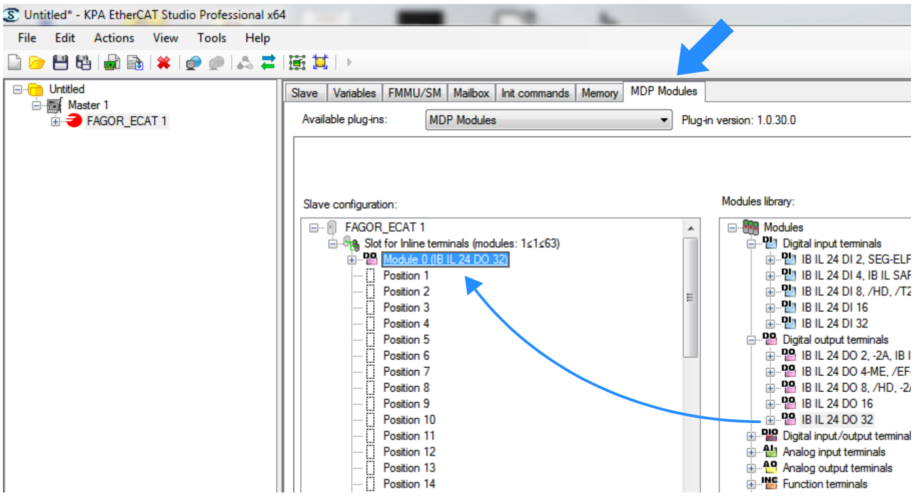
Setting the bus headers.

To add headers, place the mouse pointer over the topology master (left window), right click the mouse and select the "Append Slave" option. The configurator shows the available devices.



Setting the modules for each node.

In the "MDP modules" tab there is the list of available modules for the selected header. Drag the modules from the right window to the left window.

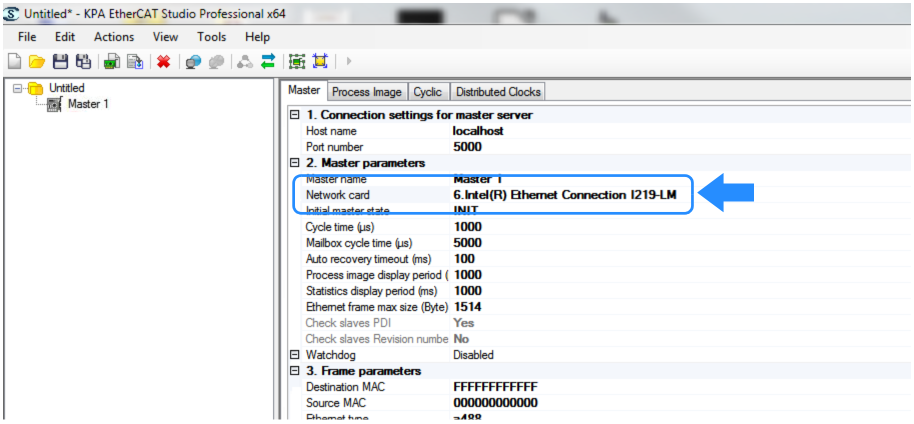


22.4 "KPA EtherCAT Studio" configurator. Configuring a topology with drives.

22.4.1 Configuring the EtherCAT bus.

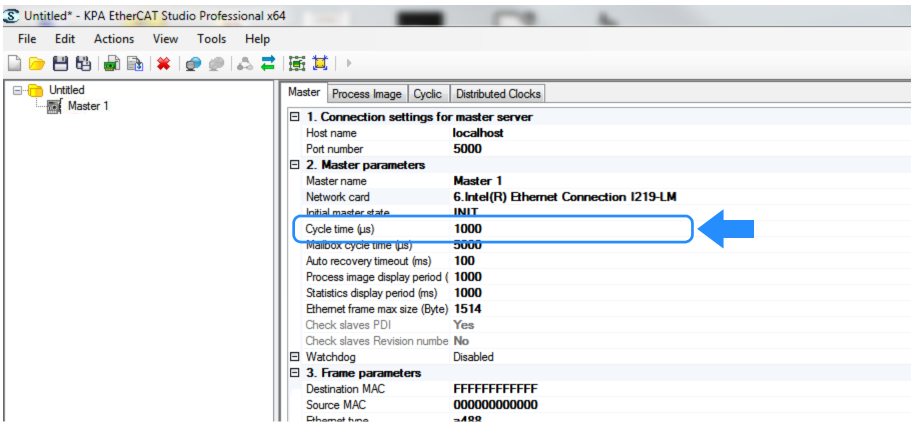
Adaptor selection for the Ethernet network.

Select the network adapter where the EtherCAT bus is connected.



Configuring the bus cycle time.

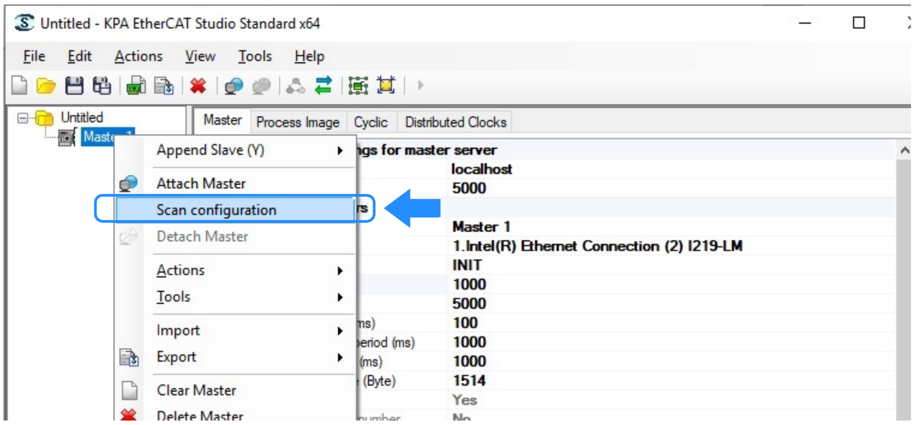
The bus cycle times must coincide with the CNC position loop time (CNCTIME parameter).



22.4.2 Configuring the topology master.

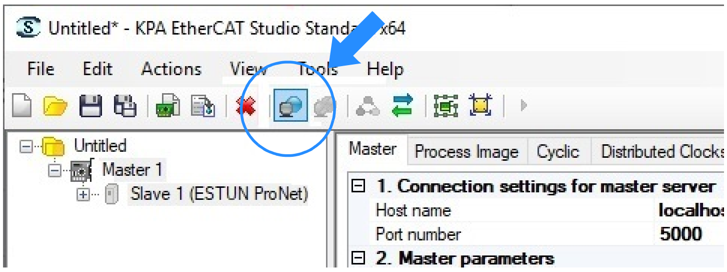
Scanning the bus to detect the headers.

To detect the headers, place the mouse pointer over the topology master (left window), click the right mouse button and select the "Scan Configuration" option. The configurator shows the detected headers.



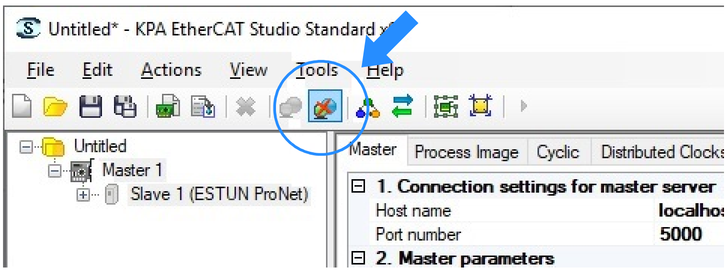
Scanning the bus to detect the resources of each node.

To detect the resources of each node, move the mouse over the topology master (left window), click the right mouse button and select the “Attach master” option; or use the following button. The configurator shows the detected modules.



Decoupling the bus master.

A preliminary step for exporting the configuration is to decouple the bus master. To do this, move the mouse over the topology master (left window), click the right mouse button and select the “Detach master” option; or use the following button.



22.4.3 Configure the slaves (variables of the cyclic channel).

Variables to configure.

For CiA 402 compliant drives, the following cyclic channel variables must be configured at the least.

- Variables sent by the CNC to the drive.

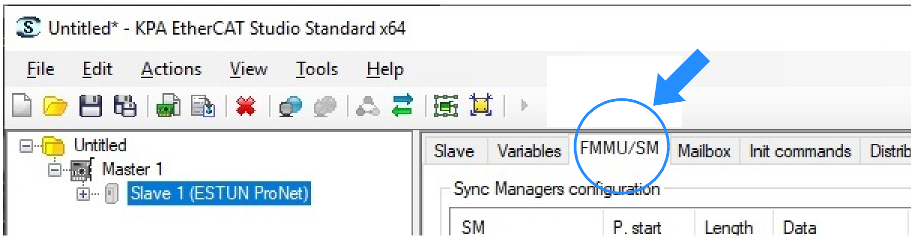
0x6040.0	Control word.
0x60FF.0	Target velocity.
0x6060.0	Modes of operation.

- Variables read by the CNC from the drive.

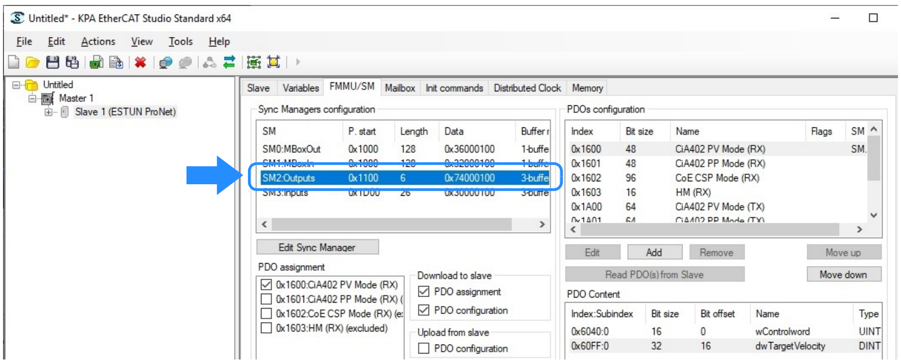
0x6041.0	Status word.
0x6064.0	Position actual value.

Variables sent by the CNC to the drive.

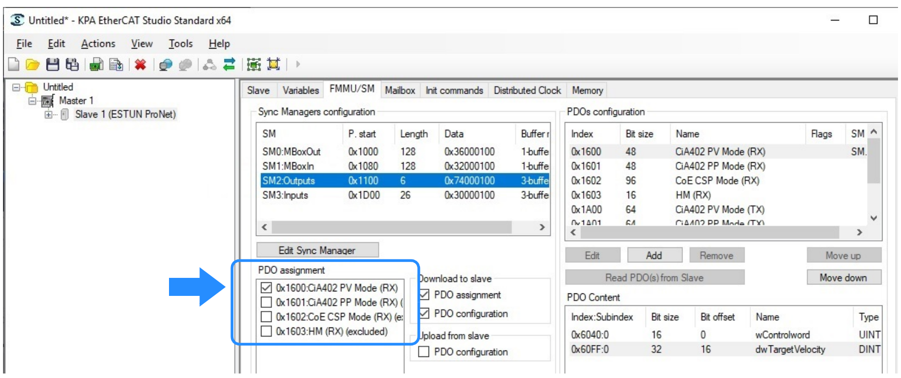
1 Move the mouse over a slave and select the FMMU/SM tab.



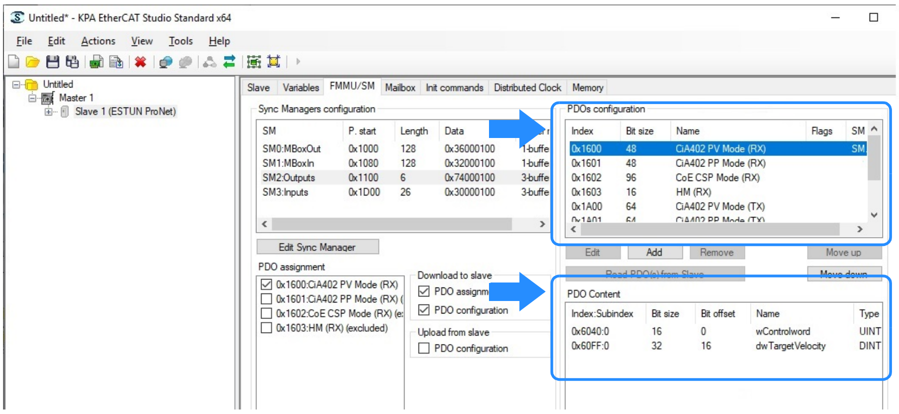
2 To configure the variables sent by the CNC to the drive, click on the field SM2:outputs



3 The lower window shows the PDO allocation and can be used to select which ones the CNC will send.

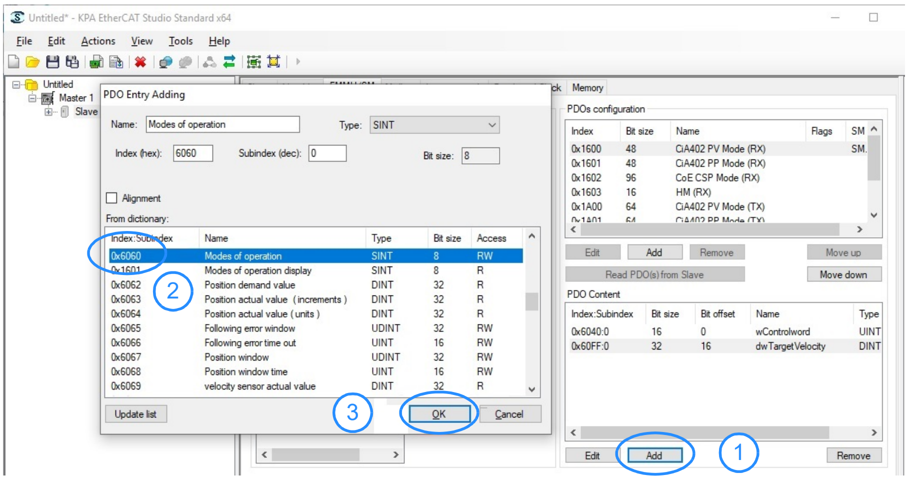


4 The upper right window displays the various PDOs and the lower window displays the content of the selected PDO. This window is used to edit the PDO, by cyclically adding variables to send to the drive or removing the unneeded variables.

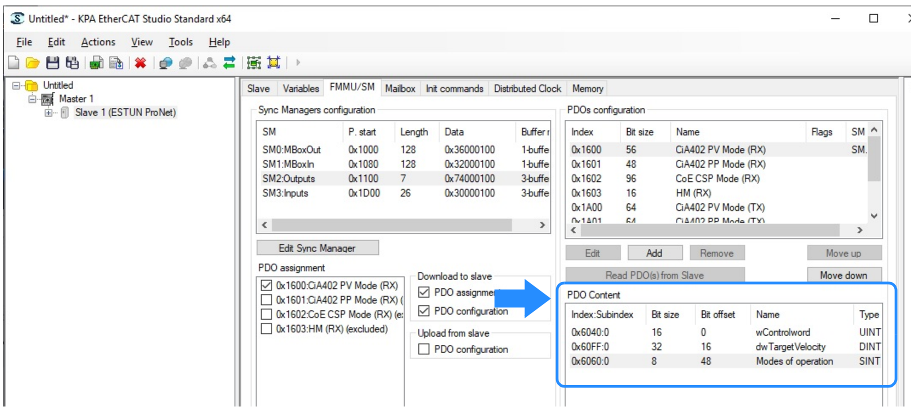


The image driver has variables 0x6040.0 and 0x60ff.0 in the PDO 0x1600.

5 By clicking the “Add” button, variables can be added to the selected PDO. In this case, it can be added to this PDO since the variable 0x6060.0 (Operation modes) must be sent to the drive. Click on the “Add” button and select the desired variable.

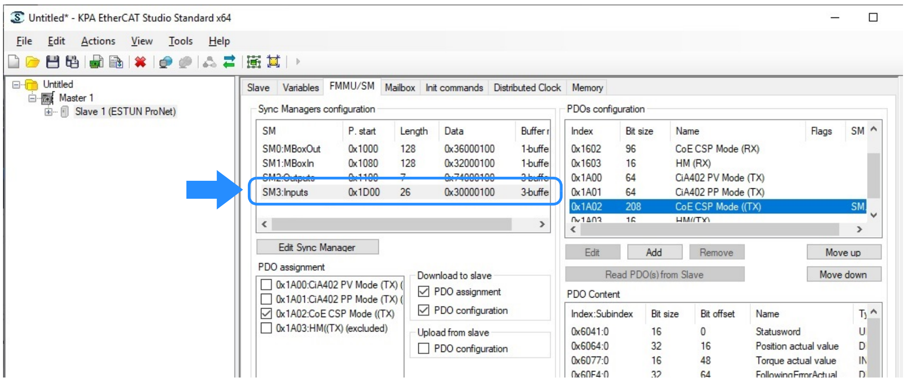


6 The variable is added to the PDO after clicking “OK”.

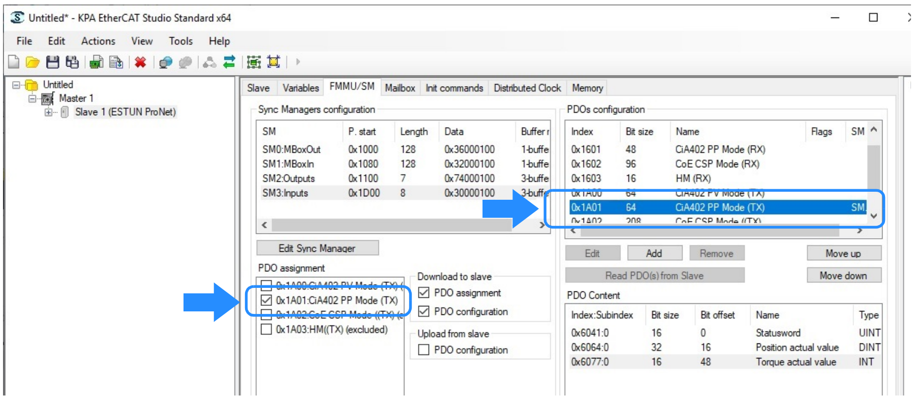


Variables read by the CNC from the drive.

1 To configure the variables read by the CNC from the drive, click on the field SM3:outputs



2 The lower window shows the PDO allocation and can be used to select which ones the CNC will read. The upper right window displays the various PDOs and the lower window displays the content of the selected PDO. This window is used to edit the PDO, by cyclically adding variables to be read from the drive or removing the unneeded variables.

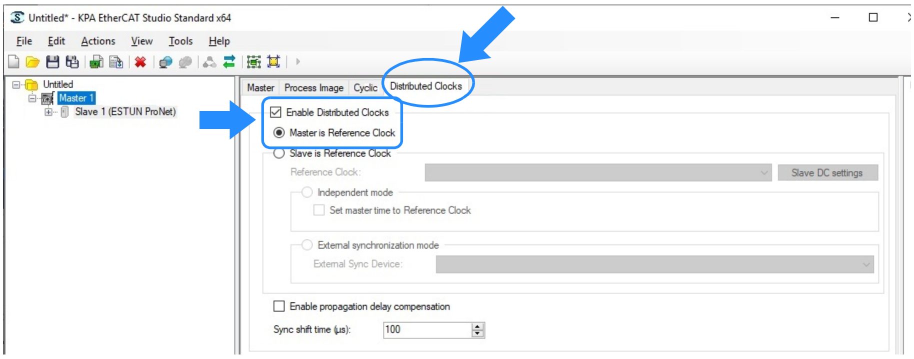


PDO 0x1A01 contains the variables that the CNC needs to read from the drive. The variable 0x6077.0 can be removed from the PDO, since it is not required.

22.4.4 Clock distribution.

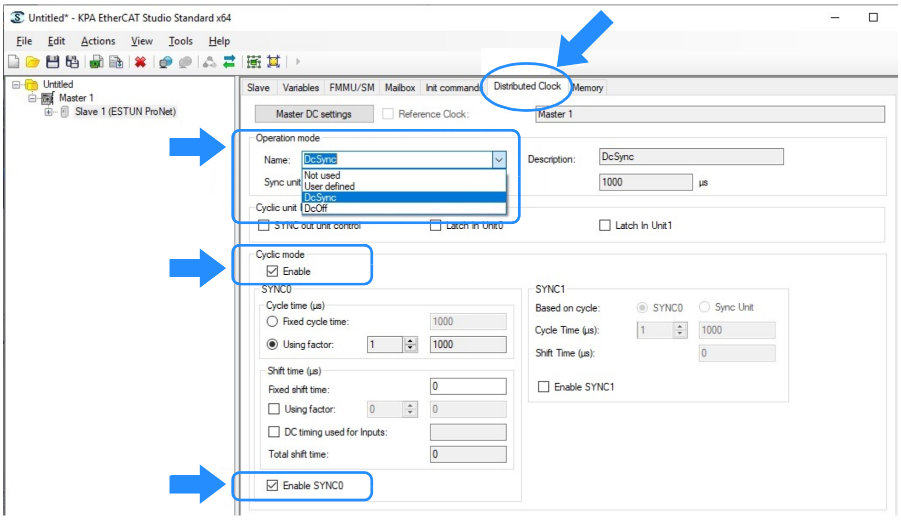
Setting the master clock.

The CNC must be set as the master clock to synchronize the bus. Move the mouse over the topology master (left window) and select the “Distributed Clocks” tab. Select the “Enable Distributed Clocks” and “Master is Reference Clock” options.



Setting the slave clock.

The drives must work in sync with the distributed clock. For each typology slave (left window), select the “Distributed Clocks” tab and set in operation mode. Refer to the manufacturer’s documentation.



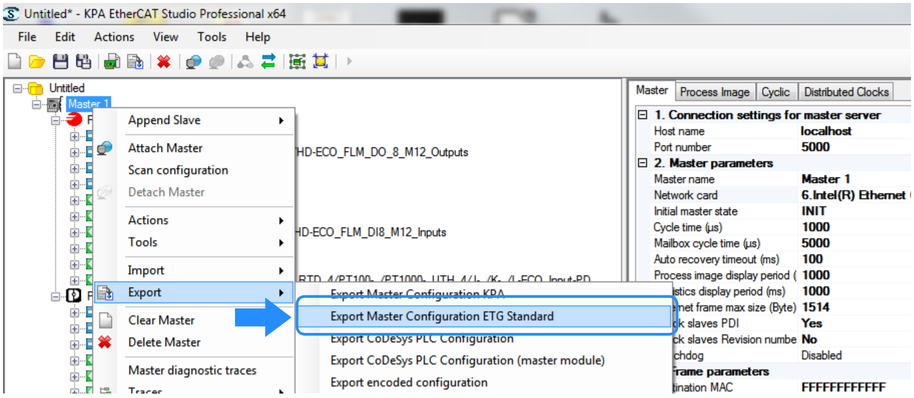
For the slave in the figure, the synchronization with the distributed clock is called “"DcSynch”.

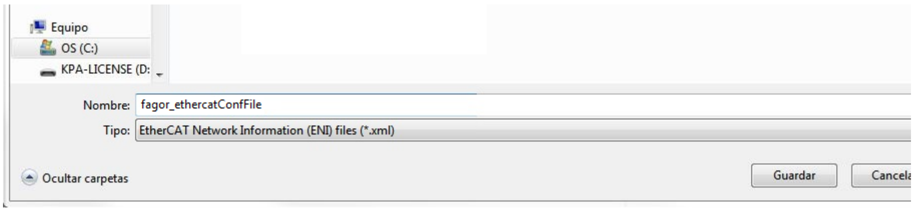
22.5 "KPA EtherCAT Studio" configurator. Export, save and copy the configuration file.

22.5.1 Export the configuration file.

To export the configuration, place the mouse pointer over the topology master (left window), click the right mouse button and select the "Export" and "Export Master Configuration ETG Standard" option. The file can be exported as any name (*.xml). This is the file that needs to be copied to the CNC.

NOTE: In Online mode, to export the configuration the bus master must be uncoupled from the bus (“Detach master” option).



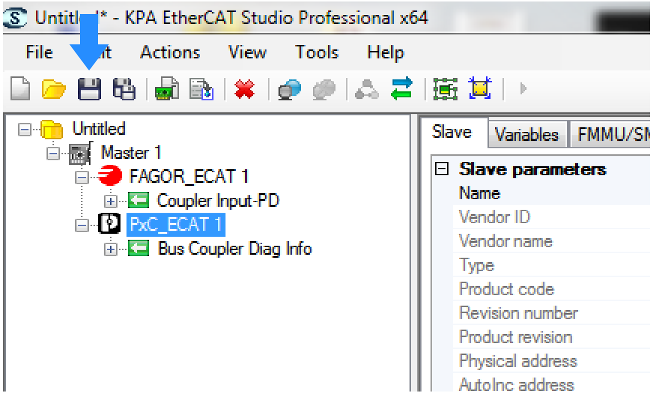


22.5.2 Copy the configuration file to the CNC.

When copying the configuration file to the CNC (xml file), first rename it as “fagor_ethercatConfFile.xml” and copy it to the ../Mtb/Data folder.

22.5.3 Save the project.

The project can be saved even if it is not completed. In a later session, we can recover the saved configuration and continue from that point. To save the project, select the File > Save option. Save the project with any name (extension *.ecsn).



22.6 Fagor EtherCAT Mapper. Map the EtherCAT resources to the PLC resources.

To configure EtherCAT drives, v2.01 or higher must be used for the "Fagor EtherCAT Mapper" application.

Fagor supplies the "Fagor EtherCAT Mapper" application to map bus resources (I/Os, drive variables, etc.) to the PLC resources (inputs, outputs or registers). From the ENI file generated by the EtherCAT configurator, this program makes a mapping proposal. The user can accept this proposal or perform the necessary modifications, all in a way that helps ensure there are no duplicated resources. Once the mapping is completed, this software creates a file with the description of the routing. This file must be copied to the folder ../Mtb/Data of the CNC with the name "resourceRouterConf.xml".

The digital inputs and outputs are mapped to resources of the same type in the PLC. Analog inputs and outputs are mapped to PLC registers. The drive variables are mapped to the PLC registers. For example:

EtherCAT.		PLC.
Module (0)	IB IL 24 DI 8 - HD-ECO	I1…I8
Module (1)	IB IL 24 DI 32 - HD-PAC	I9…I40
Module (2)	IB IL 24 DO 8 - HD-ECO	O1…O8
Module (3)	IB IL 24 DO 32 - HD-PAC	O9…O40
Module (4)	IB IL AO 2 - UI-PAC	R1…R2
Module (5)	IB IL AI 4 - U-PAC	R3...R6

Resource units of the bus and the CNC/PLC.

- The I/Os and PLC modules exchange the values for the following units.

Resources.	Units.
Live analog inputs (V).	Tenths of a millivolt.
"Analog inputs in current (mA).	Microamperes.
Live analog outputs (V).	Tenths of a millivolt.
Analog outputs in current (mA).	Microamperes.
Inputs for the temperature sensors Pt100.	Tenths of a degree.

- The drives and PLC modules exchange the values for the following units.

Resources.	Units.
Control word.	16 bit array.
Speed Reference. (*)	Tenths of rpm for BCSD drives. For third-part drives, usually in encoder pulses.
Operation mode.	Numerical value of 1 a n.
Status word.	Bit array.
Position. (*)	Encoder pulses for BCSD drives. For third-part drives, usually in encoder pulses.

(*) The units depend on the drive; refer to the manufacturer's documentation.

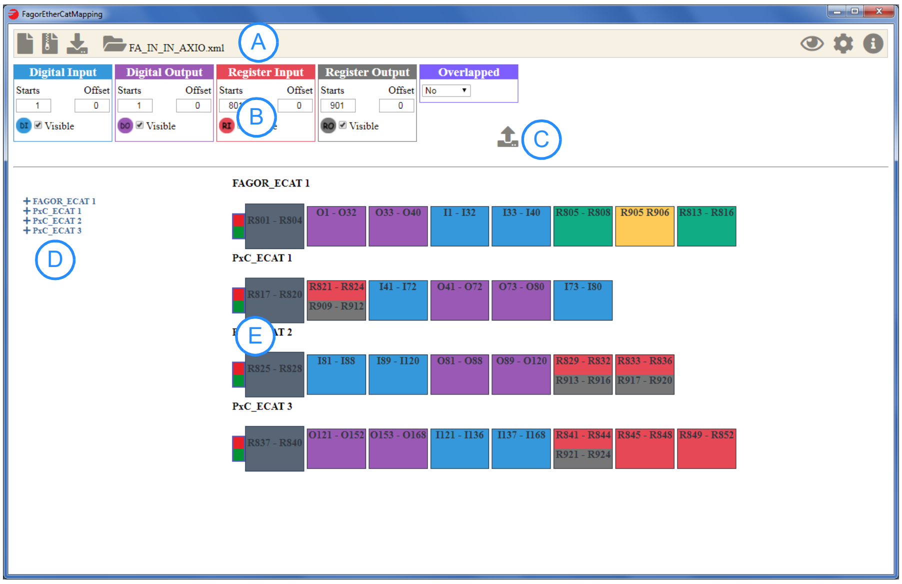
22.6.1 Install the program.

Although "Fagor EtherCAT Mapper" can be run in the same CNC, it is recommended to install it in an external PC, along with the configurator. Use a PC with Windows 7 32/64 bits or Windows 10 64 bits.

Run the "Setup fagorethercatmapper-x.x.x.exe" file (where x.x.x indicates the version) to begin the installation. Follow the instructions on the screen to complete the process. Once installed, the following icon will appear on the desktop.

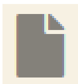
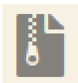
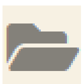
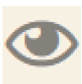
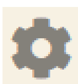
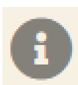
Icon.	Function.
	EtherCAT mapper of resources.

22.6.2 Interface description.



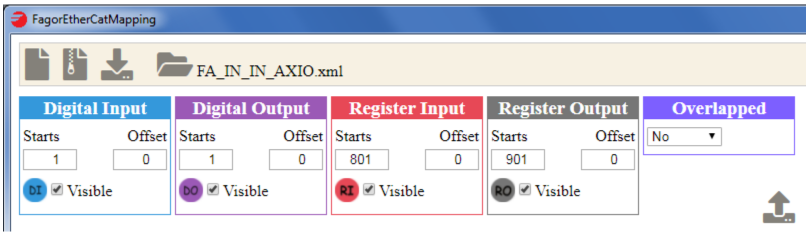
A. Icon bar and ENI file name loaded.
B. Default status bar for mapping of resources.
C. Validation of the "Overlapped" option.
D. Bus topology (tree view).
E. Bus topology (block diagram).

Icon bar and ENI file name loaded.

Block.	Color and type of resource.
	New project.
	Generate the project.
	Export routed file.
	Open a Fagor, ENI or resources file.
	View slaves.
	Settings.
	Help.
	Validation of the "Overlapped" option.

Default status bar for mapping of resources.

This bar shows which configuration is assumed by the resources when the ENI is loaded for the first time (default configuration). Then the allocation of resources can be customized manually.



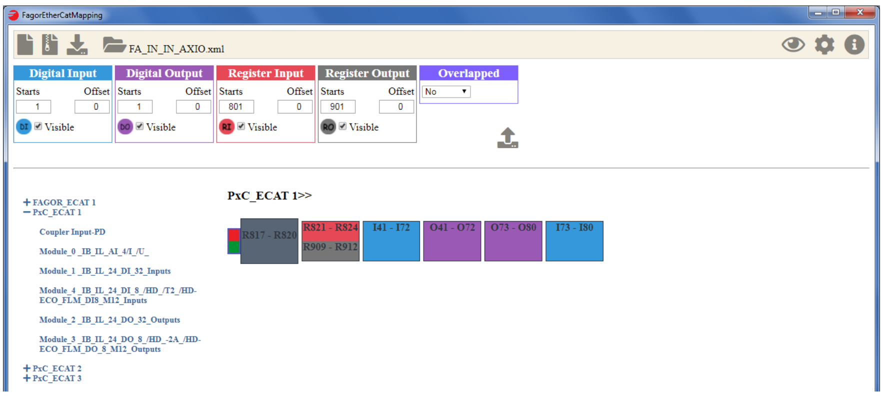
- The "Digital Input" and "Digital Output" files configure the mapping of the digital inputs and outputs.
- The "Register Input" and "Register Output" files map those resources of the EtherCAT bus which are not assumed to be digital inputs/outputs to PLC registers. For the Fagor modules, these will be the analog inputs/outputs and the inputs for the temperature sensors. For third-party modules, it will be necessary to refer to the manufacturer's manual so that they can be used correctly from the PLC.

Starts.	Number of the first input, output or register of the bus.
Offset.	Numbering offset between headers. If the offset is zero, the numbering will be consecutive between headers.
Visible.	Shows the numbering on the graphics.

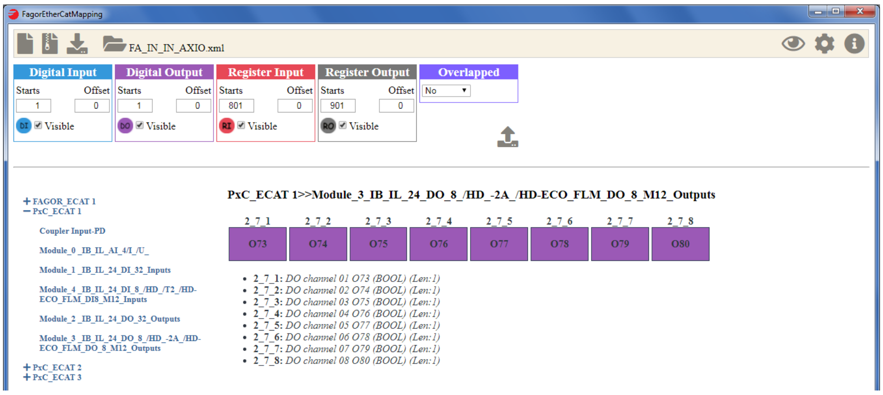
- The "Overlapped" file defines whether the layout of resources is permitted in different modules. See "22.6.5 Overlapped." on page 816.

Bus topology (tree view).

The first level consists of the bus slaves. By clicking on a slave, the program shows the modules connected to the bus. To show all the slaves, click on the "View Slaves" icon.



The second level consists of the modules that belong to the slave. By clicking on a module, the program shows its list of resources.



Bus topology (block diagram).

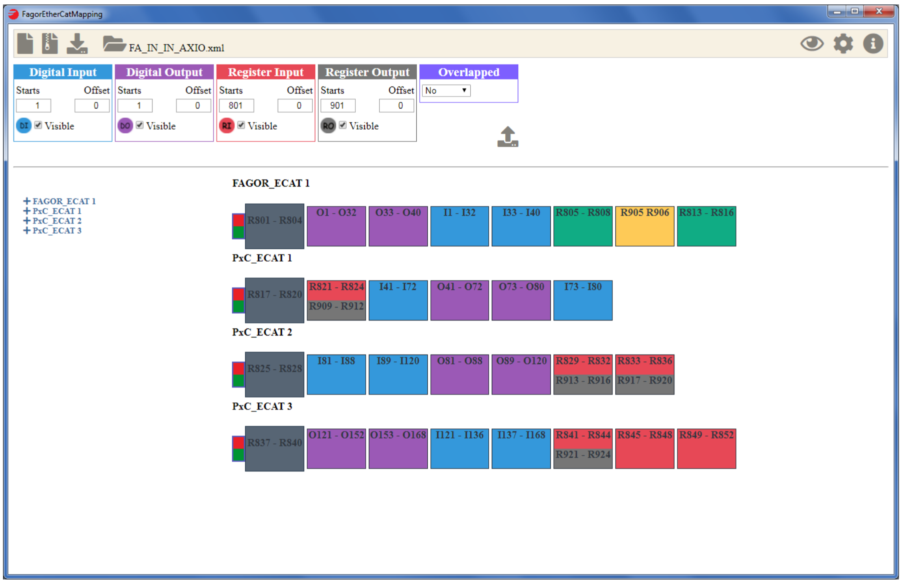
Block.	Color and type of resource.
I33 - I40	Blue. Digital inputs.
O33 - O40	Purple. Digital outputs.
R845 - R848	Red. Mapping of EtherCAT resources to PLC registers.
R917 - R920	Gray. Mapping of PLC registers to EtherCAT resources.
R805 - R808	Green color. Records of the Fagor analog inputs.
R905 R906	Yellow. Registers of the Fagor analog outputs.

Help.

For detailed information on the program operation, refer to the help for the program.

22.6.3 Open a configuration (default mapping).

Load the ENI file created with the configurator. The program allocates the resources as defined in the status bar.

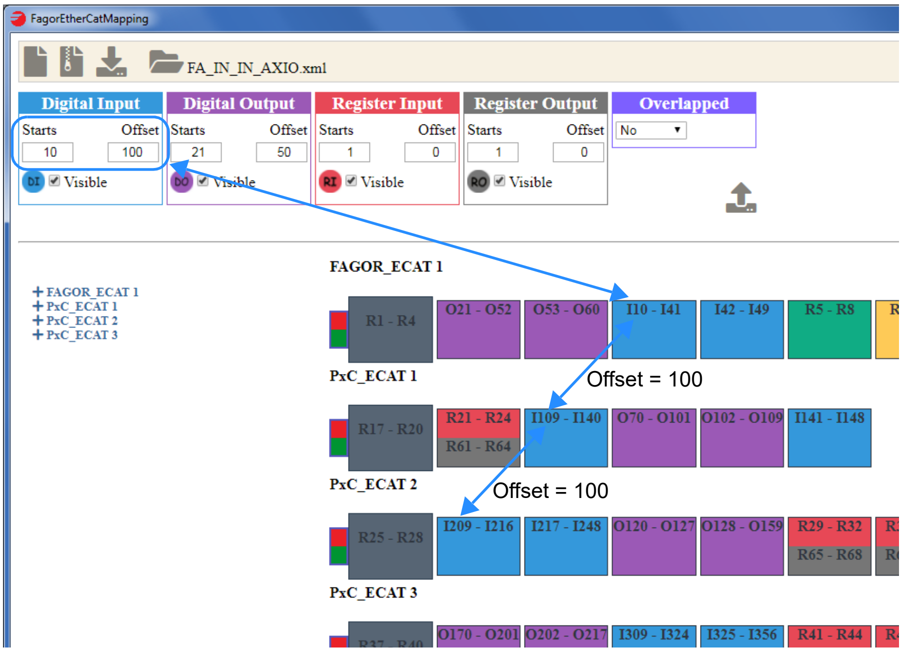


22.6.4 Change the numbering of the resources.

Change the numbering of the resources of the whole bus.

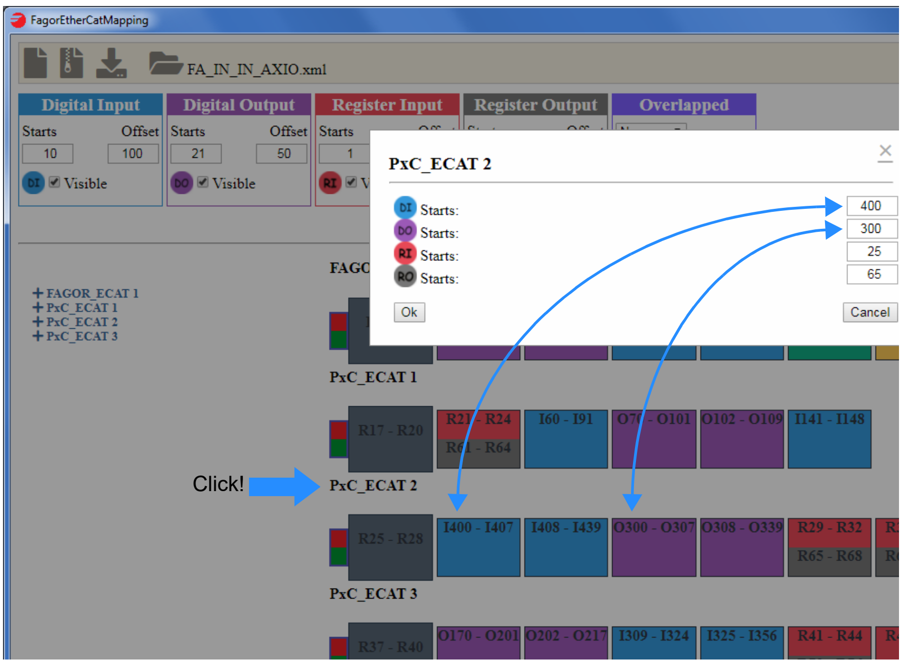
Change the resources on the top navigation bar and the program updates the resources of the whole bus. In the following configuration the numbering of the digital inputs and outputs has been changed as follows.

- Digital inputs. First = 10 Offset = 100
- Digital outputs. First = 21 Offset = 50



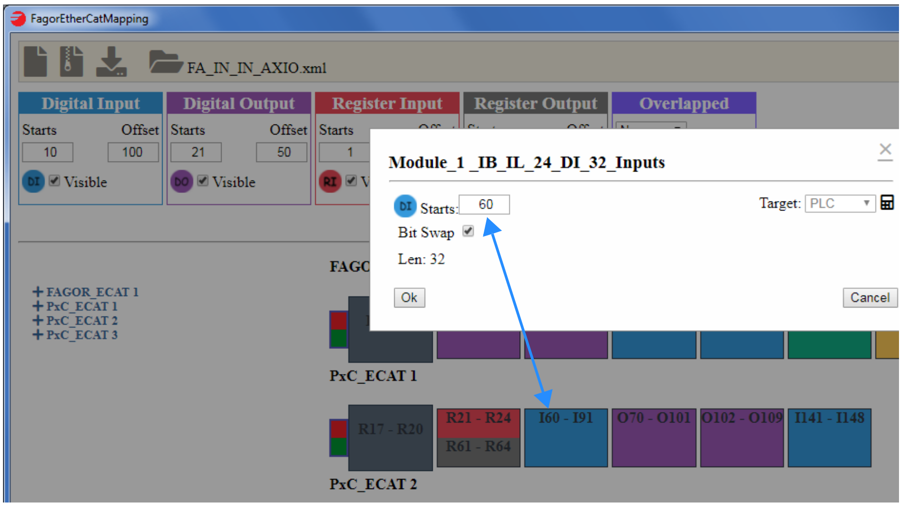
Change the numbering of the resources of a node.

In the block diagram, click on the name of the header and set the start number of the resources of the node.



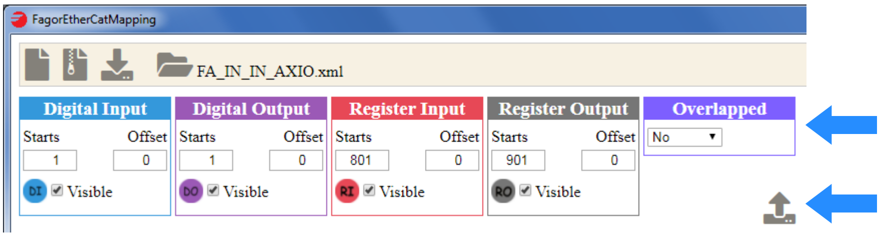
Change the numbering of the resources of a module.

In the block diagram, click on a module and set the start number of its resources; the remaining resources are numbered consecutively.



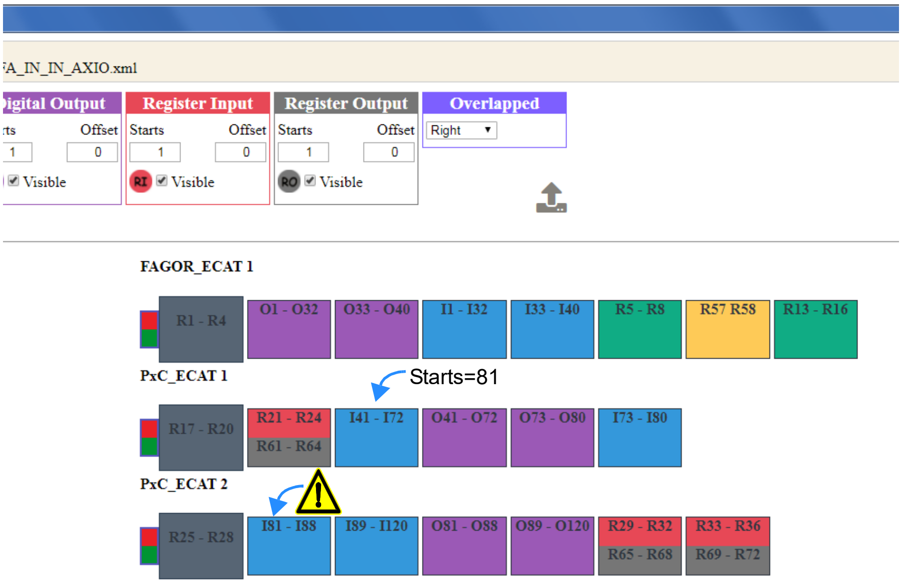
22.6.5 Overlapped.

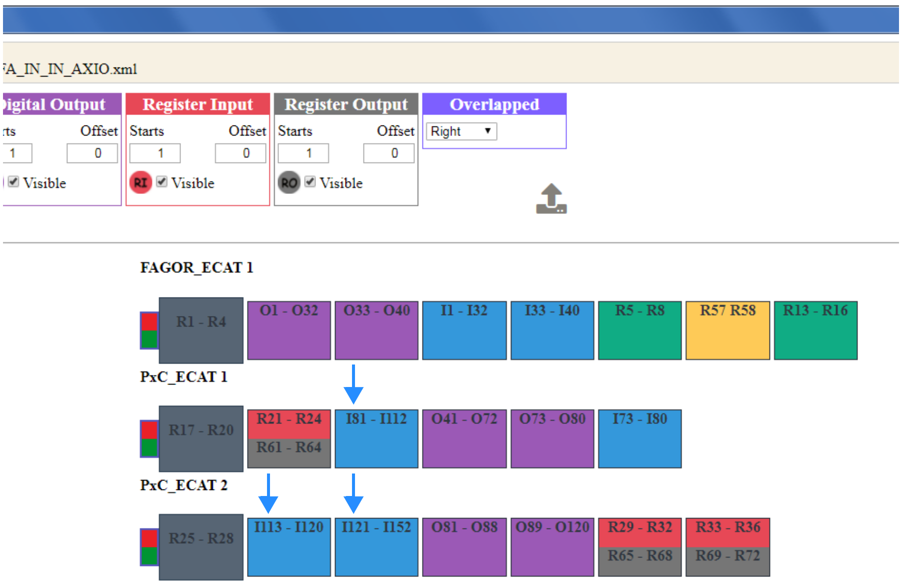
The "Overlapped" file defines whether the layout of resources is permitted in different modules. Every time this option is changed, it should be validated with the "Load filter" icon located on the upper right.



Overlapped = Right.

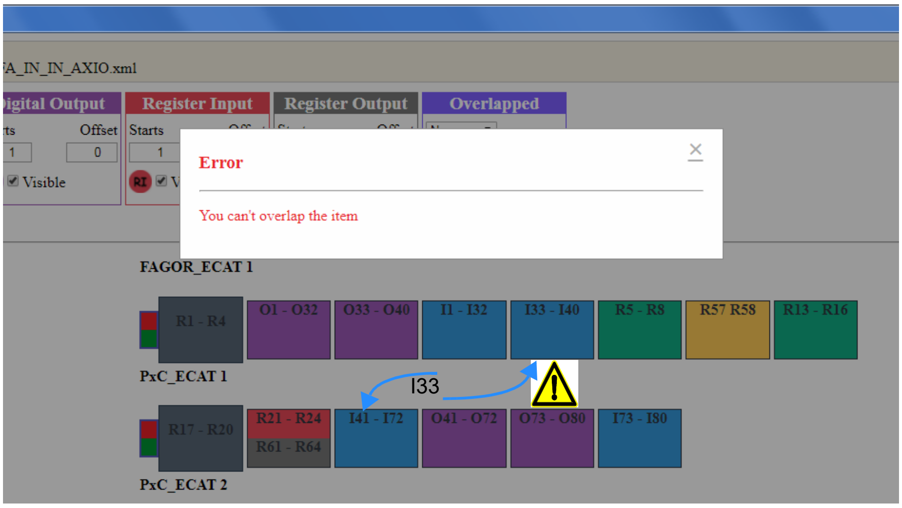
When assigning a resource a number that already exists in the bus, the program rearranges the following resources (top numbering). Resources with previous numbering (bottom numbering) are not changed.





Overlapped = No.

If assigning a resource a number already in the bus, the program will issue an error.

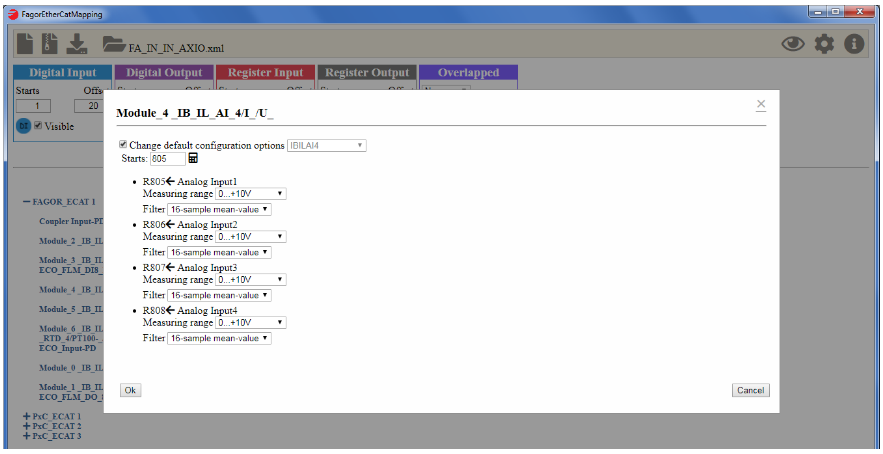


22.6.6 Configure the Fagor analog inputs and outputs.

The configuration of the work mode (range of measurement) of these modules is performed from the mapper. In the block view, click on a module to access its properties.

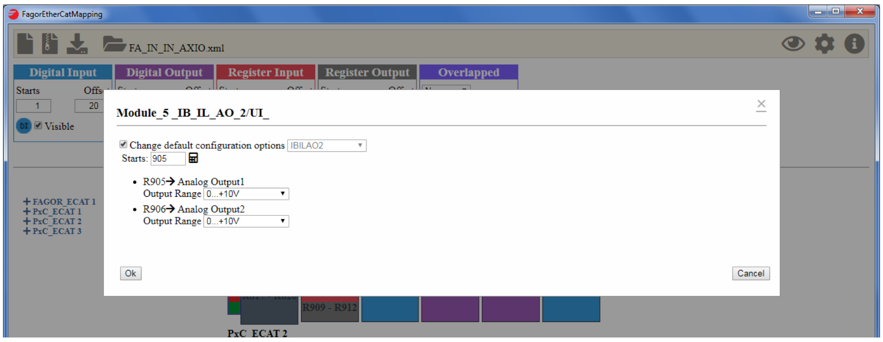
Fagor analog inputs (for example, the Fagor IB IL AI 4 - U-PAC module).

These modules, as well as the number of the first input, enable the measurement range of each input to be configured, as well as the frequency of the sampling.



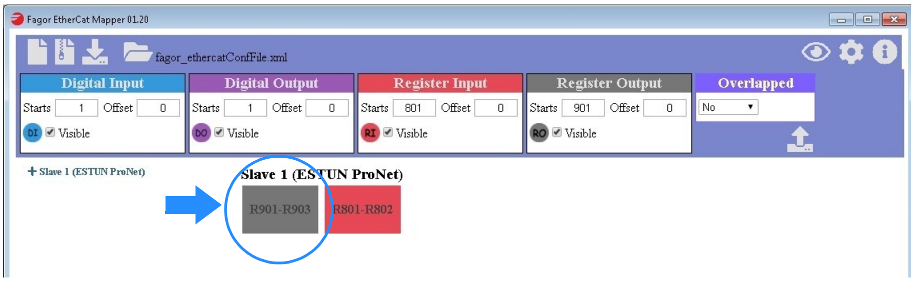
Fagor analog outputs (for example, the Fagor IB IL AO 2 - U-PAC module).

These modules, as well as the number of the first output, allow the measurement range of each output to be configured.

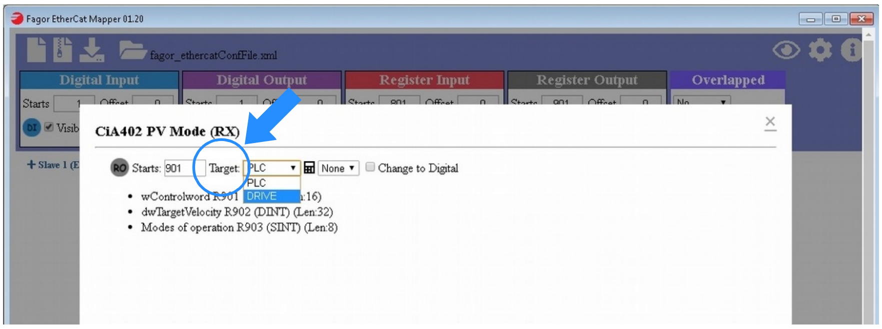


22.6.7 Configuring the resources of a EtherCAT drive.

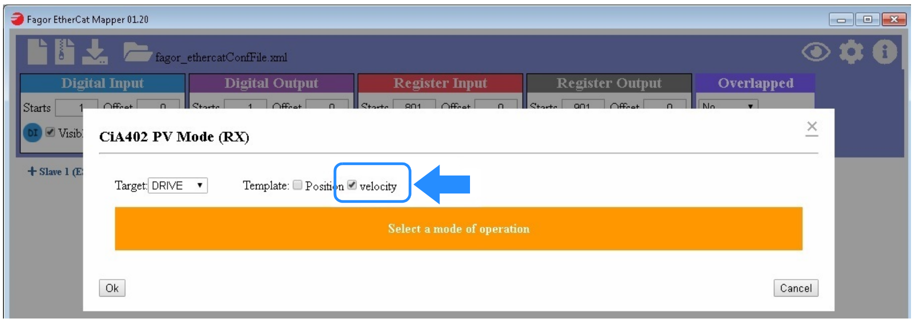
1 Click on a register module of the slave.



2 Select the “Drive” option from the Target menu.



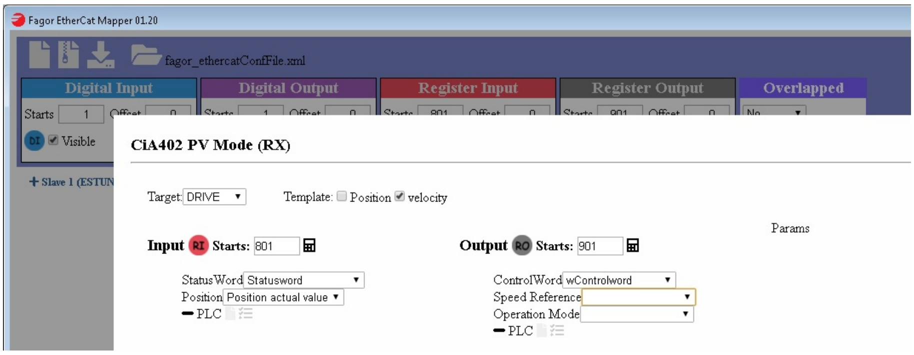
3 Select velocity profile.



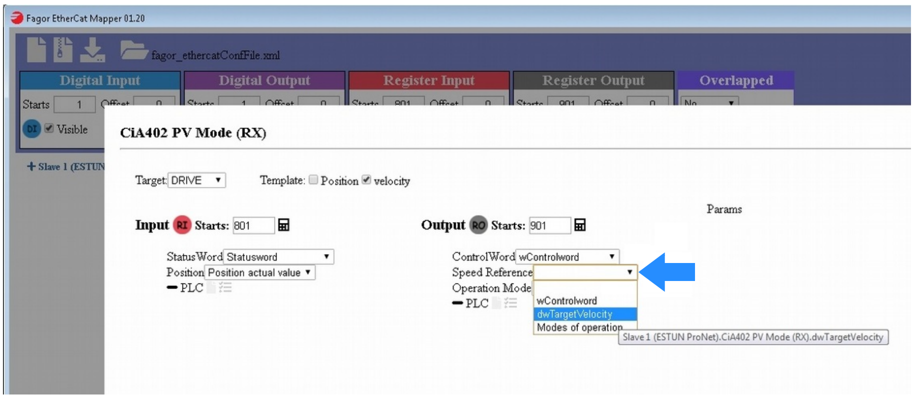
4 The next window maps the bus resources to the drive resources. Click on a module and configure the following options.

- Inputs: Control word, Speed Reference and Operation mode.
- Outputs: Status word and Position.

Each drive will have its own names for these resources. The mapper will automatically attempt to select the resources for these resources, however manual configuration may be required.



For example, for this drive, the mapper has automatically selected the resources for Status Word, Position and Control Word. Ensure that these options are the correct ones. The empty fields will have to be filled in next.

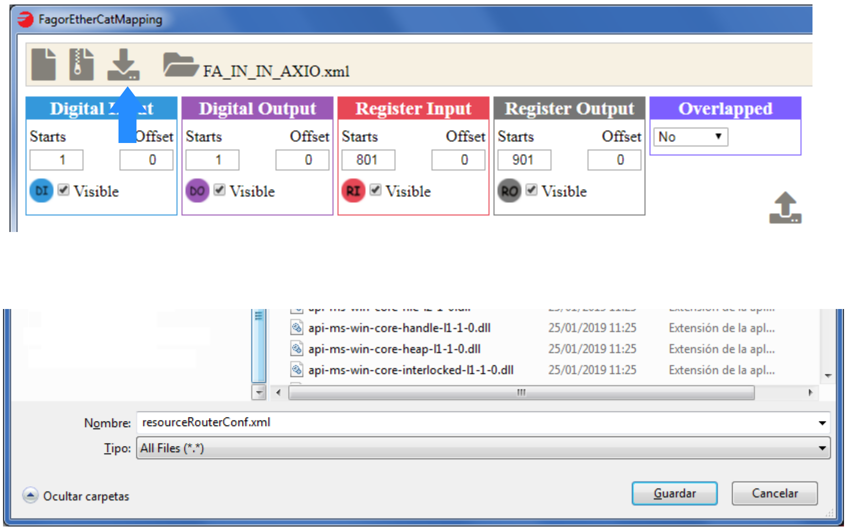


5 When the mouse is over a resource, its value found in the EtherCAT configuration file will be displayed. Make sure that the selected value is the same.

6 Now, the slave id needs to be defined. EtherCAT nodes have a registry called “Configured station alias”. The id assigned to this slave must be the same as the one defined in the “Configured station alias” register of the drive.

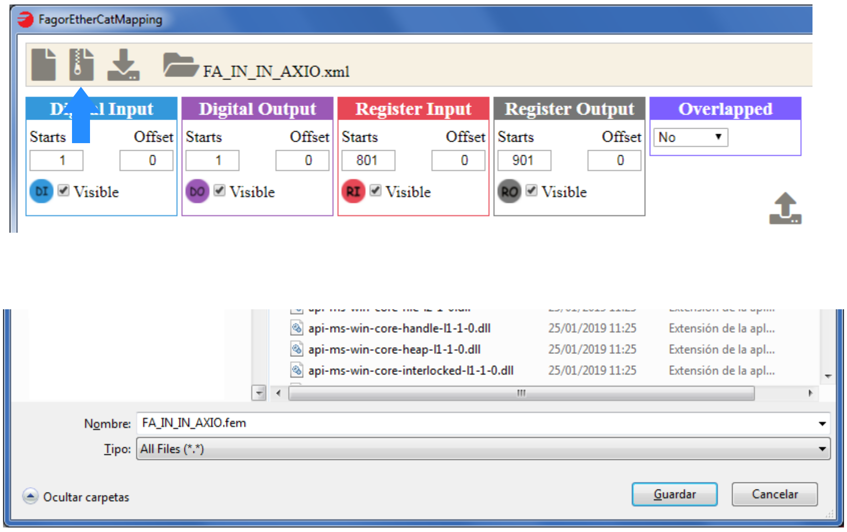
22.6.8 Exporting the routing file.

To export the routing file, select the option “Exports router file”. The file can be exported as any name (*.xml). This is the file that needs to be copied to the CNC.



22.6.9 Save the project.

The project can be saved even if it is not completed. In a later session, we can recover the saved configuration and continue from that point. To save the project, select the "Generate project" option. Save the project with any name (extension *.fem).



22.6.10 Copy the routing file to the CNC.

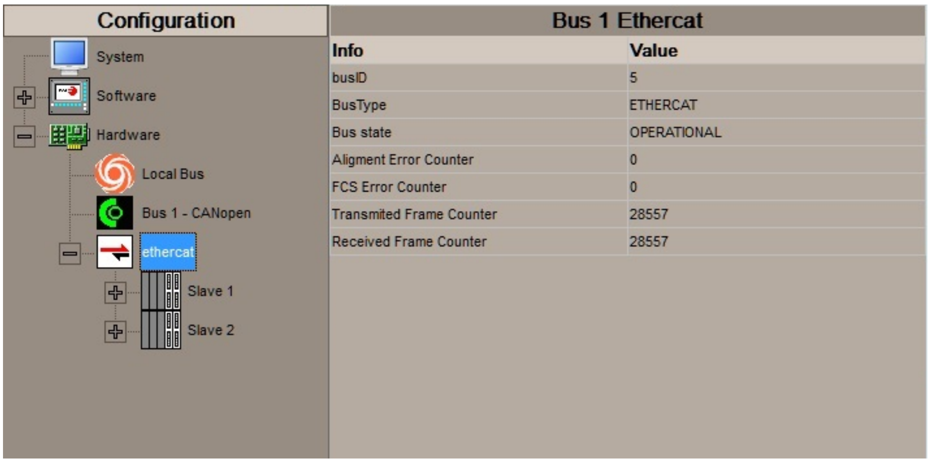
When copying the routing file to the CNC (xml file), first rename it as “resourceRouterConf.xml” and copy it to the ../Mtb/Data folder.

22.7 Diagnosis in the CNC.

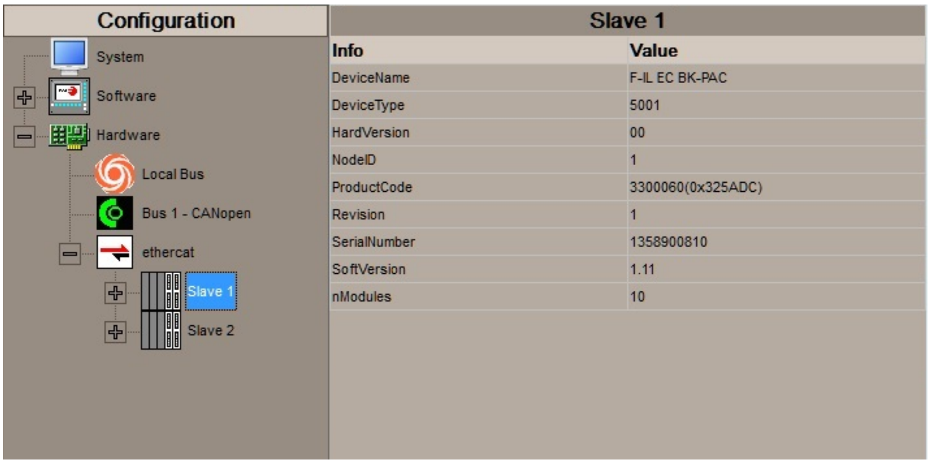
22.7.1 General information.

The diagnosis mode provides all the information required of the modules connected to the EtherCAT bus.

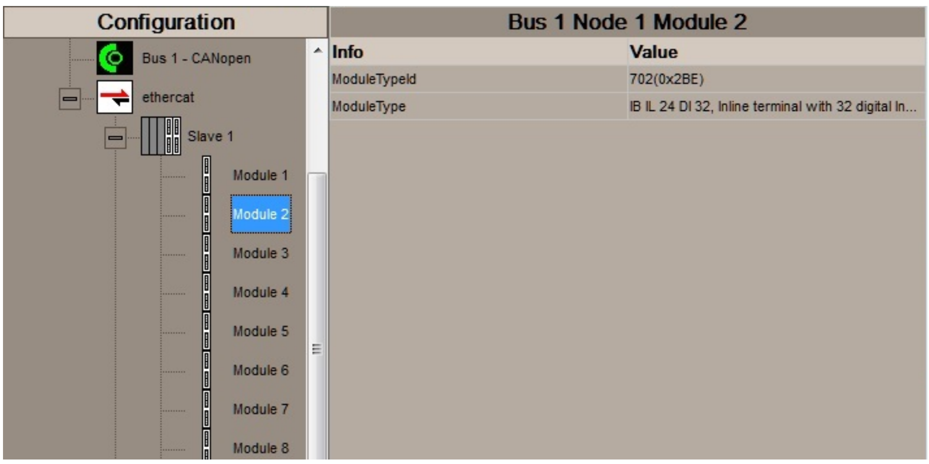
General information about the bus.



Information of the slaves.



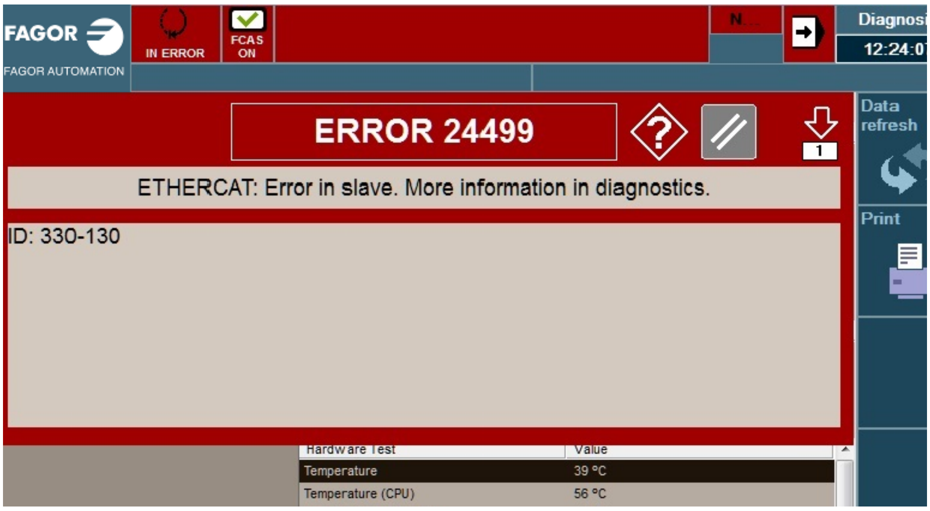
Module information.



22.7.2 Error in the slaves.

Error 24499. EtherCAT generic error.

The CNC shows the following generic error when an error in a node is detected. To find out the details of an error, refer to the diagnosis mode.



Detailed information of the module in error.

The configuration tree identifies the node in error. The right window shows the information available for the module and the error.

REMOTE OPENPCS.

23

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.

Parámetro.		Meaning.
PLCTYPE	MPG	PLC type; standard IEC-61131 and/or Fagor.

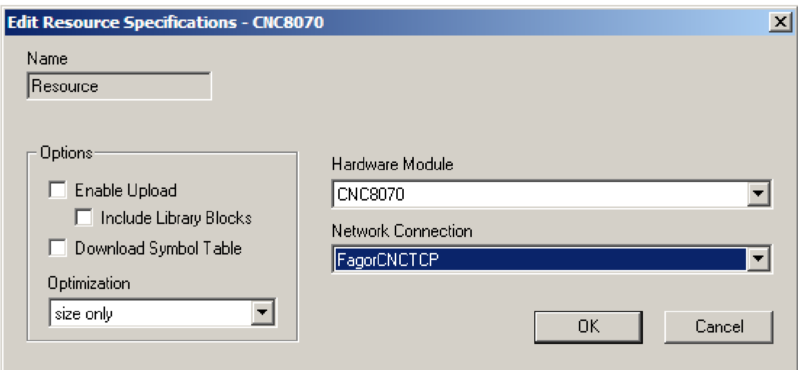
[MPG]....... General machine parameters.

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

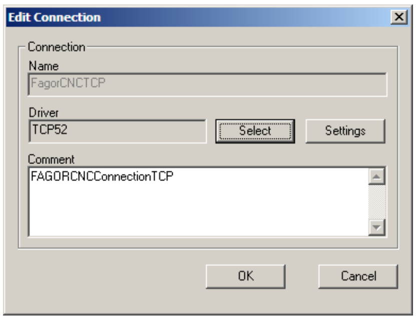
Create the hardware module of the remote OpenPCS.

The information needed by the OpenPCS to generate the proper PLC for the CNC is in the folder ..\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.

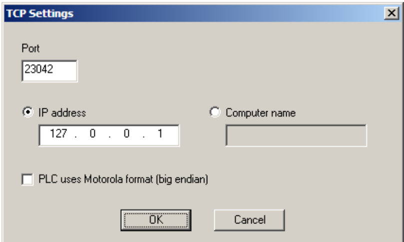


Configure the IP direction of the connection at the hardware module.

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).

MOTOR CONFIGURATION.

24

24.1 V/f (voltage/frequency) control.

The V/f control adjusts the voltage to be applied to the motor according to the adjustment characteristics at the indicated speed ω (see frequency, $f = \omega/2\pi$). The ratio between the velocity command ω (i.e. frequency) and the stator voltage is obtained according to the V/f characteristic curve. It also supports additional characteristic curves adapted to the application and whose parameters are set by the user.

The V/f control is configured from the motor machine parameters.

Parameter.	Description.
MOT_CTRL_TYPE	Type of control of the asynchronous motor.
VF_MAXVOLTAGE	Maximum voltage to be applied to the V/f control.
VF_VOLT_CURVE1	Percentage of the rated motor voltage (MOT_VOLTAGE) defining the ordinate of point 1 of the characteristic curve (Figure B).
VF_FREQ_CURVE1	Percentage of rated engine speed (MOT_SMON) defining the abscissa of point 1 of the characteristic curve (Figure B).
VF_VOLT_CURVE2	Percentage of the rated motor voltage (MOT_VOLTAGE) defining the ordinate of point 2 of the characteristic curve (Figure B).
VF_FREQ_CURVE2	Percentage of rated engine speed (MOT_SMON) defining the abscissa of point 2 of the characteristic curve (Figure B).
VF_BOOSTVOLTAGE	Percentage of the rated motor voltage (MOT_VOLTAGE) to be applied at zero speed (Figure A).

Other parameters involved.

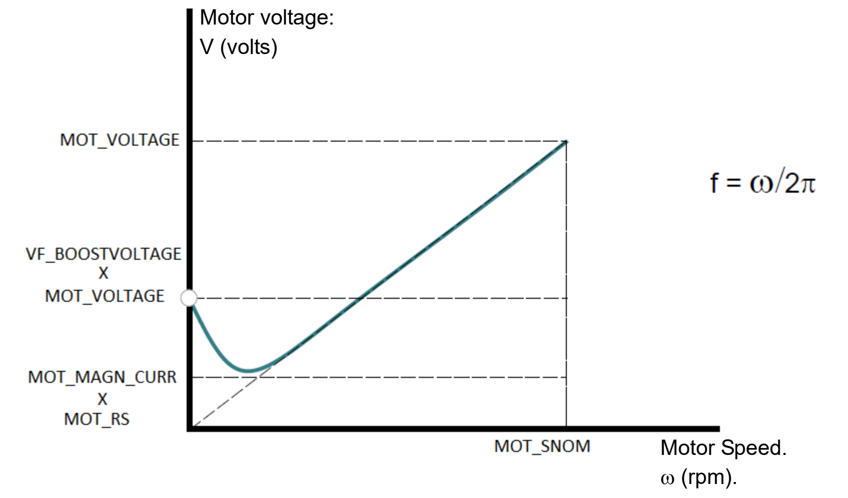
Parameter.	Description.
MOT_VOLTAGE	Rated motor voltage. Note that for a synchronous motor, this parameter is just for information purposes, it is not used in the control.
MOT_RS	Phase-neuter resistance of the stator at 20 °C.
MOT_SNOM	Rated speed of a synchronous motor or base speed (below the constant power area) of a synchronous motor.
MOT_MAGN_CURR	Motor rms current without load.

Quadratic V/f characteristic curve.

The V/f control with quadratic characteristic reduces (in comparison with the linear one) the losses at the motor and the inverter due to lower currents. At zero speed, a voltage will be added to the theoretical V/f curve, this voltage will be equal to the percentage set in VF_BOOSTVOLTAGE multiplied by the rated voltage of the motor given in MOT_VOLTAGE and the voltage will be decreasing quadratically down to zero as the speed increases until it reaches the rated speed of the motor MOT_SMON.

Apply the quadratic V/f characteristic curve whenever the inequality VF_BOOSTVOLTAGE x MOT_VOLTAGE > MOT_MAGN_CURR x MOT_RS is verified and add MOT_MAGN_CURR x MOT_RS if not satisfied at low revolutions.

Figure A.



V/f characteristic curve adapted to the application.

Apply a quadratic V/f characteristic curve adapted to the application when verifying the following inequality:

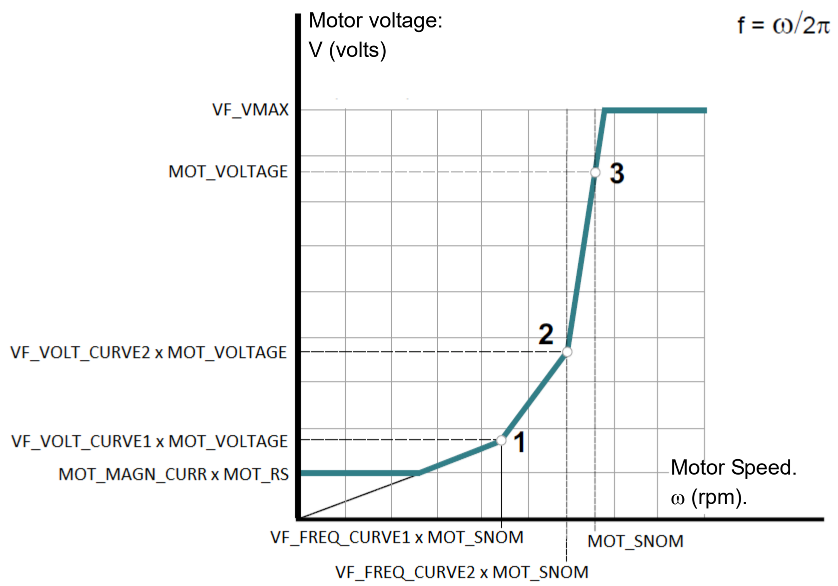
MOT_MAGN_CURR x MOT_RS > VF_BOOSTVOLTAGE x MOT_VOLTAGE

Make sure that you do not parameterise any of the following inequalities, otherwise the CNC will display the corresponding error. Modify the value of one of the parameters to prevent these inequalities from occurring and thus the error code indicated.

VF_VOLT_CURVE2 < VF_VOLT_CURVE1

VF_FREQ_CURVE2 < VF_FREQ_CURVE1

Figure B.



User notes:

User notes:

User notes:

Fagor Automation S. Coop.
Bº San Andrés, 19 - Apdo. 144
E-20500 Arrasate-Mondragón, Spain
Tel: +34 943 039 800
Fax: +34 943 791 712
E-mail: contact@fagorautomation.es
www.fagorautomation.com

AENOR
Empresa Registrada
ER-073/1994