# CNC

## 8055
## ·M· & ·EN·

### Programming manual

Ref.1705
Soft: V02.2x

**FAGOR**

**FAGOR AUTOMATION**

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

**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

Soft: V02.2x

·3·

**CHAPTER 6    PATH CONTROL**

**CHAPTER 7    ADDITIONAL PREPARATORY FUNCTIONS**

**CHAPTER 8    TOOL COMPENSATION**

**CHAPTER 9    CANNED CYCLES**

FAGOR

FAGOR AUTOMATION

CNC 8055
CNC 8055i

SOFT: V02.2x

FAGOR ⬤

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

SOFT: V02.2x

**CHAPTER 15     COORDINATE TRANSFORMATION**

**CHAPTER 16     ANGULAR TRANSFORMATION OF AN INCLINE AXIS**

**APPENDIX**

FAGOR

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

SOFT: V02.2x

**FAGOR**

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

SOFT: V02.2X

# ABOUT THE PRODUCT

## BASIC CHARACTERISTICS OF THE DIFFERENT MODELS.

|  | 8055i FL EN | 8055 FL<br>8055i FL | 8055 Power<br>8055i Power |
|---|---|---|---|
| Pendant | 8055i FL EN | 8055i FL | 8055i Power |
| Enclosure | ----- | 8055 FL | 8055 Power |
| USB | Standard | Standard | Standard |
| Block processing time | 1 ms | 3.5 ms | 1 ms |
| RAM memory | 1Mb | 1Mb | 1 Mb |
| Software for 7 axes | ----- | ----- | Option |
| TCP transformation | ----- | ----- | Option |
| C axis (Lathe) | ----- | ----- | Option |
| Y axis (Lathe) | ----- | ----- | Option |
| Look-ahead | 100 blocks | 100 blocks | 200 blocks |
| Flash Memory 512Mb / 2Gb | 512Mb | Option | Option |

## HARDWARE OPTIONS OF THE 8055I CNC

|  | Analog | Digital | Engraving |
|---|---|---|---|
| Ethernet | Option | Option | Option |
| RS232 serial line. | Standard | Standard | Standard |
| 16 digital inputs and 8 outputs (I1 to I16 and O1 to O8) | Standard | Standard | Standard |
| Another 40 digital inputs and 24 outputs (I65 to I104 and O33 to O56) | Option | Option | Option |
| Probe inputs | Standard | Standard | Standard |
| Spindle (feedback input and analog output) | Standard | Standard | Standard |
| Electronic handwheels | Standard | Standard | Standard |
| 4 axes (feedback and velocity command) | Option | Option | - - - |
| Remote CAN modules, for digital I/O expansion (RIO). | Option | Option | - - - |
| Sercos servo drive system for Fagor servo drive connection. | - - - | Option | - - - |
| CAN servo drive system for Fagor servo drive connection. | - - - | Option | - - - |

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

FAGOR

FAGOR AUTOMATION

CNC 8055
CNC 8055i

# SOFTWARE OPTIONS OF THE 8055 AND 8055I CNCS.

| | Model | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | **GP** | **M** | **MC** | **MCO** | **EN** | **T** | **TC** | **TCO** |
| Number of axes with standard software | 4 | 4 | 4 | 4 | 3 | 2 | 2 | 2 |
| Number of axes with optional software | 7 | 7 | 7 | 7 | ----- | 4 or 7 | 4 or 7 | 4 or 7 |
| Electronic threading | ----- | Stand. | Stand. | Stand. | Stand. | Stand. | Stand. | Stand. |
| Tool magazine management: | ----- | Stand. | Stand. | Stand. | ----- | Stand. | Stand. | Stand. |
| Machining canned cycles | ----- | Stand. | Stand. | ----- | Stand. | Stand. | Stand. | ----- |
| Multiple machining | ----- | Stand. | Stand. | ----- | Stand. | ----- | ----- | ----- |
| Solid graphics | ----- | Stand. | Stand. | Stand. | ----- | Stand. | Stand. | Stand. |
| Rigid tapping | ----- | Stand. | Stand. | Stand. | Stand. | Stand. | Stand. | Stand. |
| Tool life monitoring | ----- | Opt. | Opt. | Opt. | Stand. | Opt. | Opt. | Opt. |
| Probing canned cycles | ----- | Opt. | Opt. | Opt. | Stand. | Opt. | Opt. | Opt. |
| DNC | Stand. | Stand. | Stand. | Stand. | Stand. | Stand. | Stand. | Stand. |
| COCOM version | Opt. | Opt. | Opt. | Opt. | ----- | Opt. | Opt. | Opt. |
| Profile editor | Stand. | Stand. | Stand. | Stand. | ----- | Stand. | Stand. | Stand. |
| Tool radius compensation | Stand. | Stand. | Stand. | Stand. | Stand. | Stand. | Stand. | Stand. |
| Tangential control | Opt. | Opt. | Opt. | Opt. | ----- | Opt. | Opt. | Opt. |
| Retracing | ----- | Opt. | Opt. | Opt. | Stand. | Opt. | Opt. | Opt. |
| Setup assistance | Stand. | Stand. | Stand. | Stand. | Stand. | Stand. | Stand. | Stand. |
| Irregular pockets with islands | ----- | Stand. | Stand. | Stand. | ----- | ----- | ----- | ----- |
| TCP transformation | ----- | Opt. | Opt. | Opt. | ----- | ----- | ----- | ----- |
| C axis (on Lathe) | ----- | ----- | ----- | ----- | ----- | Opt. | Opt. | Opt. |
| Y axis (on Lathe) | ----- | ----- | ----- | ----- | ----- | Opt. | Opt. | Opt. |
| Telediagnosis | Opt. | Opt. | Opt. | Opt. | Stand. | Opt. | Opt. | Opt. |

FAGOR

FAGOR AUTOMATION

CNC 8055
CNC 8055i

# DECLARATION OF CONFORMITY AND WARRANTY CONDITIONS

## DECLARATION OF CONFORMITY

The declaration of conformity for the CNC is available in the downloads section of FAGOR'S corporate website at http://www.fagorautomation.com. (Type of file: Declaration of conformity).

## WARRANTY TERMS

The warranty conditions for the CNC are available in the downloads section of FAGOR's corporate website at http://www.fagorautomation.com. (Type of file: General sales-warranty conditions).

**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

**FAGOR**

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

# VERSION HISTORY

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

The version history uses the following abbreviations:

| | |
|---|---|
| INST | Installation manual |
| PRG | Programming manual |
| OPT | Operating manual |
| OPT-MC | Operating manual for the MC option. |
| OPT-TC | Operating manual for the TC option. |
| OPT-CO | Manual of the CO manual |

---

**Software V01.00**          **October 2010**

First version.

---

**Software V01.20**          **April 2011**

| List of features | Manual |
|---|---|
| Open communication. | INST |
| Improvements to Look Ahead machining. | INST |
| Blocks with helical interpolation in G51. | PRG |
| G84. Tapping with relief. | PRG |

---

**Software V01.08**          **August 2011**

| List of features | Manual |
|---|---|
| Spindle parameter OPLDECTI (P86). | INST |

---

**Software V01.30**          **September 2011**

| List of features | Manual |
|---|---|
| Gear ratio management on Sercos spindles | INST |
| Improved feedrate limit management (FLIMIT). | INST |
| New type of penetration in lathe type threading cycles. | PRG |
| Improved lathe type thread repair. Partial repair. | PRG |
| MC option: Rigid tapping with relief. | OPT-MC |
| TC option: New type of penetration in threading cycles. | OPT-TC |
| TC option: Improved thread repair. Partial and multi-entry (start) thread repair. | OPT-TC |
| TC option: Zig-zag entry to the groove at the starting point of the groove. | OPT-TC |

## Software V01.31                                              October 2011

| List of features | Manual |
|---|---|
| CNC 8055 FL Engraving model | INST / OPT / PRG |

## Software V01.40                                              January 2012

| List of features | Manual |
|---|---|
| Execution of M3, M4 and M5 using PLC marks | INST / PRG |
| Values 12 and 43 of variable OPMODE in conversational work mode. | INST / PRG |

## Software V01.60                                              December 2013

| List of features | Manual |
|---|---|
| Auto-adjustment of axis machine parameter DERGAIN. | INST |
| New value for axis machine parameter ACFGAIN (P46). | INST |
| Value 120 of the OPMODE variable. | INST / PRG |

## Software V01.65                                              January 2015

| List of features | Manual |
|---|---|
| Block processing time of 1 ms on the "CNC 8055i FL Engraving" model. | INST / OPT / PRG |

## Software V02.00                                              February 2014

| List of features | Manual |
|---|---|
| Profile machining in segments. J parameter for G66 and G68 cycles. | PRG |
| Calls to subroutines using G functions. | INST / PRG |
| Anticipated tool management. | INST |
| Managing "PNG" and "JPG" graphic elements. | INST |
| New values for parameters MAXGEAR1..4 (P2..5), SLIMIT (P66) and MAXSPEED (P0). | INST |
| Retracing function of 2000 blocks. | INST |
| Quick block search. | OPT |
| Local subroutines within a program. | PRG |
| Avoid spindle stop with M30 or RESET. Spindle parameter SPDLSTOP (P87). | INST |
| Programming T and M06 with associated with a subroutine in the same line. | PRG |
| New values of the OPMODE variable. | INST / PRG |
| New variables: DISABMOD, GGSN, GGSO, GGSP, GGSQ, CYCCHORDERR. | INST / PRG |
| Possibility to set the parameters of SERCOS nodes in a non-sequential order. | INST |
| WRITE instruction: "$" character followed by "P". | PRG |
| Cancel additive handwheel offset with G04 K0. General parameter ADIMPG (P176). | INST / PRG |
| Ethernet parameter NFSPROTO (P32). TCP or UDP protocol selection. | INST |
| Face thread repair cycle. | OPT TC |
| Penetration increment (step) in thread repair. | INST / OPT TC |
| API compliant thread. | OPT TC |
| Roughing by segments in inside profiling cycles 1 and 2. | INST / OPT TC |
| Programming the Z increment and the angle on threads. | INST / OPT TC |
| Reversal of the starting and final point of the face thread repair. | INST / OPT TC |
| Manual tool calibration without stopping the spindle during each step. | INST / OPT TC |

**Software V02.03**                                                                                      **July 2014**

| List of features | Manual |
|---|---|
| Set PAGE and SYMBOL instructions support PNG and JPG/JPEG formats. | PRG |
| New values for parameters MAXGEAR1..4 (P2..5), SLIMIT (P66), MAXSPEED (P0) and DFORMAT (P1). | INST |

**Software V02.10**                                                                                  **November 2014**

| List of features | Manual |
|---|---|
| Incremental zero offset (G158). | INST / PRG |
| Programs identified with letters. | OPT |
| Variables PRGN and EXECLEV. | INST |
| Korean language. | INST |
| Change of default value for general machine parameters: MAINOFFS (P107), MAINTASF (P162) and FEEDTYPE (P170). | INST |
| New variable EXTORG. | INST / PRG |
| Image handling via DNC. | PRG |
| Save/restore a trace of the oscilloscope. | OPT |

**Software V02.21**                                                                                      **July 2015**

| List of features | Manual |
|---|---|
| PLC library. | INST |
| Zero offsets table in ISO mode. | OPT |
| Compensation of the elastic deformation in the coupling of an axis. | INST |
| Machine axis parameter DYNDEFRQ (P103). | INST |
| Change of maximum value of axis and spindle parameter NPULSES. | INST |

**Software V02.22**                                                                                     **March 2016**

| List of features | Manual |
|---|---|
| Axis filters for movements with the handwheel. General machine parameter HDIFFBAC (P129) and machine axis parameter HANFREQ (P104). | INST |
| Change of maximum value of axis and spindle parameter NPULSES. | INST |

**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

**FAGOR**

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

# SAFETY CONDITIONS

Read the following safety measures in order to prevent harming people or damage to this product and those products connected to it.

The unit can only be repaired by personnel authorized by Fagor Automation.

Fagor Automation shall not be held responsible of any physical or material damage originated from not complying with these basic safety rules.

## PRECAUTIONS AGAINST PERSONAL HARM

- Interconnection of modules.

  Use the connection cables provided with the unit.

- Use proper Mains AC power cables

  To avoid risks, use only the Mains AC cables recommended for this unit.

- Avoid electric shocks.

  In order to avoid electrical discharges and fire hazards, do not apply electrical voltage outside the range selected on the rear panel of the central unit.

- Ground connection.

  In order to avoid electrical discharges, connect the ground terminals of all the modules to the main ground terminal. Before connecting the inputs and outputs of this unit, make sure that all the grounding connections are properly made.

- Before powering the unit up, make sure that it is connected to ground.

  In order to avoid electrical discharges, make sure that all the grounding connections are properly made.

- Do not work in humid environments.

  In order to avoid electrical discharges, always work under 90% of relative humidity (non-condensing) and 45 ºC (113º F).

- Do not operate this unit in explosive environments.

  In order to avoid risks, harm or damages, do not work in explosive environments.

Safety conditions

- 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 that could suffer or cause when installed under other conditions (residential or domestic environments).

- 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.
  - Etc.

- 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 tool.

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

  - DC relay coils. Diode type 1N4000.
  - AC relay coils. RC connected as close to the coils as possible with approximate values of R=220 $\Omega$ / 1 W y C=0,2 µF / 600 V.
  - AC motors. RC connected between phases, with values of R=300 $\Omega$ / 6 W y C=0,47 µF / 600 V.

- Use the proper power supply.

  Use an external regulated 24 Vdc power supply for the inputs and outputs.

- 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.

  It is recommended to connect them using shielded cables and connecting their shields (mesh) to the corresponding pin.

- Ambient conditions.

  The working temperature must be between +5 ºC and +40 ºC (41ºF and 104º F)

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

- Monitor enclosure (CNC 8055) or central unit ( CNC 8055i)

  Guarantee the required gaps between the monitor or the central unit and each wall of the enclosure. Use a DC fan to improve enclosure ventilation.

- Power switch.

  This power switch must be mounted in such a way that it is easily accessed and at a distance between 0.7 meters (27.5 inches) and 1.7 meters (5.5ft) off the floor.

**FAGOR** ⊐

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

## PROTECTIONS OF THE UNIT ITSELF (8055)

- "Axes" and "Inputs-Outputs" modules.

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

  They are protected by an external fast fuse (F) of 3.15 A 250V against overvoltage of the external power supply (over 33 Vdc) and against reverse connection of the power supply.

- Monitor.

  The type of protection fuse depends on the type of monitor. See identification label of the unit itself.

## PROTECTIONS OF THE UNIT ITSELF (8055I)

- Central unit.

  It has a 4 A 250V external fast fuse (F).
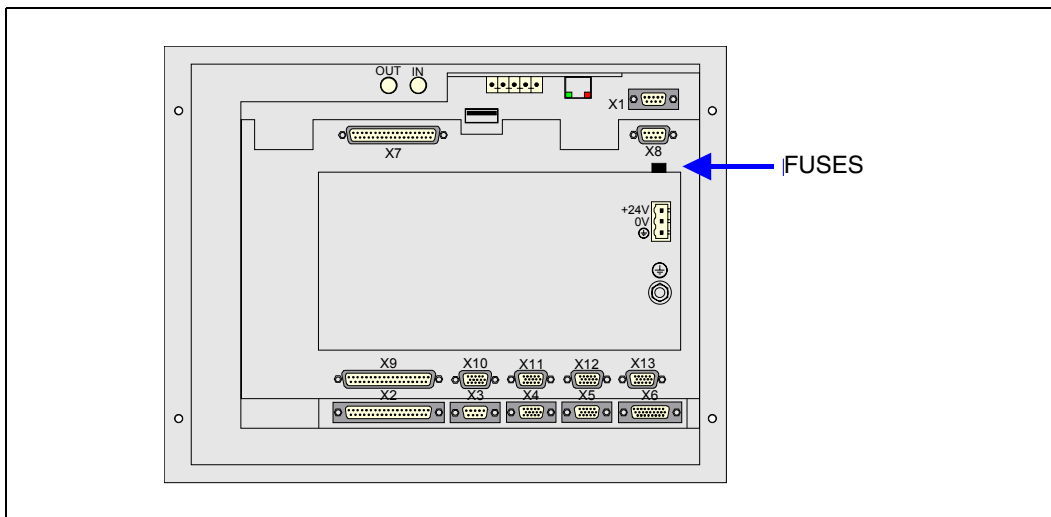


- Inputs-Outputs.

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

**FAGOR** ⟩

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

# PRECAUTIONS DURING REPAIRS

Do not manipulate 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 manipulating the connectors (inputs/outputs, feedback, etc.) make sure that the unit is not connected to AC power.

# SAFETY SYMBOLS

• Symbols that may appear in the manual.

Symbol for danger or prohibition.
It indicates actions or operations that may cause damage to people or to units.

Warning or caution symbol.
It indicates situations that could be caused by certain operations and the actions to take to prevent them.

Mandatory symbol.
It indicates actions or operations that MUST be carried out.

Information symbol.
It indicates notes, warnings and advises.

# RETURNING CONDITIONS

When sending the central nit or the remote modules, pack them in its original package and 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 pounds).

2. Attach a label indicating the owner of the unit, person to contact, type of unit and serial number.

3. In case of failure, also indicate the symptom and a short description of the failure.

4. Protect the unit wrapping it up with a roll of polyethylene or with similar material.

5. When sending the central unit, protect especially the screen.

6. Pad the unit inside the cardboard box with polyurethane foam on all sides.

7. Seal the cardboard box with packaging tape or with industrial staples.

FAGOR

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

# ADDITIONAL NOTES

Mount the CNC away from coolants, chemical products, blows, etc. which could damage it. Before turning the unit on, verify that the ground connections have been made properly.

To prevent electrical shock at the central unit of the 8055 CNC, use the proper mains AC connector at the power supply module. Use 3-wire power cables (one for ground connection).
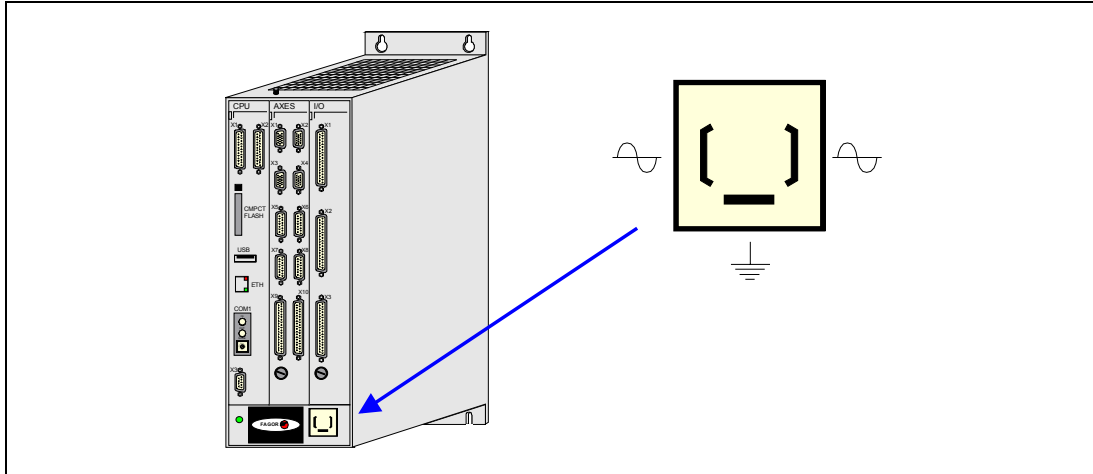


To prevent electrical shock at the monitor of the 8055 CNC, use the proper mains AC connector (A) with 3-wire power cables (one of them for ground connection).
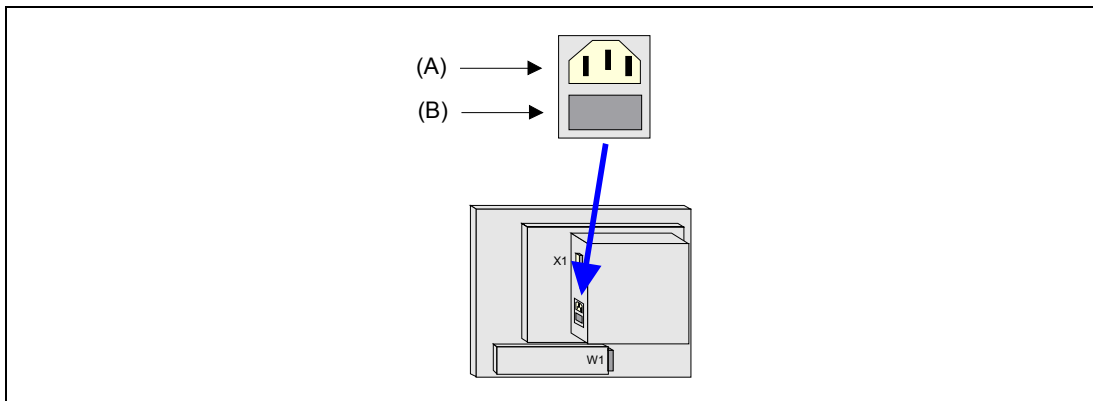


Before turning on the monitor of the 8055 CNC and verifying that the external AC line (B) fuse of each unit is the right one. See identification label of the unit itself.

In case of a malfunction or failure, disconnect it and call the technical service. Do not get into the inside of the unit.

FAGOR

FAGOR AUTOMATION

CNC 8055
CNC 8055i

**FAGOR**

FAGOR AUTOMATION

CNC 8055
CNC 8055i

# FAGOR DOCUMENTATION

**OEM manual**

It is directed to the machine builder or person in charge of installing and starting-up the CNC.

**USER-M manual**

Directed to the end user.

It describes how to operate and program in M mode.

**USER-T manual**

Directed to the end user.

It describes how to operate and program in T mode.

**MC Manual**

Directed to the end user.

It describes how to operate and program in MC mode.

It contains a self-teaching manual.

**TC Manual**

Directed to the end user.

It describes how to operate and program in TC mode.

It contains a self-teaching manual.

**MCO/TCO model**

Directed to the end user.

It describes how to operate and program in MCO and TCO mode.

**Examples-M manual**

Directed to the end user.

It contains programming examples for the M mode.

**Examples-T manual**

Directed to the end user.

It contains programming examples for the T mode.

**WINDNC Manual**

It is directed to people using the optional DNC communications software.

It is supplied in a floppy disk with the application.

**WINDRAW55 Manual**

Directed to people who use the WINDRAW55 to create screens.

It is supplied in a floppy disk with the application.

FAGOR

FAGOR AUTOMATION

CNC 8055
CNC 8055i

**FAGOR**

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

# GENERAL CONCEPTS

**1**

The CNC may be programmed at the machine (from the front panel) and from a peripheral (computer). Memory available to the user for carrying out the part programs is 1 Mbyte.

The part programs and the values in the tables which the CNC has can be entered from the front panel, from a pc (DNC) or from a peripheral.

### Entering programs and tables from the front panel.

Once the editing mode or desired table has been selected, the CNC allows you to enter data from the keyboard.

### Entering programs and tables from a Computer (DNC) or peripheral device.

The CNC allows data to be exchanged with a computer or peripheral device, using the RS232C serial line.

If this is controlled from the CNC, it is necessary to preset the corresponding table or part program directory (utilities) you want to communicate with.

Depending on the type of communication required, the serial port machine parameter "PROTOCOL" should be set.

    "PROTOCOL" = 0    if the communication is with a peripheral device.

    "PROTOCOL" = 1    if the communication is via DNC.

## 1.1    Part programs

The operating manual describes the different operating modes. Refer to that manual for further information.

### Editing a part-program

To create a part-program, access the –Edit– mode.

The new part-program edited is stored in the CNC's RAM memory. A copy of the part-programs may be stored in the hard disk (KeyCF) at a PC connected through the serial line or in the USB disk.

To transmit a program to a PC through the serial, proceed as follows:

1. Execute the "*WinDNC.exe*" application program at the PC.
2. Activate DNC communications at the CNC.
3. Select the work directory at the CNC. It is selected from the –Utilities– mode, option Directory \Serial L \Change directory

In –Edit– mode, it is possible to modify part-programs residing in the CNC's RAM memory. To modify a program stored in the hard disk (KeyCF), in a PC or in the USB disk, it must be previously copied into RAM memory.

### Executing and editing a part-program

Part-programs stored anywhere may be executed or simulated. Simulation is carried out in the –Simulation– mode, whereas the execution is done in the –Automatic– mode

When executing or simulating a part-program, bear in mind the following points:

- Only subroutines stored in the CNC's RAM memory can be executed. Therefore, to execute a subroutine stored in the hard disk (KeyCF), in a PC or in the USB disk, it must be first copied into the CNC's RAM memory.
- The GOTO and RPT instructions cannot be used in programs that are executed from a PC connected through the serial line.
- From a program in execution, it is possible to execute another program located in RAM memory, in the hard disk (KeyCF) or in a PC using the EXEC instruction.

The user customizing programs must be in RAM memory so the CNC can execute them.

### –Utilities– operating mode.

The –Utilities– mode, lets display the part-program directory of all the devices, make copies, delete, rename and even set the protections for any of them.

*Operations that may be carried out with part-programs.*

|  | RAM memory | Hard disk | DNC |
|---|---|---|---|
| See the program directory of ... | Yes | Yes | Yes |
| See the subroutine directory of ... | Yes | No | No |
| Create the work directory from ... | No | No | No |
| Change the work directory from ... | No | No | Yes |
| Edit a program from ... | Yes | Yes | No |
| Modify a program from ... | Yes | Yes | No |
| Delete a program from ... | Yes | Yes | Yes |
| Copy from/to RAM memory to/from ... | Yes | Yes | Yes |
| Copy from/to HD to/from ... | Yes | Yes | Yes |
| Copy from/to DNC to/from ... | Yes | Yes | Yes |
| Rename a program from ... | Yes | Yes | No |
| Change the comment of a program from ... | Yes | Yes | No |
| Change the protections of a program from ... | Yes | Yes | No |
| Execute a part-program from ... | Yes | Yes | Yes |
| Execute a user program from ... | Yes | Yes | No |
| Execute a PLC program from ... | Yes | No | No |
| Execute programs with GOTO or RPT instructions from ... | Yes | Yes | No |
| Execute subroutines residing in ... | Yes | No | No |
| Execute programs with the EXEC instruction, in RAM from ... | Yes | Yes | Yes |
| Execute programs with the EXEC instruction, in HD from ... | Yes | Yes | Yes |
| Execute programs with the EXEC instruction, in DNC from ... | Yes | Yes | No |
| Open programs with the OPEN instruction, in RAM from ... | Yes | Yes | Yes |
| Open programs with the OPEN instruction, in HD from ... | Yes | Yes | Yes |
| Open programs with the OPEN instruction, in DNC from ... | Yes | Yes | No |
| Via Ethernet: | | | |
| See from a PC the program directory of ... | No | Yes | No |
| See from a PC the subroutine directory of ... | No | No | No |
| See from a PC, a directory in ... | No | No | No |

(*) If it is not in RAM memory, it generates the executable code in RAM and it executes it.

## Ethernet

When having the Ethernet option and if the CNC is configured as another node within the computer network, the following operations are possible from any PC of the network:

- Access the part-program directory of the hard disk (KeyCF).
- Edit, modify, delete, rename, etc. the programs stored on the hard disk.
- Copy programs from the hard disk to the PC and vice versa.

To configure the CNC as another node within the computer network, see the installation manual.

### 1.1.1 Considerations regarding the Ethernet connection

When configuring the CNC as another node in the computer network, the programs stored in the hard disk (KeyCF) may be edited and modified from any PC.

## Instructions for setting up a PC to access CNC directories

To set up the PC to access the CNC directories, we recommend to proceed as follows.

1. Open the "Windows Explorer"

2. On the "Tools" menu, select the "Connect to Network Drives" option.

3. Select the drive, for example "D".

4. Indicate the path. The path will be the CNC name followed by the name of the shared directory.

   For example: \\FAGORCNC\CNCHD

5. When selecting the option: "Connect again when initiating the session", the selected CNC will appear on each power-up as another path of the "Windows Explorer" without having to define it again.

## Data format

This connection is established through Ethernet and, therefore, the CNC does not control the syntax of the programs while they are received or modified. However, whenever accessing the program directory of the Hard Disk (HD), the following verification takes place:

### *File name.*

The file number must always have 6 digits and the extension PIM (for milling) or PIT (for lathe).

   Examples:    001204.PIM   000100.PIM   123456.PIT    020150.PIT

If the file has been given the wrong name, for example: 1204.PIM or 100.PIT, the CNC will not change it, but it will display it with the comment "*****************". The file name cannot be modified at the CNC; it must be edited from the PC to correct the error.

### *File size.*

If the file is empty (size = 0) the CNC will display it with the comment "*********************".

The file can be edited or deleted either from the CNC or from the PC.

### *First line of the program.*

The first line of the program must have the % character, the comment associated with the file (up to 20 characters) and between the two commas (,) the program attributes O (OEM), H (hidden), M (modifiable), X (executable).

   Examples:     %Comment ,MX,

                 % ,OMX,

If the first line does not exist, the CNC will display the program with an empty comment and with the modifiable (M) and executable (X) attributes.

When the format of the first line is wrong, the CNC does not modify it, but it displays it with the comment "****************". The file can be edited or deleted either from the CNC or from the PC.

The format is incorrect when the comment has more than 20 characters, a comma (,) is missing to group the attributes or there is a strange character in the attributes.

## 1.2    **DNC connection**

The CNC offers as optional feature the possibility of working in DNC (Distributed Numerical Control), enabling communication between the CNC and a computer to carry out the following functions:

- Directory and delete commands.
- Transfer of programs and tables between the CNC and a computer.
- Remote control of the machine.
- The ability to supervise the status of advanced DNC systems.

**1.**

**GENERAL CONCEPTS**
DNC connection

**FAGOR**

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

## 1.3    Communication protocol via DNC or peripheral device

This type of communication enables program-and-table transfer commands, plus the organization of CNC directories such as the computer directory, for copying/deleting programs, etc. to be done either from the CNC or the computer.

When you want to transfer files, it is necessary to follow this protocol:

- The "%" symbol will be used to start the file, followed by the program comment (optional), of up to 20 characters.

  Then, and separated by a comma ",", comes the protection of each file, read, write, etc. These protections are optional, since their programming is not obligatory.

  To end the file header, RT (RETURN ) or LF (LINE FEED) characters should be sent separated by a comma (",").

  Example:        %Fagor Automation, MX, RT

- Following the header, the file blocks should be programmed. These will all be programmed according to the programming rules indicated in this manual. After each block, to separate it from the others, the RT (RETURN ) or LF (LINE FEED) characters should be used.

  Example:        N20 G90 G01 X100 Y200 F2000 LF

                          (RPT N10, N20) N3 LF

If communication is made with a peripheral device, you will need to send the 'end of file' command. This command is selected via the machine parameter for the serial port: "EOFCHR", and can be one of the following characters :

| | |
|---|---|
| ESC | ESCAPE |
| EOT | END OF TRANSMISSION |
| SUB | SUBSTITUTE |
| EXT | END OF TRANSMISSION |

### Image handling via DNC

While using WinDNC (version V6.01 or later), it is possible to send and receive PNG, JPG/JPEG and BMP type images via DNC.

#### *WinDNC software:*

Version V6.01 of WinDNC supports files with extension bmp, png, jpg and jpeg. The maximum length or the file name is 16 characters (including the dot and the extension).

The application scans all image type files in the work folder. When sending the files, if the name of any file is too long, it will ask the user to enter a new shorter name (up to 16 characters). It must also keep the original extension.

# CREATING A PROGRAM

**2**

A CNC program consists of a series of blocks or instructions. These blocks or instructions are made of words composed of capital letters and numerical format.

The CNC's numerical format consists of :

- The signs . (decimal points, + (plus), - (minus).
- Digits 0 1 2 3 4 5 6 7 8 9.

Programming allows spaces between letters, numbers and symbols, in addition to ignoring the numerical format if it has zero value, or a symbol if it is positive.

The numeric format of a word may be replaced by an arithmetic parameter when programming. Later on, during execution, the CNC will replace the arithmetic parameter by its value. For example, if XP3 has been programmed, during execution the CNC will replace P3 by its numerical value, obtaining results such as X20, X20.567, X-0.003, etc.

## 2.1 Program structure at the CNC

All the blocks which make up the program have the following structure:

Block header + program block + end of block

### 2.1.1 Block header

The block header is optional, and may consist of one or more block skip conditions and by the block number or label. Both must be programmed in this order.

#### *Block skip condition. "/", "/1", "/2", "/3".*

These three block skip conditions, given that "/" and "/1" are the same, they are governed by the marks BLKSKIP1, BLKSKIP2 and BLKSKIP3 of the PLC. If any of these marks is active, the CNC will not execute the block or blocks in which it has been programmed; the execution takes place in the following block.

Up to 3 skip conditions can be programmed in one block; they will be evaluated one by one, respecting the order in which they have been programmed.

The control reads 200 blocks ahead of the one being executed in order to calculate in advance the path to be run. The condition for block skip will be analyzed at the time when the block is read i.e. 200 blocks before execution.

If the block skip needs to be analyzed at the time of execution, it is necessary to interrupt the block preparation, by programming G4 in the previous block.

#### *Label or block number. N(0-99999999).*

This is used to identify the block, and is only used when block references or jumps are made. They are represented by the letter N followed by up to 8 digits (0-99999999).

No particular order is required and the numbers need not be sequential. If two or more blocks with the same label number are present in the same program, the CNC will always give priority to the first number.

Although it is not necessary to program it, by using a softkey the CNC allows the automatic programming of labels. The programmer can select the initial number and the step between labels.

**Restrictions:**

- Displaying the active block number at the top window on the screen:
  - When executing a program in ISO mode, when the label number is higher than 9999, it displays N**** .
  - On the "DISPLAY / SUBROUTINES" window, when displaying an RPT that has a label higher than 9999, it displays it with ****.
- Canned cycles G66, G67 and G68 (irregular pockets with islands) can only be edited using 4-digit labels.

### 2.1.2    Program block

This is written with commands in ISO and high level languages. To prepare a program, blocks written in both languages will be used, although each one should be edited with commands in just one language.

#### *ISO language.*

This language is specially designed to control axis movement, as it gives information and movement conditions, in addition to data on feedrate. It offers the following types of functions.

- Preparatory functions for movement, used to determine geometry and working conditions, such as linear and circular interpolations, threading, etc.
- Control functions for axis feedrate and spindle speeds.
- Tool control functions.
- Complementary functions, with technological instructions.

#### *High level language.*

This enables access to general purpose variables and to system tables and variables.

It gives the user a number of control sentences which are similar to the terminology used in other languages, such as IF, GOTO, CALL, etc. Also, it allows the use of any type of arithmetic, relational or logical expression.

It also has instructions for the construction of loops, plus subroutines with local variables. A local variable is one that is only recognized by the subroutine in which it has been defined.

It is also possible to create libraries, grouping subroutines with useful and tested functions, which can be accessed from any program.

### 2.1.3     End of block

The end of block is optional and may consist of the indication of number of repetitions of the block and of the block comment. Both must be programmed in this order.

*Number of block repetitions. N(0-9999)*

This indicates the number of times the block will be executed. The number of repetitions is represented by the letter N followed by up to 4 digits (0-9999). The active machining operation does not take place if N0 is programmed; only the movement programmed within the block takes place.

Movement blocks can only be repeated which, at the time of their execution, are under the influence of a modal subroutine. In these cases, the CNC executes the programmed move and the active machining operation (canned cycle or modal subroutine) the indicated number of times.

*Block comment*

The CNC allows you to incorporate any kind of information into all blocks in the form of a comment. The comment is programmed at the end of the block, and should begin with the character ";" (semicolon).

If a block begins with ";" all its contents will be considered as a comment, and it will not be executed.

Empty blocks are not permitted. They should contain at least one comment.

## 2.2    Local subroutines within a program

A subroutine is a part of a program which, being properly identified, can be called from any position of a program to be executed.

Local subroutines may be defined within a program. These subroutines are executed from RAM or hard disk memory.

The local subroutines are defined as part of a program. These subroutines may only be called upon from the program where it has been defined.

### Programming

The local subroutines are located at the beginning of the program, before the actual beginning of the program. Local subroutines are defined by programming (LSUB n), where n indicates the subroutine number. Followed by the contents of the subroutine.

The range of local subroutines is from 0 to 9999.

    (LSUB 0)
    (LSUB 9999)

The actual beginning of the program is identified with the % sign. Any text may follow this character.

A local subroutine may be called upon using the commands CALL, PCALL or MCALL. When executing the calls, it first looks for the subroutines defined as local in that program and having matching names. If there aren't any, it will look among the global subroutines.

To execute a local subroutine directly, program (LL n). This way, only the local subroutine will be executed. If this subroutine does not exist, it will not execute anything and it will issue an error message indicating undefined subroutine.

Up to 100 local subroutines may be defined in a program. The maximum local subroutine nesting level is 15.

***Examples:***

Example 1:

```
(LSUB9505)
X100
(RET)

%**** ; beginning of program
(CALL 9505)
M30
```

Example 2:

```
(LSUB9505)
X100
(RET)

%**** ; beginning of program
(LL9505)
M30
```

**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

*Executing programs:*

(LL n) Call to a local subroutine.
   Parameters cannot be initialized with this command.
(CALL n) Call to a local or global subroutine.
   Parameters cannot be initialized with this command.
(PCALL n ...) Call to a global or local subroutine.
   Local parameters can be initialized with this command.
(MCALL n ...) Modal call to a local or global subroutine.
   Local parameters can be initialized with this command.

*Limitations:*

A local subroutine can call a global subroutine but a global subroutine cannot call a local subroutine except if that local subroutine is defined in the root program; in other words, in the first program that is executed.

Local subroutines defined inside a program that has been called with the "EXEC" command are ignored. It only takes into account the ones defined in the root program.

It only takes into account local subroutines that are in programs that are executed from the CNC execution channel, either in ISO mode or in conversational mode. Local subroutines cannot be executed from the PLC channel.

# AXES AND COORDINATE SYSTEMS

Given that the purpose of the CNC is to control the movement and positioning of axes, it is necessary to determine the position of the point to be reached through its coordinates.

The CNC allows you to use absolute, relative or incremental coordinates throughout the same program.

## 3.1    Axis nomenclature

The axes are named according to DIN 66217.
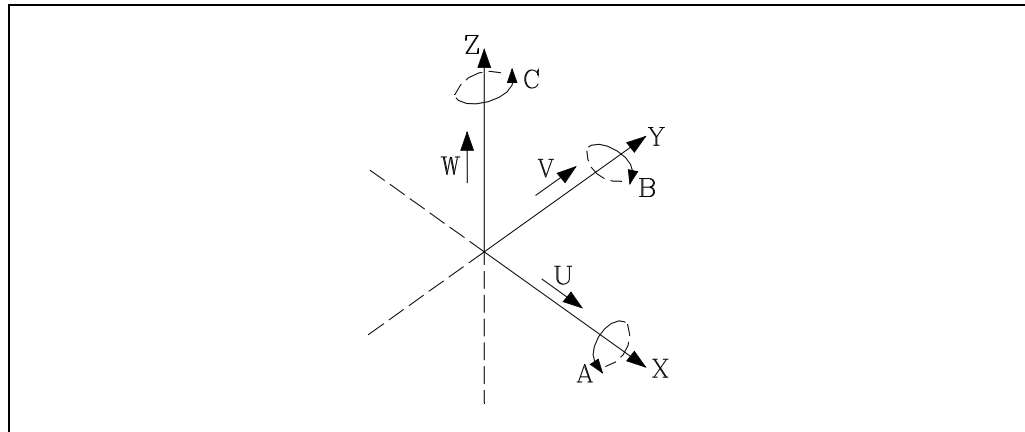
Characteristics of the system of axes:

X and Y        main movements on the main work plane of the machine.

Z                   parallel to the main axis of the machine, perpendicular to the main XY plane.

U, V, W        auxiliary axes parallel to X, Y, Z respectively.

A, B, C        Rotary axes on each axis X, Y, Z.

The drawing below shows an example of the nomenclature of the axes on a milling-profiling machine with a tilted table.



**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

·**M**· & ·**EN**· **MODELS**
**SOFT: V02.2X**
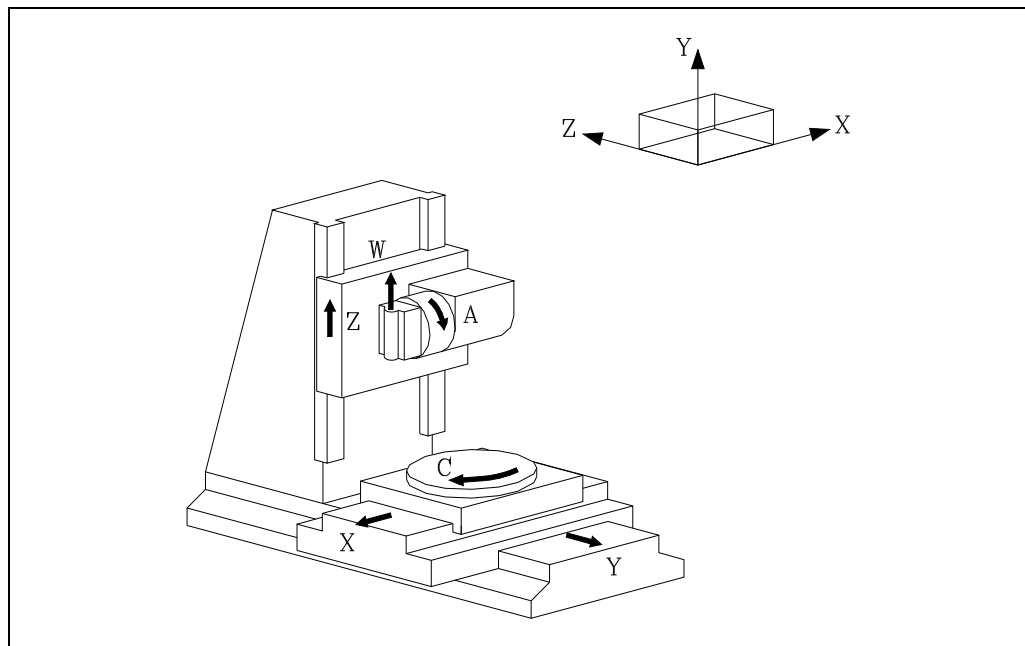
### 3.1.1    Axis selection

Of the 9 possible axes that may exist, the CNC allows the manufacturer to select up to 7 of them.

Moreover, all the axes should be suitably defined as linear/rotary, etc. through the axis machine parameters which appear in the Installation and Start-up Manual.

There is no limitation to the programming of the axes, and up to 7 axes may be interpolated at the same time.

**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

## 3.2 Plane selection (G16, G17, G18, G19)

Plane selection should be made when the following are carried out :

- Circular interpolations.
- Controlled corner rounding.
- Tangential entry and exit.
- Chamfer.
- Coordinate programming in Polar coordinates.
- Machining canned cycles.
- Rotation of the coordinate system.
- Tool radius compensation.
- Tool length compensation.

The "G" functions which enable selection of work planes are as follows :

G16 axis1 axis2 axis3. Enables selection of the desired work plane, plus the direction of G02 G03 (circular interpolation), axis1 being programmed as the abscissa axis and axis2 as the ordinate axis.

The axis3 is the longitudinal axis along which tool length compensation is applied.
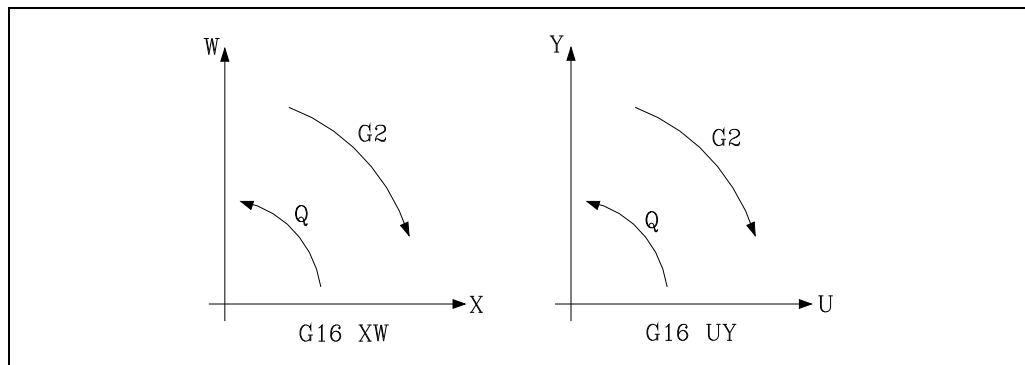


G17.          Selects the XY plane and the Z axis as longitudinal axis.

G18.          Selects the ZX plane and the Y axis as longitudinal axis.

G19.          Selects the YZ plane and the X axis as longitudinal axis.

The G16, G17, G18 and G19 functions are modal and incompatible among themselves. The G16 function should be programmed on its own within a block.
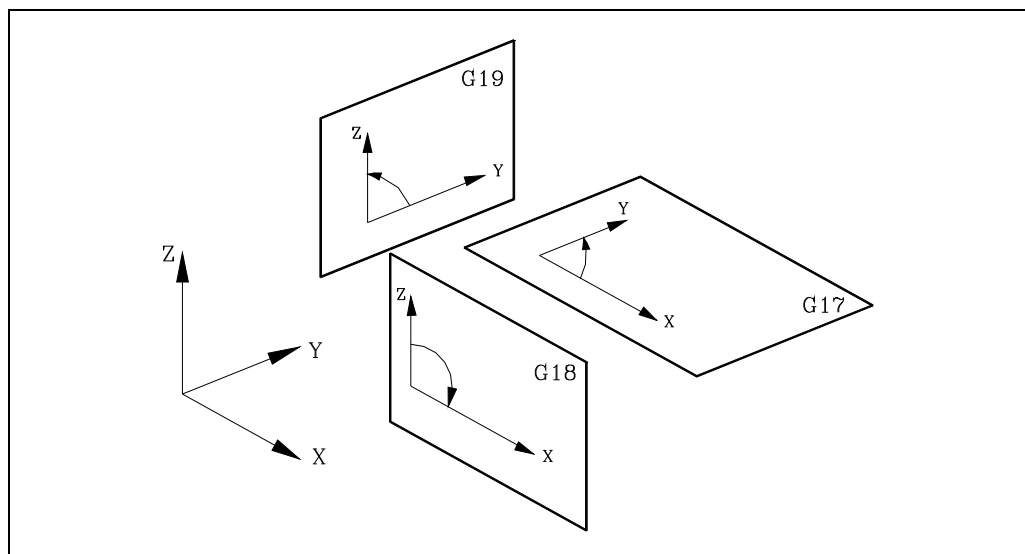


The G17, G18, and G19 functions define two of the three main axes (X, Y, Z) as belonging to the work plane, and the other as the perpendicular axis to the same.

When radius compensation is done on the work plane, and length compensation on the perpendicular axis, the CNC does not allow functions G17, G18, and G19 if any one of the X, Y, or Z axes is not selected as being controlled by the CNC.

On power-up, after executing M02, M30 or after EMERGENCY or RESET, the CNC will assume that the plane defined by the general machine parameter as "IPLANE" is the work plane.

> **i**
>
> *Function G49, coordinate transformation, must be used to machine inclined planes.*
> *See chapter "15  Coordinate transformation".*

**3.**

**AXES AND COORDINATE SYSTEMS**
Plane selection (G16, G17, G18, G19)

**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

## 3.3    Part dimensioning. Millimeters (G71) or inches (G70)

The CNC allows you to enter units of measurement with the programming, either in millimeters or inches.

It has a general machine parameter "INCHES" to define the unit of measurement of the CNC.

However, these units of measurement can be changed at any time in the program. Two functions are supplied for this purpose :

- G70. Programming in inches.
- G71. Programming in millimeters.

Depending on whether G70 or G71 has been programmed, the CNC assumes the corresponding set of units for all the blocks programmed from that moment on.

The G70 and G71 functions are modal and are incompatible.

The CNC allows you to program figures from 0.00001 to 99999.9999 with or without sign, working in millimeters (G71), called format +/-5.4, or either from 0.00001 to 3937.00787 with or without sign if the programming is done in inches (G70), called format +/-4.5.

However, and to simplify the instructions, we can say that the CNC admits +/- 5.5 format, thereby admitting +/- 5.4 in millimeters and +/- 4.5 in inches.

On power-up, after executing M02, M30 or after EMERGENCY or RESET, the CNC will assume that the system of units of measurement is the one defined by the general machine parameter "INCHES".

## 3.4   Absolute/incremental programming (G90, G91)

The CNC allows the programming of the coordinates of one point either with absolute G90 or incremental G91 values.

When working with absolute coordinates (G90), the point coordinates refer to a point of origin of established coordinates, often the part zero (datum).

When working in incremental coordinates (G91), the numerical value programmed corresponds to the movement information for the distance to be traveled from the point where the tool is situated at that time. The sign in front shows the direction of movement.

The G90/G91 functions are modal and incompatible with each other.



**Absolute coordinates:**

| G90 | X0 | Y0 | ; Point P0 |
| | X150.5 | Y200 | ; Point P1 |
| | X300 | | ; Point P2 |
| | X0 | Y0 | ; Point P0 |

**Incremental coordinates:**

| G90 | X0 | Y0 | ; Point P0 |
| G91 | X150.5 | Y200 | ; Point P1 |
| | X149.5 | | ; Point P2 |
| | X-300 | Y-200 | ; Point P0 |
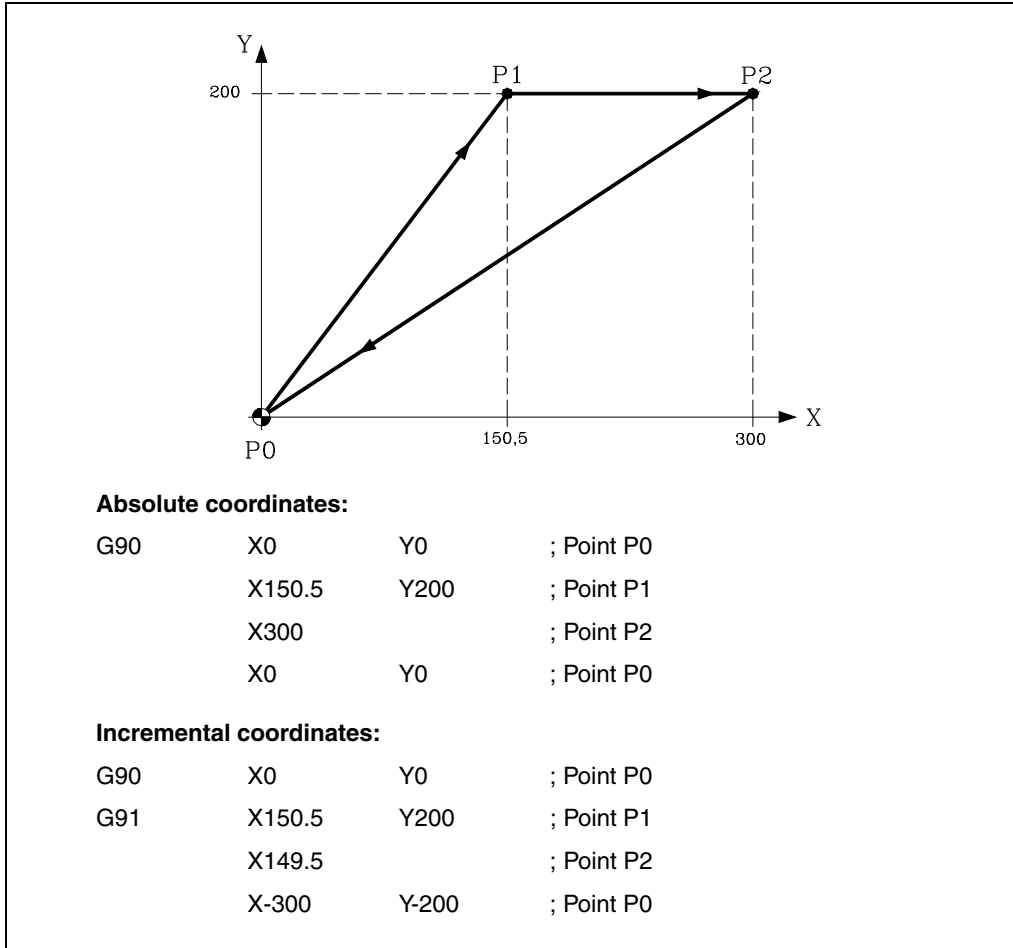
On power-up, after executing M02, M30 or after an EMERGENCY or RESET, the CNC will assume G90 or G91 according to the definition by the general machine parameter "ISYSTEM".

**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2x

## 3.5 Coordinate programming

The CNC allows the selection of up to 7 of the 9 possible axes X, Y, Z, U, V, W, A, B, C.

Each of these may be linear, linear to position only, normal rotary, rotary to position only or rotary with hirth toothing (positioning in complete degrees), according to the specification in the machine parameter of each "AXISTYPE" axis.

With the aim of always selecting the most suitable coordinate programming system, the CNC has the following types :

- Cartesian coordinates
- Polar coordinates
- Cylindrical coordinates
- Angle and Cartesian coordinate

### 3.5.1 Cartesian coordinates

The Cartesian Coordinate System is defined by two axes on the plane, and by three or more axes in space.

The origin of all these, which in the case of the axes X Y Z coincides with the point of intersection, is called Cartesian Origin or Zero Point of the Coordinate System.

The position of the different points of the machine is expressed in terms of the coordinates of the axes, with two, three, four, or five coordinates.

The coordinates of the axes are programmed via the letter of the axis (X, Y, Z, U, V, W, A, B, C, always in this order) followed by the coordinate value.

The values of the coordinates are absolute or incremental, depending on whether it is working in G90 or G91, and its programming format is ±5.5.
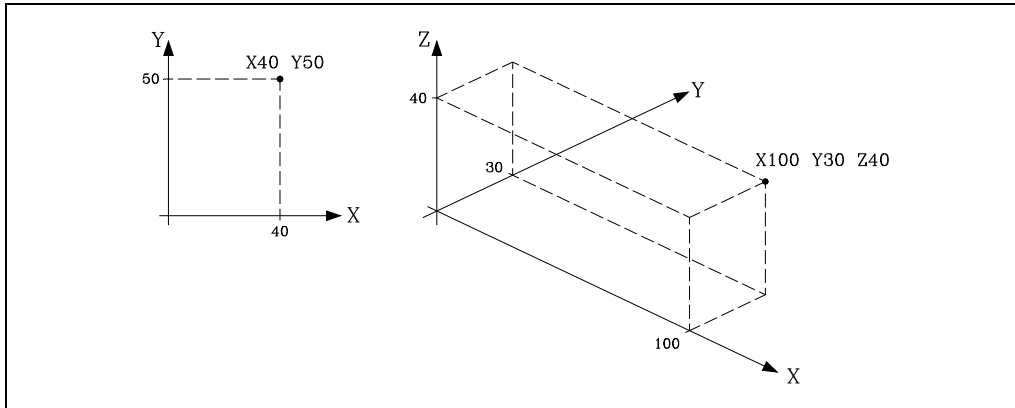
### 3.5.2 Polar coordinates

In the event of the presence of circular elements or angular dimensions, the coordinates of the different points on the plane (2 axes at the same time), it may be easier to express them in polar coordinates.

The reference point is called Polar Origin, and this will be the origin of the Polar Coordinate System.

A point on this system would be defined by :
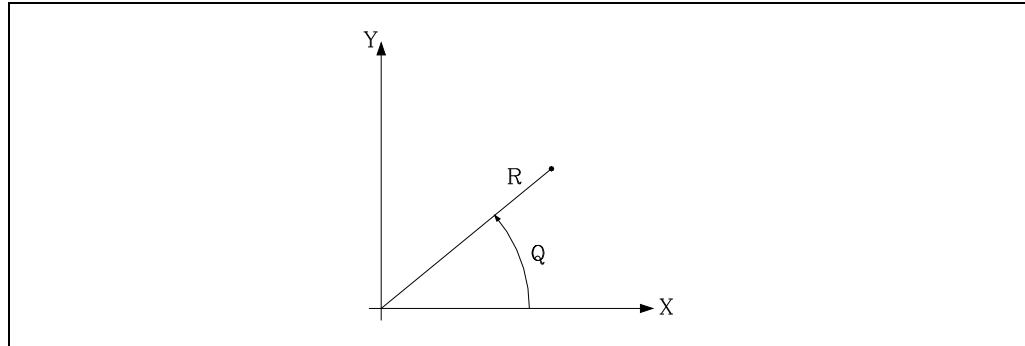


- The RADIUS (R), the distance between the polar origin and the point.

- The ANGLE (Q), formed by the abscissa axis and the line which joins the polar origin with the point. (In degrees).

The values R and Q are absolute or incremental depending on whether you are working with G90 or G91, and their programming format will be R5.5 Q±5.5. The radius value must always be positive.

The values R and Q are incremental and their programming format will be R±5.5 Q±5.5.

The R values may be negative when programming in incremental coordinates; but the resulting value assigned to the radius must always be positive.

When programming a "Q" value greater than 360º, the module will be assumed after dividing it by 360. Thus, Q420 is the same as Q60 and Q-420 is the same as Q-60.

Programming example assuming that the Polar Origin is located at the Coordinate Origin.
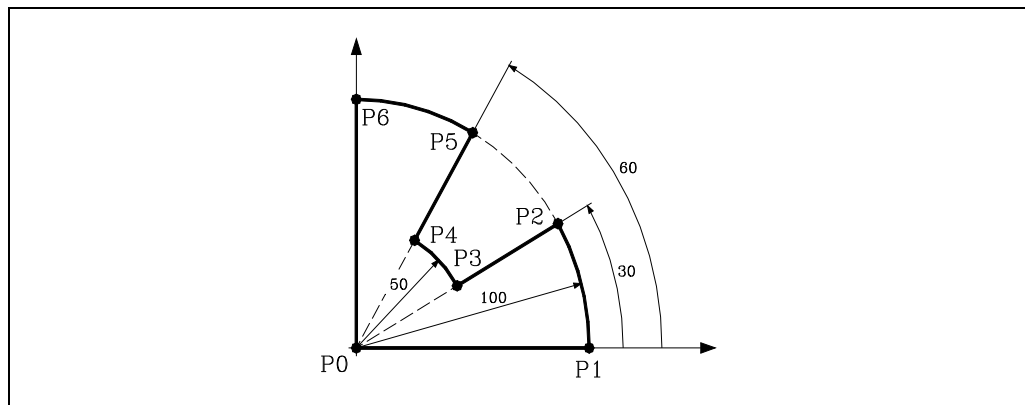


Absolute coordinates:

| G90  | X0   | Y0  | ; Point P0                              |
|------|------|-----|-----------------------------------------|
| G01  | R100 | Q0  | ; Point P1, in a straight line (G01)    |
| G01  | R50  | Q30 | ; Point P2, in an arc (G03)             |
| G01  | R50  | Q30 | ; Point P3, in a straight line (G01)    |
| G03  |      | Q60 | ; Point P4, in an arc (G03)             |
| G01  | R100 | Q60 | ; Point P5, in a straight line (G01)    |
| G03  |      | Q90 | ; Point P6, in arc (G03)                |
| G01  | R0   | Q90 | ; Point P0, in a straight line (G01)    |

Incremental coordinates:

| | | | |
|---|---|---|---|
| G90 | X0 | Y0 | ; Point P0 |
| G91 G01 | R100 | Q0 | ; Point P1, in a straight line (G01) |
| G01 | R50 | Q30 | ; Point P2, in an arc (G03) |
| G01 | R-50 | Q0 | ; Point P3, in a straight line (G01) |
| G03 | | Q30 | ; Point P2, in an arc (G03) |
| G01 | R50 | Q0 | ; Point P1, in a straight line (G01) |
| G03 | | Q30 | ; Point P6, in arc (G03) |
| G01 | R-100 | Q0 | ; Point P1, in a straight line (G01) |

The polar origin, apart from being able to be preset using function G93 (described later) can be modified in the following cases :

- On power-up, after executing M02, M30 EMERGENCY or RESET, the CNC will assume, as the polar origin, the coordinate origin of the work plane defined by the general machine parameter"IPLANE".

- Every time the work plane is changed (G16,G17,G18 or G19), the CNC assumes the coordinate origin of the new work plane selected as the polar origin.

- When executing a circular interpolation (G02 or G03), and if the general machine parameter "PORGMOVE" has a value of 1, the center of the arc will become the new polar origin.

### 3.5.3 Cylindrical coordinates

To define a point in space, the system of cylindrical coordinates can be used as well as the Cartesian coordinate system.

A point on this system would be defined by :
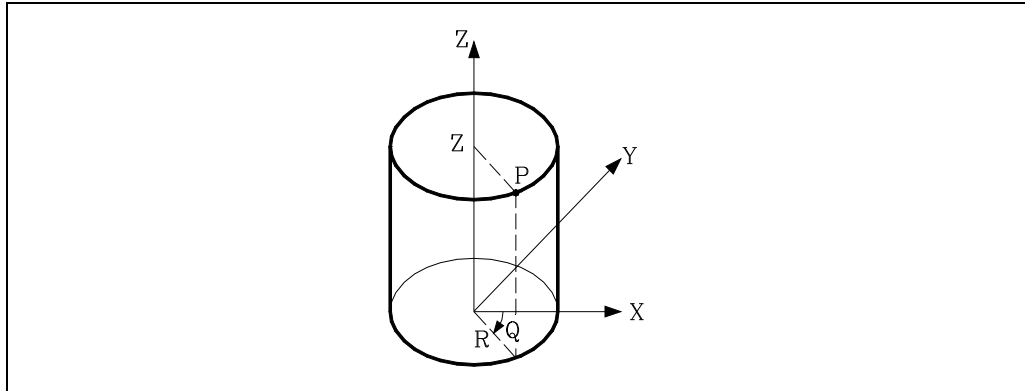


The projection of this point on the main plane, which should be defined in polar coordinates (R Q).

Rest of axes in Cartesian coordinates.

Examples:

    R30 Q10 Z100
    R20 Q45 Z10 V30 A20

### 3.5.4    Angle and Cartesian coordinate

A point on the main plane can be defined via one of its Cartesian coordinates, and the exit angle of the previous path.

Example of programming assuming that the main plane is XY:
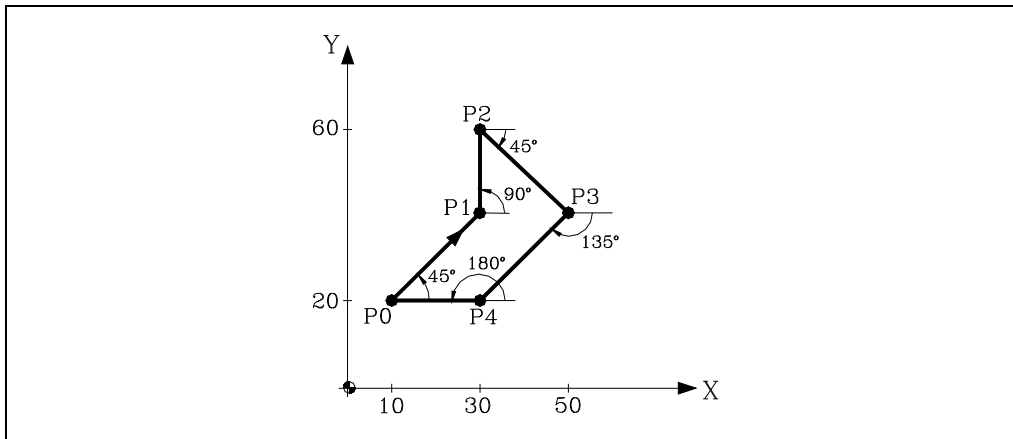


| X10   | Y20 | ; Point P0, starting point |
|-------|-----|----------------------------|
| Q45   | X30 | ; Point P1 |
| Q90   | Y60 | ; Point P2 |
| Q-45  | X50 | ; Point P3 |
| Q-135 | Y20 | ; Point P4 |
| Q180  | X10 | ; Point P0 |

To represent a point in space, the rest of the coordinates may be programmed in Cartesian coordinates.

## 3.6    Rotary axes

The types of rotary axes available are:

Normal rotary axis.

Positioning-only rotary axis.

Rotary HIRTH axis.

Each one of them can be divided into:

Rollover          When it is displayed between 0º and 360º.

Non Rollover   When it may be displayed between -99999º and 99999º.

They are all programmed in degrees. Therefore, their readings are not affected by the  inch/mm conversion.

**Normal rotary axes**

They can interpolate with linear axes.

Movement: In G00 and G01.

Rollover axis programming:

G90          The sign indicates the turning direction and the target position (between 0 and 359.9999).

G91          The sign indicates the turning direction. If the programmed movement exceeds 360º, the axis will rotate more than one turn before positioning at the desired point.

Non-rollover axis programming.

In G90 and G91 like a linear axis.

**Positioning-only rotary axis**

They cannot be interpolated with linear axes.

Movement: Always in G00 and they do not admit tool radius compensation (G41, G42).

Rollover axis programming:

G90          Always positive and in the shortest direction. End coordinate between 0 and 359.9999.

G91          The sign indicates the turning direction. If the programmed movement exceeds 360º, the axis will rotate more than one turn before positioning at the desired point.

Non-rollover axis programming.

In G90 and G91 like a linear axis.

**Rotary Hirth axis**

They work like the positioning-only axis except that they do not admit decimal position values (coordinates).

More than one hirth axis can be used, but they can only be moved one at a time.

**FAGOR**

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· Models
Soft: V02.2x

·52·

## 3.7 Work zones

The CNC provides four work zones or areas, and also limits the tool movement in each of these.

### 3.7.1 Definition of the work zones

Within each work zone, the CNC allows you to limit the movement of the tool on each axis, with upper and lower limits being defined in each axis.

G20: Defines the lower limits in the desired zone.

G21: Defines the upper limits in the desired zone.

The format to program these functions is:

G20 K X…C±5.5
G21 K X…C±5.5

Where:

| | |
|---|---|
| K | Indicates the work zone you wish to define (1, 2, 3 or 4). |
| X...C | Indicates the coordinates (upper or lower) with which you wish to limit the axes. These coordinates will be programmed with reference to machine zero (home). For safety, the axis stops 0.1mm before the programmed limit. |

It is not necessary to program all the axes, so only defined axes will be limited.
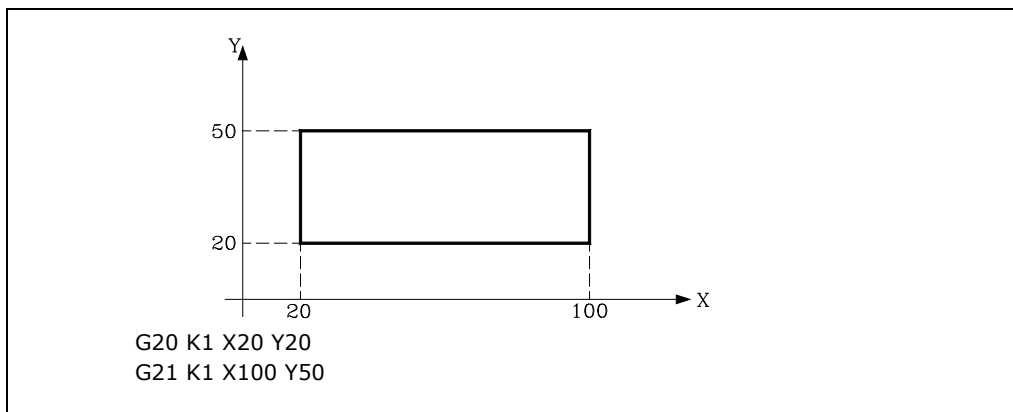


```
G20 K1 X20 Y20
G21 K1 X100 Y50
```

### 3.7.2    Using the work zones

Within each work zone, the CNC allows you to restrict the movement of the tool, either prohibiting its exit from the programmed zone (no exit zone) or its entry into the programmed zone (no entry zone).



S= 1 No entry zone          S = 2 No exit zone
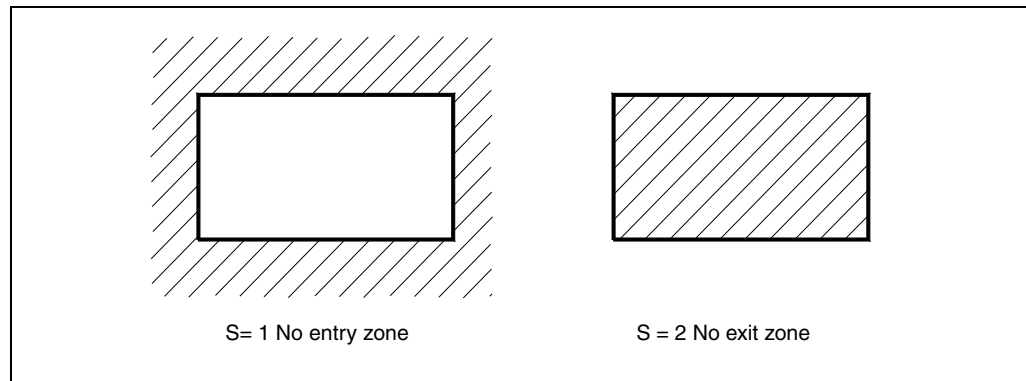
The CNC will take the dimensions of the tool into account at all times (tool offset table) to avoid it exceeding the programmed limits.

The presetting of work zones is done via Function G22, the programming format being:

G22 K S

Where:

K        Indicates the work zone you wish to set (1, 2, 3 or 4).

S        Indicates the enabling/disabling of the work zone.

S=0 disabled.

S=1 enabled as a no-entry zone.

S=2 enabled as a no-exit zone.

On power-up, the CNC will disable all work zones. However, upper and lower limits for these zones will not undergo any variation, and they can be re-enabled through the G22 function.

# REFERENCE SYSTEMS

**4**

## 4.1    Reference points

A CNC machine needs the following origin and reference points defined :

- Machine Reference Zero or home. This is set by the manufacturer as the origin of the coordinate system of the machine.

- Part zero or point of origin of the part. This is the origin point that is set for programming the measurements of the part. It can be freely selected by the programmer, and its machine reference zero can be set by the zero offset.

- Machine Reference point. This is a point on the machine established by the manufacturer around which the synchronization of the system is done. The control positions the axis on this point, instead of moving it as far as the Machine Reference Zero, taking, at this point, the reference coordinates which are defined via the axis machine parameter "REFVALUE".



| M | Machine zero |
|---|---|
| W | Part zero |
| R | Machine reference point |
| XMW, YMW, ZMW... | Coordinates of the part zero |
| XMR, YMR, ZMR... | Coordinates of machine reference point ("REFVALUE") |
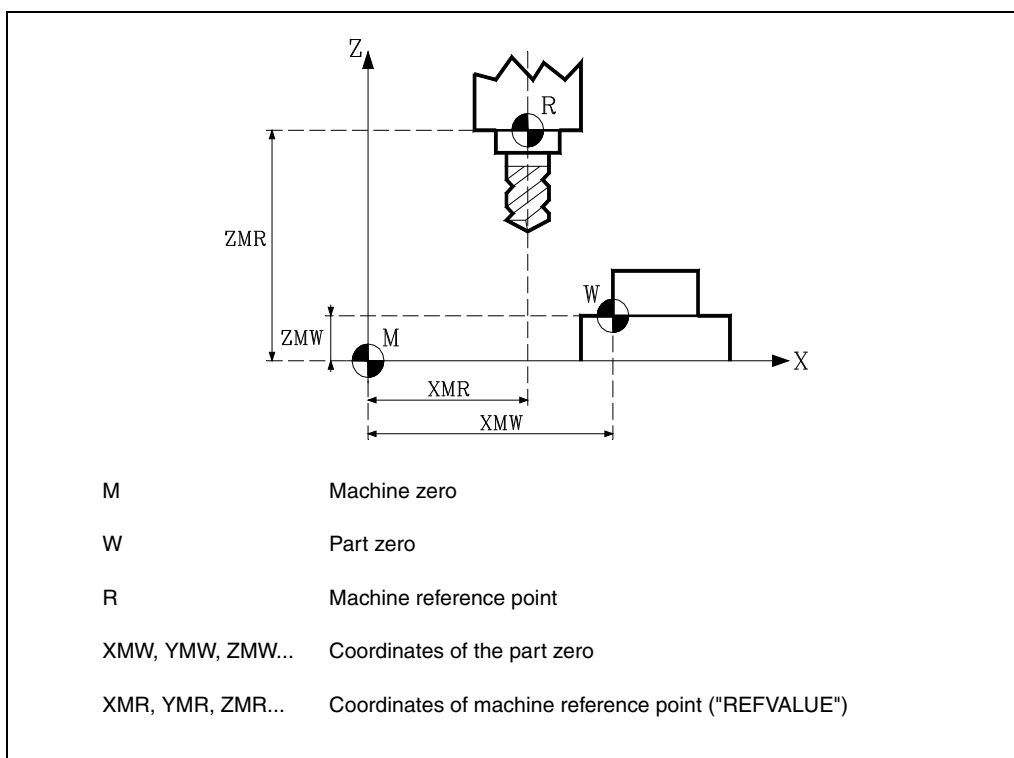
## 4.2    Machine reference (Home) search (G74)

The CNC allows you to program the machine reference search in two ways :

- Machine reference (home) search of one or more axes in a particular order.

  G74 is programmed followed by the axes in which you want to carry out the reference search. For example: G74 X Z C Y.

  The CNC begins the movement of all the selected axes which have a machine reference switch (machine axis parameter "DECINPUT") and in the direction indicated by the axis machine parameter "REFDIREC".

  This movement is carried out at the feedrate indicated by the axis machine parameter "REFEED1" for each axis until the home switch is hit.

  Next, the home search (marker pulse or home) will be carried out in the programmed order.

  This second movement will be carried out one axis at a time, at the feedrate indicated in the axis machine parameter "REFEED2" until the machine reference point is reached (i.e. the marker pulse is found).

- Home search using the associated subroutine.

  The G74 function will be programmed alone in the block, and the CNC will automatically execute the subroutine whose number appears in the general machine parameter "REFPSUB". In this subroutine it is possible to program the machine reference searches required, and also in the required order.

In a block in which G74 has been programmed, no other preparatory function may appear.

If the machine reference search is done in JOG mode, the part zero selected is lost. The coordinates of the reference point indicated in the machine axis parameter "REFVALUE" is displayed. In all other cases, the active part zero will be maintained and the CNC will display the position values with respect to that part zero.

If the G74 command is executed in MDI, the display of coordinates depends on the mode in which it is executed : Jog, Execution, or Simulation.

## 4.3    Programming with respect to machine zero (G53)

Function G53 can be added to any block that has path control functions.

It is only used when the programming of block coordinates relating to machine zero is required. These coordinates should be expressed in millimeters or inches, depending on how the general machine parameter "INCHES" is defined.

By programming G53 alone (without motion information) the current active zero offset is canceled regardless of whether it was originated by a G54-G59 or a G92 preset. This origin preset is described next.

Function G53 is not modal, so it should be programmed every time you wish to indicate the coordinates referred to machine zero.

This function temporarily cancels radius and tool length compensation.
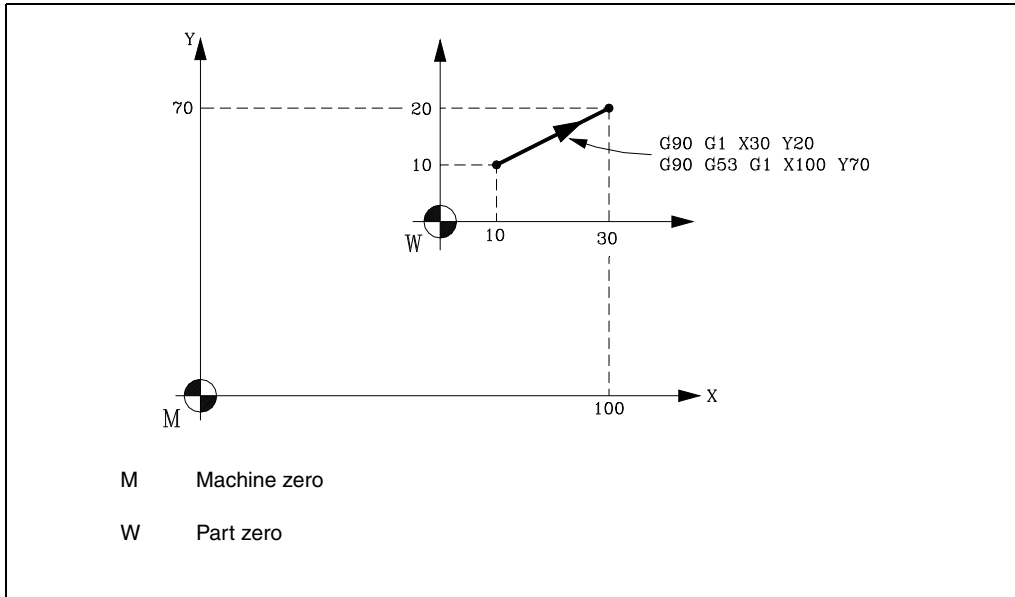


```
G90 G1  X30 Y20
G90 G53 G1 X100 Y70
```

M        Machine zero

W        Part zero

## 4.4    Coordinate preset and zero offsets

The CNC allows you to carry out zero offsets with the aim of using coordinates related to the plane of the part, without having to modify the coordinates of the different points of the part at the time of programming.

The zero offset is defined as the distance between the part zero (point of origin of the part) and the machine zero (point of origin of the machine).



| | |
|---|---|
| M | Machine zero |
| W | Part zero |

This zero offset can be carried out in one of two ways :

- Via Function G92 (coordinate preset). The CNC accepts the coordinates of the programmed axes after G92 as new axis values.
- Through the use of zero offsets (G54 ... G59, G159N1 ... G159N20); the CNC accepts as a new part zero the point located relative to machine zero at the distance indicated by the selected table(s).

Both functions are modal and incompatible, so if one is selected the other is disabled.

There is, moreover, another zero offset which is governed by the PLC. This offset is always added to the zero offset selected and is used (among other things) to correct deviations produced as a result of expansion, etc.
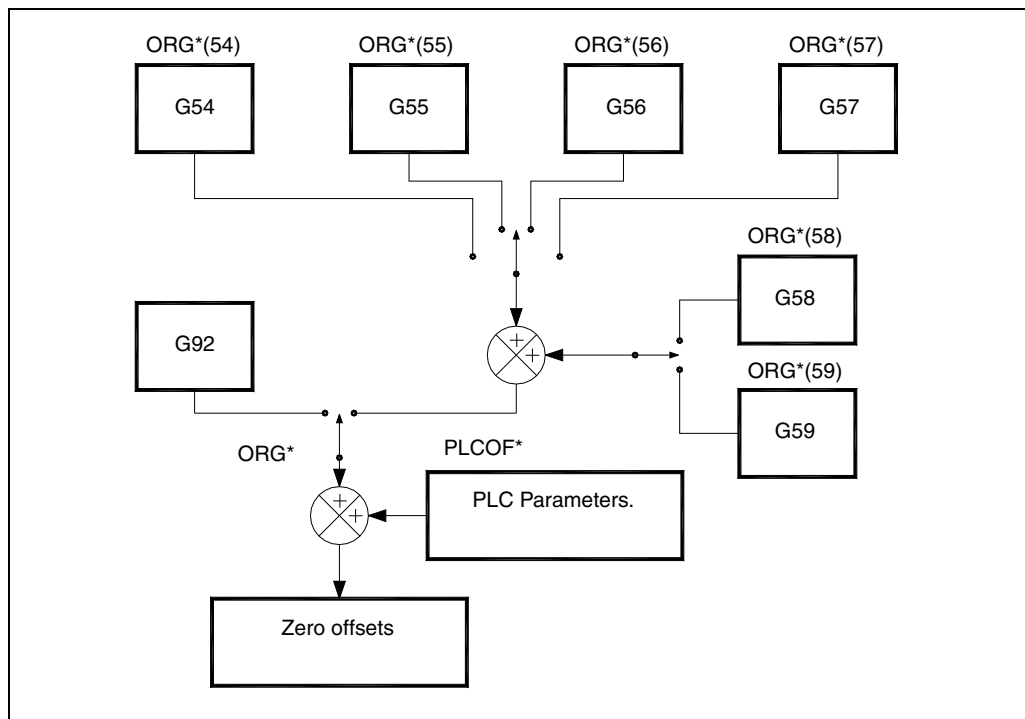
### 4.4.1 Coordinate preset and S value limitation (G92)

Via Function G92 one can select any value in the axes of the CNC, in addition to limiting the spindle speed.

- Coordinate preset.

  When carrying out a zero offset via Function G92, the CNC assumes the coordinates of the axes programmed after G92 as new axis values.

  No other function can be programmed in the block where G92 is defined, the programming format being :

  G92 X…C ±5.5



```
; Positioning in P0
G90 X50 Y40
; Preset P0 as part zero
G92 X0 Y0
; Programming according to part coordinates
G91 X30
X20 Y20
X-20 Y20
X -30
Y-40
```
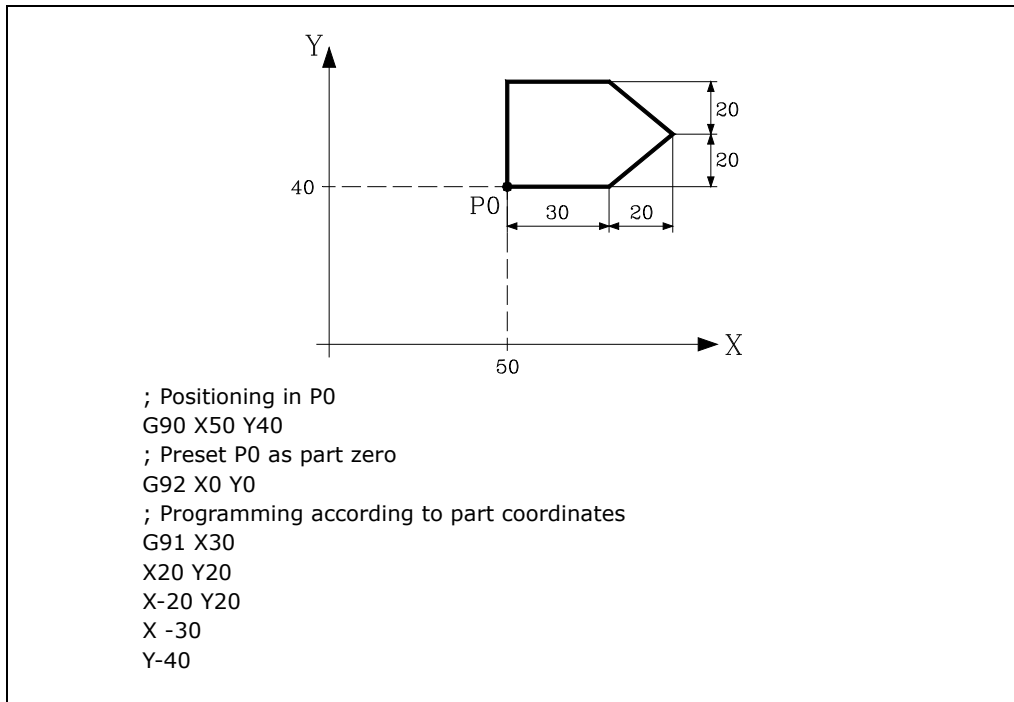
- Spindle speed limitation

  When executing a "G92 S5.4" type block, the CNC limits the spindle speed from that instant on to the value set by S5.4.

  If later on, a block is to be executed at a greater "S", the CNC will execute that block at the maximum "S" set with function G92S.

  Neither is it possible to exceed this maximum value from the keyboard on the front panel.

### 4.4.2    Zero offsets (G54..G59 and G159)

The CNC has a table of zero offsets, in which several zero offsets can be selected. The aim is to generate certain part zeros independently of the part zero active at the time.

Access to the table can be obtained from the front panel of the CNC (as explained in the Operating Manual), or via the program using high-level language commands.

There are two types of zero offsets:

- Absolute zero offsets (G54 ... G57, G159N1 ... G159N20), must be referred to machine zero (home).
- Incremental zero offsets (G58, G59).

Functions G54, G55, G56, G57, G58 & G59 must be programmed alone in the block, and work in the following way:

When one of the G54, G55, G56, G57 functions is executed, the CNC applies the zero offset programmed with respect to machine zero, canceling the possible active zero offsets.

If one of the incremental offsets G58 or G59 is executed, the CNC adds its values to the absolute zero offset active at the time. Previously canceling the additive offset which might be active.

You can see (in the following example) the zero offsets which are applied when the program is executed.

G54          Applies zero offsets G54== G54

G58          Applies zero offsets G58== G54+G58

G59          Cancels G58 and adds G59== G54+G59

G55          Cancels whatever and applies G55== G55

Once a Zero Offset has been selected, it will remain active until another one is selected or until a home search is carried out (G74) in JOG mode. This zero offset will remain active even after powering the CNC off and back on.

This kind of zero offsets established by program is very useful for repeated machining operations at different machine positions.

Example: The zero offset table is initialized with the following values:

G54:     X200     Y100
G55:     X160     Y 60
G56:     X170     Y110
G58:     X-40     Y-40
G59:     X-30     Y 10
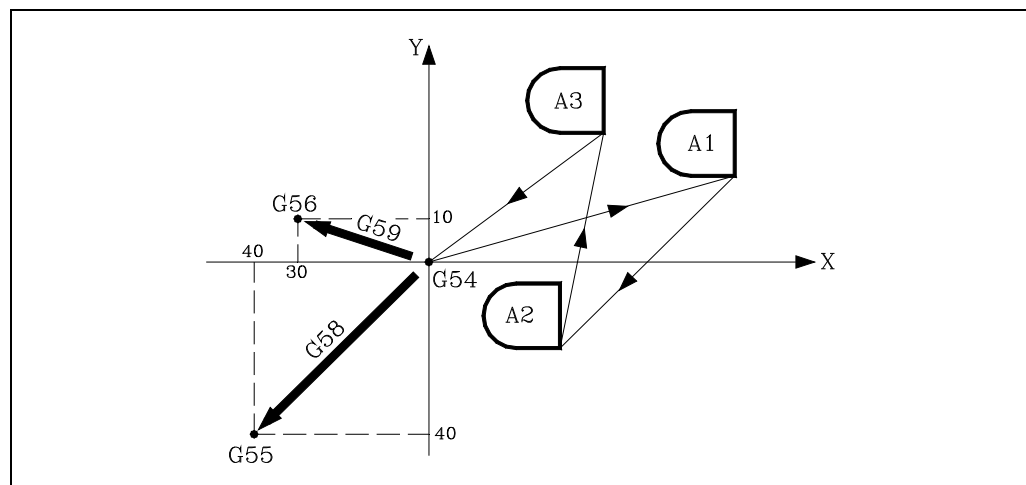
Using absolute zero offsets:

| | |
|---|---|
| G54 | ; Applies G54 offset |
| Profile execution | ; Executes profile A1 |
| G55 | ; Applies G55 offset |
| Profile execution | ; Executes profile A2 |
| G56 | ; Applies G56 offset |
| Profile execution | ; Executes profile A3 |

Using incremental zero offsets:

| | |
|---|---|
| G54 | ; Applies G54 offset |
| Profile execution | ; Executes profile A1 |
| G58 | ; Applies offsets G54+G58 |
| Profile execution | ; Executes profile A2 |
| G59 | ; Applies offsets G54+G59 |
| Profile execution | ; Executes profile A3 |

## Function G158 (incremental zero offset)

The G158 instruction may be used to program and activate an incremental offset in a program. This feature is used to define new part zeros in the same program without having to set them previously in the offset table or use high level instructions.

When applying an incremental zero offset, the CNC adds it to the absolute zero offset active at a time.

### *Programming:*

Incremental zero offset are defined by program using function G158 followed by the values of the zero offset to be applied on each axis. To cancel the incremental zero offset, program function G158 without axes in the block. To cancel the incremental zero offset only on particular axes, program a 0 (zero) incremental offset for each one of them.
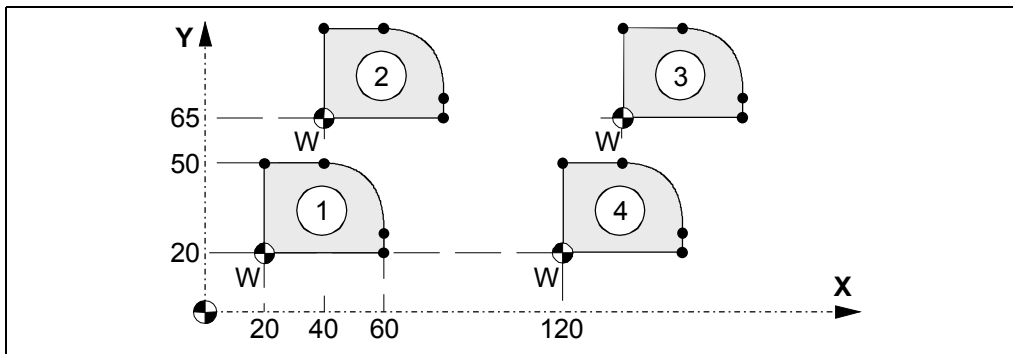


| | X | Y |
|---|---|---|
| G54 (G159N1) | 20 | 20 |
| G55 (G159N2) | 120 | 20 |

| | |
|---|---|
| N100 G54 | (It applies the first zero offset) |
| … | (Machining of profile 1) |
| N200 G158 X20 Y45 | (Apply incremental zero offset) |
| … | (Machining of profile 2) |
| N300 G55 | (It applies the second zero offset. G158 stays active) |
| … | (Machining of profile 3) |
| N400 G158 | (Cancel incremental zero offset. G55 stays active) |
| … | (Machining of profile 4) |

| | X | Z |
|---|---|---|
| G54 (G159N1) | 0 | 420 |
| G55 (G159N2) | 0 | 330 |

| N100 G54 | (It applies the first absolute zero offset) |
|---|---|
| ... | (Machining of profile A1) |
| N200 G158 Z-90 | (Apply incremental zero offset) |
| ... | (Machining of profile A2) |
| N300 G55 | (It applies the second absolute zero offset) |
| | (The incremental zero offset stays active) |
| ... | (Machining of profile A3) |
| N200 G158 Z-180 | (It applies the second incremental zero offset) |
| ... | (Machining of profile A4) |
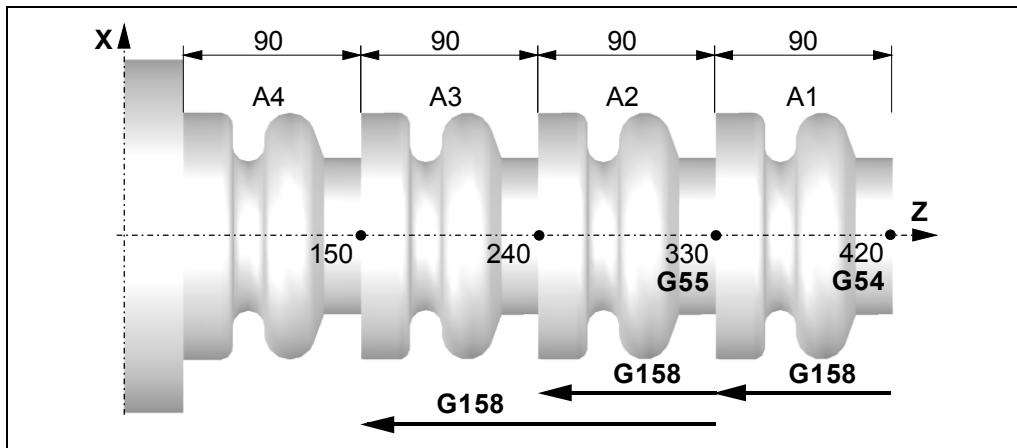
Only one incremental zero may be active at a time for each axis; therefore, applying an incremental zero offset on an axis cancels the one that was active on that axis. The offsets on the rest of the axes are not affected.
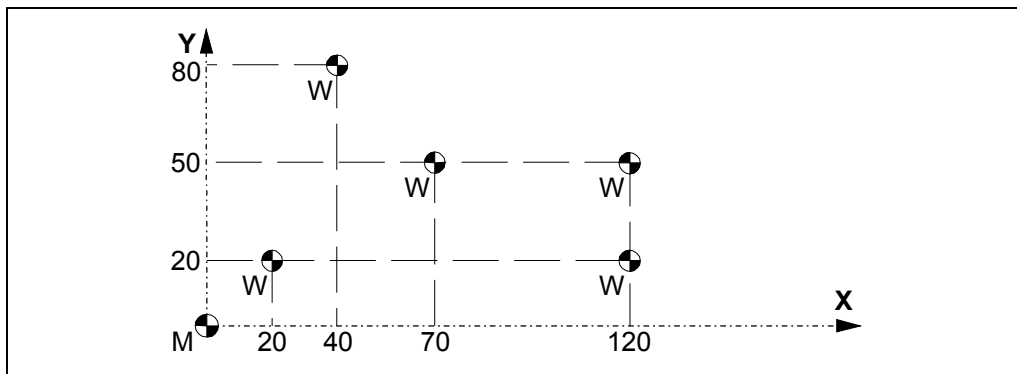


| | X | Y |
|---|---|---|
| G54 (G159N1) | 20 | 20 |

| N100 G54 | (Apply absolute zero offset) |
|---|---|
| N200 G158 X20 Y60 | (It applies the first incremental zero offset) |
| N300 G158 X50 Y30 | (It applies the second incremental zero offset) |
| N400 G158 X100 | (It applies the third incremental zero offset) |
| N500 G158 Y0 | (It applies the fourth incremental zero offset) |
| N600 G158 X0 | (Cancel incremental zero offset) |

The incremental zero offset is not canceled after applying a new absolute zero offset (G54-G57 or G159Nx).

As described earlier, only one incremental zero offset may be active; therefore, instructions G58 and G59 are incompatible with G158. This way, the last incremental zero offset programmed cancels the incremental zero offset that is currently active.

Programming the G158 function alone in the block or G158 with a 0 value in the axes cancels the incremental zero offset G158 activated earlier. Those instructions also cancel the incremental zero offsets G58/G59 that are currently active.

### Considerations:

An incremental zero offset, by itself, does not cause any axis movement.

When homing an axis in JOG mode, the incremental zero offset for that axis is canceled.

### Function properties:

G158 is modal and incompatible with G53.

On power-up, the CNC assumes the incremental zero offset that was active when the CNC was turned off. On the other hand, the incremental zero offset is neither affected by functions M02 and M30 nor by RESETTING the CNC.

### Display in the zero offset table:

In ISO mode and conversational mode, the zero offset table is one line over the the G54 position where it identifies the G158 with its values X, Y, Z, etc.

This line cannot be modified from the table, it can only be modified by programming G158.

## Function G159

To apply any zero offset defined in the table.

The first six zero offsets are the same as programming G54 through G59, except that the values of G58 and G59 are absolute. This is because function G159 cancels functions G54 through G57 and, consequently, there is no active zero offset to add the G58 or G59 to.

Function G159 is programmed as follows:

G159 Nn        Where n is a number from 1 to 20 that indicates the number of the zero offset being applied.

Function G159 is modal, it is programmed alone in the block and is incompatible with functions G53, G54, G55, G56, G57, G58, G59 and G92.

On power-up, the CNC assumes the zero offset that was active when the CNC was turned off. On the other hand, the zero offset is neither affected by functions M02 and M30 nor by RESET.

This function is displayed in the history like G159Nn where the n is the active zero offset.

### Examples:

G159 N1        It applies the first zero offset. It is the same as programming G54.

G159 N6        It applies the sixth zero offset. It is the same as programming G59, but it is applied in absolute.

G159 N20       It applies the 20th zero offset.

## 4.5    Polar origin preset (G93)

Function G93 allows you to preset any point from the work plane as a new origin of polar coordinates.

This function must be programmed alone in the block, its programming format being :

     G93 I±5.5 J±5.5

Parameters I & J respectively define the abscissa and ordinate axes, of the new origin of polar coordinates referred to part zero.



Example, assuming that the tool is at X0 Y0

| G93 | | I35 | J30 | ; Preset P3 as polar origin. |
|---|---|---|---|---|
| G90 | G01 | R25 | Q0 | ; Point P1, in a straight line (G01). |
| | G03 | | Q90 | ; Point P2, in arc (G03). |
| | G01 | X0 | Y0 | ; Point P0, in a straight line (G01) |
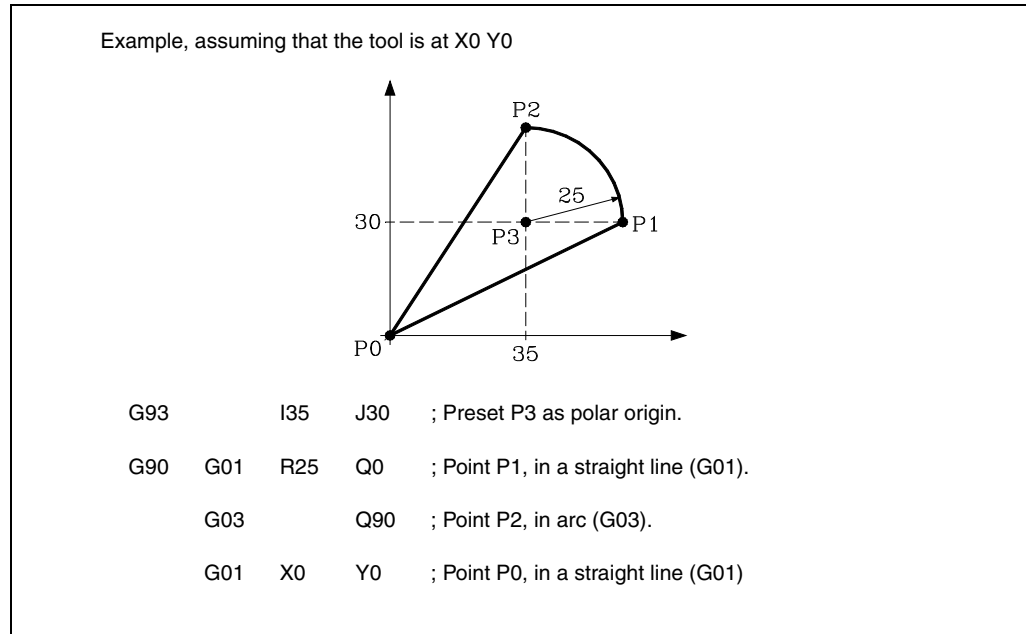
If G93 is only programmed in a block, the point where the machine is at that moment becomes the polar origin.

On power-up; or after executing M02, M30; or after an EMERGENCY or RESET; the CNC assumes the currently active part zero as polar origin.

When selecting a new work plane (G16, G17, G18, G19), the CNC assumes as polar origin the part zero of that plane.

> *The CNC does not modify the polar origin when defining a new part zero; but it modifies the values of the variables: "PORGF" y "PORGS".*
>
> *If, while selecting the general machine parameter "PORGMOVE" a circular interpolation is programmed (G02 or G03), the CNC assumes the center of the arc as the new polar origin.*

# ISO CODE PROGRAMMING

**5**

A block programmed in ISO language can consist of:

- Preparatory (G) functions
- Axis coordinates (X...C)
- Feedrate (F)
- Spindle speed (S)
- Tool number (T)
- Tool offset number (D)
- Auxiliary functions (M)

This order should be maintained within each block, although it is not necessary for every block to contain the information.

The CNC allows you to program figures from 0.00001 to 99999.9999 with or without sign, working in millimeters (G71), called format +/-5.4, or either from 0.00001 to 3937.00787 with or without sign if the programming is done in inches (G70), called format +/-4.5.

However, and to simplify the instructions, we can say that the CNC admits +/- 5.5 format, thereby admitting +/- 5.4 in millimeters and +/- 4.5 in inches.

Any function with parameters can also be programmed in a block, apart from the number of the label or block. Thus, when the block is executed the CNC substitutes the arithmetic parameter for its value at that time.

## 5.1 Preparatory functions

Preparatory functions are programmed using the letter G followed by up to 3 digits (G0 - G319).

They are always programmed at the beginning of the body of the block and are useful in determining the geometry and working condition of the CNC.

*Table of G functions used in the CNC.*

| Function | M | D | V | Meaning | Section |
|----------|---|---|---|---------|---------|
| G00 | * | ? | * | Rapid traverse | 6.1 |
| G01 | * | ? | * | Linear interpolation | 6.2 |
| G02 | * | | * | Clockwise circular (helical) interpolation | 6.3 / 6.7 |
| G03 | * | | * | Counterclockwise circular (helical) interpolation | 6.3 / 6.7 |
| G04 | | | | Dwell/interruption of block preparation | 7.1 / 7.2 |
| G05 | * | ? | * | Round corner | 7.3.2 |
| G06 | | | * | Circle center in absolute coordinates | 6.4 |
| G07 | * | ? | | Square corner | 7.3.1 |
| G08 | | | * | Arc tangent to previous path | 6.5 |
| G09 | | | * | Arc defined by three points | 6.6 |
| G10 | * | * | | Mirror image cancellation | 7.5 |
| G11 | * | | * | Mirror image on X axis | 7.5 |
| G12 | * | | * | Mirror image on Y axis | 7.5 |
| G13 | * | | * | Mirror image on Z axis | 7.5 |
| G14 | * | | * | Mirror image in the programmed directions | 7.5 |
| G15 | * | | * | Longitudinal axis selection | 8.2 |
| G16 | * | | * | Main plane selection by two addresses and longitudinal axis | 3.2 |
| G17 | * | ? | * | Main plane X-Y and longitudinal Z | 3.2 |
| G18 | * | ? | * | Main plane Z-X and longitudinal Y | 3.2 |
| G19 | * | | * | Main plane Y-Z and longitudinal X | 3.2 |
| G20 | | | | Definition of lower work zone limits | 3.7.1 |
| G21 | | | | Definition of upper work zone limits. | 3.7.1 |
| G22 | | | * | Enable/disable work zones. | 3.7.2 |
| G28 | * | | * | Second spindle selection | 5.4 |
| G29 | * | * | | Main spindle selection | 5.4 |
| G28-G29 | | | * | Axis toggle | 7.9 |
| G30 | * | | * | Spindle synchronization (offset) | 5.5 |
| G32 | * | | * | Feedrate "F" as an inverted function of time. | 6.15 |
| G33 | * | | * | Electronic threading | 6.12 |
| G34 | | | | Variable-pitch threading | 6.13 |
| G36 | | | * | Corner rounding | 6.10 |
| G37 | | | * | Tangential entry | 6.8 |
| G38 | | | * | Tangential exit | 6.9 |
| G39 | | | * | Chamfer | 6.11 |
| G40 | * | * | | Cancellation of tool radius compensation | 8.1 |
| G41 | * | | * | Left-hand tool radius compensation | 8.1 |
| G41 N | * | | * | Collision detection | 8.3 |
| G42 | * | | * | Right-hand tool radius compensation | 8.1 |
| G42 N | * | | * | Collision detection | 8.3 |
| G43 | * | ? | * | Tool length compensation | 8.2 |
| G44 | * | ? | | Cancellation of tool length compensation | 8.2 |
| G45 | * | | * | Tangential control (G45) | 6.16 |
| G47 | | | * | Tool movement according to the tool coordinate system. | 15.2 |
| G48 | * | | * | TCP transformation | 15.3 |
| G49 | * | | * | Inclined plane definition | 15.1 |
| G50 | * | | * | Controlled corner rounding | 7.3.3 |
| G51 | * | | * | Look-Ahead | 7.4 |

| Function | M | D | V | Meaning | Section |
|---|---|---|---|---|---|
| G52 | | | * | Movement until making contact | 6.14 |
| G53 | | | * | Programming with respect to machine zero | 4.3 |
| G54 | * | | * | Absolute zero offset 1 | 4.4.2 |
| G55 | * | | * | Absolute zero offset 2 | 4.4.2 |
| G56 | * | | * | Absolute zero offset 3 | 4.4.2 |
| G57 | * | | * | Absolute zero offset 4 | 4.4.2 |
| G58 | * | | * | Additive zero offset 1 | 4.4.2 |
| G59 | * | | * | Additive zero offset 2 | 4.4.2 |
| G60 | | | * | Multiple machining in a straight line | 10.1 |
| G61 | | | * | Multiple machining in rectangular pattern | 10.2 |
| I* | | | * | Grid pattern canned cycle | 10.3 |
| G63 | | | * | Multiple machining in a circular pattern | 10.4 |
| G64 | | | * | Multiple machining in an arc | 10.5 |
| G65 | | | * | Machining programmed with an arc-chord | 10.6 |
| G66 | | | * | Irregular pocket canned cycle | 11.1 / 11.2 |
| G67 | | | * | Irregular pocket roughing | 11.1.2 |
| G68 | | | * | Irregular pocket finishing | 11.1.3 |
| G69 | * | | * | Drilling canned cycle with variable peck | 9.6 |
| G70 | * | ? | * | Programming in inches | 3.3 |
| G71 | * | ? | | Programming in millimeters | 3.3 |
| G72 | * | | * | General and specific scaling factor | 7.6 |
| G73 | * | | * | Rotation of the coordinate system | 7.7 |
| G74 | | | * | Machine reference (home) search | 4.2 |
| G75 | | | * | Probing move until touching | 12.1 |
| G76 | | | * | Probing move while touching | 12.1 |
| G77 | * | | * | Axis coupling (slaving) | 7.8.1 |
| G77S | * | | * | Spindle synchronization | 5.5 |
| G78 | * | * | | Cancellation of axis coupling (slaving) | 7.8.2 |
| G78S | * | * | | Cancellation of spindle synchronization | 5.5 |
| G79 | | | | Canned cycle parameter modification | 9.2.1 |
| G80 | * | * | | Canned cycle cancellation | 9.3 |
| G81 | * | | * | Drilling canned cycle | 9.7 |
| G82 | * | | * | Drilling canned cycle with dwell | 9.8 |
| G83 | * | | * | Deep-hole drilling canned cycle with constant peck | 9.9 |
| G84 | * | | * | Tapping canned cycle | 9.10 |
| G85 | * | | * | Reaming canned cycle | 9.11 |
| G86 | * | | * | Boring canned cycle with withdrawal in G00 | 9.12 |
| G87 | * | | * | Rectangular pocket canned cycle. | 9.13 |
| G88 | * | | * | Circular pocket canned cycle | 9.14 |
| G89 | * | | * | Boring canned cycle with withdrawal in G01 | 9.15 |
| G90 | * | ? | | Absolute programming: | 3.4 |
| G91 | * | ? | * | Incremental programming | 3.4 |
| G92 | | | | Coordinate preset / spindle speed limit | 4.4.1 |
| G93 | | | | Polar origin preset | 4.5 |
| G94 | * | ? | | Feedrate in millimeters (inches) per minute | 5.2.1 |
| G95 | * | ? | * | Feedrate in millimeters (inches) per revolution. | 5.2.2 |
| G96 | * | | * | Constant cutting point speed | 5.2.3 |
| G97 | * | * | | Constant tool center speed | 5.2.4 |
| G98 | * | * | | Withdrawal to the starting plane at the end of the canned cycle | 9.5 |
| G99 | * | | * | Withdrawal to the reference plane at the end of the canned cycle | 9.5 |
| G145 | * | | * | Temporary cancellation of tangential control | 6.17 |

**5.**

**ISO CODE PROGRAMMING**
Preparatory functions

FAGOR

FAGOR AUTOMATION

CNC 8055
CNC 8055i

·M· & ·EN· MODELS
SOFT: V02.2X

| Function | M | D | V | Meaning | Section |
|----------|---|---|---|---------|---------|
| G159 | * | | | Absolute zero offsets | *4.4* |
| G210 | * | | * | Bore milling canned cycle | *9.16* |
| G211 | * | | * | Inside thread milling canned cycle. | *9.17* |
| G212 | * | | * | Outside thread milling canned cycle. | *9.18* |

M means modal, i.e. the G function, once programmed, remains active until another incompatible G function is programmed or until an M02, M30, EMERGENCY or RESET is executed or the CNC is turned off and back on.

D means BY DEFAULT, i.e. they will be assumed by the CNC when it is powered on, after executing M02, M30 or after EMERGENCY or RESET.

In those cases indicated by ?, it should be understood that the DEFAULT of these G functions depends on the setting of the general machine parameters of the CNC.

The letter V means that the G code is displayed next to the current machining conditions in the execution and simulation modes.

**5.**

**ISO CODE PROGRAMMING**
Preparatory functions

**FAGOR** 

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

·68·

## 5.2 Feedrate F

The machining feedrate can be selected from the program. It remains active until another feedrate is programmed. It is represented by the letter F and Depending on whether it is working in G94 or G95, it is programmed in mm/minute (inches/minute) or in mm/revolution (inches/revolution).

Its programming format is 5.5; in other words, 5.4 when programmed in mm and 4.5 when programmed in inches.

The maximum operating feedrate of the machine, limited on each axis by the axis machine parameter "MAXFEED", may be programmed via code F0, or by giving F the corresponding value.

The programmed feedrate F is effective working in linear (G01) or circular (G02, G03) interpolation. If function F is not programmed, the CNC assumes the feedrate to be F0. When working in rapid travel (G00), the machine will move at the rapid feedrate indicated by the axis machine parameter "G00FEED", apart from the F programmed.

The programmed feedrate F may be varied between 0% and 255% via the PLC, or by DNC, or between 0% and 120% via the switch located on the Operator Panel of the CNC.

The CNC, however, is equipped with the general machine parameter "MAXFOVR" to limit maximum feedrate variation.

If you are working in rapid travel (G00), rapid feedrate will be fixed at 100%, alternatively it can be varied between 0% and 100%, depending on how the machine parameter "RAPIDOVR" is set.

When functions G33 (electronic threading), G34 (variable-pitch threading) or G84 (tapping canned cycle) are executed the feedrate cannot be modified; it works at 100% of programmed F.

### 5.2.1 Feedrate in mm/min or inches/min (G94)

From the moment the code G94 is programmed, the control takes that the feedrates programmed through F5.5 are in mm/min or inches/mm.

If the moving axis is rotary, the CNC interprets that the programmed feedrate is in degrees/minute.

If an interpolation is made between a rotary and a linear axis, the programmed feedrate is taken in mm/min or inches/min, and the movement of the rotary axis (programmed in degrees) will be considered programmed in millimeters or inches.

The relationship between the feedrate of the axis component and the programmed feedrate "F" is the same as that between the movement of the axis and the resulting programmed movement.

$$\text{Feedrate component} = \frac{\text{Feedrate F x Movement of axis}}{\text{Resulting programmed movement}}$$

Example:

On a machine which has linear X and Y axes and rotary C axis, all located at point X0 Y0 C0, the following movement is programmed :

    G1 G90 X100 Y20 C270 F10000

You get:

$$Fx = \frac{F \cdot \Delta x}{\sqrt{(\Delta x)^2 + (\Delta y)^2 + (\Delta c)^2}} = \frac{10000 \times 100}{\sqrt{100^2 + 20^2 + 270^2}} = 3464,7946$$

$$Fy = \frac{F \cdot \Delta y}{\sqrt{(\Delta x)^2 + (\Delta y)^2 + (\Delta c)^2}} = \frac{10000 \times 20}{\sqrt{100^2 + 20^2 + 270^2}} = 692,9589$$

$$Fc = \frac{F \cdot \Delta c}{\sqrt{(\Delta x)^2 + (\Delta y)^2 + (\Delta c)^2}} = \frac{10000 \times 270}{\sqrt{100^2 + 20^2 + 270^2}} = 9354,9455$$

Function G94 is modal i.e. once programmed it stays active until G95 is programmed.

On power-up, after executing M02, M30 or following EMERGENCY or RESET, the CNC assumes function G94 or G95 according to how the general machine parameter "IFEED" is set.

### 5.2.2 Feedrate in mm/rev.or inches/rev (G95)

From the moment when the code G95 is programmed, the control assumes that the feedrates programmed through F5.5 are in mm/rev or inches/mm.

This function does not affect the rapid moves (G00) which will be made in mm/min or inch/min. By the same token, it will not be applied to moves made in the JOG mode, during tool inspection, etc.

Function G95 is modal i.e. once programmed it stays active until G94 is programmed.

On power-up, after executing M02, M30 or following EMERGENCY or RESET, the CNC assumes function G94 or G95 according to how the general machine parameter "IFEED" is set.

**5.**

**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

### 5.2.3    Constant surface speed (G96)

When G96 is programmed the CNC takes the F5.5 feedrate as corresponding to the cutting point of the tool on the part.

By using this function, the finished surface is uniform in curved sections.

In this manner (working in function G96) the speed of the center of the tool in the inside or outside curved sections will change in order to keep the cutting point constant.

Function G96 is modal i.e. once programmed it stays active until G97 is programmed.

On power-up, after executing M02, M30 or following EMERGENCY or RESET, the CNC assumes function G97.

### 5.2.4    Constant tool center speed (G97)

When G97 is programmed the CNC takes the programmed F5.5 feedrate as corresponding to the feedrate of the center of the tool.

In this manner (working in function G97) the speed of the cutting point on the inside or outside curved sections is reduced, keeping the speed of the center of the tool constant.

Function G97 is modal i.e. once programmed it stays active until G96 is programmed.

On power-up, after executing M02, M30 or following EMERGENCY or RESET, the CNC assumes function G97.

## 5.3 Spindle turning speed (S)

The turning speed of the spindle is programmed directly in rpm via code S5.4.

The maximum value is limited by spindle machine parameters "MAXGEAR1", MAXGEAR2, MAXGEAR 3 and MAXGEAR4", in each case depending on the spindle range selected.

It is also possible to limit this maximum value from the program by using function G92 S5.4.

The programmed turning speed S may be varied from the PLC, DNC, or by the SPINDLE keys "+" and "-" on the Operator Panel of the CNC.

This speed variation is made between the maximum and minimum values established by spindle machine parameters "MINSOVR" and "MAXSOVR".

The incremental pitch associated with the SPINDLE keys "+" and "-" on the CNC Operator Panel in order to vary the programmed S value is fixed by the spindle machine parameter "SOVRSTEP".

When functions G33 (electronic threading), G34 (variable-pitch threading) or G84 (tapping canned cycle) are executed the programmed speed cannot be modified; it works at 100% of programmed S.

**FAGOR**

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

## 5.4     Spindle selection (G28, G29)

This CNC can govern two spindles: the main one and the second one. They both can be operative simultaneously, but only one can be controlled at a time.

This selection is made using functions G28 and G29.

G28: Second spindle selection.

G29: Main spindle selection.

Once the desired spindle has been selected, it can be acted upon from the keyboard or by means of the following functions:

M3, M4, M5, M19

S****

G33, G34, G94, G95, G96, G97

Both spindles can work in open and closed loop.

Functions G28 and G29 are modal and incompatible with each other.

Function G28 and G29 must be programmed alone in the block.

No more information can be programmed in that block. On power-up, after executing an M02, M30 or after an EMERGENCY or RESET, the CNC assumes function G29 (selects the main spindle).

### *Operating example for when 2 spindles are used.*

On power-up, the CNC assumes function G29 selecting the main spindle.

All the keyboard actions and by spindle related functions affect the main spindle.

Example: S1000 M3

Main spindle clockwise at 1000 rpm.

To select the second spindle, execute function G28.

From then on, All the keyboard actions and spindle related functions affect the second spindle.

The main spindle remains in its previous status.

Example: S1500 M4

Second spindle turns counterclockwise at 1500 rpm.

The main spindle keeps turning at 1000 rpm.

To select the main spindle again, execute function G29.

From then on, all the keyboard actions and spindle related functions affect the main spindle.

The second spindle stays in its previous status.

Example: S2000

The main spindle keeps turning clockwise but at 2000 rpm.

The second spindle keeps turning at 1500 rpm.

## 5.5 Synchronized spindles (G30, G77S, G78S)

With function G77S, two spindles (main and secondary) may be synchronized in speed; this synchronism may be cancelled with function G78S. Always program G77S and G78S because functions G77, G78 to slave and unslave the axes.

When the spindles are synchronized in speed, the second one turns at the same speed as the main spindle.

Function G77S may be executed at any time, open loop (M3, M4) or closed loop (M19), the spindles may even have different ranges (gears).

General output "SYNSPEED (M5560)" will be high while the spindle are in synch (same speed).

When this synchronism is cancelled (G78S), the second spindle recovers its previous speed and status (M3, M4, M5, M19) and the main spindle stays in the current status.

If while in synchronism, an S is programmed greater than the maximum allowed, the CNC applies the maximum value while they are synchronized. When canceling this synchronism, the limit is no longer applied and the main spindle assumes the programmed speed.

While the spindles are synchronized in speed, function G77S active, with G30 they may also be synchronized in position and set an angular offset between them so the second spindle follows the main spindle at this set offset distance.

Programming format: G30 D ±359.9999 (offset in degrees)

For example, with G30 D90 the second spindle will turn 90º behind the main spindle.

### *Considerations:*

Before activating the synchronism, both spindles must be homed (referenced).

To synchronized the spindles in position (G30) they must be synchronized in speed already (G77S).

To synchronize two spindles, the SERVOSON and SERVOSO2 signals must be activated already. When the spindle synchronization is active, only the main spindle signals PLCCNTL, SPDLINH, SPDLREV, etc., will be addressed. Also, if a thread is wanted to be made, only the feedback and the input of the main spindle will be used.

While the spindle synchronism is active, it is possible to:

- Execute functions G94, G95, G96, G97, M3, M4, M5, M19 S***.
- Change the spindle speed via DNC, PLC or CNC (S).
- Change the spindle speed override via DNC, PLC, CNC or keyboard.
- Change the spindle speed limit via DNC, PLC or CNC (G92 S)

But the following cannot be done:

- Toggle the spindles: G28, G29.
- Change gears: M41, M42, M43, M44.

## 5.6      Tool number (T) and tool offset (D)

With the "T" function, it is possible to select the tool and with the "D" function it is possible to select the offset associated with it. When defining both parameters, the programming order is T D. For example: T6 D17



If the machine has a tool magazine, the CNC looks up the "Tool magazine table" to know the position occupied by the selected tool and the desired one.

If the "D" function has not be defined, it looks up the "Tool table" to know the "D" offset associated with it.

It examines the "tool offset table" and assumes the tool dimensions corresponding to the "D" offset.
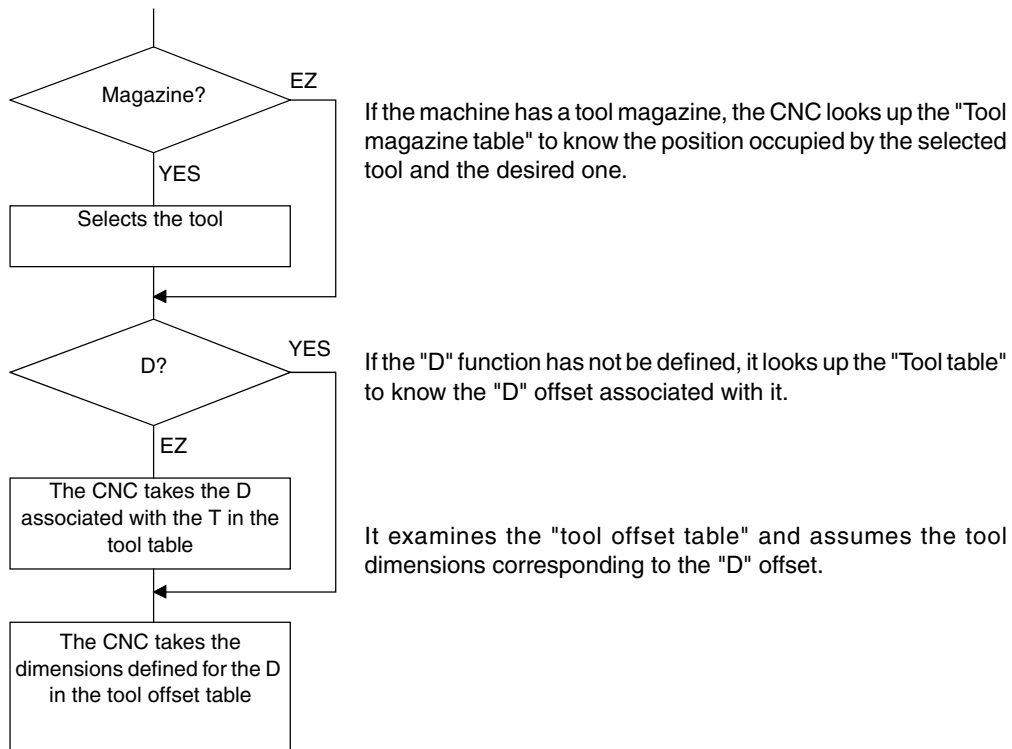
To access, check and define these tables, refer to the operating manual.

### *How to use the T and D functions*

- The "T" and "D" functions may be programmed alone or together as shown in the following example:

    T5 D18      Selects tool 5 and assumes the dimensions of tool offset 18.

    D22         Tool 5 stays selected and it assumes the dimensions of tool offset 22.

    T3          Selects tool 3 and assumes the dimensions of the offset associated with that tool.

- When having a tool magazine where the same position is occupied by more than one tool, do the following:

    Use the "T" function to refer to the magazine position and the "D" function to the dimensions of the tool located in that position.

    Thus, for example, programming T5 D23 means selecting the turret position 5 and assuming the geometry and dimensions of tool offset 23.

### *Tool length and radius compensation.*

The CNC looks up the "tool offset table" and assumes the tool dimensions corresponding to the active "D" offset.

Length compensation is applied at all times, whereas radius compensation must be selected by the operator by means of functions G40, G41, G42.

Length compensation is applied at all times, whereas tool length compensation must be selected by the operator by means of functions G43, G44.

If there is no tool selected or D0 is defined, neither tool length nor radius compensation is applied.

For further information, refer to chapter 8 "tool compensation" in this manual..

## 5.7    Auxiliary function (M)

The miscellaneous functions are programmed by means of the M4 Code, it being possible to program up to 7 functions in the same block.

When more than one function has been programmed in one block, the CNC executes these correlatively to the order in which they have been programmed.

The CNC is provided with an M functions table with "NMISCFUN" (general machine parameter) components, specifying for each element:

- The number (0-9999) of the defined miscellaneous M function.

- The number of the subroutine which is required to associate to this miscellaneous function.

- An indicator which determines if the M function is executed before or after the movement block in which it is programmed.

- An indicator which determines if the execution of the M function interrupts block preparation or not.

- An indicator which determines if the M function is executed or not, after the execution of the associated subroutine.

- An indicator which determines if the CNC must wait for the signal AUX END or not (Executed M signal, coming from the PLC), to continue the execution of the program.

If, when executing the M miscellaneous function, this is not defined in the M functions table, the programmed function will be executed at the beginning of the block and the CNC will wait for the AUX END to continue the execution of the program.

Some of the miscellaneous functions are assigned an internal meaning in the CNC.

If, while executing the associated subroutine of an "M" miscellaneous function, there is a block containing the same "M", this will be executed but not the associated subroutine.

> **i**
>
> *All the miscellaneous "M" functions which have an associated subroutine must be programmed alone in a block.*
> *In the case of functions M41 through M44 with associated subroutine, the S that generates the gear change must be programmed alone in the block. Otherwise, the CNC will display error 1031.*

### 5.7.1 M00. Program stop

When the CNC reads code M00 in a block, it interrupts the program. To start up again, press CYCLE START.

We recommend that you set this function in the table of M functions, in such a way that it is executed at the end of the block in which it is programmed.

### 5.7.2 M01. Conditional program stop

This is identical to M00, except that the CNC only takes notice of it if the signal M01 STOP from the PLC is active (high logic level).

### 5.7.3 M02. End of program

This code indicates the end of program and carries out a "General Reset" function of the CNC (returning it to original state). It also carries out the M05 function.

We recommend that you set this function in the table of M functions, in such a way that it is executed at the end of the block in which it is programmed.

### 5.7.4 M30. End of program with return to the first block

Identical to M02 except that the CNC returns to the first block of the program.

### 5.7.5 M03, M4, M5. Spindle start and stop

## M03. Clockwise spindle rotation

This code represents clockwise spindle start. As explained in the corresponding section, the CNC automatically executes this code in the machining canned cycles.

It is recommended to set this function in the table of M functions, so that it is executed at the beginning of the block in which it is programmed.

## M04. Counterclockwise spindle rotation

This code represents counterclockwise spindle start. We recommend that you set this function in the table of M functions, so that it is executed at the beginning of the block in which it is programmed.

## M05. Spindle stop

We recommend that you set this function in the table of M functions, in such a way that it is executed at the end of the block in which it is programmed.

## Execution of M03, M04 and M05 using PLC marks

Auxiliary functions M03, M04 and M05 may be executed using the following PLC marks:

- First spindle: PLCM3 (M5070), PLCM4 (M5071) and PLCM5 (M5072).
- Second spindle: PLCM3SP2 (M5073), PLCM4SP2 (M5074) and PLCM5SP2 (M5075).

- Auxiliary spindle: PLCM45 (M5076) to stop the auxiliary spindle and PLCM45S (M5077) to start it up.

The PLC activates these marks to indicate to the CNC that it must execute the corresponding M function at the indicated spindle.

If that spindle is not the main spindle at that moment, it changes the M in the execution history, it activates the corresponding PLC mark DM3/4//5 and executes the transfer with the PLC (it writes the M number in the MBCD1 (R550) register, it activates the MSTROBE signal, it waits for the AUXEND signal to go up and it cancels the MSTROBE signal; if the M function is set as not to wait for AUXEND in the M function table, it waits for the time period set with MINAENDW and cancels the MSTROBE).

When acting upon the secondary spindle, it executes the same operation, but having activated the mark S2MAIN (M5536) first and then cancelling it at the end. This operation is done automatically; i.e. there is no need to program it at the PLC.

Even if the M3, M4 or M5 function has an associated subroutine in the M function table, that subroutine will not be executed when they are executed using the PLC marks.

When executing M3, M4 or M5 using the PLC marks, the gear change that could be required by the new S is not output to the PLC even if the gear change is automatic.

> *If the CNC is started up with no gear being active yet, because no M3 or M4 has been executed in the main channel, the CNC will issue an error message even if it has been configured as AUTOGEAR.*

The CNC will admit the M functions from the PLC as long as it is not in an error state or with LOPEN (M5506) high, whether there is an execution active or not, either in manual or automatic. If the M function is executed during a tool inspection and the spindle turning direction is changed, the change will be identified in repositioning and it will offer the choice to change it again.

If when activating the M3, M4 or M5 marks via PLC, the main channel is transferring something to the PLC, the PLC keeps the mark active until the CNC can attend to it. Once the M function is executed, the CNC deactivates the mark.

In the following cases, the CNC ignores these PLC marks and deletes the mark so the request is not left as pending:

- When the spindle is threading in electronic threading (G33).
- When it is doing a rigid tapping or a regular tapping.
- When the CNC is in error state or when LOPEN (M5506) is high.

If several marks of different spindles are activated at the same time, the below order will be followed: first the first spindle, then the second spindle and finally the auxiliary spindle.

When receiving contradictory marks, they will all be ignored. When receiving several marks at the same time and there is a stop (PLCM5 / PLCM45) between them, only this will be considered and the rest will be ignored and not saved.

If the spindle has M19TYPE=1, the spindle is homed with the first M3 or M4 after start-up, as long as that M function is executed in manual or automatic mode. If the M function is executed using one of the PLC marks, the spindle will not be homed.

When activating the PLC marks while homing the spindle, the PLC command stays waiting for the homing operation to be over. If the home search is associated with the first M3 or M4 after start-up, the PLC command stays waiting for the homing operation is over.

If there are synchronized spindles, it acts upon the command of the main and secondary spindles at the same time.

While executing the M function, it is possible to abort the process by deactivating the PLC mark that has initiated it.

### *Note:*

The PLCM5 mark is used to handle the open-door safety operation defined by Fagor Automation.

### 5.7.6    M06. Tool change code

If the general machine parameter "TOFFM06" (indicating that it is a machining center) is active, the CNC sends instructions to the tool changer and updates the table corresponding to the tool magazine.

It is recommended to set this function in the table of M functions, so that the subroutine corresponding to the tool changer installed in the machine is executed.

The functions T and M06 may be programmed in the same block, regardless if they have an associated subroutine or not. In a block where the functions T and M06 are programmed, nothing else may be programmed.

**5.7.7 M19. Spindle orientation**

With this CNC it is possible to work with the spindle in open loop (M3, M4) and with the spindle in closed loop (M19).

In order to work in closed loop, it is necessary to have a rotary encoder installed on the spindle of the machine.

To switch from open loop to closed loop, execute function M19 or M19 S±5.5. The CNC will act as follows:

- If the spindle has a home switch, the CNC modifies the spindle speed until it reaches the one set by spindle machine parameter "REFEED1".

  It then searches for actual marker pulse (Io) of the spindle encoder at the turning speed set by spindle machine parameter REFEED2.

  And, finally, it positions the spindle at the programmed S±5.5 point.

- If the spindle does not have a home switch, it searches the encoder marker pulse at the turning speed set by spindle machine parameter REFEED2.

  And, then, it positions the spindle at the programmed S±5.5 point.

If only M19 is executed, the spindle is oriented to position "I0".

To, now, orient the spindle to another position, program M19 S±5.5, the CNC will not perform the home search since it is already in closed loop and it will orient the spindle to the indicated position. (S±5.5).

The S±5.5 code indicates the spindle position, in degrees, from the spindle reference point (marker pulse).

The sign indicates the counting direction and the 5.5 value is always considered to be absolute coordinates regardless of the type of units currently selected.

Example:

S1000 M3

    Spindle in open loop.

M19 S100

    The spindle switches to closed loop. Home search and positioning (orientation) at 100º.

M19 S -30

    The spindle orients to -30º, passing through 0º.

M19 S400

    The spindle turns a whole revolution and positions at 40º.

> **i** *During the M19 process the screen will display the warning: "M19 in execution"*

### 5.7.8    M41, M42, M43, M44. Spindle gear change

The CNC offers 4 spindle speed ranges M41, M42, M43 and M44 with maximum speed limits set by the spindle machine parameters "MAXGEAR1", "MAXGEAR2", "MAXGEAR3" and "MAXGEAR4".

If machine parameter "AUTOGEAR" is set so the CNC executes the range change automatically, M41 thru M44 will be sent out automatically by the CNC without having to be programmed.

If this machine parameter is set for non-automatic gear change, M41 thru M44 will have to be programmed every time a gear change is required. Bear in mind that the maximum velocity command value assigned to machine parameter "MAXVOLT" corresponds to the maximum speed indicated for each one of the speed ranges (machine parameters "MAXGEAR1" thru "MAXGEAR4").

Regardless of whether the gear change is automatic or not, functions M41 through M44 may have an associated subroutine. If the function M41 through M44 is programmed and then an S corresponding to that gear, it does not generate the automatic gear change and it does not execute the associated subroutine.

### 5.7.9 M45. Auxiliary spindle / Live tool

In order to use this miscellaneous function, it is necessary to set one of the axes of the machine as auxiliary spindle or live tool (general machine parameter P0 thru P7).

To use the auxiliary spindle or live tool, execute the command: M45 S±5.5 where S indicates the turning speed in rpm and the sign indicates the turning direction.

The CNC will output the velocity command corresponding to the selected speed according to the value assigned to the machine parameter "MAXSPEED" for the auxiliary spindle.

To stop the auxiliary spindle, program M45 or M45 S0.

Whenever the auxiliary spindle or live tool is active, the CNC will let the PLC know by activating the general logic output "DM45" (M5548).

Also, it is possible to set the machine parameter for the auxiliary spindle "SPDLOVR" so the Override keys of the front panel can modify the currently active turning speed of the auxiliary spindle.

# PATH CONTROL

The CNC allows you to program movements on one axis only or several at the same time.

Only those axes which intervene in the required movement are programmed. The programming order of the axes is as follows :

X, Y, Z, U, V, W, A, B, C

## 6.1    Rapid traverse (G00)

The movements programmed after the G00 are executed using the rapid feedrate found in the machine axis parameter "G00FEED".

Independently of the number of axis which move, the resulting path is always a straight line between the starting point and the final point.



```
X100  Y100                    ; Starting point

G00 G90 X400 Y300             ;  Programmed path
```
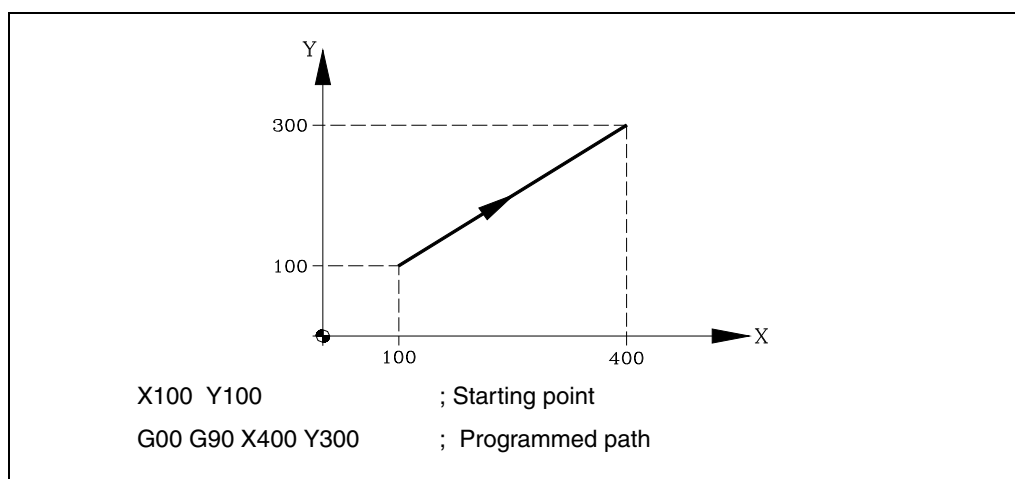
It is possible, via the general machine parameter "RAPIDOVR", to establish if the feedrate override % switch (when working in G00) operates from 0% to 100%, or whether it stays constant at 100%.

When G00 is programmed, the last "F" programmed is not cancelled i.e. when G01, G02 or G03 are programmed again "F" is recovered.

G00 is modal and incompatible with G01, G02, G03, G33 G34 and G75. Function G00 can be programmed as G or G0.

On power-up, after executing M02, M30 or after EMERGENCY or RESET, the CNC assumes code G00 or G01, depending on how general machine parameter "IMOVE" has been set.

## 6.2      Linear interpolation (G01)

The movements programmed after G01 are executed according to a straight line and at the programmed feedrate "F".

When two or three axes move simultaneously the resulting path is a straight line between the starting point and the final point.

The machine moves according to this path to the programmed feedrate "F". The CNC calculates the feedrates of each axis so that the resulting path is the "F" value programmed.
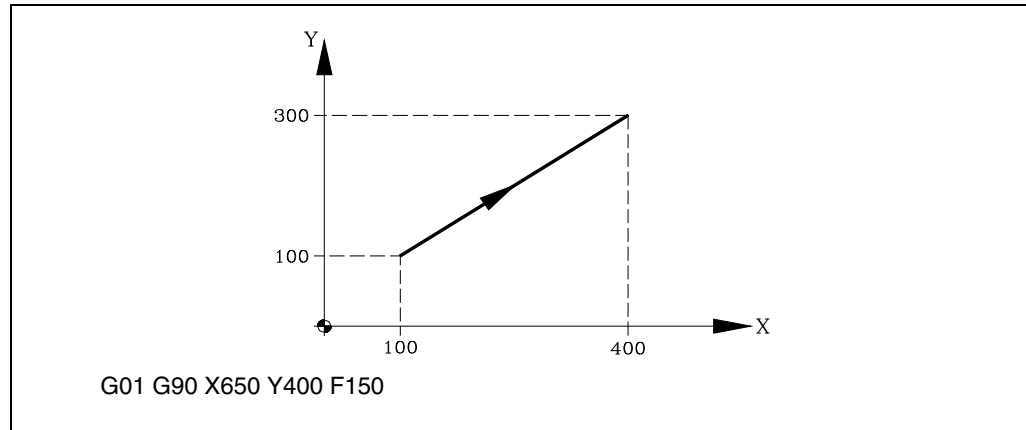


G01 G90 X650 Y400 F150

The programmed feedrate "F" may vary between 0% and 120% via the switch located on the Control Panel of the CNC, or by selecting between 0% and 255% from the PLC, or via the DNC or the program.

The CNC, however, is equipped with the general machine parameter "MAXFOVR" to limit maximum feedrate variation.

With this CNC, it is possible to program a positioning-only axis in a linear interpolation block. The CNC will calculate the feedrate for this positioning-only axis so it reaches the target coordinate at the same time as the interpolating axes.

Function G01 is modal and incompatible with G00, G02, G03, G33 and G34. Function G01 can be programmed as G1.

On power-up, after executing M02, M30 or after EMERGENCY or RESET, the CNC assumes code G00 or G01, depending on how general machine parameter "IMOVE" has been set.

## 6.3    Circular interpolation (G02, G03)

There are two ways of carrying out circular interpolation:

G02: Clockwise circular interpolation.

G03: Counterclockwise circular interpolation.

Movements programmed after G02 and G03 are executed in the form of a circular path and at the programmed feedrate "F".

Clockwise (G02) and counterclockwise (G03) definitions are established according to the system of coordinates shown below:
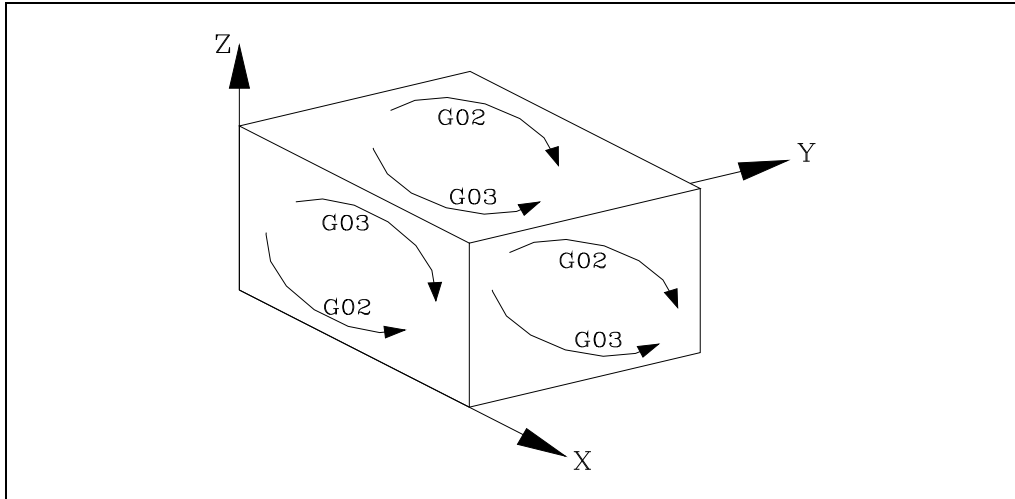


This system of coordinates refers to the movement of the tool on the part.

Circular interpolation can only be executed on a plane. The form of definition of circular interpolation is as follows :

## Cartesian coordinates

The coordinates of the endpoint of the arc and the position of the center with respect to the starting point are defined according to the axes of the work plane.

The center coordinates are defined in radius by the letters I, J, or K, each one of these being associated to the axes as follows: When not defining the center coordinates, the CNC assumes that their value is zero.



| Axes X, U, A | ==> | I |
| Axes Y, V, B | ==> | J |
| Axes Z, W, C | ==> | K |
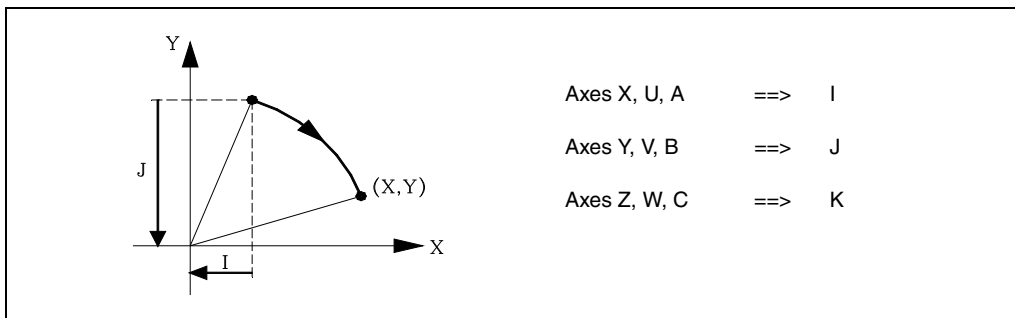
Programming format:

| Plane XY: | G02(G03) | X±5.5 | Y±5.5 | I±6.5 | J±6.5 |
| Plane ZX: | G02(G03) | X±5.5 | Z±5.5 | I±6.5 | K±6.5 |
| Plane YZ: | G02(G03) | Y±5.5 | Z±5.5 | J±6.5 | K±6.5 |

The programming order of the axes is always maintained regardless of the plane selected,, as are the respective center coordinates.

| | | | | | |
|---|---|---|---|---|---|
| Plane AY: | G02(G03) | Y±5.5 | A±5.5 | J±6.5 | I±6.5 |
| Plane XU: | G02(G03) | X±5.5 | U±5.5 | I±6.5 | I±6.5 |

## Polar coordinates

It is necessary to define the angle to be traveled Q and the distance from the starting point to the center (optional), according to the axes of the work plane.

The center coordinates are defined by the letters I, J, or K, each one of these being associated to the axes as follows:

| | | |
|---|---|---|
| Axes X, U, A | ==> | I |
| Axes Y, V, B | ==> | J |
| Axes Z, W, C | ==> | K |

If the center of the arc is not defined, the CNC will assume that it coincides with the current polar origin.

Programming format:

| | | | | |
|---|---|---|---|---|
| Plane XY: | G02(G03) | Q±5.5 | I±6.5 | J±6.5 |
| Plane ZX: | G02(G03) | Q±5.5 | I±6.5 | K±6.5 |
| Plane YZ: | G02(G03) | Q±5.5 | J±6.5 | K±6.5 |

## Cartesian coordinates with radius programming

The coordinates of the endpoint of the arc and radius R are defined.

Programming format:

| | | | | |
|---|---|---|---|---|
| Plane XY: | G02(G03) | X±5.5 | Y±5.5 | R±6.5 |
| Plane ZX: | G02(G03) | X±5.5 | Z±5.5 | R±6.5 |
| Plane YZ: | G02(G03) | Y±5.5 | Z±5.5 | R±6.5 |

If a complete circle is programmed, with radius programming, the CNC will show the corresponding error, as infinite solutions exist.

If an arc is less than 180o, the radius is programmed with a plus sign, and a minus sign if it is more than 180o.
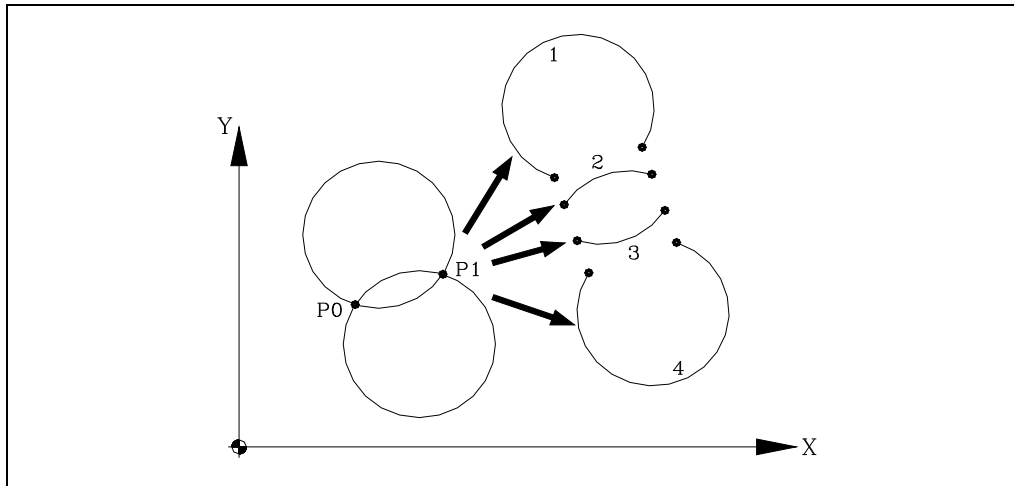
If P0 is the starting point and P1 the endpoint, there are 4 arcs which have the same value passing through both points.

Depending on the circular interpolation G02 or G03, and on the radius sign, the relevant arc is defined. Thus the programming format of the sample arcs is as follows:

| Arc 1 | G02 X.. Y.. R- .. |
|-------|-------------------|
| Arc 2 | G02 X.. Y.. R+.. |
| Arc 3 | G03 X.. Y.. R+.. |
| Arc 4 | G03 X.. Y.. R- .. |

## Execution of the circular interpolation

The CNC calculates, depending on the programmed arc, the radii of the starting point and endpoint. Although both of them should be "exactly" the same, general parameter "CIRINERR" allows a certain calculation tolerance by establishing the maximum difference between these two radii. When exceeding this value, the CNC will issue the corresponding error message.

In all these programming cases, the CNC checks that the center or radius coordinates do not exceed 214748.3647mm. Otherwise, the CNC will display the corresponding error.

The programmed feedrate "F" may vary between 0% and 120% via the switch located on the Control Panel of the CNC, or by selecting between 0% and 255% from the PLC, or via the DNC or the program.

The CNC, however, is equipped with the general machine parameter "MAXFOVR" to limit maximum feedrate variation.

If the general machine parameter "PORGMOVE" has been selected and a circular interpolation (G02 or G03) is programmed, the CNC assumes the center of the arc to be a new polar origin.

Functions G02 and G03 are modal and incompatible both among themselves and with G00, G01, G33 and G34. Functions G02 and G03 can be programmed as G2 and G3.

Also, function G74 (home search) and G75 (probing) cancel the G02 and G03 functions.

On power-up, after executing M02, M30 or after EMERGENCY or RESET, the CNC assumes code G00 or G01, depending on how general machine parameter "IMOVE" has been set.

## Programming examples
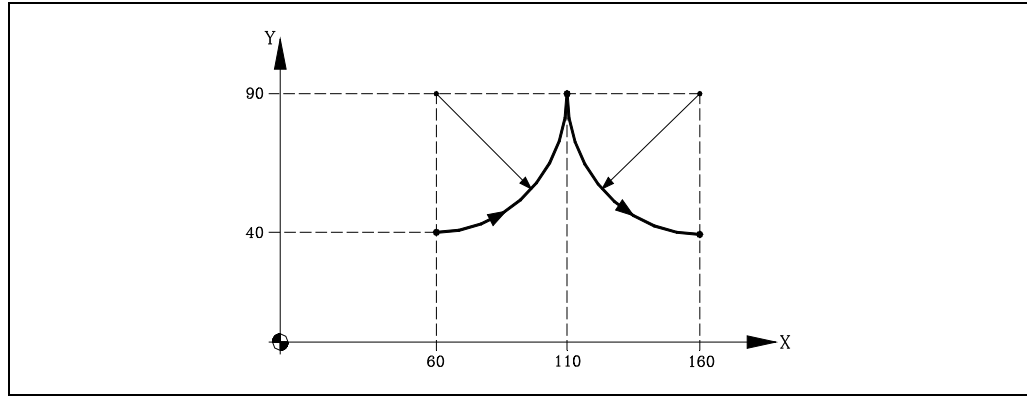
Various programming modes are analyzed below, point X60 Y40 being the starting point.

Cartesian coordinates:

```
G90 G17 G03 X110 Y90 I0 J50
X160 Y40 I50 J0
```

Polar coordinates:

```
G90 G17 G03 Q0 I0 J50
Q-90 I50 J0
```

or:

```
G93 I60 J90              ; defines polar center
G03 Q0
G93 I160 J90             ; defines new polar center
Q-90
```

Cartesian coordinates with radius programming:

```
G90 G17 G03 X110 Y90 R50
X160 Y40 R50
```

### *Programming of a (complete) circle in just one block:*
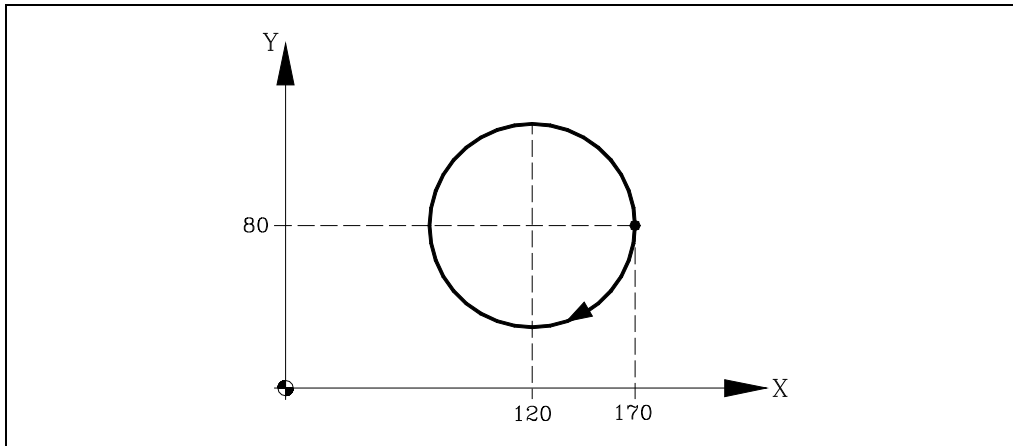


Various programming modes analyzed below, point X170 Y80 being the starting Point.

Cartesian coordinates:

    G90 G17 G02 X170 Y80 I-50 J0

or:

    G90 G17 G02 I-50 J0

Polar coordinates.

    G90 G17 G02 Q36 0I-50 J0

or:

    G93 I120 J80               ; defines polar center
    G02 Q360

Cartesian coordinates with radius programming:

    A complete circle cannot be programmed as there is an infinite range of solutions.

## 6.4 Circular interpolation with absolute arc center coordinates (G06)

By adding function G06 to a circular interpolation block you can program the coordinates of the center of the arc (I,J, or K) in absolute coordinates i.e. with respect to the zero origin and not to the beginning of the arc.

Function G06 is not modal, so it should be programmed any time the coordinates of the center of the arc are required in absolute coordinates. G06 can be programmed as G6.
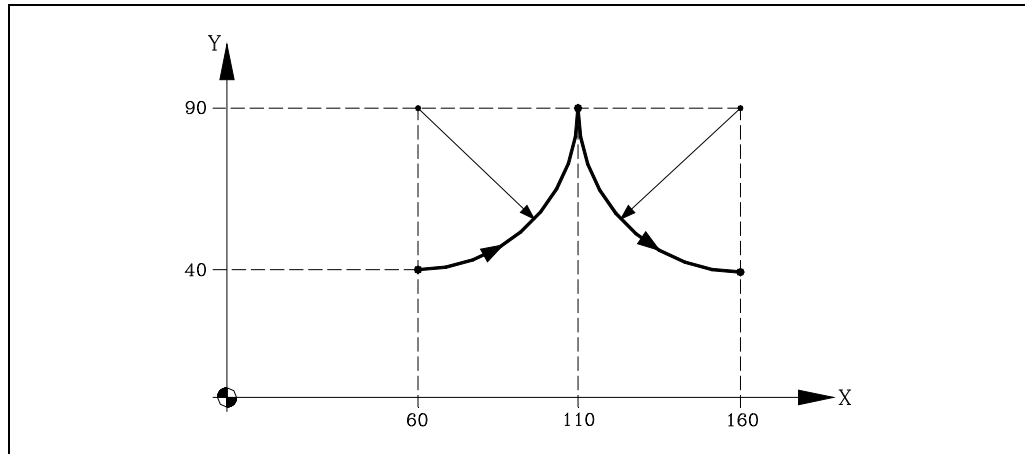


Various programming modes are analyzed below, point X60 Y40 being the starting point.

Cartesian coordinates:

    G90 G17 G06 G03 X110 Y90 I60 J90
    G06 X160 Y40 I160 J90

Polar coordinates:

    G90 G17 G06 G03 Q0 I60 J90
    G06 Q-90 I160 J90

## 6.5 Arc tangent to previous path (G08)

Via function G08 you can program an arc tangential to the previous path without having to program the coordinates (I.J &K) of the center.

Only the coordinates of the endpoint of the arc are defined, either in polar coordinates or in Cartesian coordinates according to the axes of the work plane.
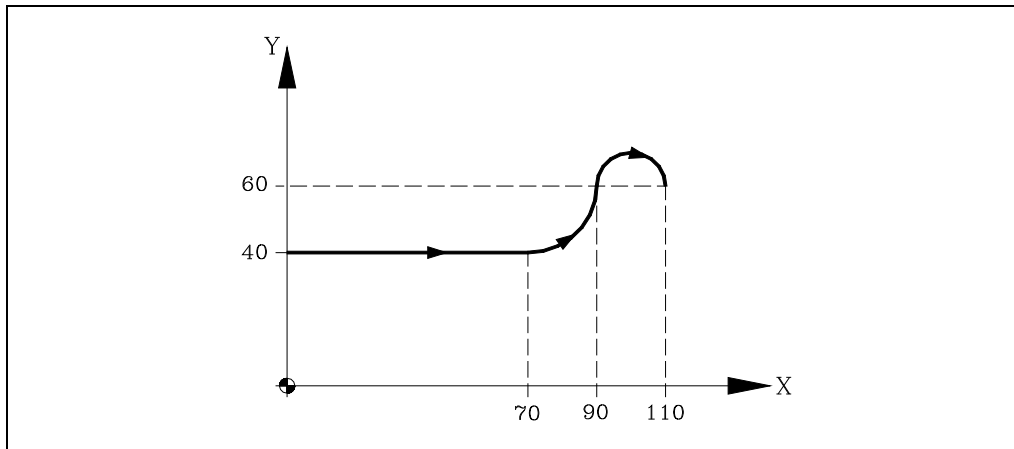


Supposing that the starting point is X0 Y40, you wish to program a straight line, then an arc tangential to the line and finally an arc tangential to the previous one.

    G90 G01 X70

    G08 X90 Y60              ; Arc tangent to previous path.

    G08 X110 Y60             ; Arc tangent to previous path.

Function G08 is not modal, so it should always be programmed if you wish to execute an arc tangential to the previous path. Function G08 can be programmed as G8.

Function G08 enables the previous path to be a straight line or an arc and does not alter its history. The same function G01, G02 or G03 stays active after the block is finished.

> ⚠ *When using function G08 it is not possible to execute a complete circle, as an infinite range of solutions exists. The CNC displays the corresponding error code.*

## 6.6     Arc defined by three points (G09)

Through function G09 you can define an arc by programming the endpoint and an intermediate point (the starting point of the arc is the starting point of the movement). In other words, instead of programming the coordinates of the center, you program any intermediate point.

The endpoint of the arc is defined in Cartesian or polar coordinates, and the intermediate point is always defined in Cartesian coordinates by the letters I,J, or K, each one being associated to the axes as follows:

| | | |
|---|---|---|
| Axes X, U, A | ==> | I |
| Axes Y, V, B | ==> | J |
| Axes Z, W, C | ==> | K |

In Cartesian coordinates:

| | | | | | |
|---|---|---|---|---|---|
| G17 | G09 | X±5.5 | Y±5.5 | I±5.5 | J±5.5 |

Polar coordinates:

| | | | | | |
|---|---|---|---|---|---|
| G17 | G09 | R±5.5 | Q±5.5 | I±5.5 | J±5.5 |

Example:
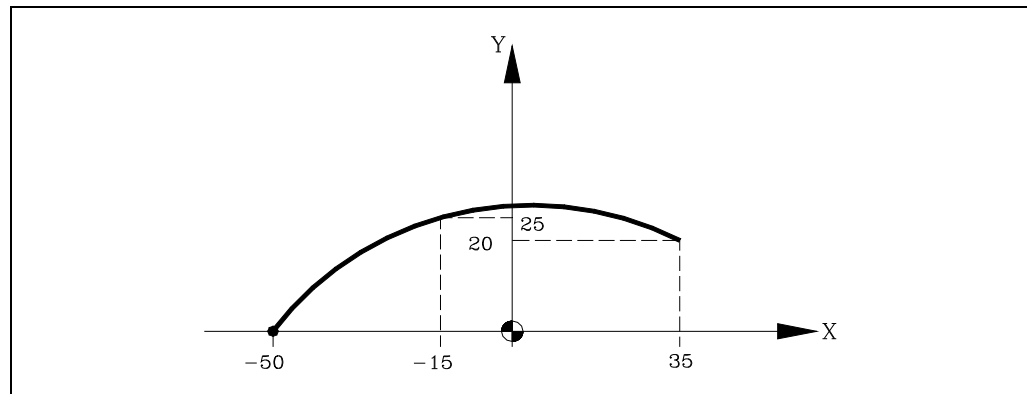


Being initial point X-50 Y0.

    G09 X35 Y20 I-15 J25

Function G09 is not modal, so it should always be programmed if you wish to execute an arc defined by three points. Function G09 can be programmed as G9.

When G09 is programmed it is not necessary to program the direction of movement (G02 or G03).

Function G09 does not alter the history of the program. The same G01, G02 or G03 function stays active after finishing the block.

> ⚠️ *Function G09 may not be used to programmed a full circle because all three points must be different. The CNC displays the corresponding error code.*

**FAGOR**

FAGOR AUTOMATION

**CNC 8055
CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2x

·**94**·

## 6.7    Helical interpolation

A helical interpolation consists in a circular interpolation in the work plane while moving the rest of the programmed axes.
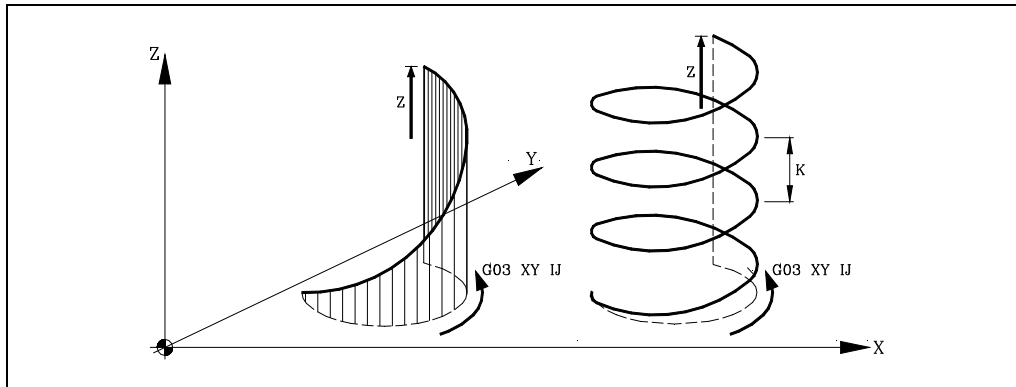


The helical interpolation is programmed in a block where the circular interpolation must be programmed by means of functions: G02, G03, G08 or G09.

    G02 X Y I J Z
    G02 X Y R Z A
    G03 Q I J A B
    G08 X Y Z
    G09 X Y I J Z

If the helical interpolation is supposed to make more than one turn, the linear movement of another axis must also be programmed (one axis only).

On the other hand, the pitch along the linear axis must also be set (format 5.5) by means of the I, J and K letters. Each one of these letters is associated with the axes as follows:

    Axes X, U, A        ==>    I

    Axes Y, V, B        ==>    J

    Axes Z, W, C        ==>    K

    G02 X Y I J Z K
    G02 X Y R Z K
    G03 Q I J A I
    G08 X Y B J
    G09 X Y I J Z K

---

**Example:**



Programming a helical interpolation where the starting point is X0 Y0 Z0.

As the example shows, it is not necessary to program the last point (X, Y):

    G03 I15 J0 Z18 K5

---
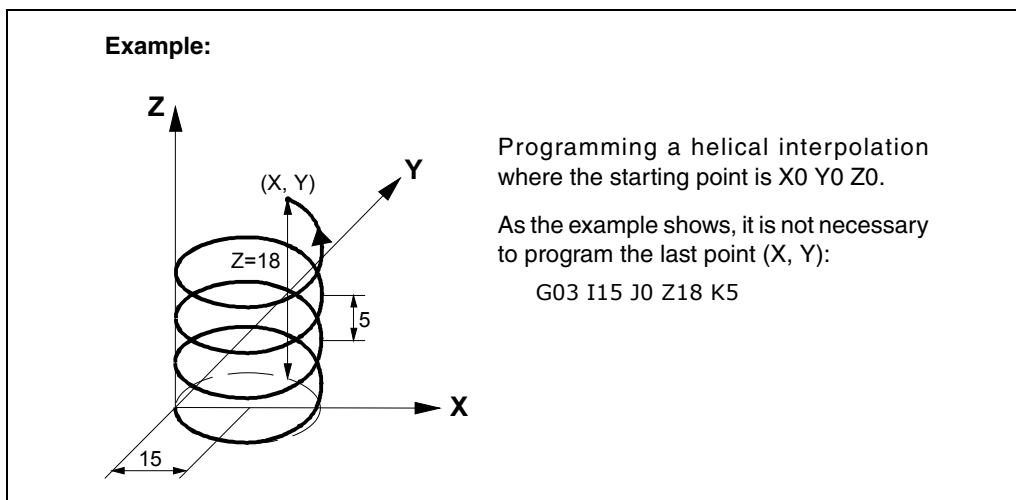
It is now possible to program helical interpolations while Look Ahead is active (G51). Thanks to this, CAN/CAM programs that contain this type of paths may be executed while look-ahead is active.

## 6.8    Tangential entry at the beginning of a machining operation (G37)

Via function G37 you can tangentially link two paths without having to calculate the intersection points.

Function G37 is not modal, so it should always be programmed if you wish to start a machining operation with tangential entry:
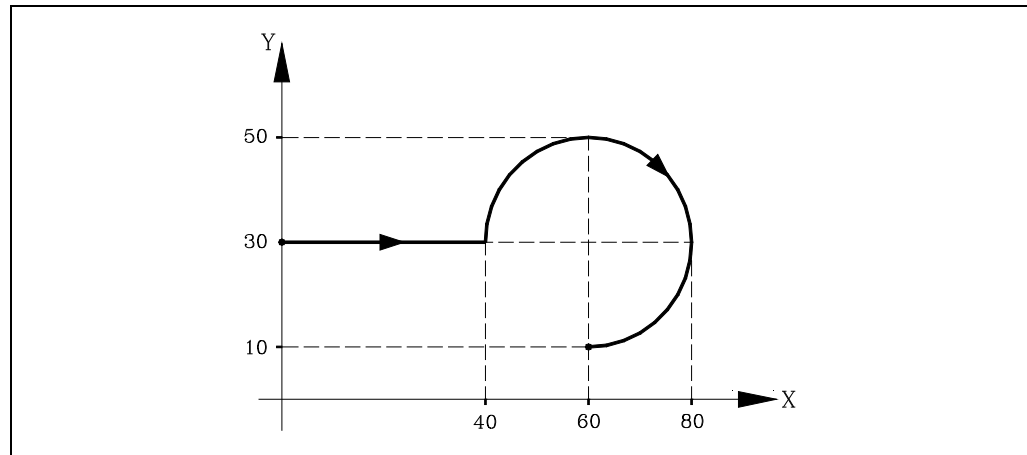


If the starting point is X0 Y30 and you wish to machine an arc (the path of approach being straight) you should program:

```
G90 G01 X40
G02 X60 Y10 I20 J0
```

If, however, in the same example you require the entrance of the tool to the part to be machined tangential to the path and describing a radius of 5 mm, you should program:
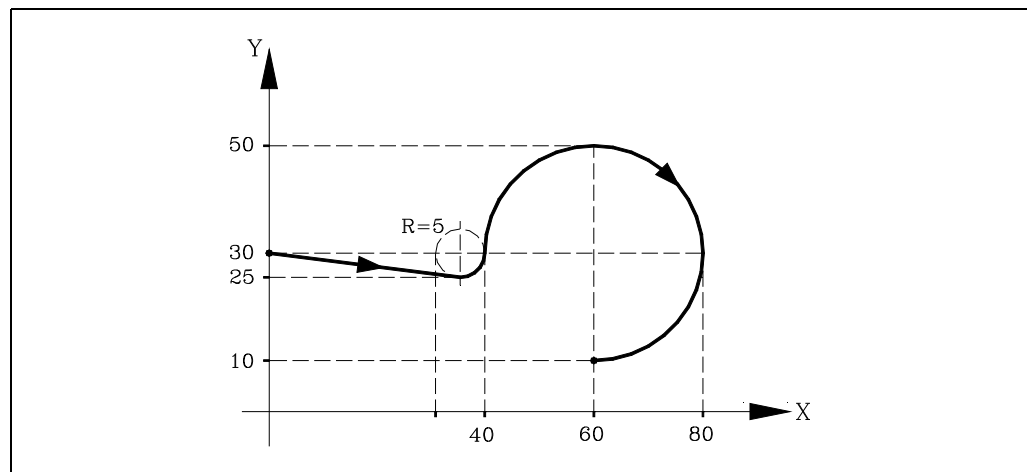


```
G90 G01 G37 R5 X40
G02 X60 Y10 I20 J0
```

As can be seen in the figure, the CNC modifies the path so that the tool starts to machine with a tangential entry to the part.

You have to program Function G37 plus value R in the block which includes the path you want to modify.

R5.5 should appear in all cases following G37, indicating the radius of the arc which the CNC enters to obtain tangential entry to the part. This R value must always be positive.

Function G37 should only be programmed in the block which includes a straight-line movement (G00 or G01). If you program in a block which includes circular movement (G02 or G03), the CNC displays the corresponding error.

## 6.9    Tangential exit at the end of a machining operator (G38)

Function G38 enables the ending of a machining operation with a tangential exit of the tool. The path should be in a straight line (G00 or G01). Otherwise, the CNC will display the corresponding error.

Function G38 is not modal, so it should be programmed whenever a tangential exit of the tool is required.

Value R 5.5 should always appear after G38. It also indicates the radius of the arc which the CNC applies to get a tangential exit from the part. This R value must always be positive.
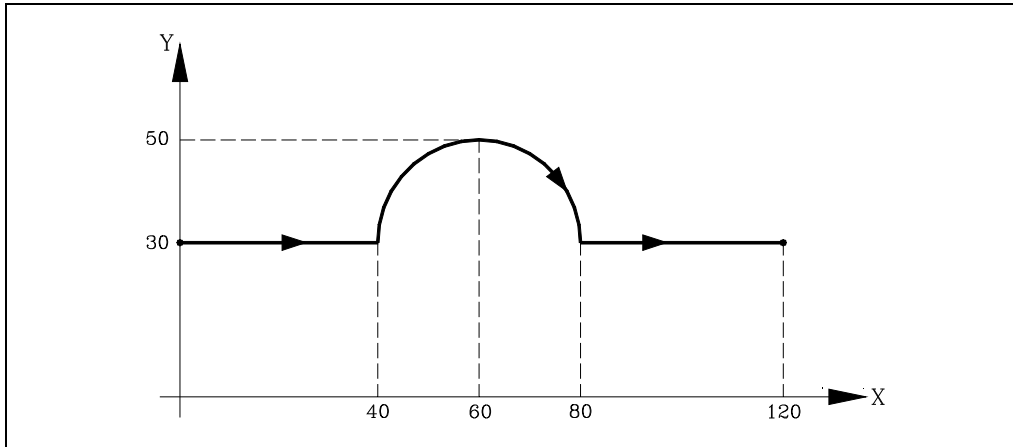


If the starting point is X0 Y30 and you wish to machine an arc (with the approach and exit paths in a straight line), you should program :

```
G90 G01 X40
G02 X80 I20 J0
G00 X120
```

If, however, in the same example you wish the exit from machining to be done tangentially and describing a radius of 5 mm, you should program :

```
G90 G01 X40
G02 G38 R5 X80 I20 J0
G00 X120
```

**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2x

## 6.10    Automatic radius blend (G36)

In milling operations, it is possible to round a corner via function G36 with a determined radius, without having to calculate the center nor the start and end points of the arc.

Function G36 is not modal, so it should be programmed whenever controlled corner rounding is required.

This function should be programmed in the block in which the movement the end you want to round is defined.

The R5.5 value should always follow G36. It also indicates the rounding radius which the CNC applies to get the required corner rounding. This R value must always be positive.
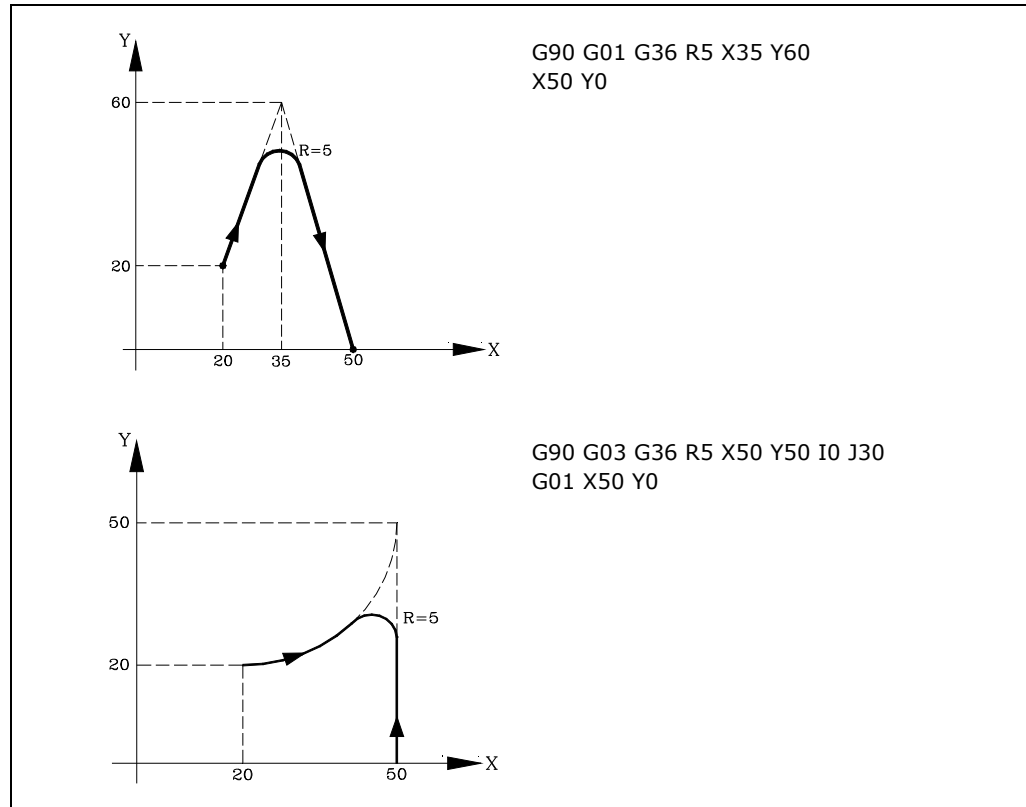
G90 G01 G36 R5 X35 Y60
X50 Y0

G90 G03 G36 R5 X50 Y50 I0 J30
G01 X50 Y0

## 6.11    Chamfer (G39)

In machining operations it is possible (using G39) to chamfer corners between two straight lines, without having to calculate intersection points.

Function G39 is not modal, so it should be programmed whenever the chamfering of a corner is required.

This function should be programmed in the block in which the movement whose end you want to chamfer is defined.

The R5.5 value should always follow G39. It also indicates the distance from the end of the programmed movement as far as the point where you wish to carry out the chamfering. This R value must always be positive.
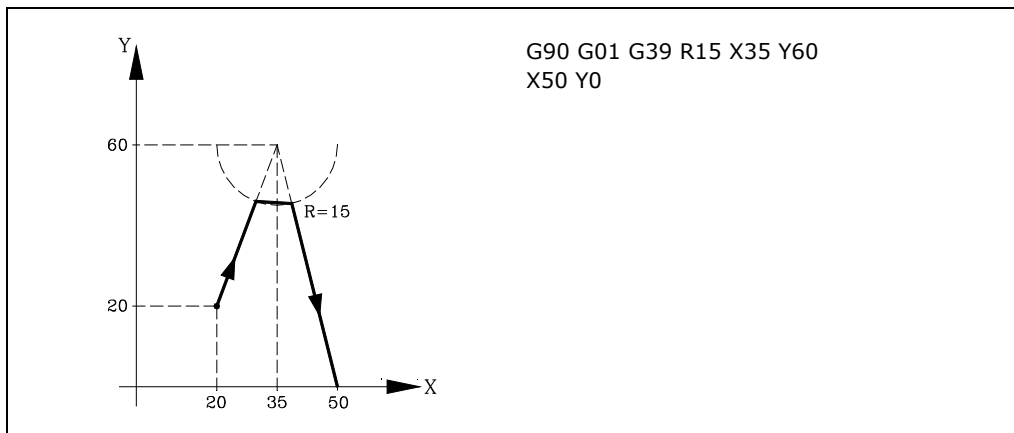


```
G90 G01 G39 R15 X35 Y60
X50 Y0
```

## 6.12    Threading (G33)

If the machine spindle is equipped with a rotary encoder, you can thread with a tool tip via function G33.

Although this threading is often done along the entire length of an axis, the CNC enables threading to be done interpolating more than one axis at a time.

Programming format:

G33 X.....C L Q

| | | |
|---|---|---|
| X...C ±5.5 | | End point of the thread |
| L 5.5 | | Thread pitch |
| Q ±3.5 | | Optional. It indicates the spindle angular position (±359.9999) of the thread's starting point. If not programmed, a value of 0 is assumed. |

### *Considerations:*

Whenever G33 is executed, if s.m.p. M19TYPE (P43) =0, and before making the servo-driven thread, the CNC references the spindle (home search).

Spindle machine parameter M19TYPE (P43) must be set to "1" in order to be able to program parameter Q (angular spindle position).

When executing function G33 Q (s.m.p. M19TYPE (P43) =1), and before executing the thread, a home search must be made for the spindle after the last power-up.

When executing function G33 Q (s.m.p. M19TYPE (P43) =1), and the s.m.p. DECINPUT (P31) =NO, a home search is not needed for the spindle, since the CNC will be homed automatically after power-up when turning the spindle in M3 or in M4 for the first time.

This search will be carried out at the feedrate set by s.m.p. REFEED2 (P35). After finding home, the spindle will speed up or slow down to the programmed speed without stopping.

If the spindle has motor feedback with a SINCOS encoder (without reference mark), the home search will be done directly at the programmed S speed without going through the speed set by s.m.p. REFEED2.

If after power-up, an M19 is executed before an M3 or M4, that M19 will be executed without homing the spindle when executing the first M3 or M4.

If the feedback device does not have the reference mark synchronized, the home search in M3 might not coincide with the home search in M4. This does not happen with FAGOR feedback.

If the threads are blended together in round corner, only the first one can have an entry angle (Q).

While function G33 is active, neither the programmed feedrate "F" nor the programmed Spindle speed "S" can be varied. They will both be set to 100%.

Function G33 is modal and incompatible with G00, G01, G02, G03, G34 and G75.

On power-up, after executing M02, M30 or after EMERGENCY or RESET, the CNC assumes code G00 or G01, depending on how general machine parameter "IMOVE" has been set.

**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

*Example:*

We would like to a make a thread in a single pass in X0 Y0 Z0, with a depth of 100 mm and a pitch of 5 mm using a threadcutting tool located in Z10.
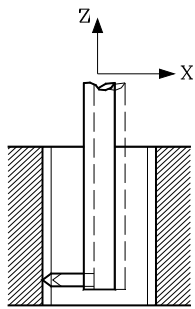


| | |
|---|---|
| G90 G0 X Y Z | ; Positioning |
| G33 Z -100 L5 | ; Threading |
| M19 | ; Spindle orientation |
| G00 X3 | ; Cutter withdrawal |
| Z30 | ; Withdrawal (exit the hole) |
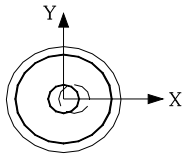
**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2x

## 6.13    Variable pitch threads (G34)

To make variable-pitch threads, the spindle of the machine must have a rotary encoder.

Although this threading is often done along the entire length of an axis, the CNC enables threading to be done interpolating more than one axis at a time.

Programming format:

G34 X.....C L Q K

| | | |
|---|---|---|
| X...C ±5.5 | End point of the thread | |
| L 5.5 | Thread pitch | |
| Q ±3.5 | Optional. It indicates the spindle angular position (±359.9999) of the thread's starting point. If not programmed, a value of "0" is assumed. | |
| K ±5.5 | Thread pitch increase or decrease per spindle turn. | |

### *Considerations:*

Whenever G34 is executed and before making the thread, the CNC references the spindle (home search) and positions the spindle at the angular position indicated by parameter Q.

Parameter "Q" is available when spindle machine parameter "M19TYPE" has been set to "1".

When working in round corner mode (G05), it is possible to blend different threads in the same part.

While function G34 is active, neither the programmed feedrate "F" nor the programmed Spindle speed "S" can be varied. They will both be set to 100%.

Function G34 is modal and incompatible with G00, G01, G02, G03, G33 and G75.

On power-up, after executing M02, M30 or after EMERGENCY or RESET, the CNC assumes code G00 or G01, depending on how general machine parameter "IMOVE" has been set.

**Blending a fixed-pitch thread (G33) with a variable-pitch thread (G34).**

The starting thread pitch (L) of the G34 must coincide with the thread pitch of the G33.

The pitch increase in the first turn of the spindle in variable pitch will be half and increment (K/2) and in the rest of the turns it will be the full increment K.

**Blending a variable-pitch thread (G34) with a fixed-pitch thread.**

It is used to finish a variable-pitch thread (G34) with a portion of the thread keeping the final pitch of the previous thread.

Since it is very complicated to calculate the final thread pitch, the fixed-pitch thread is not programmed with G33 but with G34 ... L0 K0. The CNC calculates the pitch.

**Blending two variable-pitch threads (G34).**

Two variable-pitch threads (G34) cannot be blended together.

## 6.14    Move to hardstop (G52)

By means of function G52 it is possible to program the movement of an axis until running into an object. This feature may be interesting for forming machines, live tailstocks, bar feeders, etc.

The programming format is:

G52 X..C ±5.5

After G52, program the desired axis as well as the target coordinate of the move.

The axis will move towards the programmed target coordinate until running into something. If the axis reaches the programmed target coordinate without running into the hardstop it will stop.

Function G52 is not modal; therefore, it must be programmed every time this operation is to be carried out.

Also, it assumes functions G01 and G40 modifying the program history. It is incompatible with functions G00, G02, G03, G33, G34, G41, G42, G75 and G76.

## 6.15    Feedrate "F" as an inverted function of time (G32)

There are instances when it is easier to define the time required by the various axes of the machine to reach the target point instead of defining a common feedrate for all of them.

A typical case may be when a linear axis (X, Y, Z) has to move together (interpolated) with a rotary axis programmed in degrees.

Function G32 indicates that the "F" functions programmed next set the time it takes to reach the target point.

In order for a greater value of "F" to indicate a greater feedrate, the value assigned to "F" is defined as "Inverted function of time" and it is assumed as the activation of this feature.

"F" units: 1/min

Example: G32 X22 F4

indicates that the movement must be executed in 1/4 minute; i.e. in 0.25 minutes.

Function G32 is modal and incompatible with G94 and G95.

On power-up, after executing M02, M30 or after an Emergency or Reset, the CNC assumes G94 or G95 depending on the setting of general machine parameter "IFFED".

### *Considerations:*

The CNC variable PROGFIN will show the feedrate programmed as an inverted function of time and variable FEED will show the resulting feedrate in mm/min or inches/min.

If the resulting feedrate of any axis exceeds the maximum value set by machine parameter "MAXFEED", the CNC will apply this maximum value.

The programmed "F" is ignored on G00 movements. All the movements will be carried out at the feedrate set by axis machine parameter "G00FEED".

When programming "F0" the movement will be carried out at the feedrate set by axis machine parameter "MAXFEED".

Function G32 may be programmed and executed in the PLC channel.

Function G32 is canceled in JOG mode.

**FAGOR**

FAGOR AUTOMATION

**CNC 8055
CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2x

·104·

## 6.16    Tangential control (G45)

The "Tangential Control" function keeps an axis always in the same orientation with respect to the programmed path.



Orientation parallel to the path          Orientation perpendicular to the path
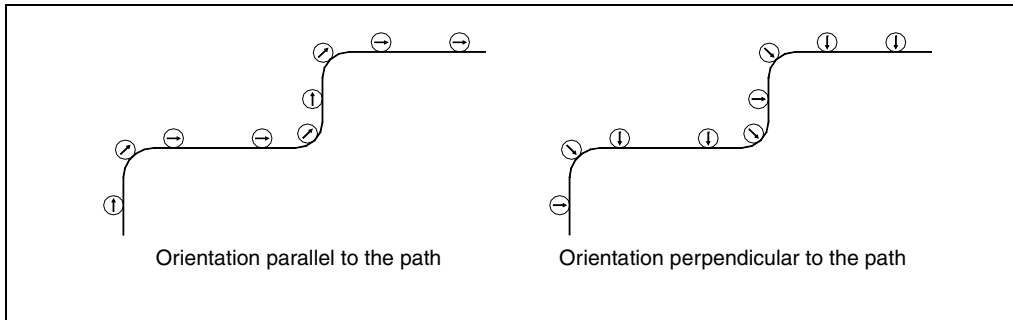
The path is defined by the axes of the active plane. The axis maintaining the orientation must be a rotary rollover axis (A, B or C).

Programming format:

G45   Axis Angle

> Axis       Axis maintaining the orientation (A, B or C).
>
> Angle      Indicates the angular position in degrees with respect to the path (±359.9999). If not programmed, "0" will be assumed.
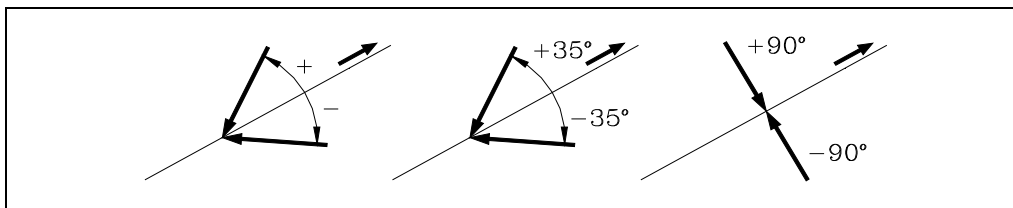


To cancel this function, program G45 alone (without defining the axis).

Every time G45 (tangential control) is activated, the CNC acts as follows:

1. Positions the tangential axis, with respect to the first section in the programmed position.
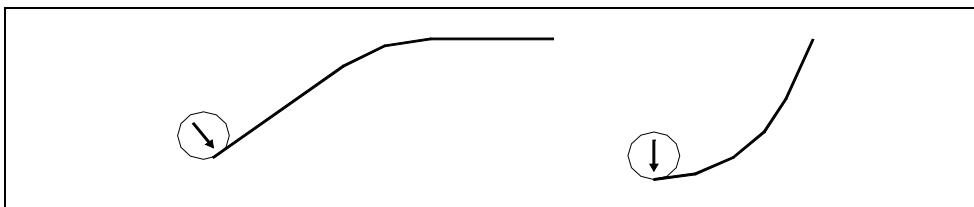


2. The interpolation of the plane axes starts once the tangential axis has been positioned.

3. The orientation of the tangential axis is maintained in linear sections whereas in circular interpolations the programmed orientation is maintained throughout the whole travel.
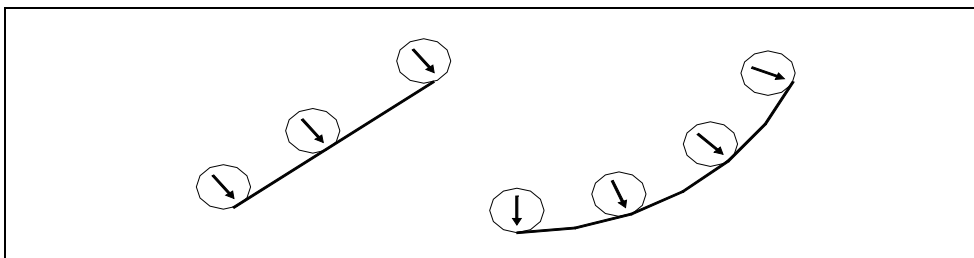
**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

4. If the joint of sections requires a new orientation of the tangential axis, the following takes place:

· **·1·** Ends the current section.

· **·2·** Orients the tangential axis with respect to the next section.

· **·3·** Resumes the execution.
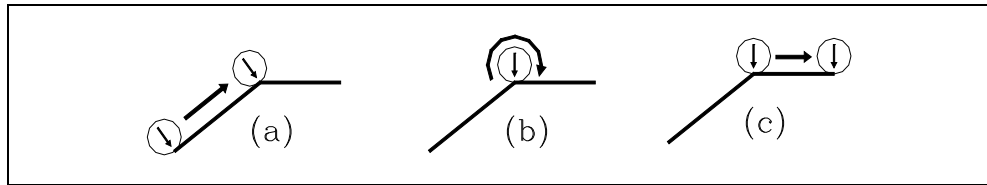


When working in round corner (G05), the tool orientation is not maintained at the corners since it begins before ending the current section.

It is recommended to work in square corner (G07). However, to work in round corner (G05), function G36 (automatic radius blend) should be used in order to also maintain tool orientation at the corners.

4. To cancel this function, program G45 alone (without defining the axis).

Even when the tangential axis takes the same orientation by programming 90° or -270°, the turning direction in a direction change depends on the programmed value.
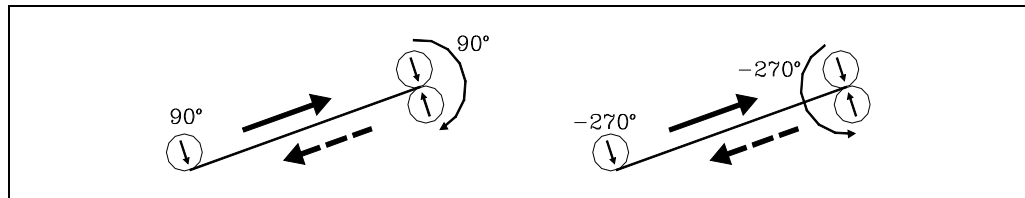
### 6.16.1 Considerations about the G45 function

Tangential control, G45, is optional. It can only be executed in the main channel and is compatible with:

- Tool radius and length compensation (G40, 41, 42, 43, 44).
- Mirror image (G10, 11, 12, 13 14).
- Gantry axes , including the gantry axis associated with the tangential rotary axis.

The maximum feedrate while orienting the tangential axis is defined by machine parameter MAXFEED for that axis.

While tangential control is active, tool inspection is also possible. When accessing tool inspection, the tangential control is deactivated, the axes are free and when quitting tool inspection, tangential control may be activated again.

While in JOG mode, tangential control may be activated in MDI mode and the axes may be moved by programming blocks in MDI.

Tangential control is canceled when jogging the axes with the jog keys (not in MDI). Once the movement is over, tangential control is recovered.

Likewise, the following is NOT possible:

- To define as tangential axis, one of the plane axes, the longitudinal axis or any other axis which is not rotary.
- To jog the tangential axis in JOG mode or by program using another G code while tangential control is active.
- Inclined planes.

The TANGAN variable is read-only, from the CNC, PLC and DNC, associated with function G45. It indicates the angular position, in degrees, referred to the programmed path.

Also, general logic output TANGACT (M5558) indicates to the PLC that function G45 is active.

Function G45 is modal and is canceled when executing G45 alone (without defining the axis), on power-up, after executing an M02 or M30 or after an EMERGENCY or RESET.

**6.**

**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2x

## 6.17    G145. Temporary cancellation of tangential control

Function G145 may be used to cancel the tangential control (G415) temporarily:

### G145 K0

It cancels the tangential control temporarily. Function G45 stays in the history and the new function G145 comes up in it.

If no G45 has been programmed, G145 is ignored. If K is not programmed, K0 will be assumed.

### G145 K1

It resumes the tangential control of the axis with the angle it had before it was canceled. Then, G145 disappears from history.

# ADDITIONAL PREPARATORY FUNCTIONS

**7**

## 7.1    Interruption of block preparation (G04)

The CNC reads up to 20 blocks ahead of the one it is executing, with the aim of calculating beforehand the path to be followed.

Each block is evaluated (in its absence) at the time it is read, but if you wish to evaluate it at the time of execution of the block you use function G04.

This function holds up the preparation of blocks and waits for the block in question to be executed in order to start the preparation of blocks once more.

A case in point is the evaluation of the "status of block-skip inputs" which is defined in the block header.

**Example:**

```
.
.
   G04              ; Interrupts block preparation
/1 G01 X10 Y20      ; block-skip condition "/1"
.
.
```

Function G04 is not modal, so it should be programmed whenever you wish to interrupt block preparation.

It should be programmed on its own and in the block previous to the one in which the evaluation in execution is required. Function G04 can be programmed as G4.

Every time G04 is programmed, active radius and length compensation are cancelled.

For this reason, care needs to be taken when using this function, because if it is introduced between machining blocks which work with compensation, unwanted profiles may be produced.

**Example:**

The following program blocks are performed in a section with G41 compensation.

```
    ...
        N10 X50 Y80
        N15 G04
   /1  N17 M10
        N20 X50 Y50
        N30 X80 Y50
    ...
```

Block N15 interrupts block preparation and the execution of block N10 will finish at point A.
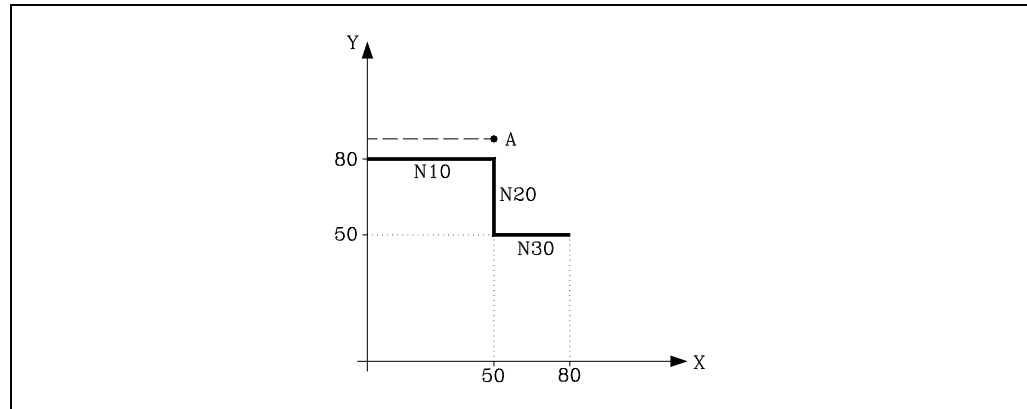


Once the execution of block N15 has been carried out, the CNC continues preparing blocks starting from block N17.

As the next point corresponding to the compensated path is point "B", the CNC will move the tool to this point, executing path "A-B".
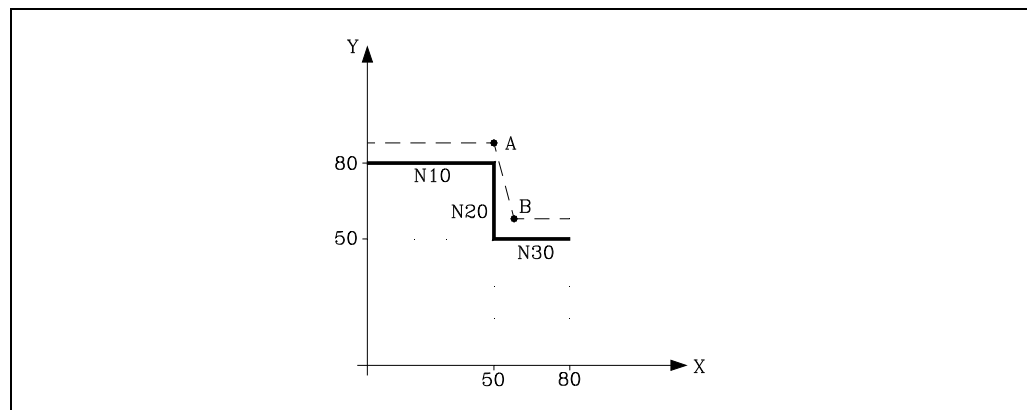


As you can see, the resulting path is not the required one, so we recommend avoiding the use of function G04 in sections which work with compensation.

**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

·110·

### 7.1.1 G04 K0: Block preparation interruption and coordinate update

The function associated with G04 K0 may be used to update the coordinates of the axes of the channel after finishing particular PLC routines.

The PLC routines that require updating the coordinates of the axes of the channel are the following:

• PLC routine using the SWITCH* marks.

• PLC routines where an axis goes into DRO mode and then back into normal axis mode during the execution of part programs.

G04 operation:

| Function | Description |
|----------|-------------|
| G04 | Interrupts block preparation. |
| G04 K50 | It executes a dwell 50 hundredths of a second. |
| G04 K0 or G04 K | It interrupts block preparation and updates the CNC coordinates to the current position.<br>(G4 K0 works in the CNC and PLC channel). |

If bit 10 of the g.m.p. ADIMPG (P176) =1, with the instruction G04 K0, the coordinates are initialized and the offset applied with the additive handwheel is eliminated on all the axes that had an offset.

The coordinates will be initialized to the real coordinates of the machine and the offset will be deleted without moving any machine axis at all.

## 7.2    Dwell (G04 K)

A dwell can be programmed via function G04 K.

The dwell value is programmed in hundredths of a second via format K5 (1..99999).

**Example:**

```
G04 K50      ; Dwell of 50 hundredths of a second (0.5 seconds)
G04 K200     ; Dwell of 200 hundredths of a second (2 seconds)
```

Function G04 K is not modal, so it should be programmed whenever a dwell is required. Function G04 K can be programmed as G4 K.

The dwell is executed at the beginning of the block in which it is programmed.

**Note:**  When programming G04 K0 or G04 K, instead of applying a delay, it only interrupts block preparation and it will refresh the coordinates. See *"7.1.1  G04 K0: Block preparation interruption and coordinate update"* on page 111.

## 7.3 Working with square (G07) and round (G05,G50) corners

### 7.3.1 G07 (square corner)

When working in G07 (square corner) the CNC does not start executing the following program block until the position programmed in the current block has been reached.

The CNC considers that the programmed position has been reached when the axis is within the "INPOSW" (in-position zone or dead band) from the programmed position.



G91 G01 G07 Y70 F100
X90
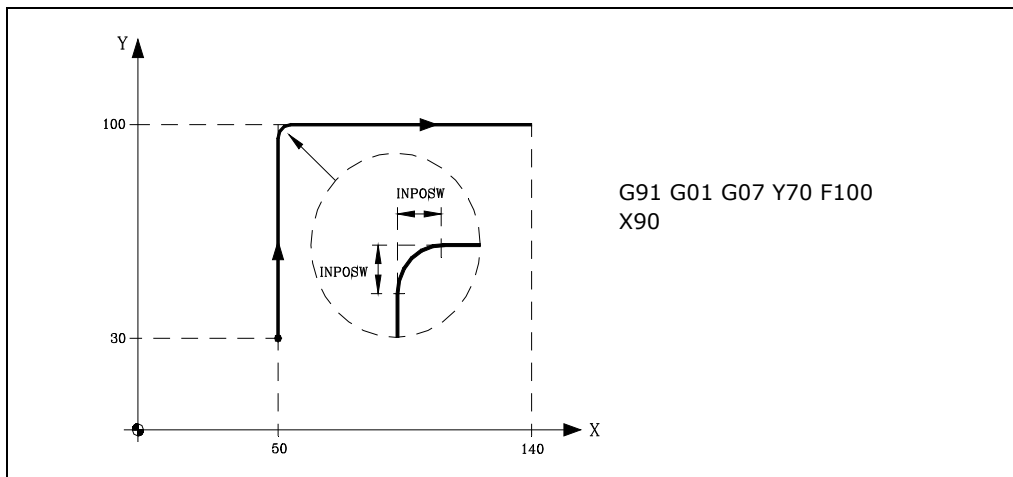
The theoretical and real profile coincide, obtaining square corners, as seen in the figure.

Function G07 is modal and incompatible with G05, G50 and G51. Function G07 can be programmed as G7.

On power-up, after executing M02, M30 or after EMERGENCY or RESET, the CNC assumes code G05 or G07 depending on how the general machine parameter "ICORNER" is set.

**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

### 7.3.2 G05 (round corner)

When working in G05 (round corner), the CNC starts executing the following block of the program as soon as the theoretical interpolation of the current block has concluded. It does not wait for the axes to physically reach the programmed position.

The distance prior to the programmed position where the CNC starts executing the next block depends on the actual axis feedrate.



```
G91 G01 G05 Y70 F100
X90
```
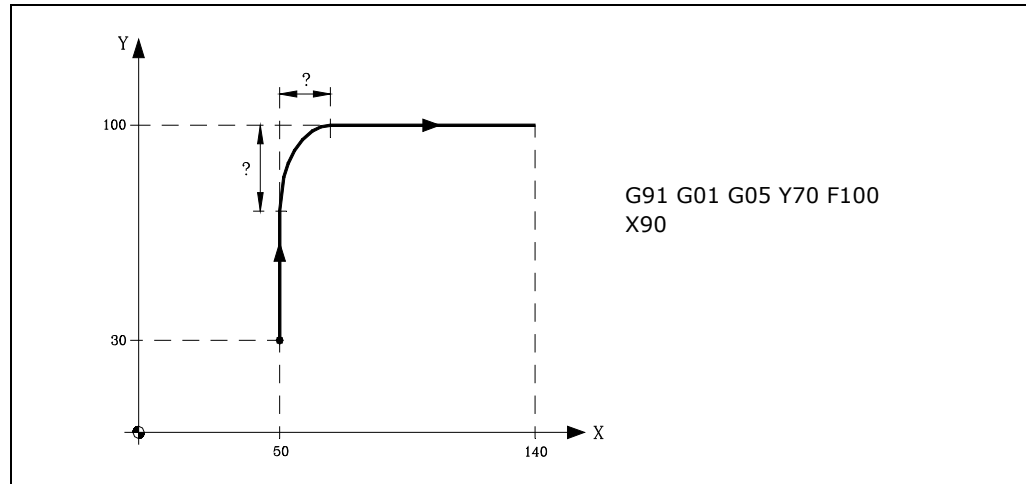
Via this function round corners can be obtained, as shown in the figure.

The difference between the theoretical and real profiles depends on the programmed feedrate value "F". The higher the feedrate, the greater the difference between both profiles.

Function G05 is modal and incompatible with G07, G50 and G51. Function G05 can be programmed as G5.

On power-up, after executing M02, M30 or after EMERGENCY or RESET, the CNC assumes code G05 or G07 depending on how the general machine parameter "ICORNER" is set.

### 7.3.3 Controlled round corner (G50)

When working in G50 (controlled round corner); once the theoretical interpolation of the current block has concluded, the CNC waits for the axis to enter the area defined by machine parameter "INPOSW2" and it then starts executing the following block of the program.



G91 G01 G50 Y70 F100
X90
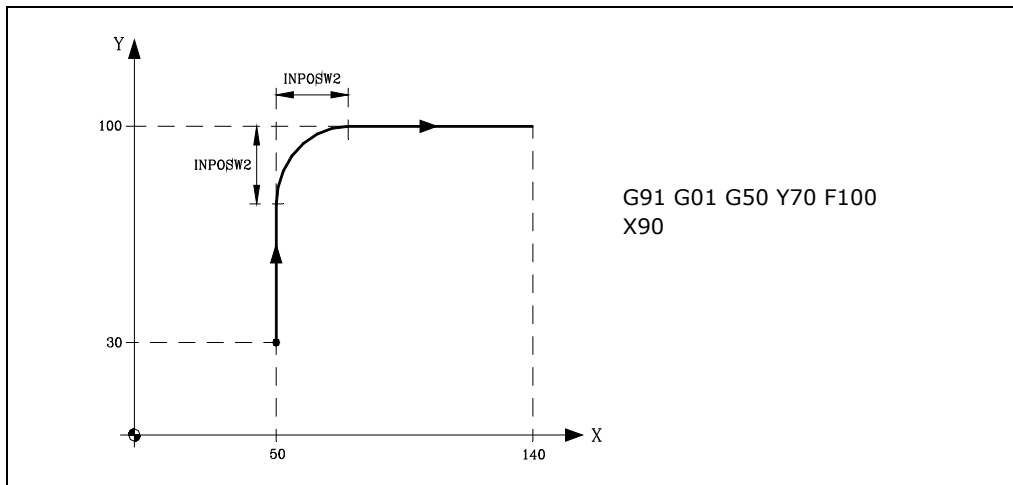
Function G50 assures that the difference between the theoretical and actual paths stays smaller than what was set by machine parameter "INPOSW2".

On the other hand, when working in G05, the difference between the theoretical and real profiles depends on the programmed feedrate value "F". The higher the feedrate, the greater the difference between both profiles.

Function G50 is modal and incompatible with G07, G05 and G51.

On power-up, after executing M02, M30 or after EMERGENCY or RESET, the CNC assumes code G05 or G07 depending on how the general machine parameter "ICORNER" is set.

**7.**

**ADDITIONAL PREPARATORY FUNCTIONS**

Working with square (G07) and round (G05,G50) corners

**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2x

## 7.4    Look-ahead (G51)

Programs consisting of very small movement blocks (CAM, digitizing, etc.) tend to run very slowly. Those programs may be executed at high machining speed using the "Look-Ahead" function.

The look-ahead function analyzes in advance the path to be machined (up to 75 blocks) in order to calculate the maximum feedrate for each section of the path. This function provides smoother and faster machining in programs with very small movements, even in the order of microns.

When operating with "Look-Ahead", it is a good idea to adjust the axes so their following error (lag) is as small as possible because the contouring error will be at least equal to the minimum following error.

### *Programming format:*

The programming format is:

G51 [A] E B

| A (0-255) | It is optional and it defines the percentage of acceleration to be applied. |
|---|---|
| | When not programmed or programmed with a "0" value, the CNC assumes the acceleration value set by machine parameter for each axis. |

| E (5.5) | Maximum contouring error. |
|---|---|
| | The lower this parameter value is, the lower the machining feedrate will be. |

| B (0-180) | It makes it possible to machine square corners using the Look-ahead function. |
|---|---|
| | It indicates the angular value (in degrees) of the programmed corners, below which, the machining operation will be carried out in square corner mode. |
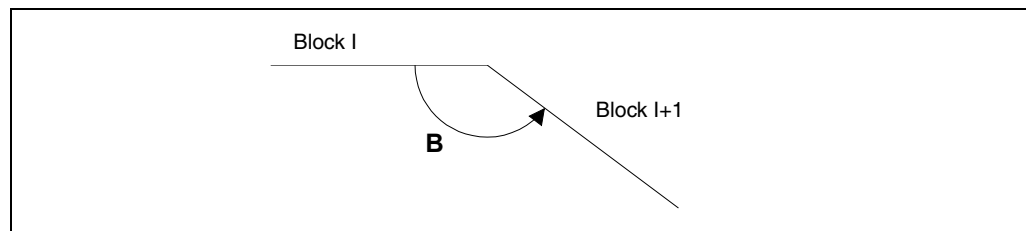


Parameter "A" permits using a standard working acceleration and another one to be used when executing with Look-Ahead.

If parameter "B" is not programmed, the square corner management at the corners is canceled.

The square-corner management at the corners is valid for the look-ahead algorithm with jerk management and for the look-ahead algorithm without jerk management.

### *Considerations for execution:*

When calculating the feedrate, the CNC takes the following into account:

- The programmed feedrate.
- The curvature and the corners.
- The maximum feedrate of the axes.
- The maximum accelerations.
- The jerk.

If any of the circumstances listed below occurs while executing with Look-Ahead, the CNC slows down to "0" at the previous block and it recovers the machining conditions for Look-Ahead in the next motion block.

- Motionless block.
- Execution of auxiliary functions (M, S, T).
- Single block execution mode.
- MDI mode.
- Tool inspection mode.

If a Cycle Stop, Feed-Hold, etc. occurs while executing in Look-Ahead mode, the machine may not stop at the current block, several additional blocks will be necessary to stop with the permitted deceleration.

To prevent motionless blocks from causing a square-corner effect, change bit 0 of general machine parameter MANTFCON (P189).

### *Function properties:*

Function G51 is modal and incompatible with G05, G07 and G50. Should any of them be programmed, function G51 will be canceled and the new one will be selected.

Function G51 must be programmed alone in a block and there must be no more information in that block.

On power-up, after executing an M02, M30, of after an EMERGENCY or RESET, the CNC will cancel G51, if it was active, and it will assume G05 or G07 according to the setting of general machine parameter "ICORNER".

On the other hand, the CNC will issue Error 7 (Incompatible G functions) when programming any of the following functions while G51 is active:

| | |
|---|---|
| G33 | Electronic threading (G33) |
| G34 | Variable-pitch threading. |
| G52 | Move to hardstop. |
| G95 | Feedrate per revolution. |

### 7.4.1 Advanced look-ahead algorithm (integrating Fagor filters)

This mode is indicated when machining accuracy is required, especially if Fagor filters have been set for the axes by machine parameters.

The advanced look-ahead algorithm calculates the feedrates at the corners so as to take the effect of those filters into consideration. When programming G51 E, the contouring errors when machining corners will be closer to the value programmed in that G51 function depending on the filters.

To activate the advanced look-ahead algorithm, use bit 15 of g.m.p. LOOKATYP (P160).

## Considerations

- If there are no Fagor filters set by machine parameters in the axes of the main channel, activating the advanced look-ahead algorithm will internally activate FIR filters of the 5th order and a frequency of 30 Hz in all the axes of the channel.

- If there are Fagor filters set by machine parameters, activating the advanced look-ahead algorithm will keep the values of those filters as long as their frequency is not higher than 30Hz

  If their frequency is higher than 30 Hz, it will assume the values of 5th order and a frequency of 30 Hz.

  If there are several filters defined in the axes of the channel, the one with the lowest frequency will be assumed as long as it is not higher than 30Hz.

- Even if the advanced look-ahead algorithm (using Fagor filters) is active with bit 15 of g.m.p. LOOKATYP (P160), it will not start working in the following cases:

  - If g.m.p. IPOTIME (P73) = 1.

  - If any of the axes of the main channel has a.m.p. SMOTIME (P58) not equal to 0.

  - If any of the axes of the main channel has defined by parameter a non-Fagor type filter, a.m.p. TYPE (P71) not equal to 2.

  In these cases, when activating G51, the CNC will issue the corresponding error message.

### 7.4.2 Look-ahead operation with Fagor filters active

This option makes it possible to use Fagor filters with Look-ahead (not advanced look-ahead algorithm). It will only be considered if the advanced look-ahead algorithm is deactivated; that is, if bit 15 of g.m.p. LOOKATYP (P160)=0.

To activate/deactivate this option, use bit 13 of g.m.p. LOOKATYP (P160).

## Effect of Fagor filters when machining circles

When machining circles, the error will be smaller when using Fagor filters than without using these filters:



Programmed movement.

Real (actual) movement using Fagor filters.

Real (actual) movement without using Fagor filters.

**7.**
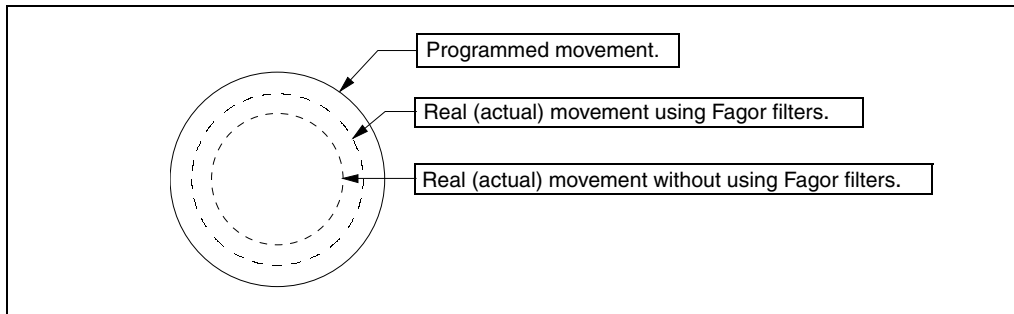
ADDITIONAL PREPARATORY FUNCTIONS
Look-ahead (G51)

**FAGOR**
FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

## 7.5    Mirror image (G10, G11. G12, G13, G14)

The functions to activate the mirror image are the following.

G10:          Cancel mirror image.

G11:          Mirror image on X axis.

G12:          Mirror image on the Y axis.

G13:          Mirror image on the Z axis

G14:          Mirror image on any axis (X..C), or in several at the same time.

**Examples:**

    G14 W
    G14 X Z A B

When the CNC works with mirror images, it executes the movements programmed in the axes that have mirror image selected, with the sign changed.
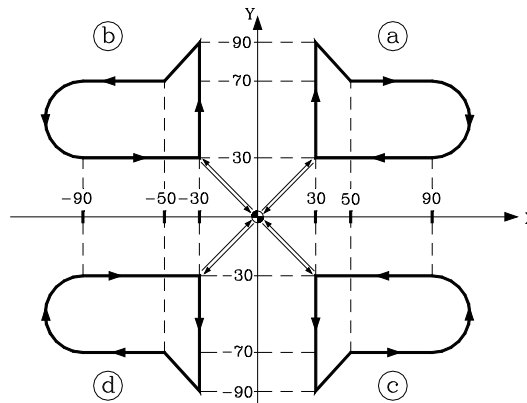


The following subroutine defines the machining of part "a".

    G91 G01 X30 Y30 F100
    Y60
    X20 Y-20
    X40
    G02 X0 Y-40 I0 J-20
    G01 X-60
    X-30 Y-30

The programming of all parts would be :

| Execution of subroutine | ; machines "a" |
|---|---|
| G11 | ; Mirror image on X axis. |
| Execution of subroutine | ; machines "b" |
| G10 G12 | ; Mirror image on Y axis. |
| Execution of subroutine; | machines "c" |
| G11 | ; Mirror image on X and Y axes. |
| Execution of subroutine; | Machines "d" |
| M30 | ; End of program |

Functions G11, G12, G13, and G14 are modal and incompatible with G10.

G11, G12, and G13 can be programmed in the same block, because they are not incompatible with each other. Function G14 must be programmed alone in the block.

If function G73 (pattern rotation) is also active in a mirror image program, the CNC first applies the mirror image function and then the pattern rotation.

If while one of the mirror imaging functions (G11, G12, G13, and G14) is active, a new coordinate origin (part zero) is preset with G92, this new origin will not be affected by the mirror imaging function.

On power-up, after executing M02, M30 or after EMERGENCY or RESET, the CNC assumes code G10.

## 7.6      Scaling factor (G72)

By using function G72 you can enlarge or reduce programmed parts.

In this way, you can produce families of parts which are similar in shape but of different sizes with a single program.

Function G72 should be programmed on its own in a block. There are two formats for programming G72 :

- Scaling factor applied to all axes.
- Scaling factor applied to one or more axes.

## 7.6.1    Scaling factor applied to all axes.

The programming format is:

G72 S5.5

Following G72 all coordinates programmed are multiplied by the value of the scaling factor defined by S until a new G72 scaling factor definition is read or the definition is canceled.

Programming example (starting point X-30 Y10)
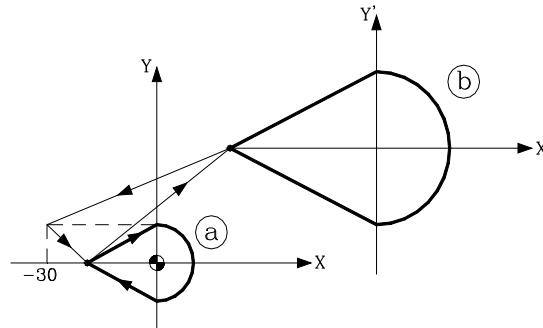


The following subroutine defines the machining of the part.

G90 X-19 Y0
G01 X0 Y10 F150
G02 X0 Y-10 I0 J-10
G01 X-19 Y0

The programming of the parts would be :

Execution of subroutine Machines "a".

G92 X-79 Y-30          ; Coordinate preset
                       (zero offset)
G72 S2                 ; Applies a scaling factor of 2

Execution of subroutine Machines "b".

G72 S1                 ; Cancel the scaling factor
M30                    ; End of program

Examples of application of the scaling factor.
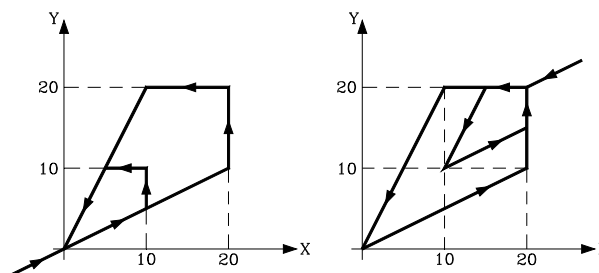


| G90 G00 X0 Y0 | G90 G00 X20 Y20 |
|---|---|
| N10 G91 G01 X20 Y10 | N10 G91 G01 X-10 |
| Y10 | Y-20 X-10 |
| X-10 | X20 Y10 |
| N20 X-10 Y-20 | N20 Y10 |
| ; Scaling factor | ; Scaling factor |
| G72 S0.5 | G72 S0.5 |
| ; Repeats from block 10 to block 20 | ; Repeats from block 10 to block 20 |
| (RPT N10,20) | (RPT N10,20) |
| M30 | M30 |

Function G72 is modal and is cancelled when another scaling factor with a value of S1 is programmed, or on power-up, after executing M02, M30 or after EMERGENCY or RESET.

### 7.6.2 Scaling factor applied to one or more axes.

The programming format is:

G72 X…C 5.5

After G72 the axis or axes and the required scaling factor are programmed.

All blocks programmed after G72 are treated by the CNC as follows :

1. The CNC calculates the movement of all the axes in relation to the programmed path and compensation.

2. It then applies the scaling factor indicated to the calculated movement of the corresponding axis or axes.

If the scaling factor is applied on one or more axes, the CNC will apply the scaling factor indicated both to the movement of the corresponding axis or axes and to their feedrate.

If, within the same program, both scaling factor types are applied, the one applied to all the axes and the one for one or several axes, the CNC applies a scaling factor equal to the product of the two scaling factors programmed for this axis to the axis or axes affected by both types.

Function G72 is modal and will be cancelled when the CNC is turned on, after executing M02, M30 or after an EMERGENCY or RESET.

*This type of scaling factor is ignored when simulating without moving the axes.*

Application of the scaling factor to a plane axis, working with tool radius compensation.
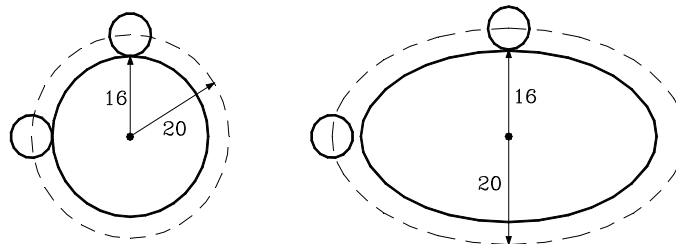


As it can be observed, the tool path does not coincide with the required path, as the scaling factor is applied to the calculated movement.

If a scaling factor equal to $360/2\pi R$ is applied to a rotary axis, R being the radius of the cylinder on which you wish to machine, this axis can be considered linear, and any figure with tool radius compensation can be programmed on the cylindrical surface.

7.

ADDITIONAL PREPARATORY FUNCTIONS
Scaling factor (G72)

FAGOR

FAGOR AUTOMATION

CNC 8055
CNC 8055i

·M· & ·EN· MODELS
SOFT: V02.2x

·124·
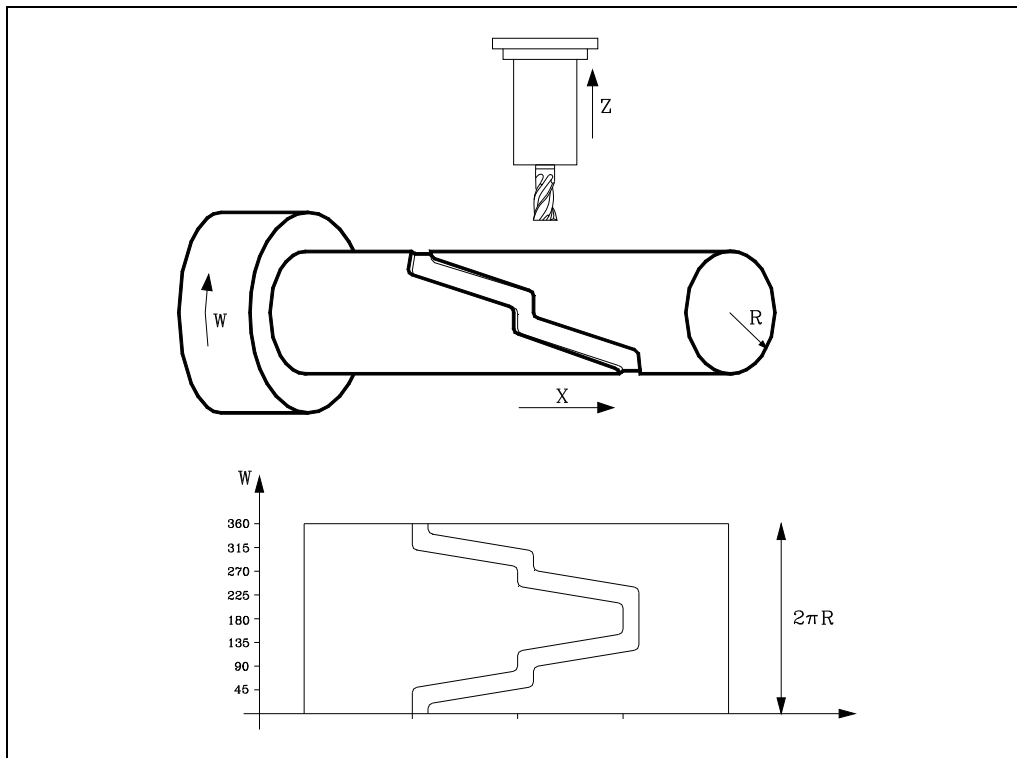
## 7.7    Pattern rotation (G73)

Function G73 enables you to turn the system of coordinates, taking either the coordinates origin or the programmed rotation center as the active rotation center.

The format which defines the rotation is the following :

G73 Q+/5.5 I±5.5 J±5.5

Where:

Q          Indicates the rotation angle in degrees.

I, J        The are optional and define the abscissa and ordinate respectively of the rotation center. If they are not defined, the coordinate origin will be taken as the rotation center.

The "I" and "J" values are defined in absolute coordinates and referred to the coordinate origin of the work plane. These coordinates are affected by the active scaling factor and mirror images.
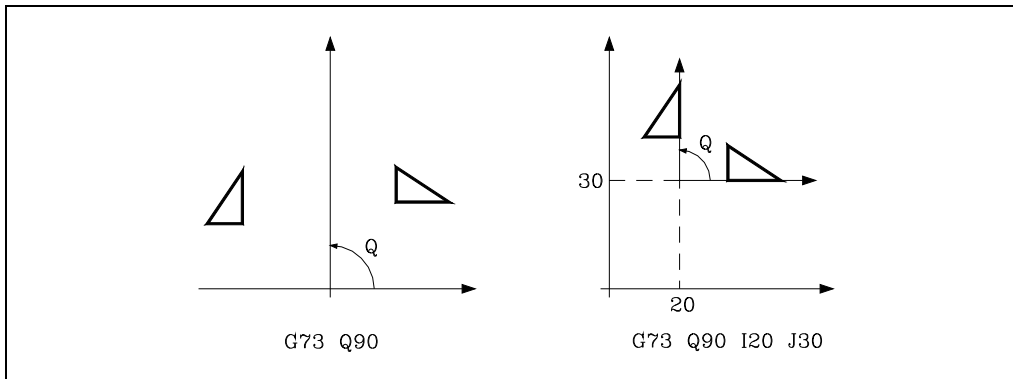


You should remember that G73 is incremental i.e. the different Q values programmed add up.
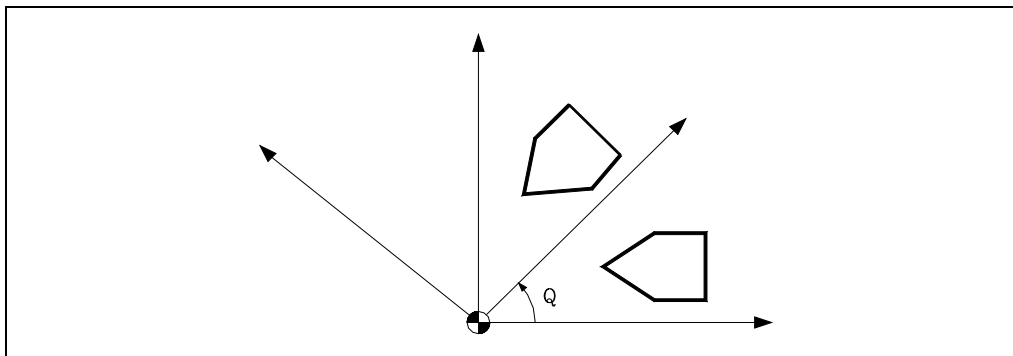


Function G73 should be programmed on its own in a block.

**FAGOR**

FAGOR AUTOMATION

**CNC 8055
CNC 8055i**

**7.**

**ADDITIONAL PREPARATORY FUNCTIONS**

Pattern rotation (G73)

Assuming that the starting point is X0 Y0, you get :
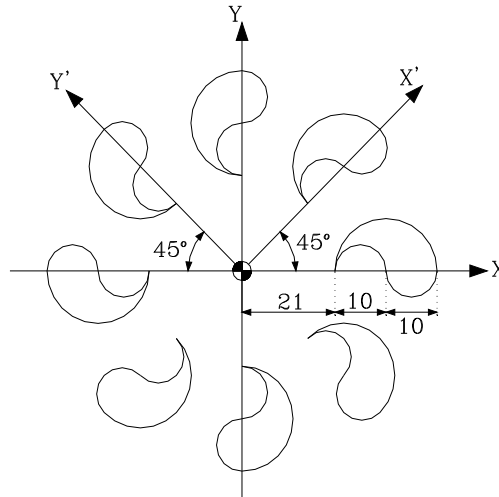


| | |
|---|---|
| N10 G01 X21 Y0 F300 | ; Positioning at starting point |
| G02 Q0 I5 J0 | |
| G03 Q0 I5 J0 | |
| Q180 I-10 J0 | |
| N20 G73 Q45 | ; Coordinate (pattern) rotation |
| (RPT N10, N20) N7 | ; Repeat from block 10 to 20 seven times |
| M30 | ; End of program |

In a program which rotates the coordinate system, if any mirror image function is also active the CNC first applies the mirror image function and then the turn.

The pattern rotation function can be cancelled either by programming G72 (on its own, without angle value) or via G16, G17, G18, or G19, or on power-up, after executing M02, M30 or after EMERGENCY or RESET.

**FAGOR**

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

## 7.8 Electronic axis coupling/uncoupling

The CNC enables two or more axes to be coupled together. The movement of all axes is subordinated to the movement of the axis to which they were coupled.

There are three possible ways of coupling axes :

- Mechanical coupling. This is imposed by the manufacturer of the machine, and is selected via the axis machine parameter "GANTRY".

- By PLC. This enables the coupling and uncoupling of each axis through logic input on the CNC "SYNCHRO1", "SYNCHRO2", "SYNCHRO3", "SYNCHRO4", and "SYNCHRO5". Each axis is coupled to the one indicated in the axis machine parameter "SYNCHRO".

- By program. This enables electronic coupling and uncoupling between two or more axes, through functions G77 and G78.

### 7.8.1   Electronic axis coupling, slaving, (G77)

Function G77 allows the selection of both the master axis and the slaved axis (axes). The programming format is as follows :

G77 <Axis 1> <Axis 2> <Axis 3> <Axis 4> <Axis 5>

Where <Axis 2>, <Axis 3>, <Axis 4> and <Axis 5> will indicate the axes to be coupled to <Axis 1>. You have to define <Axis 1> and <Axis 2>, the programming of the rest of the axes being optional.

**Example:**

G77 X Y U     ; Couples the Y and U axes to the X axis

The following rules should be observed when slaving axes electronically:

- You may use one or two different electronic couplings.

  G77 X Y U     ; Couples the Y and U axes to the X axis

  G77 V Z         ; Couples the Z axis to the V axis.

- You cannot couple one axis to two others at the same time.

  G77 V Y         ; Couples the Y axis to the V axis.

  G77 X Y         ; Gives an error signal, because the Y axis is coupled to the V axis.

- You can couple several axes to one in successive steps.

  G77 X Z         ; Couples the Z axis to the X axis.

  G77 X U         ; Couples the U axis to the X axis — Z U coupled to X.

  G77 X Y         ; Couples the Y axis to the X axis —Y Z U coupled to X.

- A pair of axes which are already coupled to each other cannot be coupled to another axis.

  G77 Y U         ; Couples the U axis to the Y axis.

  G77 X Y         ; Issues an error because Y axis is coupled to U axis.

### 7.8.2    Cancellation of the electronic axis coupling, slaving, (G78)

Function G78 enables you to uncouple all the axes that are coupled (slaved), or only uncouple indicated axes.

| | |
|---|---|
| G78 | Uncouples all slaved axes. |
| G78 <Axis1> <Axis2> <Axis3> <Axis4> | Only uncouples the indicated axes. |

**Example.**

| | |
|---|---|
| G77 X Y U | ; Couples the Y and U axes to the X axis |
| G77 V Z | ; Couples the Z axis to the V axis |
| G78 Y | ; Uncouples Y axis, but U stays coupled to X and Z to V. |
| G78 | ; Uncouples all axes |

**7.**

**ADDITIONAL PREPARATORY FUNCTIONS**

Electronic axis coupling/uncoupling

**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

·M· & ·EN· Models
Soft: V02.2x

## 7.9    Axes toggle G28-G29

With this feature, on machines having two machining tables, it is possible to use a single part-program to make the same parts on both tables.
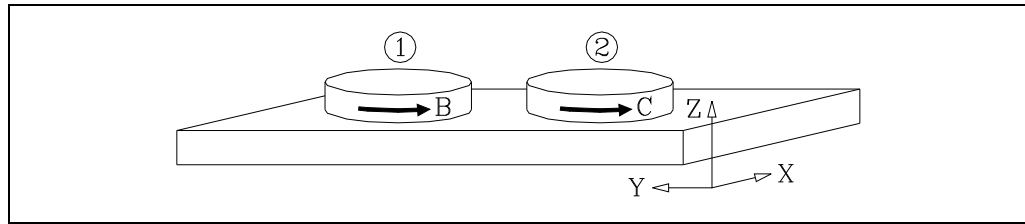


With function G28 the axes can be toggled from one to the other in such way that after that instruction all the movements associated with the first axis next to G28 will take place on the second axis next to G28 and vice versa.

Programming format:

    G28 (axis 1) (axis 2)

To cancel the toggle, execute function G29 followed by one of the axes to be toggled back. Up to three pairs of axes may be toggled at the same time.

The main axes cannot be toggled while the C axis is active on a lathe.

On power-up, after executing an M30 or after an emergency or reset, the axes are toggled back as long as G48 or G49 is not active.

Example. Let us suppose that the part program is defined for table 1.

1. Execute the part-program on table 1
2. G28 BC. Toggle the "B" and "C" axes
3. Zero offset for machining on table 2.
4. Execute the part-program.
    − It will be executed on table 2.
    − In the meantime, replace the part made on table 1 with a new one.
5. G29 B. Toggle the BC axes back.
6. Cancel zero offset to machine on table 1.
7. Execute the part-program.
    − It will be executed on table 1.
    − In the meantime, replace the part made on table 2 with a new one.

# TOOL COMPENSATION

# 8

The CNC has a tool offset table, its number of components being defined via the general machine parameter "NTOFFSET". The following is specified for each tool offset:

- Tool radius in work units in R±5.5 format.

- Tool length in work units in L±5.5 format.

- Wear of tool radius, in work units, in I±5.5 format. The CNC adds this value to the theoretical radius (R) to calculate the real radius (R+I).

- Wear of tool length, in work units, in K±5.5 format. The CNC adds this value to the theoretical length (L) to calculate the real length (L+K).

When tool radius compensation is required (G41 or G42), the CNC applies the sum of R+I values of the selected tool offset as the compensation value.

When tool length compensation is required (G43), the CNC applies the sum of L+K values of the selected tool offset as the compensation value.

## 8.1 Tool radius compensation (G40, G41, G42)

In normal milling operations, it is necessary to calculate and define the path of the tool taking its radius into account so that the required dimensions of the part are achieved.

Tool radius compensation allows the direct programming of part contouring and of the tool radius without taking the dimensions of the tool into account.

The CNC automatically calculates the path the tool should follow based on the contour of the part and the tool radius value stored in the tool offset table.

There are three preparatory functions for tool radius compensation:

G40: Cancellation of tool radius compensation

G41: Left-hand tool radius compensation.

G42: Tool radius compensation to the right of the part.
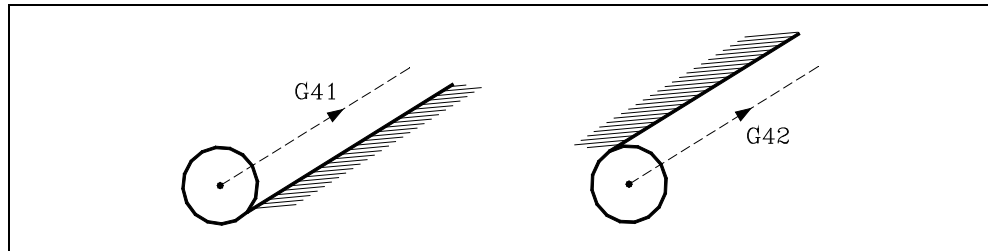


G41    The tool is to the left of the part, depending on the machining direction.

G42    The tool is to the right of the part, depending on the machining direction.

Tool values R, L, I, K should be stored in the tool offset table before starting machining, or should be loaded at the beginning of the program via assignments to variables TOR, TOL, TOI, TOK.

Once the plane in which compensation will be applied has been chosen via codes G16, G17, G18, or G19, this is put into effect by G41 or G42, assuming the value of the tool offset selected via code D, or (in its absence) by the tool offset shown in the tool table for the selected tool (T).

Functions G41 and G42 are modal and incompatible to each other. They are cancelled by G40, G04 (interruption of block preparation), G53 (programming with reference to machine zero), G74 (home search), machining canned cycles (G81, G82, G83, G84, G85, G86, G87, G88, G89) and also on power-up, after executing M02, M30 or after EMERGENCY or RESET.

**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2x

·132·

### 8.1.1 Beginning of tool radius compensation

Once the plane in which tool radius compensation has been selected (via G16, G17, G18, or G19), functions G41 or G42 must be used to activate it.

G41: Left-hand tool radius compensation.

G42: Tool radius compensation to the right of the part.

In the same block (or a previous one) in which G41 or G42 is programmed, functions T, D, or only T must be programmed so that the tool offset value to be applied can be selected from the tool offset table. If no tool offset is selected, the CNC takes D0 with R0 L0 I0 K0.

When the new selected tool has function M06 associated with it and this, in turn, has an associated subroutine, the CNC will first process the motion block of that subroutine as the starting block of the compensation.

If in that subroutine, a block is executed that contains function G53 (programming in machine coordinates), the previously programmed G41 or G42 is canceled.

The selection of tool radius compensation (G41 or G42) can only be made when functions G00 or G01 are active (straight-line movements).

If the compensation is selected while G02 or G03 are active, the CNC will display the corresponding error message.

The following pages show different cases of starting tool radius compensation, in which the programmed path is represented by a solid line and the compensated path with a dotted line.

#### *Beginning of the compensation without programmed movement*

After activating the compensation, it could happen that the plane axes do not get involved in the first motion block either because they have not been programmed or because the same point as the tool position has been programmed or because a null incremental move has been programmed.

In this case, the compensation is applied in the current tool position; depending on the first movement programmed in the plane, the tool moves perpendicular to the path on its starting point.

The first movement programmed in the plane may be either linear or circular.

# STRAIGHT-STRAIGHT path

COMPTYPE=0          COMPTYPE=1

## STRAIGHT-CURVED path



COMPTYPE=0       COMPTYPE=1

## 8.1.2 Sections of tool radius compensation

The CNC reads up to 20 blocks ahead of the one it is executing, with the aim of calculating beforehand the path to be followed. When working with tool radius compensation, the CNC needs to know the next programmed movement to calculate the path to follow; therefore, no more than 17 blocks can be programmed in a row.

The diagrams (below) show the different paths followed by a tool controlled by a programmed CNC with tool radius compensation. The programmed path is shown with solid line and the compensated path with dashed line.
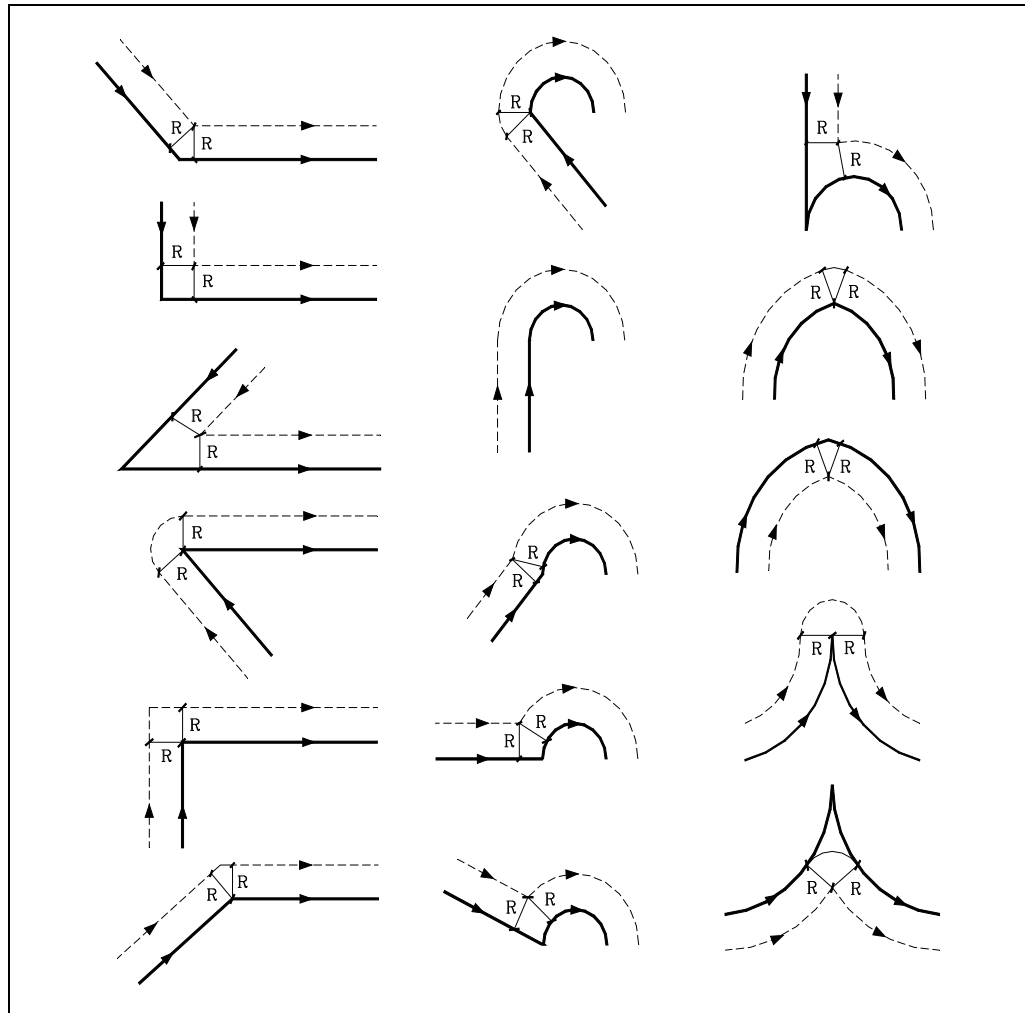


The way the various paths are blended (joined) depends on the setting of machine parameter COMPMODE.

- If it is set to ·0·, the compensation method depends on the angle between paths.

  With an angle between paths of up to 300º, both paths are joined with straight sections. In the rest of the cases, both paths are joined with arcs.

- If it is set to ·1·, both paths are joined with arcs.

- If it is set to ·2·, the compensation method depends on the angle between paths.

  With an angle between paths of up 300º, it calculates the intersection. In the rest of the cases it is compensated like COMPMODE = 0.

### 8.1.3    Cancellation of tool radius compensation

Tool radius compensation is canceled by using function G40.

It should be remembered that canceling radius compensation (G40) can only be done in a block in which a straight-line movement is programmed (G00 or G01).

If G40 is programmed while functions G02 or G03 are active, the CNC displays the corresponding error message.
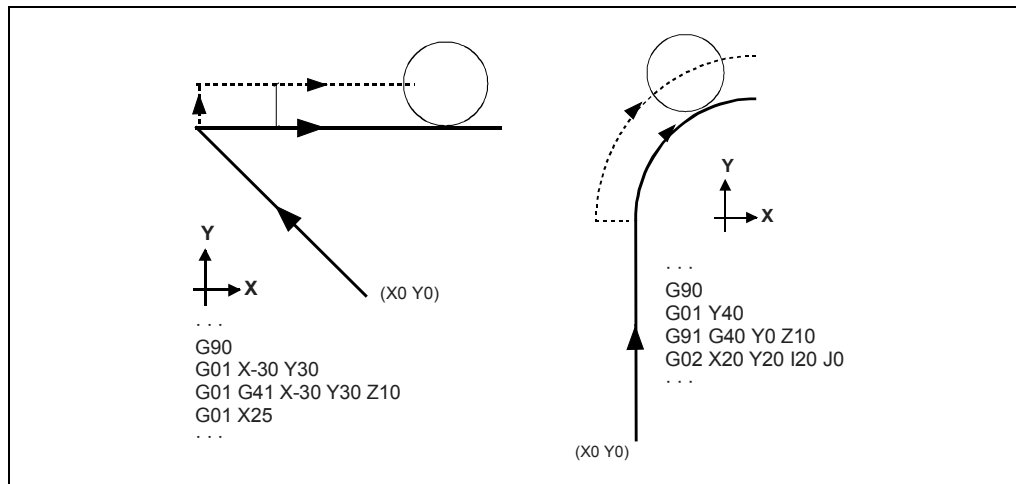
The following pages show different cases of canceling tool radius compensation, in which the programmed path is represented by a solid line and the compensated path with a dotted line.

#### *End of the compensation without programmed movement:*

After cancelling the compensation, it could happen that the plane axes do not get involved in the first motion block either because they have not been programmed or because the same point as the tool position has been programmed or because a null incremental move has been programmed.

In this case, the compensation is canceled in the current tool position; depending on the last movement executed in the plane, the tool moves to the end point without compensating the programmed path.



```
                              (X0 Y0)




Y
  X
. . .
G90
G01 X-30
G01 G40 X-30
G01 X25 Y-25
. . .
```

```
                    (X0 Y0)

              Y
                X
. . .
G90
G03 X-20 Y-20 I0 J-20
G91 G40 Y0
G01 X-20
. . .
```

**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

## STRAIGHT-STRAIGHT path

COMPTYPE=0          COMPTYPE=1

## CURVED-STRAIGHT path



COMPTYPE=0 COMPTYPE=1

### *Example of machining with radius compensation:*



The programmed path is shown with solid line and the compensated path with dashed line.

| Tool radius | 10mm |
| --- | --- |
| Tool number | T1 |
| Tool offset number | D1 |

```
; Preset
G92 X0 Y0 Z0
; Tool, offset and spindle start at S100
G90 G17 S100 T1 D1 M03
; Begins compensation
G41 G01 X40 Y30 F125Y70
X90
Y30
X40
; Cancels compensation
G40 G00 X0 Y0
M30
```

**Example of machining with radius compensation:**



The programmed path is shown with solid line and the compensated path with dashed line.

| | |
|---|---|
| Tool radius | 10mm |
| Tool number | T1 |
| Tool offset number | D1 |

```
; Preset
G92 X0 Y0 Z0
; Tool, offset and spindle start at S100
G90 G17 F150 S100 T1 D1 M03
; Begins compensation
G42 G01 X30 Y30
X50
Y60
X80
X100 Y40
X140
X120 Y70
X30
Y30
; Cancels compensation
G40 G00 X0 Y0
M30
```

**Example of machining with radius compensation:**



The programmed path is shown with solid line and the compensated path with dashed line.

| | |
|---|---|
| Tool radius | 10mm |
| Tool number | T1 |
| Tool offset number | D1 |
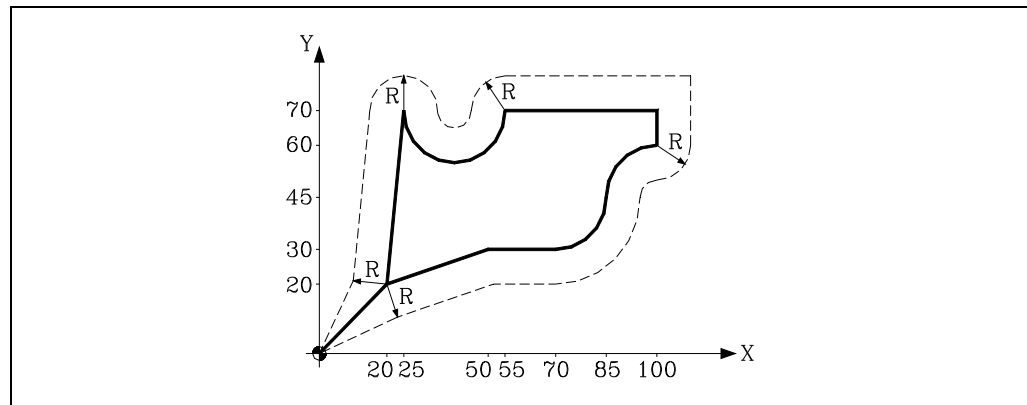
```
; Preset
G92 X0 Y0 Z0
; Tool, offset and spindle start at S100
G90 G17 F150 S100 T1 D1 M03
; Begins compensation
G42 G01 X20 Y20
X50 Y30
X70
G03 X85Y45 I0 J15
G02 X100 Y60 I15 J0
G01 Y70
X55
G02 X25 Y70 I-15 J0
G01 X20 Y20
; Cancels compensation
G40 G00 X0 Y0 M5
M30
```

### 8.1.4  Change of type of radius compensation while machining

The compensation may be changed from G41 to G42 or vice versa without having to cancel it with G40. It may be changed in any motion block and even in a motionless one, i.e. without moving the axes of the plane or by programming the same point twice.

It compensates independently the last movement before the change and the first one after the change. To change the type of compensation, the different cases are solved according to the following criteria:

**A.** The compensated paths cut each other.

The programmed paths are compensated each on its corresponding side. The side change takes place in the intersection point between both paths.

**B.** The compensated paths do not cut each other.

An additional section is inserted between the two paths. From the point perpendicular to the first path at the end point up to the point perpendicular to the second path at the starting point. Both points are located at a distance R from the programmed path.

Here is a summary of the different cases:
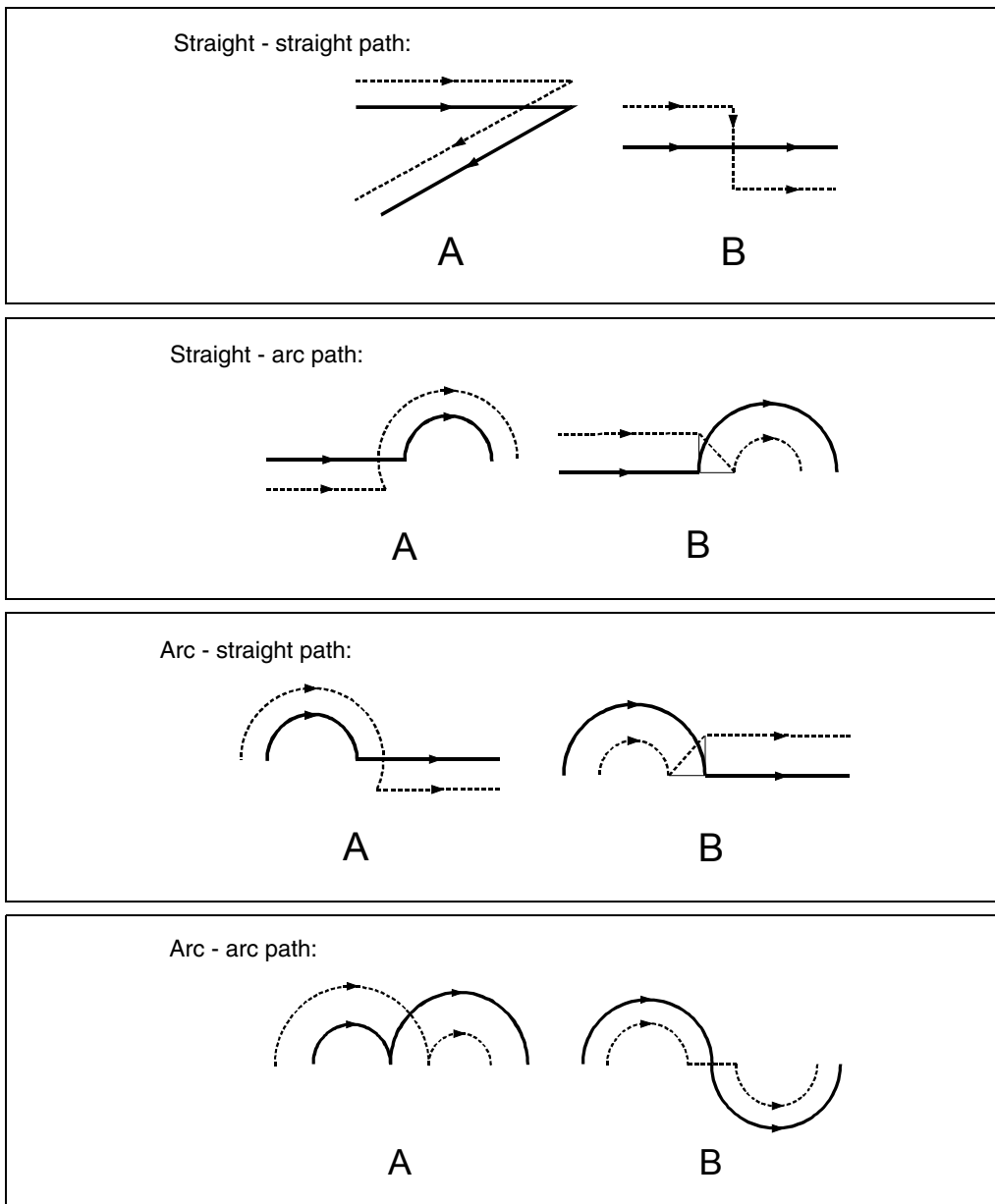
Straight - straight path:



A       B

Straight - arc path:



A       B

Arc - straight path:



A       B

Arc - arc path:



A       B

**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2x

## 8.2    Tool length compensation (G43, G44, G15)

With this function it is possible to compensate possible differences in length between the programmed tool and the tool being used.

The tool length compensation is applied on to the axis indicated by function G15 or, in its absence, to the axis perpendicular to the main plane.

If G17, tool length compensation on the Z axis.

If G18, tool length compensation on the Y axis

If G19, tool length compensation on the X axis.

Whenever one of functions G17, G18 or G19 is programmed, the CNC assumes as new longitudinal axis (upon which tool length compensation will be applied) the one perpendicular to the selected plane.

On the other hand, if function G15 is executed while functions G17, G18 or G19 are active, the new longitudinal axis (selected with G15) will replace the previous one.

The function codes used in length compensation are as follows:

G43: Tool length compensation.

G44: Cancellation of tool length compensation.

Function G43 only indicates that a longitudinal compensation is to be applied. The CNC starts applying it when the longitudinal (perpendicular) axis starts moving.

```
; Preset
G92 X0 Y0 Z50
 ; Tool, offset ...
G90 G17 F150 S100 T1 D1 M03
; Selects compensation
G43 G01 X20 Y20
X70
; Begins compensation
Z30
```

When G43 is programmed, the CNC compensates the length in accordance with the value of the tool offset selected with code D, or (in its absence) the tool offset shown in the tool table for the selected tool (T).

Tool values R, L, I, K should be stored in the tool offset table before starting machining, or should be loaded at the beginning of the program via assignments to variables TOR, TOL, TOI, TOK.

If no tool offset is selected, the CNC takes D0 with R0 L0 I0 K0.

Function G43 is modal and can be canceled via G44 and G74 (home search). If general machine parameter "ILCOMP=0", it is also canceled on power-up, after executing M02, M30 or after EMERGENCY or RESET.

G53 (programming with respect to machine zero) temporarily cancels G43 only while executing a block which contains a G53.

Length compensation can be used together with canned cycles, although here care should be taken to apply this compensation before starting the cycle.

FAGOR

FAGOR AUTOMATION

CNC 8055
CNC 8055i

·M· & ·EN· MODELS
SOFT: V02.2x

·144·

### *Example of machining with tool length compensation:*
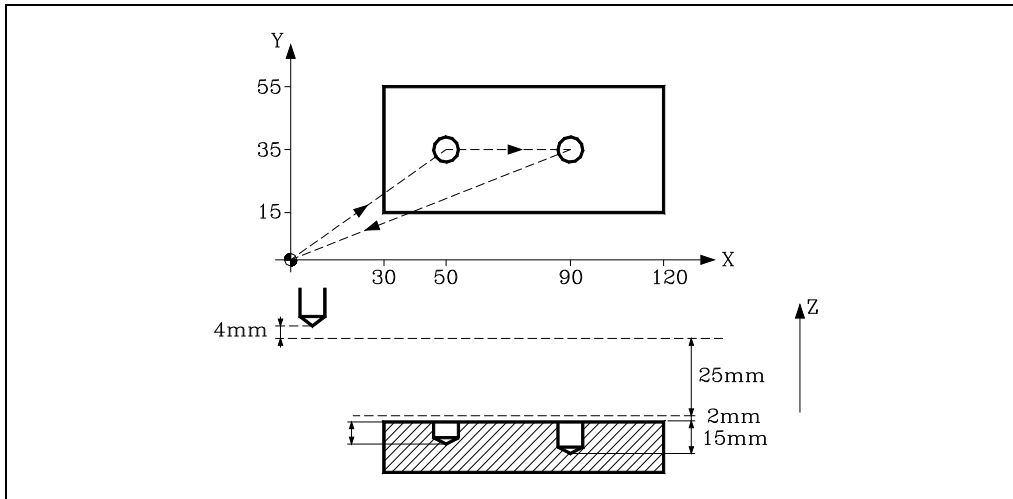


It is assumed that the tool used is 4 mm shorter than the programmed one.

| | |
|---|---|
| Tool length | -4mm |
| Tool number | T1 |
| Tool offset number | D1 |

```
; Preset
G92 X0 Y0 Z0
; Tool, offset ...
G91 G00 G05 X50 Y35 S500 M03
; Begins compensation
G43 Z-25 T1 D1
G01 G07 Z-12 F100
G00 Z12
X40
G01 Z-17
; Cancels compensation
G00 G05 G44 Z42 M5
G90 G07 X0 Y0
M30
```

**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2x

## 8.3     Collision detection (G41 N, G42 N)

Using this option, the CNC analyzes in advance the blocks to be executed in order to detect loops (profile intersections with itself) or collisions of the programmed profile. The number of blocks to be analyzed (up to 50) may be defined by the user.

The example shows machining errors (E) due to a collision in the programmed profile. This type of errors may be avoided using collision detection.
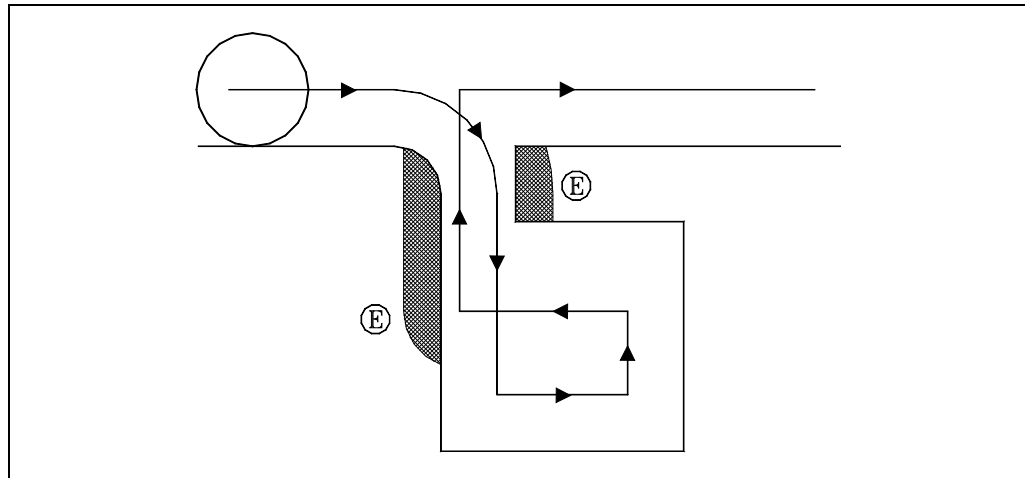


When detecting a loop or a collision, the blocks that caused it will not be executed and a warning will be issued for each loop or collision eliminated.

Possible cases: step on a straight path, a step in a circular path and tool radius compensation too large.

The information contained in the eliminated blocks, not being the moving in the active plane, will be executed (including the movements of the axes).

Block detection is defined and activated with tool radius compensation functions G41 and G42. A new parameter N (G41 N y G42 N) has been added to activate the feature and define the number of blocks to analyze.

Possible values from N3 to N50. Without "N", or with N0, N1 and N2 , it behaves like in older versions.

In CAD generated programs that are made up of lots of very short blocks, it is recommended to use very low N values (around 5) so as not to jeopardize block processing time.

When this function is active, the history of active G functions shows G41 N or G42 N.

FAGOR

FAGOR AUTOMATION

CNC 8055
CNC 8055i

·M· & ·EN· MODELS
SOFT: V02.2x

·146·

# CANNED CYCLES

These canned cycles can be performed on any plane, the depth being along the axis selected as longitudinal via function G15 or, in its absence, along the axis perpendicular to this plane.

The CNC offers the following machining canned cycles :

| | |
|---|---|
| G69 | Complex deep hole drilling |
| G81 | Drilling canned cycle. |
| G82 | Drilling cycle with dwell. |
| G83 | Deep hole drilling canned cycle with constant peck (drilling step). |
| G84 | Tapping canned cycle. |
| G85 | Reaming canned cycle. |
| G86 | Boring cycle with withdrawal in G00 |
| G87 | Rectangular pocket canned cycle. |
| G88 | Circular pocket canned cycle. |
| G89 | Boring cycle with withdrawal in G01 |
| G210 | Bore milling canned cycle |
| G211 | Inside thread milling canned cycle. |
| G212 | Outside thread milling canned cycle. |

It also offers the following functions that can be used with the machining canned cycles:

| | |
|---|---|
| G79 | Change of canned cycle parameters. |
| G98 | Return to the starting plane at the end of the canned cycle |
| G99 | Return to the reference plane at the end of the canned cycle. |

## 9.1    Canned cycle definition

A canned cycle is defined by the G function indicating the canned cycle and its corresponding parameters.

A canned cycle cannot be defined in a block which has nonlinear movements (G02, G03, G08, G09, G33 or G34).

Also, a canned cycle cannot be executed while function G02, G03, G33 or G34 is active. The CNC will issue the corresponding error message.

However, once a canned cycle has been defined in a block and following blocks, functions G02, G03, G08 or G09 can be programmed.

## 9.2 Influence zone of a canned cycle

Once a canned cycle has been defined it remains active, and all blocks programmed after this block are under its influence while it is not cancelled.

In other words, every time a block is executed in which some axis movement has been programmed, the CNC will carry out (following the programmed movement) the machining operation which corresponds to the active canned cycle.

If, in a movement block within the area of influence of a canned cycle, the number of times a block is executed (repetitions) "N" is programmed at the end of the block, the CNC repeats the programmed positioning and the machining operation corresponding to the canned cycle the indicated number of times.

If a number of repetitions (times) "N0" is programmed, the machining operation corresponding to the canned cycle will not be performed. The CNC will only carry out the programmed movement.

If, within the area of influence of a canned cycle, there is a block which does not contain any movement, the machining operation corresponding to the defined canned cycle will not be performed, except in the calling block.

| | |
|---|---|
| G81... | Definition and execution of the canned cycle (drilling). |
| G90 G1 X100 | The X axis moves to X100, where the hole is to be drilled. |
| G91 X10 N3 | The CNC runs the following operation 3 times. |
| | • Incremental move to X10. |
| | • Runs the cycle defined above. |
| G91 X20 N0 | Incremental move only to X20 (no drilling). |

### 9.2.1    G79. Modification of the canned cycle parameters

The CNC allows one or several parameters of an active canned cycle to be modified by programming the G79 function, without any need for redefining the canned cycle.

The CNC will continue to maintain the canned cycle active and will perform the following machinings of the canned cycle with the updated parameters.

The G79 function must be programmed alone in a block, and this block must not contain any more information.

Next 2 programming examples are shown assuming that the work plane is formed by the X and Y axes, and that the longitudinal axis (perpendicular) is the Z axis:
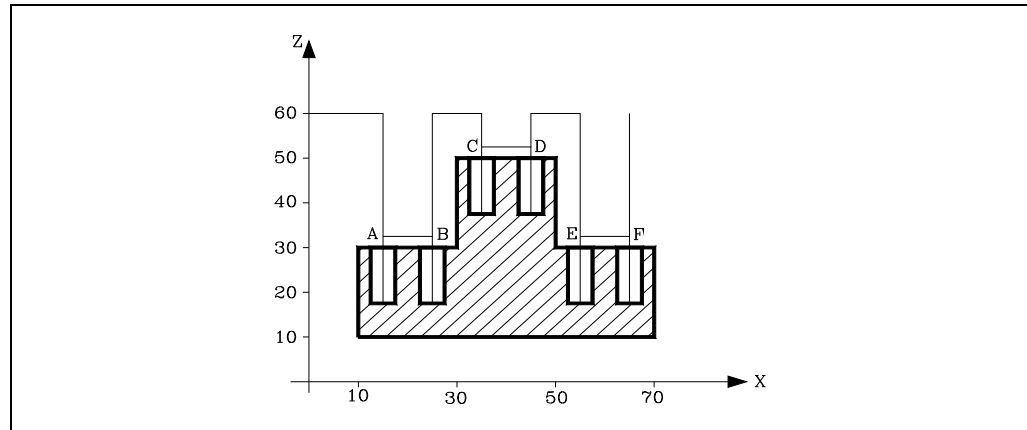


```
T1
M6
; Starting point.
G00 G90 X0 Y0 Z60
; Defines drilling cycle. Drills in A.
G81 G99 G91 X15 Y25 Z-28 I-14
; Drills in B.
G98 G90 X25
; Modifies reference plane and machining depth.
G79 Z52
; Drills in C.
G99 X35
; Drills in D.
G98 X45
; Modifies reference plane and machining depth.
G79 Z32
; Drills in E.
G99 X55
; Drills in F.
G98 X65
M30
```
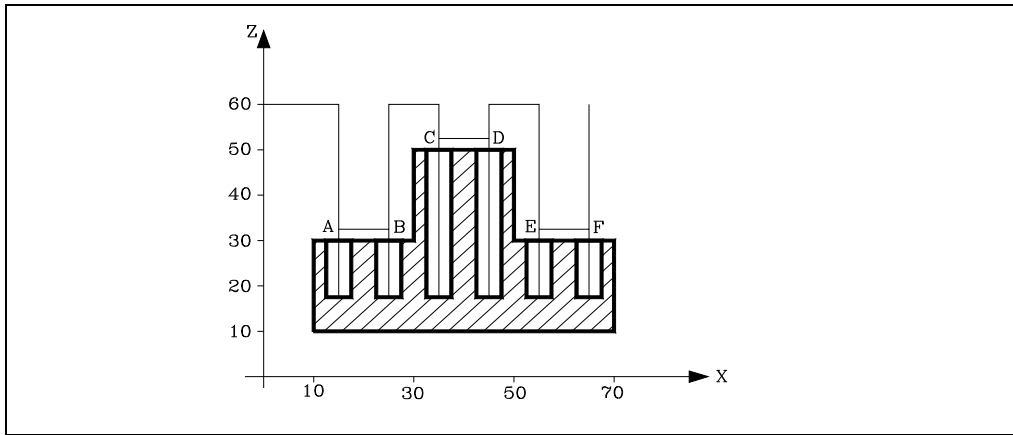
```
T1
M6
; Starting point.
G00 G90 X0 Y0 Z60
; Defines drilling cycle. Drills in A.
G81 G99 X15 Y25 Z32 I18
; Drills in B.
G98 X25
; Modifies reference plane.
G79 Z52
; Drills in C.
G99 X35
; Drills in D.
G98 X45
; Modifies reference plane.
G79 Z32
; Drills in E.
G99 X55
; Drills in F.
G98 X65
M30
```

## 9.3    Canned cycle cancellation

A canned cycle can be canceled via :

- Function G80, which can be programmed in any block.
- After defining a new canned cycle. This will cancel and replace any other that may be active.
- After executing M02, M30, or after EMERGENCY or RESET.
- When searching home with function G74.
- Selecting a new work plane via functions G16, G17, G18, or G19.

**9.**

**CANNED CYCLES**
Canned cycle cancellation

**FAGOR**

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

·152·

## 9.4    Some general points to consider

- A canned cycle may be defined anywhere in the program, that is, in the main program as well as in a subroutine.

- Calls to subroutines can be made from a block within the influence of a canned cycle without implying the cancellation of the canned cycle.

- The execution of a canned cycle will not alter the history of previous "G" functions.

- Nor will the spindle turning direction be altered. A canned cycle can be entered with any turning direction (M03 or M04), leaving in the same direction in which the cycle was entered.

  Should a canned cycle be entered with the spindle stopped, it will start in a clockwise direction (M03), and maintain the same turning direction until the cycle is completed.

- Should it be required to apply a scaling factor when working with canned cycles, it is advisable that this scale factor be common to all the axes involved.

- The execution of a canned cycle  cancels radius compensation (G41 and G42). It is equivalent to G40.

- If tool length compensation (G43) is to be used, this function must be programmed in the same block or in the one before the definition of the canned cycle.

  The CNC applies the tool length compensation when the longitudinal (perpendicular) axis starts moving. Therefore, it is recommended to position the tool outside the canned cycle area when defining function G43 for the canned cycle.

- The execution of any canned cycle will alter the global parameter P299.

**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

## 9.5 Machining canned cycles

In all machining cycles there are three coordinates along the longitudinal axis to the work plane which, due to their importance, are discussed below:

- Initial plane coordinate. This coordinate is given by the position which the tool occupies with respect to machine zero when the cycle is activated.

- Coordinate of the reference plane. This is programmed in the cycle definition block and represents an approach coordinate to the part. It can be programmed in absolute coordinates or in incremental, in which case it will be referred to the initial plane.

- Machining depth coordinate. This is programmed in the cycle definition block. It can be programmed in absolute coordinates or in incremental coordinates, in which case it will be referred to the reference plane.

There are two functions which allow to select the type of withdrawal of the longitudinal axis after machining.

- G98: Selects the withdrawal of the tool as far as the initial plane, once the indicated machining has been done.

- G99: Selects the withdrawal of the tool as far as the reference plane, once the indicated machining has been done.

These functions can be used both in the cycle definition block and the blocks which are under the influence of the canned cycle. The initial plane will always be the coordinate which the longitudinal axis had when the cycle was defined.

The structure of a canned cycle definition block is as follows:

| G** | Machining point | Parameters | F S T D M | N**** |
|-----|-----------------|------------|-----------|-------|

It is possible to program the starting point in the canned cycle definition block (except the longitudinal axis), both in polar coordinates and in Cartesian coordinates.

After defining the point at which it is required to carry out the canned cycle (optional), the functions and parameters corresponding to the canned cycle will be defined, and afterwards, if required, the complementary functions F S T D M are programmed.

When programming, at the end of the block, the number of times a block is to be executed "N", the CNC performs the programmed move and the machining operation corresponding to the active canned cycle the indicated number of times.

If "N0" is programmed, it will not execute the machining operation corresponding to the canned cycle. The CNC will only carry out the programmed movement.

The general operation for all the cycles is as follows:

1. If the spindle was previously running, it maintains the turning direction. If it was not in movement, it will start by turning clockwise (M03).

2. Positioning (if programmed) at the starting point for the programmed cycle.

3. Rapid movement of the longitudinal axis from the initial plane to the reference plane.

4. Execution of the programmed machining cycle.

5. Rapid withdrawal of the longitudinal axis to the initial plane or reference plane, depending on whether G98 or G99 has been programmed.

    The explanation of each cycle assumes that the work plane is formed by the X and Y axes, and that the longitudinal axis (perpendicular) is the Z axis:

## Programming in other planes

The programming format is always the same, it does not depend on the work plane. Parameters XY indicate the coordinate in the work plane (X = abscissa, Y = ordinate) and the penetration along the longitudinal axis.

The following examples show how to drill in X and Y in both directions.

Function G81 defines the drilling canned cycle. It is defined with these parameters:

X     Coordinate of the point to be machined along the abscissa axis.

Y     Coordinate of the point to be machined along the ordinate axis.

I     Drilling depth.

K     Dwell at the bottom.

In the following examples, the part surface has a 0 coordinate, the holes are 8 mm deep and the reference coordinate is 2 mm above the surface.

### *Example 1:*
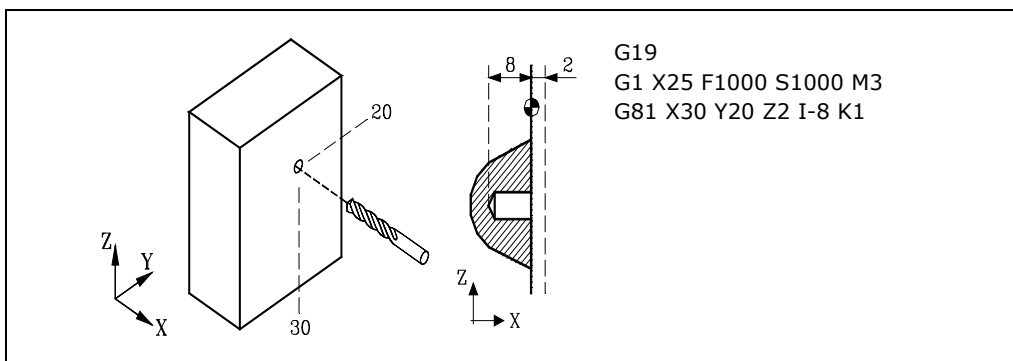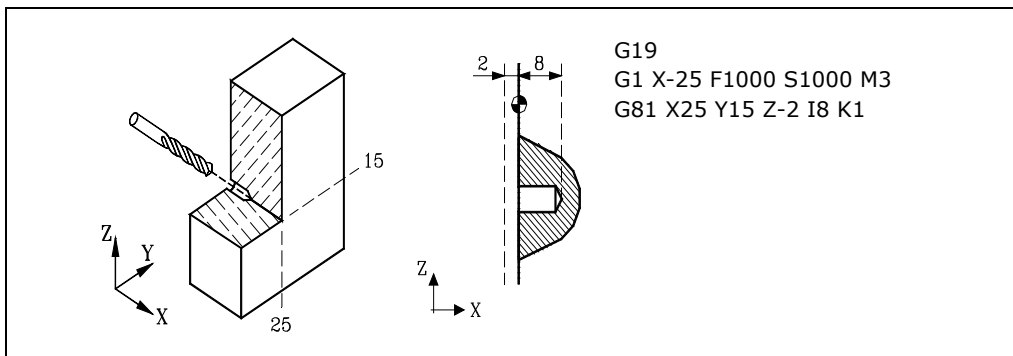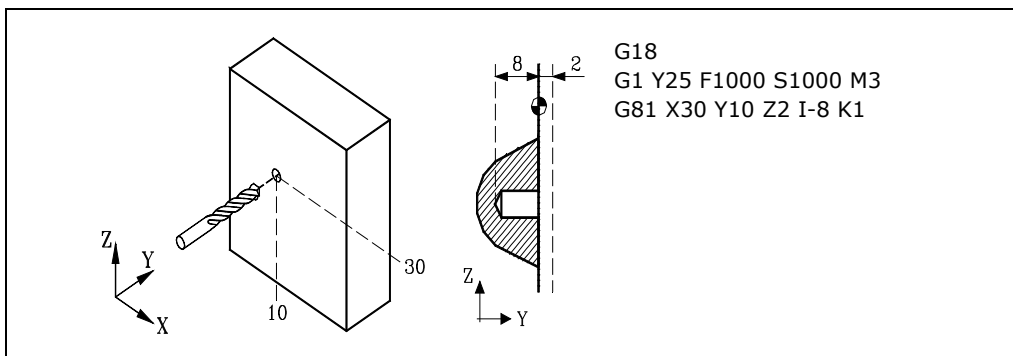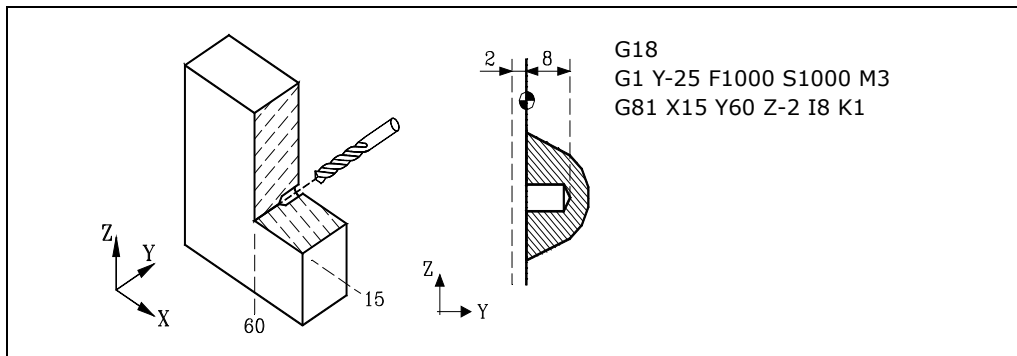


```
G19
G1 X25 F1000 S1000 M3
G81 X30 Y20 Z2 I-8 K1
```

### *Example 2:*



```
G19
G1 X-25 F1000 S1000 M3
G81 X25 Y15 Z-2 I8 K1
```

### *Example 3:*



```
G18
G1 Y25 F1000 S1000 M3
G81 X30 Y10 Z2 I-8 K1
```

*Example 4:*



```
G18
G1 Y-25 F1000 S1000 M3
G81 X15 Y60 Z-2 I8 K1
```

**FAGOR**

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

·**156**·

## 9.6    G69. Drilling canned cycle with variable peck

This cycle makes successive drilling steps until the final coordinate is reached. The tool withdraws a fixed amount after each drilling operation, it being possible to select that every J drillings it withdraws to the reference plane. A dwell can also be programmed after every drilling.

Working in Cartesian coordinates, the basic structure of the block is as follows:

G69 G98/G99 X Y Z I B C D H J K L R



### [ G98/G99 ] Withdrawal plane

G98    The tool withdraws to the Initial Plane, once the hole has been drilled.

G99    The tool withdraws to the Reference Plane, once the hole has been drilled.

### [ X/Y±5.5 ] Machining coordinates

These are optional and define the movement of the axes of the main plane to position the tool at the machining point.

This point can be programmed in Cartesian coordinates or in polar coordinates, and the coordinates may be absolute or incremental, according to whether the machine is operating in G90 or G91.

### [ Z±5.5 ] Reference plane

Defines the reference plane coordinate. It can be programmed in absolute coordinates or incremental coordinates, in which case it will be referred to the initial plane.

If not programmed, it assumes as reference plane the current position of the tool.

### [ I±5.5 ] Drilling depth.

Defines the total drilling depth. It can be programmed in absolute coordinates or incremental coordinates and in this case will be referred to the part surface.

### [ B5.5 ] Drilling peck (step)

Defines the drilling step in the axis longitudinal to the main plane.

### [ C5.5 ] Approach to the previous drilling

Defines the approach distance the longitudinal axis will move in rapid (G00) from the previous drilling step to start the next drilling step.

If not programmed, a value of 1 mm is assumed. If programmed with a 0 value, the CNC will display the corresponding error message.

### [ D5.5 ] Reference plane

Defines the distance between the reference plane and the surface of the part where the drilling is to be done.

In the first drilling, this amount will be added to "B" drilling step. If not programmed, a value of 0 is assumed.

**FAGOR**

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2x

### [ H±5.5 ] Withdrawal after drilling

Distance or coordinate the longitudinal axis returns to, in rapid (G00), after each drilling step.

"J" other than 0 means the distance and "J=0" indicates the relief coordinate or absolute coordinate it withdraws to.

If not programmed, the longitudinal axis returns to the reference plane.

### [ J4 ] Drilling passes to withdraw to the starting plane

Defines after how many drilling pecks the tool returns to the reference plane in G00. A value between 0 and 9999 may be programmed.

When not programmed or programmed with a "0" value, it returns to the position indicated by H (relief position) after each drilling peck.
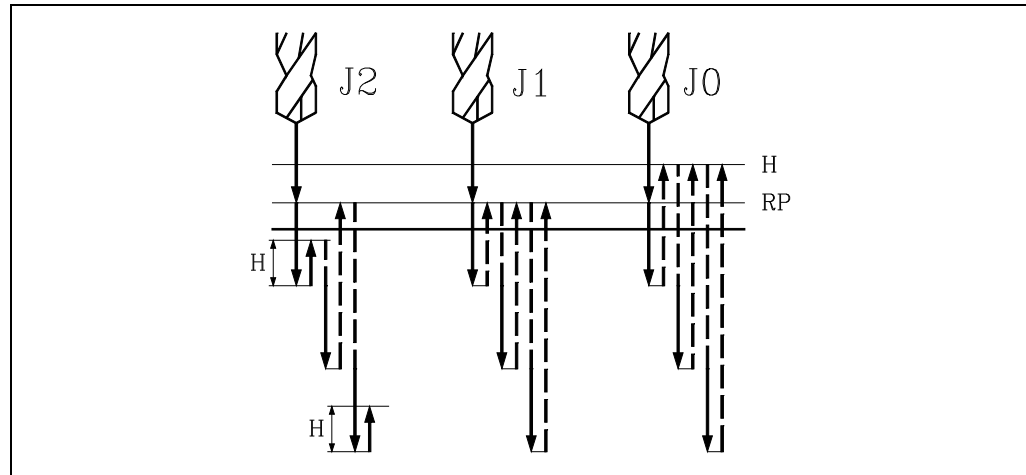


- With "J" greater than 1, after each peck, the tool returns the distance indicated by "H" and every "J" pecks to the reference plane (RP).
- With J1, it will return to the reference plane (RP) after each peck .
- With J0, it will return to the relief position indicated by H.

### [ K5 ] Dwell

Defines the dwell, in hundredths of a second, after each drilling peck until the withdrawal begins. If not programmed, the CNC will take the value of "K0".

### [ L5.5 ] Minimum drilling peck (step)

It defines the minimum value for the drilling peck. This paramater is used with "R" values other than 1. If not programmed or programmed with a 0 value, it assumes 1 mm.

### [ R5.5 ] Reduction factor for the drilling pecks.

Factor that reduces the drilling peck (step) "B". If it is not programmed or is programmed with a value of "0", it assumes a value of "1".

If R =1, all the drilling pecks will be the same (the programmed "B" value).

If R is other than 1, the first drilling peck will be "B", the second one "R B", The third one "R (RB)" and so on. In other words, from the second peck on, the new peck will be the product of the R factor times the previous peck.

If R is selected with a value other than 1, the CNC will not allow smaller steps than that programmed in L.

### 9.6.1 Basic operation

1. If the spindle was previously running, it maintains the turning direction. If it was not in movement, it will start by turning clockwise (M03).

2. Rapid movement of the longitudinal axis from the initial plane to the reference plane.
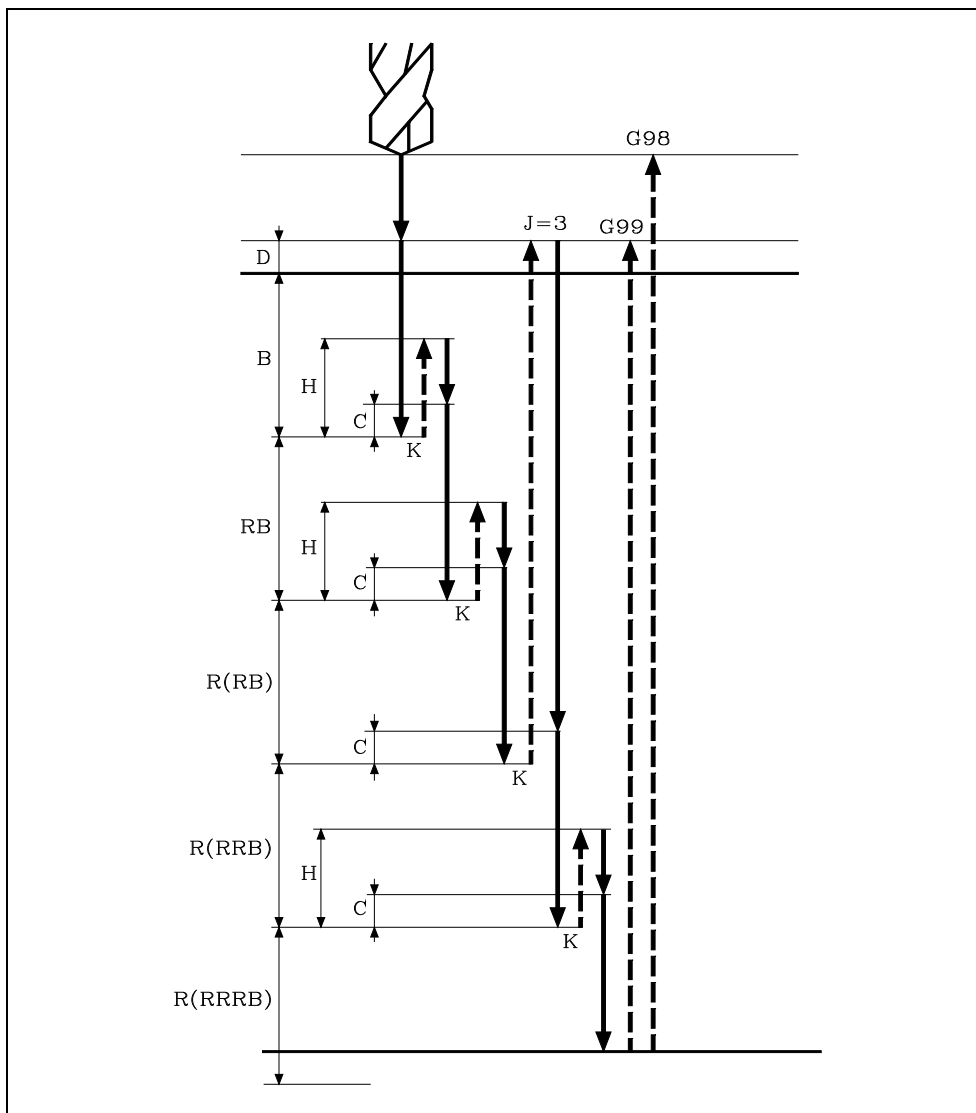


3. First drilling operation. The drilling axis moves in G01 to the programmed incremental depth "B + D".

4. Drilling loop. The following steps will be repeated until the programmed machining depth "I" is reached.

    ·1· Dwell K, in hundredths of a second, if it has been programmed.

    ·2· Withdrawal of the longitudinal axis in rapid (G00) as far as the reference plane, if the number of drillings programmed in J were made, otherwise it withdraws the distance programmed in "H".

    ·3· Rapid approach (G00) of the longitudinal axis to a "C" distance from the next peck.

    ·4· New drilling step. G1 move of the longitudinal axis to the next incremental depth "B and R".

    This movement will be carried out either in G07 or G50 depending on the value assigned to the longitudinal axis "INPOSW2(P51)"

    If P51=0, in G7 (square corner). If P51=1, in G50 (controlled round corner).

5. Dwell K, in hundredths of a second, if it has been programmed.

6. Withdrawal at rapid feedrate (G00) of the longitudinal axis to the initial or reference plane, depending on whether G98 or G99 has been programmed.

The first drilling penetration is done in G07 or G50 depending on the value assigned to the parameter of the longitudinal axis "INPOSW2 (P51)" and to parameter "INPOSW1 (P19)". This is important to join a drilling operation with another one in multiple drilling so the tool path is faster and smoother.

If INPOSW2 < INPOSW1 in G07 (square corner).

If INPOSW2 >= INPOSW1 in G50 (controlled round corner).

If a scaling factor is applied to this cycle, it should be borne in mind that this scaling factor will only affect the reference plane coordinates and drilling depth.

Therefore, and due to the fact that parameter "D" is not affected by the scaling factor, the surface coordinate of the part will not be proportional to the programmed cycle.

> Programming example assuming that the work plane is formed by the X and Y axes, that the Z axis is the longitudinal axis and that the starting point is X0 Y0 Z0:
>
> ; Tool selection.
> T1
> M6
> ; Starting point.
> G0 G90 X0 Y0 Z0
> ; Canned cycle definition.
> G69 G98 G91 X100 Y25 Z-98 I-52 B12 C2 D2 H5 J2 K150 L3 R0.8 F100 S500 M8
> ; Cancels the canned cycle.
> G80
> ; Positioning.
> G90 X0 Y0
> ; End of program.
> M30

## Tool withdrawal

While machining, the CNC lets withdraw the tool to the starting plane stopping the spindle when the tool reaches the starting plane.

Activating PLC mark RETRACYC (M5065) stops the main axis and the tool is withdrawn without stopping the spindle. The spindle stops when the tool is retracted, once it reaches the starting plane.

## Options after tool withdrawal

Once the tool has been retracted, the user will have the following options:

- Finish the hole.
- Go to the next hole.
- Go into tool inspection.

After this, the CNC will display the following message:

"To end the cycle, press START to skip to the next SKIPCYCL".

### *Finish the hole:*

To finish the hole, press the [START] key.

It goes down in G0 with spindle running to 1 mm before the coordinate where the hole stopped. From then on, it continues at the F and S programmed in the cycle.

### *Go to the next hole:*

To go to the next hole, activate the PLC mark SKIPCYCL.

The CNC will display the following message:

"Press [START] to continue".

Pressing the [START] key, the CNC considers the cycle to be finished and continues with the next block.

*Go into tool inspection:*

If you don't wish to finish the hole nor go to the next hole, it is possible to go into a standard tool inspection.

In this case, a block must be selected and a standard repositioning must be done before resuming the execution of the program.

After inspecting the tool, once repositioning is done, the following choices will be offered:

- Resume the interrupted cycle.
- Skip the interrupted cycle and resume at the next block.

## 9.7    G81. Drilling canned cycle

This cycle drills at the point indicated until the final programmed coordinate is reached. It is possible to program a dwell at the bottom of the drill hole.

Working in Cartesian coordinates, the basic structure of the block is as follows:

G81 G98/G99 X Y Z I K



### [ G98/G99 ] Withdrawal plane

G98    The tool withdraws to the Initial Plane, once the hole has been drilled.

G99    The tool withdraws to the Reference Plane, once the hole has been drilled.

### [ X/Y±5.5 ] Machining coordinates

These are optional and define the movement of the axes of the main plane to position the tool at the machining point.

This point can be programmed in Cartesian coordinates or in polar coordinates, and the coordinates may be absolute or incremental, according to whether the machine is operating in G90 or G91.

### [ Z±5.5 ] Reference plane

Defines the reference plane coordinate. It can be programmed in absolute coordinates or incremental coordinates, in which case it will be referred to the initial plane.

If not programmed, it assumes as reference plane the current position of the tool.

### [ I±5.5 ] Drilling depth.

Defines the total drilling depth. It can be programmed in absolute coordinates or incremental coordinates and in this case will be referred to the reference plane.

### [ K5 ] Dwell

Defines the dwell, in hundredths of a second, after each drilling peck until the withdrawal begins. If not programmed, the CNC will take the value of "K0".

### 9.7.1 Basic operation

1. If the spindle was previously running, it maintains the turning direction. If it was not in movement, it will start by turning clockwise (M03).

2. Rapid movement of the longitudinal axis from the initial plane to the reference plane.

3. Drill the hole. Movement of the longitudinal axis at work feedrate, to the bottom of the hole programmed in "I".

4. Dwell K, in hundredths of a second, if it has been programmed.

5. Withdrawal at rapid feedrate (G00) of the longitudinal axis to the initial or reference plane, depending on whether G98 or G99 has been programmed.

Programming example assuming that the work plane is formed by the X and Y axes, that the Z axis is the longitudinal axis and that the starting point is X0 Y0 Z0:
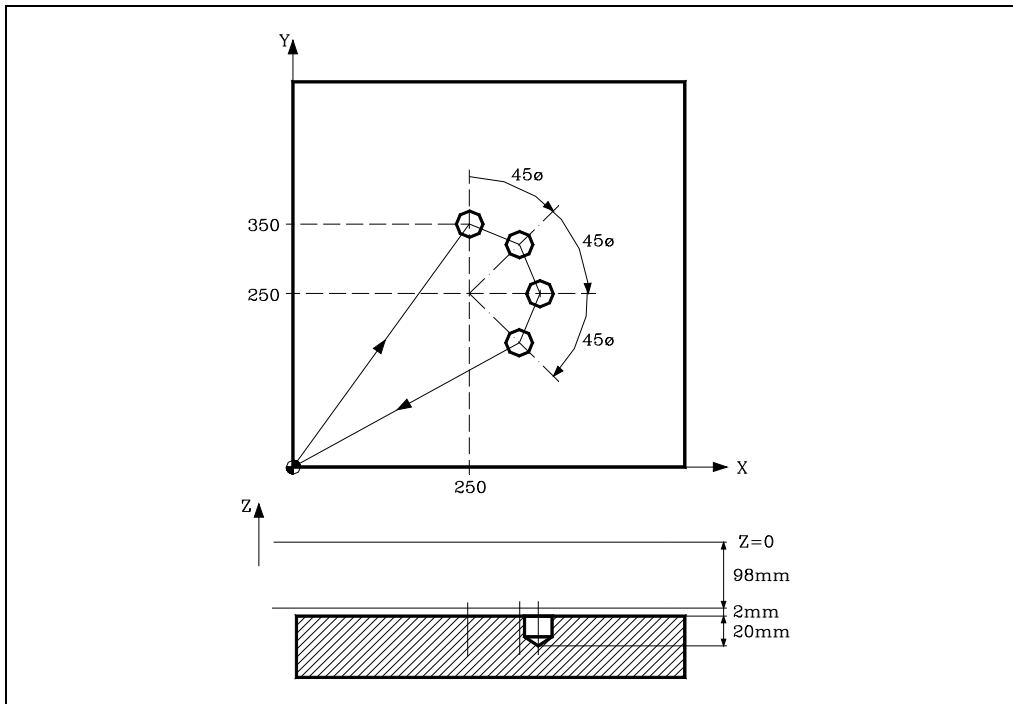


```
; Tool selection.
T1
M6
; Starting point.
G0 G90 X0 Y0 Z0
; Canned cycle definition.
G81 G98 G00 G91 X250 Y350 Z-98 I-22 F100 S500
; Polar coordinate origin.
G93  I250 J250
; Turn and canned cycle, 3 times
Q-45 N3
; Cancels the canned cycle.
G80
; Positioning.
G90 X0 Y0
; End of program.
M30
```

**Tool withdrawal**

While machining, the CNC lets withdraw the tool to the starting plane stopping the spindle when the tool reaches the starting plane.

Activating PLC mark RETRACYC (M5065) stops the main axis and the tool is withdrawn without stopping the spindle. The spindle stops when the tool is retracted, once it reaches the starting plane.

## Options after tool withdrawal

Once the tool has been retracted, the user will have the following options:

- Finish the hole.
- Go to the next hole.
- Go into tool inspection.

After this, the CNC will display the following message:

"To end the cycle, press START to skip to the next SKIPCYCL".

### *Finish the hole:*

To finish the hole, press the [START] key.

It goes down in G0 with spindle running to 1 mm before the coordinate where the hole stopped. From then on, it continues at the F and S programmed in the cycle.

### *Go to the next hole:*

To go to the next hole, activate the PLC mark SKIPCYCL.

The CNC will display the following message:

"Press [START] to continue".

Pressing the [START] key, the CNC considers the cycle to be finished and continues with the next block.

### *Go into tool inspection*

If you don't wish to finish the hole nor go to the next hole, it is possible to go into a standard tool inspection.

In this case, a block must be selected and a standard repositioning must be done before resuming the execution of the program.

After inspecting the tool, once repositioning is done, the following choices will be offered:

- Resume the interrupted cycle.
- Skip the interrupted cycle and resume at the next block.

## 9.8    G82. Drilling canned cycle with dwell

This cycle drills at the point indicated until the final programmed coordinate is reached. Then it executes a dwell at the bottom of the drill hole.

Working in Cartesian coordinates, the basic structure of the block is as follows:

G82 G98/G99 X Y Z I K



### [ G98/G99 ] Withdrawal plane

G98    The tool withdraws to the Initial Plane, once the hole has been drilled.

G99    The tool withdraws to the Reference Plane, once the hole has been drilled.

### [ X/Y±5.5 ] Machining coordinates

These are optional and define the movement of the axes of the main plane to position the tool at the machining point.

This point can be programmed in Cartesian coordinates or in polar coordinates, and the coordinates may be absolute or incremental, according to whether the machine is operating in G90 or G91.

### [ Z±5.5 ] Reference plane

Defines the reference plane coordinate. It can be programmed in absolute coordinates or incremental coordinates, in which case it will be referred to the initial plane.

If not programmed, it assumes as reference plane the current position of the tool.

### [ I±5.5 ] Drilling depth.

Defines the total drilling depth. It can be programmed in absolute coordinates or incremental coordinates and in this case will be referred to the reference plane.

### [ K5 ] Dwell

Defines the dwell, in hundredths of a second, after each drilling until the withdrawal begins. Should this not be programmed, the CNC will take a value of K0.

### 9.8.1    Basic operation

1. If the spindle was previously running, it maintains the turning direction. If it was not in movement, it will start by turning clockwise (M03).

2. Rapid movement of the longitudinal axis from the initial plane to the reference plane.

3. Drill the hole. Movement of the longitudinal axis at work feedrate, to the bottom of the hole programmed in "I".

4. Dwell time K in hundredths of a second.

5. Withdrawal at rapid feedrate (G00) of the longitudinal axis to the initial or reference plane, depending on whether G98 or G99 has been programmed.

Programming example assuming that the work plane is formed by the X and Y axes, that the Z axis is the longitudinal axis and that the starting point is X0 Y0 Z0:
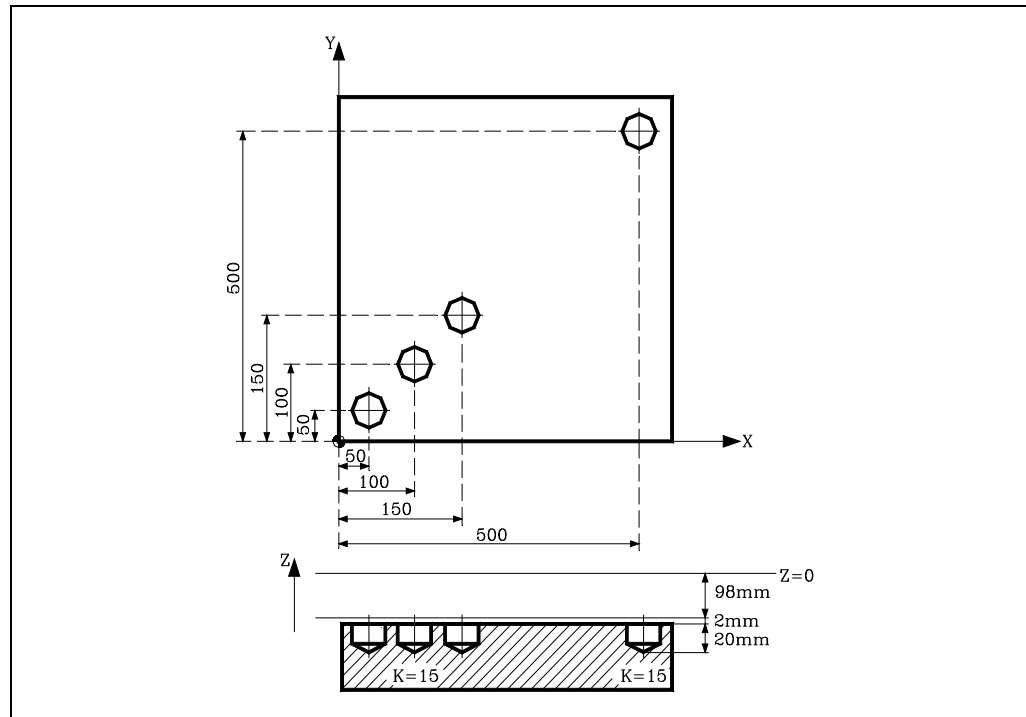
```
; Tool selection.
T1
M6
; Starting point.
G0 G90 X0 Y0 Z0
; Canned cycle definition. Three machining operations are carried out.
G82 G99 G91 X50 Y50 Z-98 I-22 K15 F100 S500 N3
; Positioning and canned cycle
G98 G90 G00 X500 Y500
; Cancels the canned cycle.
G80
; Positioning.
G90 X0 Y0
; End of program.
M30
```

## Tool withdrawal

While machining, the CNC lets withdraw the tool to the starting plane stopping the spindle when the tool reaches the starting plane.

Activating PLC mark RETRACYC (M5065) stops the main axis and the tool is withdrawn without stopping the spindle. The spindle stops when the tool is retracted, once it reaches the starting plane.

## Options after tool withdrawal

Once the tool has been retracted, the user will have the following options:

- Finish the hole.
- Go to the next hole.
- Go into tool inspection.

After this, the CNC will display the following message:

"To end the cycle, press START to skip to the next SKIPCYCL".

### *Finish the hole:*

To finish the hole, press the [START] key.

It goes down in G0 with spindle running to 1 mm before the coordinate where the hole stopped. From then on, it continues at the F and S programmed in the cycle.

### *Go to the next hole:*

To go to the next hole, activate the PLC mark SKIPCYCL.

The CNC will display the following message:

"Press [START] to continue".

Pressing the [START] key, the CNC considers the cycle to be finished and continues with the next block.

### *Go into tool inspection*

If you don't wish to finish the hole nor go to the next hole, it is possible to go into a standard tool inspection.

In this case, a block must be selected and a standard repositioning must be done before resuming the execution of the program.

After inspecting the tool, once repositioning is done, the following choices will be offered:

- Resume the interrupted cycle.
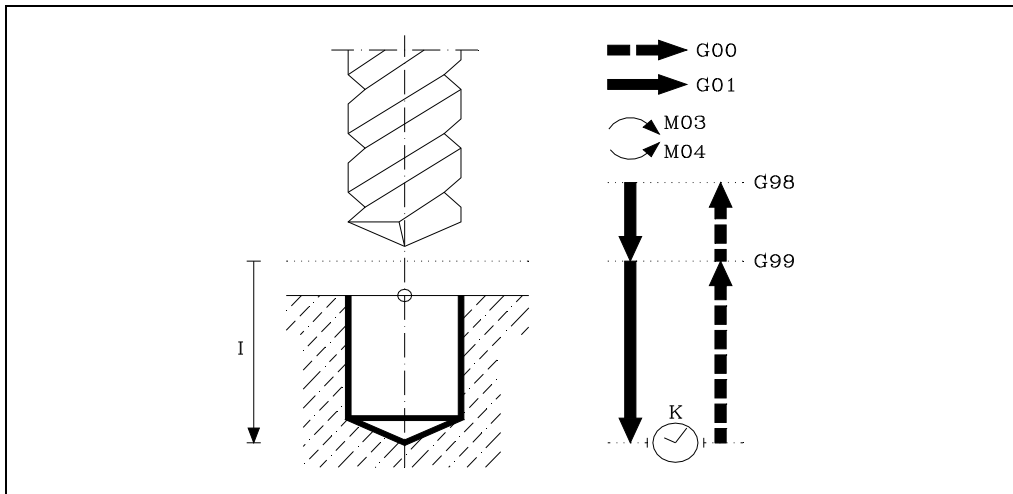- Skip the interrupted cycle and resume at the next block.

## 9.9    G83. Deep-hole drilling canned cycle with constant peck

This cycle makes successive drilling steps until the final coordinate is reached.

The tool withdraws as far as the reference plane after each drilling step.

Working in Cartesian coordinates, the basic structure of the block is as follows:

    G83 G98/G99 X Y Z I J



### [ G98/G99 ] Withdrawal plane

G98    The tool withdraws to the Initial Plane, once the hole has been drilled.

G99    The tool withdraws to the Reference Plane, once the hole has been drilled.

### [ X/Y±5.5 ] Machining coordinates

These are optional and define the movement of the axes of the main plane to position the tool at the machining point.

This point can be programmed in Cartesian coordinates or in polar coordinates, and the coordinates may be absolute or incremental, according to whether the machine is operating in G90 or G91.

### [ Z±5.5 ] Reference plane

Defines the reference plane coordinate. It can be programmed in absolute coordinates or incremental coordinates, in which case it will be referred to the initial plane.

If not programmed, it assumes as reference plane the current position of the tool.

### [ I±5.5 ] Depth of each drilling peck

Defines the value of each drilling step according to the axis longitudinal to the main plane.

### [ J4 ] Drilling passes to withdraw to the starting plane

Defines the number of steps which the drill is to make. A value between 1 and 9999 may be programmed.
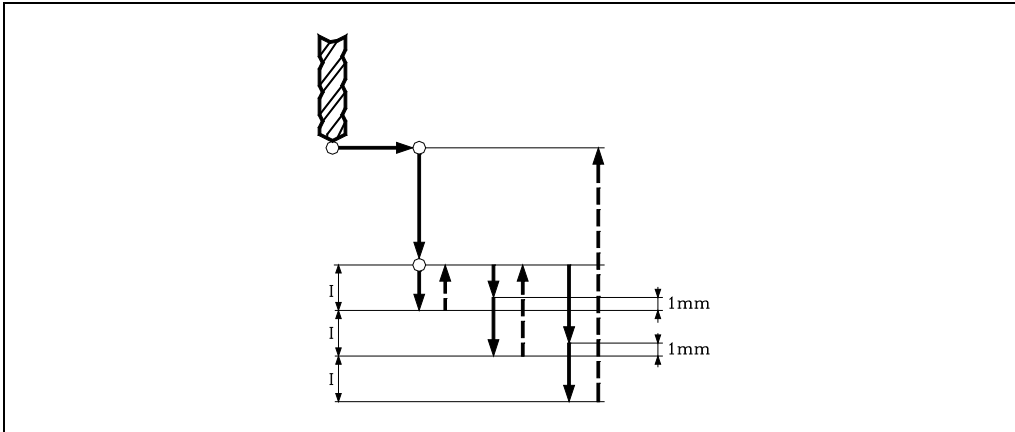
### 9.9.1    Basic operation

1.  If the spindle was previously running, it maintains the turning direction. If it was not in movement, it will start by turning clockwise (M03).

2.  Rapid movement of the longitudinal axis from the initial plane to the reference plane.

3.  First drilling operation. The longitudinal axis moves in G01 to the programmed incremental depth "I".

4.  Drilling loop. The following steps will be repeated "J-1" times as in the previous step the first programmed drilling was done.

    ·1· Withdrawal of the longitudinal axis in rapid (G00) to the reference plane.

    ·2· Longitudinal axis approach in rapid (G00):

    If INPOSW2 < INPOSW1, down to 1mm off the previous drilling peck.

    If not, down to twice the value of INPOSW2.

    ·3· New drilling step. G01 movement of the longitudinal axis to the incremental depth programmed in "I".

    If INPOSW2=0 in G7. If not, in G50.

5.  Withdrawal at rapid feedrate (G00) of the longitudinal axis to the initial or reference plane, depending on whether G98 or G99 has been programmed.

The first drilling penetration is done in G07 or G50 depending on the value assigned to the parameter of the longitudinal axis "INPOSW2 (P51)" and to parameter "INPOSW1 (P19)". This is important to join a drilling operation with another one in multiple drilling so the tool path is faster and smoother.

If INPOSW2 < INPOSW1 in G07 (square corner).

If INPOSW2 >= INPOSW1 in G50 (controlled round corner).

If a scaling factor is applied to this cycle, drilling will be performed proportional to that programmed, with the same step "I" programmed, but varying the number of steps "J".

Programming example assuming that the work plane is formed by the X and Y axes, that the Z axis is the longitudinal axis and that the starting point is X0 Y0 Z0:
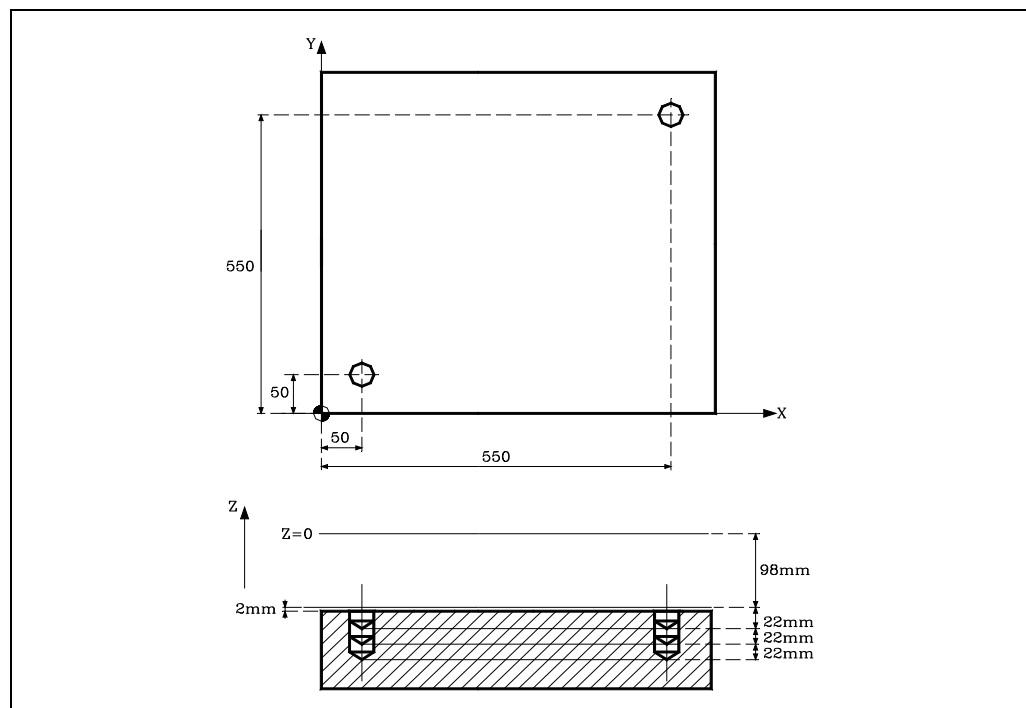
**9.**

**CANNED CYCLES**

G83. Deep-hole drilling canned cycle with constant peck

**FAGOR**

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

·170·

```
; Tool selection.
T1
M6
; Starting point.
G0 G90 X0 Y0 Z0
; Canned cycle definition.
G83 G99 X50 Y50 Z-98 I-22 J3 F100 S500 M4
; Positioning and canned cycle
G98 G90 G00 X500 Y500
; Cancels the canned cycle.
G80
; Positioning.
G90 X0 Y0
; End of program.
M30
```

## Tool withdrawal

While machining, the CNC lets withdraw the tool to the starting plane stopping the spindle when the tool reaches the starting plane.

Activating PLC mark RETRACYC (M5065) stops the main axis and the tool is withdrawn without stopping the spindle. The spindle stops when the tool is retracted, once it reaches the starting plane.

## Options after tool withdrawal

Once the tool has been retracted, the user will have the following options:

• Finish the hole.

• Go to the next hole.

• Go into tool inspection.

After this, the CNC will display the following message:

"To end the cycle, press START to skip to the next SKIPCYCL".

### Finish the hole:

To finish the hole, press the [START] key.

It goes down in G0 with spindle running to 1 mm before the coordinate where the hole stopped. From then on, it continues at the F and S programmed in the cycle.

### Go to the next hole:

To go to the next hole, activate the PLC mark SKIPCYCL.

The CNC will display the following message:

"Press [START] to continue".

Pressing the [START] key, the CNC considers the cycle to be finished and continues with the next block.

### Go into tool inspection

If you don't wish to finish the hole nor go to the next hole, it is possible to go into a standard tool inspection.

In this case, a block must be selected and a standard repositioning must be done before resuming the execution of the program.

After inspecting the tool, once repositioning is done, the following choices will be offered:

• Resume the interrupted cycle.

• Skip the interrupted cycle and resume at the next block.

## 9.10 G84. Tapping canned cycle

This cycle taps at the point indicated until the final programmed coordinate is reached. General logic output "TAPPING" (M5517) stays active while executing this cycle.

Due to the fact that the tapping tool turns in two directions (one when tapping and the other when withdrawing from the thread), by means of the machine parameter of the spindle "SREVM05" it is possible to select whether the change in turning direction is made with the intermediate spindle stop, or directly.

General machine parameter "STOPAP(P116)" indicates whether general inputs /STOP, /FEEDHOL and /XFERINH are enabled or not while executing function G84.

It is possible to program a dwell before each reversal of the spindle turning direction, i.e., at the bottom of the thread hole and when returning to the reference plane.

Using parameters B and H, the threading may be done with relief for chip breakage.

The tapping with relief is done in successive approaches until the programmed total depth is reached. After every approach, it withdraws for chip relief. In this case, the dwell (K) is only applied on the last pass, not on the relief passes.

Working in Cartesian coordinates, the basic structure of the block is as follows:

G84 G98/G99 X Y Z I K R J B H
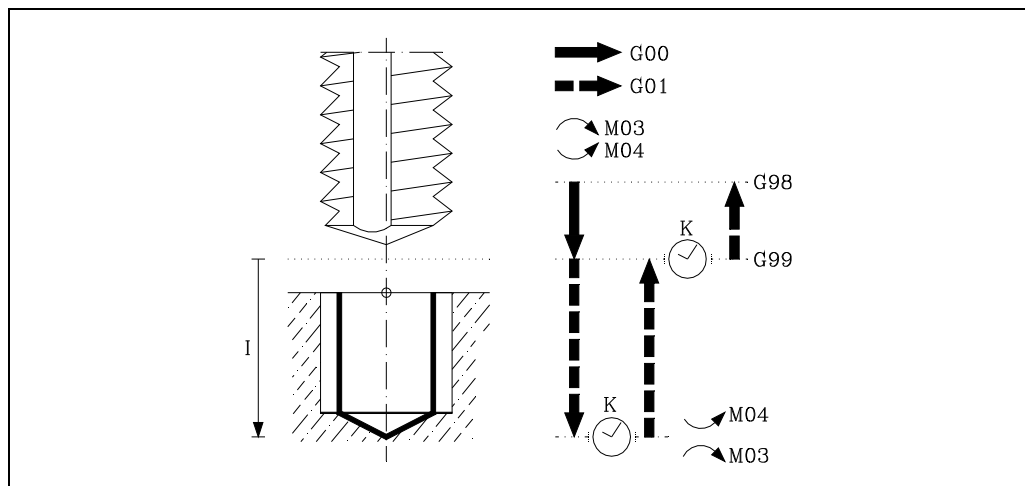


### [ G98/G99 ] Withdrawal plane

G98     The tool withdraws to the Initial Plane, once the hole has tapped.

G99     The tool withdraws to the Reference Plane, once the hole has tapped.

### [ X/Y±5.5 ] Machining coordinates

These are optional and define the movement of the axes of the main plane to position the tool at the machining point.

This point can be programmed in Cartesian coordinates or in polar coordinates, and the coordinates may be absolute or incremental, according to whether the machine is operating in G90 or G91.

### [ Z±5.5 ] Reference plane

Defines the reference plane coordinate. It can be programmed in absolute coordinates or incremental coordinates, in which case it will be referred to the initial plane.

If not programmed, it assumes as reference plane the current position of the tool.

### [ I±5.5 ] Thread depth.

Defines tapping depth. It can be programmed in absolute coordinates or incremental coordinates and in this case will be referred to the reference plane.

### [ K5 ] Dwell

Defines the dwell, in hundredths of a second, after the tapping until the withdrawal begins. If not programmed, the CNC will take the value of "K0".

### [ R ] Type of tapping

Defines the type of tapping to be carried out.

R0     Regular tapping.

R1     Rigid tapping. The CNC stops the spindle with M19 and orients it to begin tapping.

R1     Rigid tapping. If the spindle is turning in M3 or M4, the CNC does not stop it nor orient it to begin tapping. This option does not allow thread repair, even if the part has not been released because the thread start (entry) will coincide with the one machined earlier.

### [ J5.5 ] Withdrawal feedrate factor

When rigid tapping, the returning feedrate will be J times the tapping feedrate. When not programmed or programmed J1, they will both be the same.

In order to execute a rigid tapping, the spindle must be ready to operate in closed loop; in other words, that it must have a servo drive-motor system with rotary encoder.

When rigid tapping, the CNC interpolates the movement of the longitudinal axis with the spindle rotation.

### [ B5.5 ] Penetration step during tapping with relief.

It is optional and sets the penetration step during tapping with relief. This parameter is ignored when programming R=0 or R=2. Tapping with relief is only possible when programming R=1.

When not programmed, tapping will be done in a single pass. When programmed with a 0 value, it issues the corresponding error.

### [ H5.5 ] Withdrawal distance after each penetration step.

This regress will be performed at a rate that takes into account the factor programmed for J. This parameter will be ignored if R=0 or R=2 is programmed or if parameter B has not been programmed.

If not programmed or programmed with a 0 value, the withdrawal will be done to the Z reference plane coordinate.

**9.**

**CANNED CYCLES**

G84. Tapping canned cycle

**FAGOR**

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2x

### 9.10.1 Basic operation

1. If the spindle was previously running, it maintains the turning direction. If it was not in movement, it will start by turning clockwise (M03).

2. Rapid movement of the longitudinal axis from the initial plane to the reference plane.

3. Movement of the longitudinal axis and at the working feedrate, to the bottom of the machined section, producing the threaded hole. The canned cycle will execute this movement and all later movements at 100% of F feedrate and the programmed S speed.

   If rigid tapping is selected (parameter R=1), the CNC will activate the general logic output "RIGID" (M5521) to indicate to the PLC that a rigid tapping block is being executed.

4. Spindle stop (M05). This will only be performed when the spindle machine parameter "SREVM05" is selected and parameter "K" has a value other than "0"..

5. Dwell, if parameter "K" has been programmed.

6. The spindle reverses turning direction.

7. Withdrawal, at J times the working feedrate, of the longitudinal axis to the reference plane. Once this coordinate has been reached, the canned cycle will assume the selected FEEDRATE OVERRIDE and the SPINDLE OVERRIDE.

   If rigid tapping is selected (parameter R=1), the CNC will activate the general logic output "RIGID" (M5521) to indicate to the PLC that a rigid tapping block is being executed.

8. Spindle stop (M05). This will only be performed if the spindle machine parameter "SREVM05" is selected.

9. Dwell, if parameter "K" has been programmed.

10. Reverse the spindle turning direction restoring the initial turning direction.

11. Withdrawal, at rapid feedrate (G00), of the longitudinal axis as far as the initial plane if G98 has been programmed.

Programming example assuming that the work plane is formed by the X and Y axes, that the Z axis is the longitudinal axis and that the starting point is X0 Y0 Z0:



**9.**

**CANNED CYCLES**

G84. Tapping canned cycle

**FAGOR** ✦

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

·174·

```
; Tool selection.
T1
M6
; Starting point.
G0 G90 X0 Y0 Z0
; Canned cycle definition. Three machining operations are carried out.
G84 G99 G91 X50 Y50 Z-98 I-22 K150 F350 S500 N3
; Positioning and canned cycle
G98 G90 G00 X500 Y500
; Cancels the canned cycle.
G80
; Positioning.
G90 X0 Y0
; End of program.
M30
```

## Tool withdrawal

While machining, the CNC lets withdraw the tool to the starting plane stopping the spindle when the tool reaches the starting plane.

Activating PLC mark RETRACYC (M5065) stops the main axis and the spindle, the tool is withdrawn changing the direction of the axis and spindle while keeping the machining F and S values. This withdrawal will be to the starting plane.

The stopping and starting sequence of the spindle and the axis when tapping keeps the same synchronism and dwells as during the execution of the canned cycle.

## Options after tool withdrawal

Once the tool has been retracted, the user will have the following options:

• Finish the hole.
• Go to the next hole.
• Go into tool inspection.

After this, the CNC will display the following message:

"To end the cycle, press START to skip to the next SKIPCYCL".

### *Finish the hole:*

To finish the hole, press the [START] key.

The hole is repeated from the starting plane under the same F and S conditions without stopping at the point where it stopped.

### *Go to the next hole:*

To go to the next hole, activate the PLC mark SKIPCYCL.

The CNC will display the following message:

"Press [START] to continue".

Pressing the [START] key, the CNC considers the cycle to be finished and continues with the next block.

### *Go into tool inspection*

If you don't wish to finish the hole nor go to the next hole, it is possible to go into a standard tool inspection.

In this case, a block must be selected and a standard repositioning must be done before resuming the execution of the program.

After inspecting the tool, once repositioning is done, the following choices will be offered:

- Resume the interrupted cycle.
- Skip the interrupted cycle and resume at the next block.

## 9.11    G85. Reaming canned cycle

This cycle reams at the point indicated until the final programmed coordinate is reached.

It is possible to program a dwell at the bottom of the machined hole.

Working in Cartesian coordinates, the basic structure of the block is as follows:

G85 G98/G99 X Y Z I K
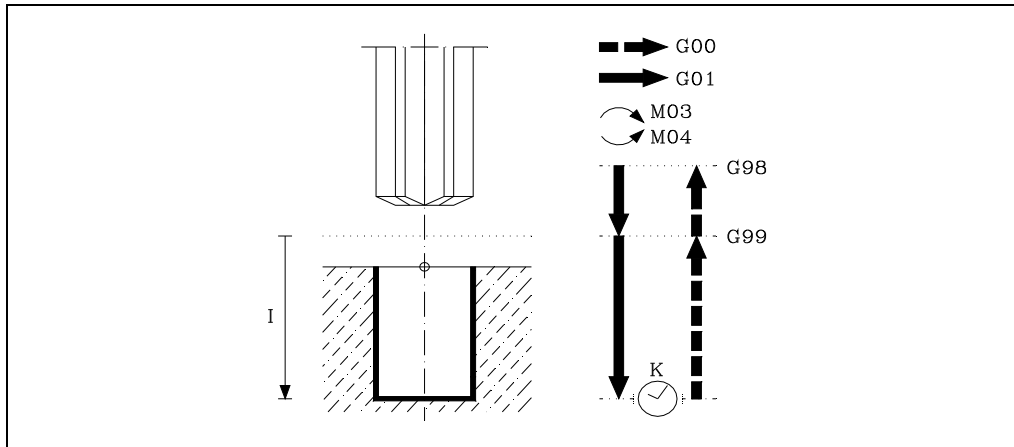


**[ G98/G99 ] Withdrawal plane**

G98    The tool withdraws to the Initial Plane, once the hole has been reamed.

G99    The tool withdraws to the Reference Plane, once the hole has been reamed.

**[ X/Y±5.5 ] Machining coordinates**

These are optional and define the movement of the axes of the main plane to position the tool at the machining point.

This point can be programmed in Cartesian coordinates or in polar coordinates, and the coordinates may be absolute or incremental, according to whether the machine is operating in G90 or G91.

**[ Z±5.5 ] Reference plane**

Defines the reference plane coordinate. It can be programmed in absolute coordinates or incremental coordinates, in which case it will be referred to the initial plane.

If not programmed, it assumes as reference plane the current position of the tool.

**[ I±5.5 ] Reaming depth.**

Defines reaming depth. It can be programmed in absolute coordinates or incremental coordinates and in this case will be referred to the reference plane.

**[ K5 ] Dwell**

Defines the dwell, in hundredths of a second, after the reaming until the withdrawal begins. If not programmed, the CNC will take the value of "K0".
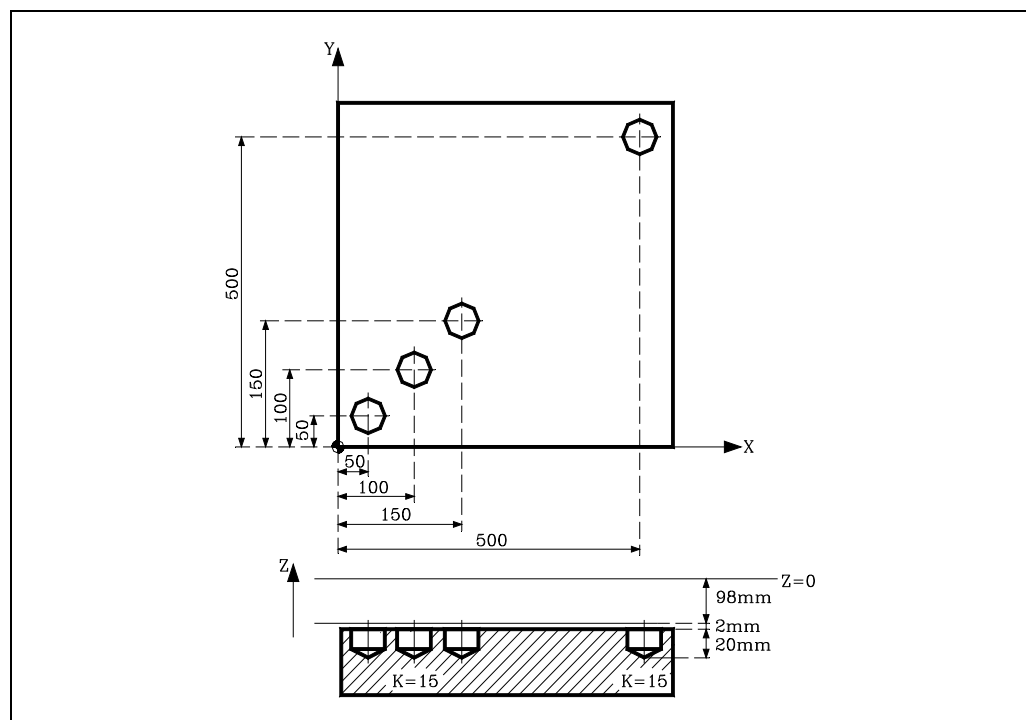
### 9.11.1 Basic operation

1. If the spindle was previously running, it maintains the turning direction. If it was not in movement, it will start by turning clockwise (M03).

2. Rapid movement of the longitudinal axis from the initial plane to the reference plane.

3. Movement at the working feedrate (G01) of the longitudinal axis to the bottom of the machined hole, and reaming.

4. Dwell, if parameter "K" has been programmed.

5. Withdrawal at working feedrate, of the longitudinal axis as far as the reference plane.

6. Withdrawal, at rapid feedrate (G00), of the longitudinal axis as far as the initial plane if G98 has been programmed.

Programming example assuming that the work plane is formed by the X and Y axes, that the Z axis is the longitudinal axis and that the starting point is X0 Y0 Z0:

```
; Tool selection.
T1
M6
; Starting point.
G0 G90 X0 Y0 Z0
; Canned cycle definition.
G85 G98 G91 X250 Y350 Z-98 I-22 F100 S500
; Cancels the canned cycle.
G80
; Positioning.
G90 X0 Y0
; End of program.
M30
```

## 9.12    G86. Boring cycle with withdrawal in G00

This cycle bores at the point indicated until the final programmed coordinate is reached. It is possible to program a dwell at the bottom of the machined hole.

After the quail has penetrated, it is possible to orient the spindle and retract the quail before the exit movement, thus preventing the part from scratching. This is only available when using spindle orientation.

Working in Cartesian coordinates, the basic structure of the block is as follows:

G86 G98/G99 X Y Z I K Q D E

### [ G98/G99 ] Withdrawal plane

G98     The tool withdraws to the Initial Plane, once the hole has been bored.

G99     The tool withdraws to the Reference Plane, once the hole has been bored.

### [ X/Y±5.5 ] Machining coordinates

These are optional and define the movement of the axes of the main plane to position the tool at the machining point.

This point can be programmed in Cartesian coordinates or in polar coordinates, and the coordinates may be absolute or incremental, according to whether the machine is operating in G90 or G91.

### [ Z±5.5 ] Reference plane

Defines the reference plane coordinate. It can be programmed in absolute coordinates or incremental coordinates, in which case it will be referred to the initial plane.

If not programmed, it assumes as reference plane the current position of the tool.

### [ I±5.5 ] Reaming depth.

Defines boring depth. It can be programmed in absolute coordinates or incremental coordinates and in this case will be referred to the reference plane.

### [ K5 ] Dwell

Defines the dwell, in hundredths of a second, after boring until the withdrawal begins. If not programmed, the CNC will take the value of "K0".

### [ Q±5.5 ] Spindle position for the withdrawal

It defines the spindle position, in degrees, to separate the cutter from the wall of the hole.

If not programmed, the withdrawal will be done without separating the cutter from the wall of the hole, the spindle stopped and in rapid.

**[ D±5.5 ] Gap between the cutter and the wall of the hole on the X axis**

Defines the gap between the cutter and the wall of the hole on the X axis for the withdrawal.

If not programmed, the cutter does not separate from the wall of the hole along the X axis.

Both Q and D must be programmed for the cutter to separate from the wall of the hole.

**[ E±5.5 ] Gap between the cutter and the wall of the hole on the Y axis**

Defines the gap between the cutter and the wall of the hole on the Y axis for the withdrawal.

If not programmed, the cutter does not separate from the wall of the hole along the Y axis.

Both Q and E must be programmed for the cutter to separate from the wall of the hole.

**9.**

**CANNED CYCLES**

G86. Boring cycle with withdrawal in G00

### 9.12.1 Basic operation

1. If the spindle was previously running, it maintains the turning direction. If it was not in movement, it will start by turning clockwise (M03).

2. Rapid movement of the longitudinal axis from the initial plane to the reference plane.

3. Movement at the working feedrate (G01) of the longitudinal axis to the bottom of the machined hole, and boring.

4. Dwell, if parameter "K" has been programmed.

5. Spindle orientation to the position programmed in parameter Q.

6. The movement of the tool, for interpolated movements and slow feedrates, the distances programmed for the parameters D and E. If the values are not programmed correctly it could cause the blade to collide with the wall instead of pulling away from it.

7. Tool withdrawal, in rapid (G00), to the starting plane or to the reference plane depending on whether G98 or G99 has been programmed.

8. Tool movement, in a rapid interpolated movement, the distances programmed in parameters D and E with opposite signs (undoing the movement made in step 6).

9. When done with the withdrawal, the spindle will start in the same direction as it was turning before.

Programming example assuming that the work plane is formed by the X and Y axes, that the Z axis is the longitudinal axis and that the starting point is X0 Y0 Z0:

```
; Tool selection.
T1
M6
; Starting point.
G0 G90 X0 Y0 Z0
; Canned cycle definition.
G86 G98 G91 X250 Y350 Z-98 I-22 K20 F100 S500
; Cancels the canned cycle.
G80
; Positioning.
G90 X0 Y0
; End of program.
M30
```

## 9.13    G87. Rectangular pocket canned cycle.

This cycle executes a rectangular pocket at the point indicated until the final programmed coordinate is reached.

It is possible to program, in addition to milling pass and feedrate, a final finishing step with its corresponding milling feedrate.

In order to obtain a good finish in the machining of the pocket walls, the CNC will apply a tangential entry and exit to the last milling step during each cutting operation.

Working in Cartesian coordinates, the basic structure of the block is as follows:

G87 G98/G99 X Y Z I J K B C D H L V
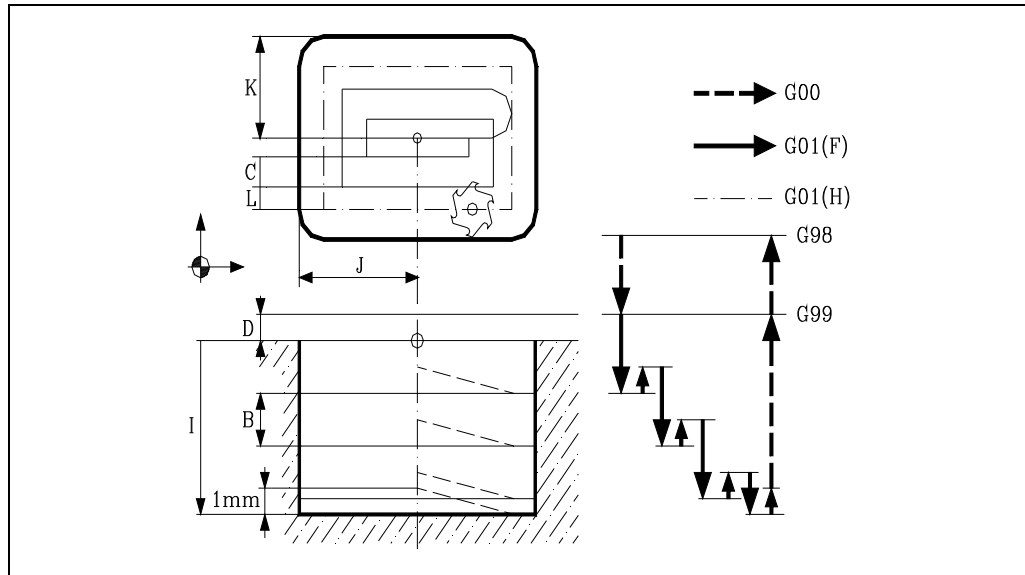


### [ G98/G99 ] Withdrawal plane

G98    The tool withdraws to the Initial Plane, once the pocket has been made.

G99    The tool withdraws to the Reference Plane, once the pocket has been made.

### [ X/Y±5.5 ] Machining coordinates

These are optional and define the movement of the axes of the main plane to position the tool at the machining point.

This point can be programmed in Cartesian coordinates or in polar coordinates, and the coordinates may be absolute or incremental, according to whether the machine is operating in G90 or G91.

### [ Z±5.5 ] Reference plane

Defines the reference plane coordinate.

When programmed in absolute coordinates, it will be referred to the part zero and when programmed in incremental coordinates, it will be referred to the starting plane (P.P.).

If not programmed, it assumes as reference plane the current position of the tool. Thus, the starting plane (P.P.) and the reference plane (P.R.) will be the same.
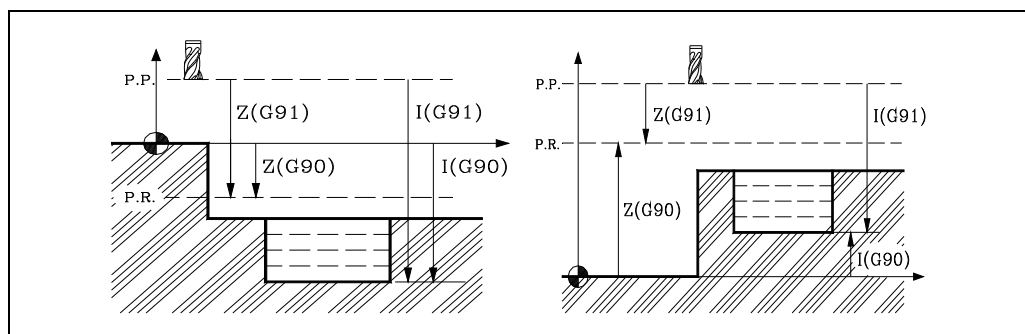
**[ I±5.5 ] Machining depth.**

Defines the machining depth.

When programmed in absolute coordinates, it will be referred to the part zero and when programmed in incremental coordinates, it will be referred to the starting plane (P.P.).

**[ J±5.5 ] Half the width of the pocket along the abscissa axis**

Defines the distance from the center to the edge of the pocket according to the abscissa axis. The sign indicates the pocket machining direction.



J with "+" sign          J with "-" sign
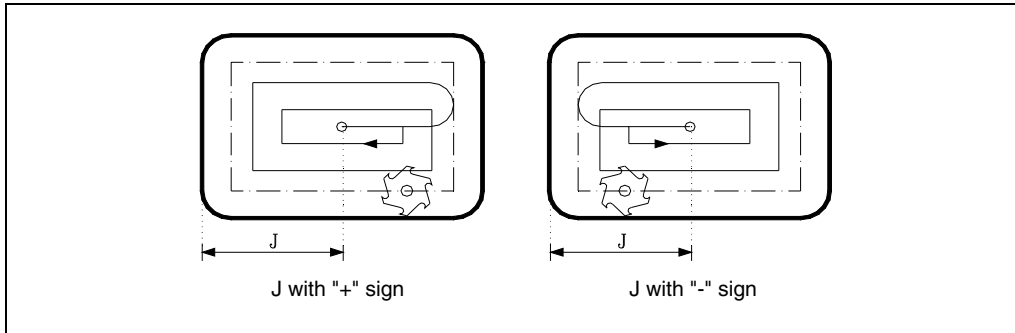
**[ K5.5 ] Half the width of the pocket along the ordinate axis**

Defines the distance from the center to the edge of the pocket according to the ordinate axis.
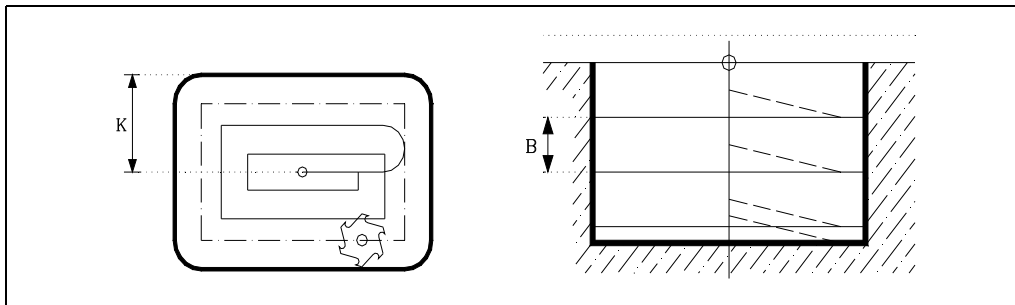


**[ B±5.5 ] Penetration step**

Defines the cutting depth according to the longitudinal axis.

If this is programmed with a positive sign, the entire cycle will be executed with the same machining pass, this being equal to or less than that programmed.

If programmed with a negative sign, the whole pocket is machined with the given pass (step) except the last pass that machines the rest.

**[ C±5.5 ] Milling pass**

Defines the milling pass along the main plane.

If the value is positive, the entire cycle will be executed with the same milling step, this being equal to or less than that programmed.

If the value is negative, the entire pocket will be executed with the given step, except for the last step which will machine whatever remains.



If not programmed, it assumes a value of 3/4 of the diameter of the selected tool.

If programmed with a value greater than the tool diameter, the CNC issues the relevant error message.

If programmed with a 0 value, the CNC will display the corresponding error message.

### [ D5.5 ] Reference plane

Defines the distance between the reference plane and the surface of the part where the pocket is to be made.

During the first deepening operation this amount will be added to incremental depth "B". If not programmed, a value of 0 is assumed.



### [ H.5.5 ] Feedrate for the finishing pass

Defines the working feedrate during the finishing pass.

If not programmed or programmed with a 0 value, it assumes the value of the machining feedrate.

### [ L±5.5 ] Finishing stock.

Defines the value of the finishing pass, along the main plane.

If the value is positive, the finishing pass is made on a square corner (G07).

If the value is negative, the finishing pass is made on a rounded corner (G05).



If not programmed or programmed with a 0 value, it does not run the finishing pass.

### [ V.5.5 ] Tool penetrating feedrate.

Defines the tool penetrating feedrate.

If not programmed or programmed with a 0 value, it assumes 50% of the feedrate in the plane (F).

## 9.13.1 Basic operation

1. If the spindle was previously running, it maintains the turning direction. If it was not in movement, it will start by turning clockwise (M03).

2. Rapid movement (G0) of the longitudinal axis from the starting plane to the reference plane.

3. First penetrating operation. Movement of longitudinal axis at the feedrate indicated by "V" to the incremental depth programmed in "B+D".

4. Milling of the pocket surface at work feedrate in the passes defined by "C" up to a distance "L" (finishing pass) from the pocket wall.

5. Finishing pass milling, "L" at the work feedrate defined by "H".

6. Once the finishing pass has been completed, the tool withdraws at the rapid feedrate (G00) to the center of the pocket, the longitudinal axis being separated 1 mm (0.040 inch) from the machined surface.



7. New milling surfaces until reaching the total depth of the pocket.

   ·1· Movement of the longitudinal axis at the feedrate indicated by "V", up to a distance "B" from the previous surface.

   ·2· Milling of the new surface following the steps indicated in points 4, 5 and 6.

8. Withdrawal at rapid feedrate (G00) of the longitudinal axis to the initial or reference plane, depending on whether G98 or G99 has been programmed.

**FAGOR**

FAGOR AUTOMATION

CNC 8055
CNC 8055i

·M· & ·EN· MODELS
SOFT: V02.2X

### Programming example ·1·

Let us suppose a work plane formed by the X and Y axis, Z being the longitudinal axis and the starting point X0 Y0 Z0.

```
; Tool selection.
(TOR1=6, TOI1=0)
T1 D1
M6
; Starting point
G0 G90 X0 Y0 Z0
; Canned cycle definition
G87 G98 X90 Y60 Z-48 I-90 J52.5 K37.5 B12 C10 D2 H100 L5 V100 F300 S1000 M03
; Cancels the canned cycle
G80
; Positioning
G90 X0 Y0
; End of program
M30
```

### Programming example ·2·

Let us suppose a work plane formed by the X and Y axis, Z being the longitudinal axis and the starting point X0 Y0 Z0.
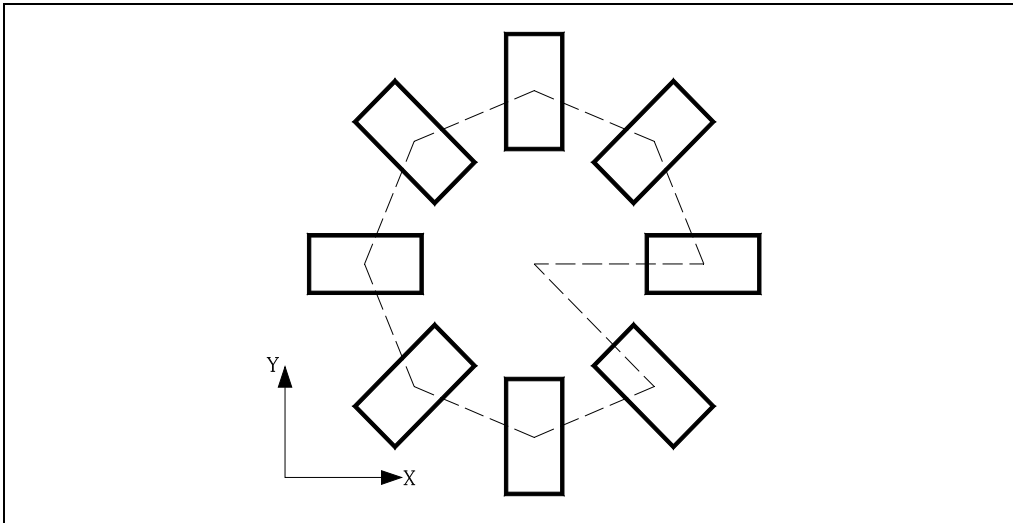
```
; Tool selection.
(TOR1=6, TOI1=0)
T1 D1
M6
; Starting point
G0 G90 X0 Y0 Z0
; Work plane.
G18
; Canned cycle definition
N10 G87 G98 X200 Y-48 Z0 I-90 J52.5 K37.5 B12 C10 D2 H100 L5 V50 F300
; Coordinate rotation
N20 G73 Q45
; Repeats the select blocks 7 times.
(RPT N10,N20) N7
; Cancels the canned cycle.
G80
; Positioning
G90 X0 Y0
; End of program
M30
```

**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

## 9.14    G88. Circular pocket canned cycle

This cycle executes a circular pocket at the point indicated until the final programmed coordinate is reached.

It is possible to program, in addition to milling pass and feedrate, a final finishing step with its corresponding milling feedrate.

Working in Cartesian coordinates, the basic structure of the block is as follows:

G88 G98/G99 X Y Z I J B C D H L V



### [ G98/G99 ] Withdrawal plane

G98    The tool withdraws to the Initial Plane, once the pocket has been made.

G99    The tool withdraws to the Reference Plane, once the pocket has been made.

### [ X/Y±5.5 ] Machining coordinates

These are optional and define the movement of the axes of the main plane to position the tool at the machining point.

This point can be programmed in Cartesian coordinates or in polar coordinates, and the coordinates may be absolute or incremental, according to whether the machine is operating in G90 or G91.

### [ Z±5.5 ] Reference plane

Defines the reference plane coordinate.

It may be programmed either in absolute or incremental coordinates, in which case it will be referred to the starting plane. If not programmed, it assumes as reference plane the current position of the tool.

### [ I±5.5 ] Machining depth

Defines the machining depth. It may be programmed either in absolute or incremental coordinates, in which case it will be referred to the reference plane.

**[ J±5.5 ] Pocket radius**

Defines the radius of the pocket. The sign indicates the pocket machining direction.



J with "+" sign          J with "-" sign
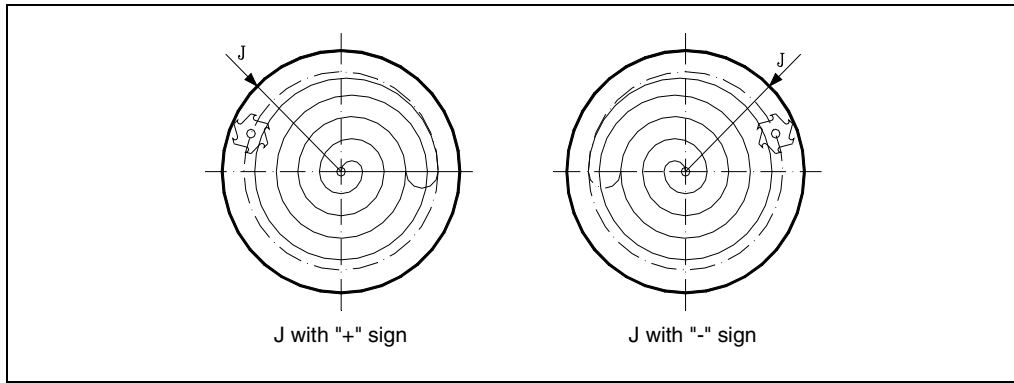
**[ B±5.5 ] Penetration step**

Defines the cutting pass along the longitudinal axis to the main plane.

- If this value is positive, the entire cycle will be executed with the same machining pass, this being equal to or less than that programmed.
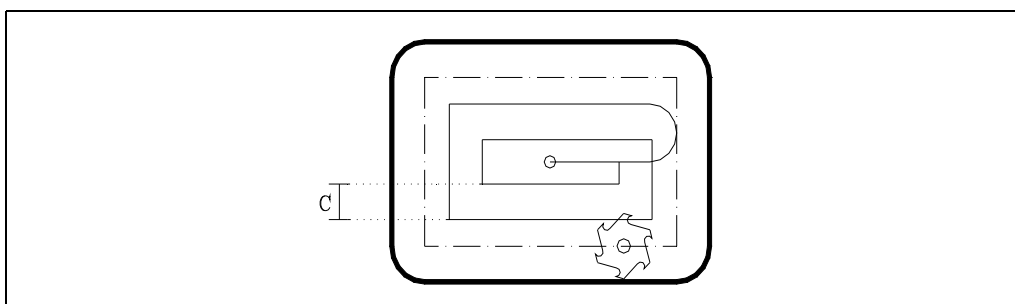- If the value is negative, the entire pocket will be executed with the given step, except for the last step which will machine whatever remains.



**[ C±5.5 ] Milling pass**

Defines the milling pass along the main plane.

- If the value is positive, the entire cycle will be executed with the same milling step, this being equal to or less than that programmed.
- If the value is negative, the entire pocket will be executed with the given step, except for the last step which will machine whatever remains.

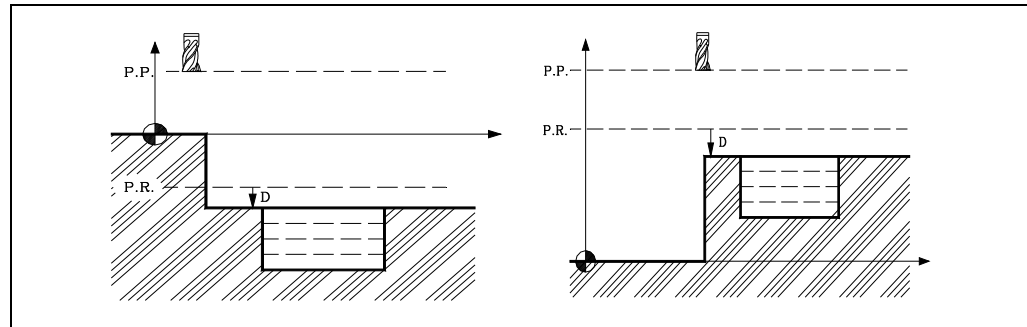If not programmed, it assumes a value of 3/4 of the diameter of the selected tool.

If programmed with a value greater than the tool diameter, the CNC issues the relevant error message.

If programmed with a 0 value, the CNC will display the corresponding error message.

**[ D5.5 ] Reference plane**

Defines the distance between the reference plane and the surface of the part where the pocket is to be made.

During the first deepening operation this amount will be added to incremental depth "B". If not programmed, a value of 0 is assumed.



**[ H5.5 ] Feedrate for the finishing pass**
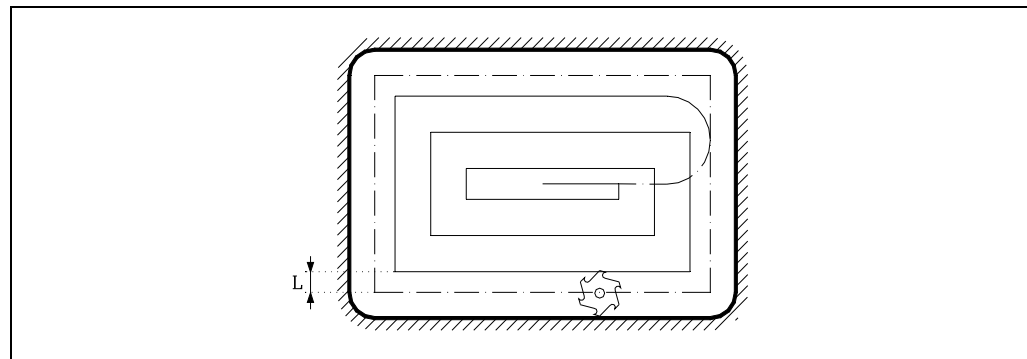
Defines the working feedrate during the finishing pass.

If not programmed or programmed with a 0 value, it assumes the value of the machining feedrate.

**[ L5.5 ] Finishing stock**

Defines the value of the finishing pass, along the main plane.

If not programmed or programmed with a 0 value, it does not run the finishing pass.

**[ V.5.5 ] Tool penetrating feedrate.**

Defines the tool penetrating feedrate.

If not programmed or programmed with a 0 value, it assumes 50% of the feedrate in the plane (F).
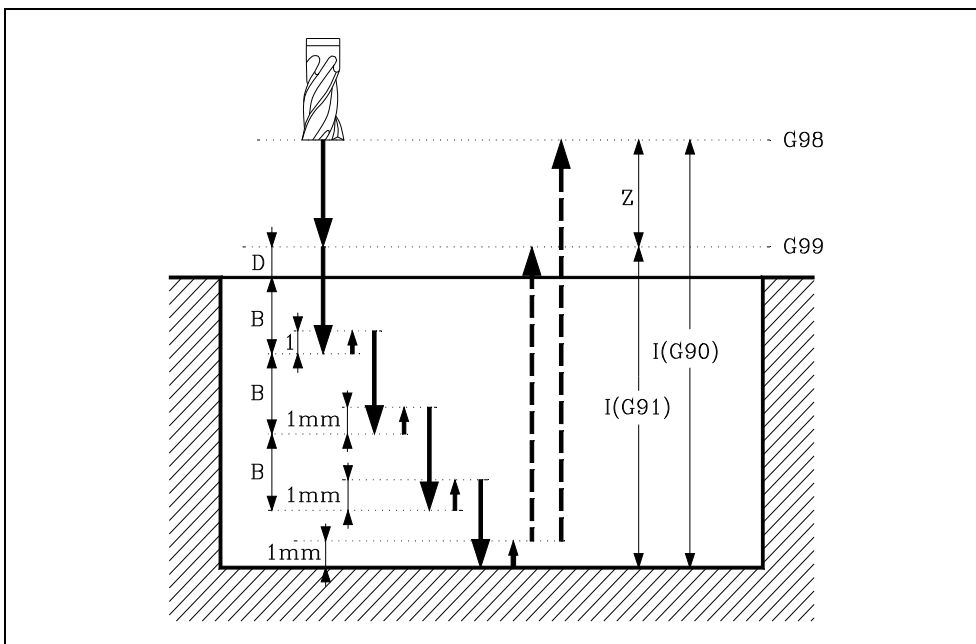
### 9.14.1   Basic operation

1. If the spindle was previously running, it maintains the turning direction.

   If it was not in movement, it will start by turning clockwise (M03).

2. Rapid movement (G0) of the longitudinal axis from the starting plane to the reference plane.

3. First penetrating operation. Movement of longitudinal axis at the feedrate indicated by "V" to the incremental depth programmed in "B+D".

4. Milling of the pocket surface at work feedrate in the passes defined by "C" up to a distance "L" (finishing pass) from the pocket wall.

5. Finishing pass milling, "L" at the work feedrate defined by "H".

6. Once the finishing pass has been completed, the tool withdraws at the rapid feedrate (G00) to the center of the pocket, the longitudinal axis being separated 1 mm (0.040 inch) from the machined surface.



7. New milling surfaces until reaching the total depth of the pocket.

   ·1· Movement of the longitudinal axis at the feedrate indicated by "V", up to a distance "B" from the previous surface.

   ·2· Milling of the new surface following the steps indicated in points 4, 5 and 6.

8. Withdrawal at rapid feedrate (G00) of the longitudinal axis to the initial or reference plane, depending on whether G98 or G99 has been programmed.

### Programming example ·1·

Let us suppose a work plane formed by the X and Y axis, Z being the longitudinal axis and the starting point X0 Y0 Z0.



```
; Tool selection.
(TOR1=6, TOI1=0)
T1 D1
M6
; Starting point
G0 G90 X0 Y0 Z0
; Canned cycle definition
G88 G98 G00 G90 X90 Y80 Z-48 I-90 J70 B12 C10 D2 H100 L5 V100 F300 S1000 M03
; Cancels the canned cycle.
G80
; Positioning
G90 X0 Y0
; End of program
M30
```

## 9.15    G89. Boring cycle with withdrawal at work feedrate (G01)

This cycle bores at the point indicated until the final programmed coordinate is reached.

It is possible to program a dwell at the bottom of the machined hole.

Working in Cartesian coordinates, the basic structure of the block is as follows:

G89 G98/G99 X Y Z I K



### [ G98/G99 ] Withdrawal plane

G98    The tool withdraws to the Initial Plane, once the hole has been bored.

G99    The tool withdraws to the Reference Plane, once the hole has been bored.

### [ X/Y±5.5 ] Machining coordinates

These are optional and define the movement of the axes of the main plane to position the tool at the machining point.

This point can be programmed in Cartesian coordinates or in polar coordinates, and the coordinates may be absolute or incremental, according to whether the machine is operating in G90 or G91.

### [ Z±5.5 ] Reference plane

Defines the reference plane coordinate. It can be programmed in absolute coordinates or incremental coordinates, in which case it will be referred to the initial plane.

If not programmed, it assumes as reference plane the current position of the tool.

### [ I±5.5 ] Machining depth

Defines boring depth. It can be programmed in absolute coordinates or incremental coordinates and in this case will be referred to the reference plane.

### [ K5 ] Dwell

Defines the dwell, in hundredths of a second, after boring until the withdrawal begins. If not programmed, the CNC will take the value of "K0".

**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2x

### 9.15.1    Basic operation

1. If the spindle was previously running, it maintains the turning direction. If it was not in movement, it will start by turning clockwise (M03).

2. Rapid movement of the longitudinal axis from the initial plane to the reference plane.

3. Movement at the working feedrate (G01) of the longitudinal axis to the bottom of the machined hole, and boring.

4. Dwell, if parameter "K" has been programmed.

5. Withdrawal at working feedrate, of the longitudinal axis as far as the reference plane.

6. Withdrawal, at rapid feedrate (G00), of the longitudinal axis as far as the initial plane if G98 has been programmed.

#### *Programming example ·1·*

Let us suppose a work plane formed by the X and Y axis, Z being the longitudinal axis and the starting point X0 Y0 Z0.

```
; Tool selection.
T1 D1
M6
; Starting point
G0 G90 X0 Y0 Z0
; Canned cycle definition
G89 G98 G91 X250 Y350 Z-98 I-22 K20 F100 S500
; Cancels the canned cycle.
G80
; Positioning
G90 X0 Y0
; End of program
M30
```

**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2x

## 9.16 G210. Bore milling canned cycle

This cycle may be used to increase the diameter of a hole through a helical movement of the tool. Besides this, if the tool allows it, it is also possible to mill a hole without having to drill it first.

Working in Cartesian coordinates, the basic structure of the block is as follows:

G210 G98/G99 X Y Z D I J K B



### [ G98/G99 ] Withdrawal plane

G98    The tool withdraws to the Initial Plane, once the hole has been milled.

G99    The tool withdraws to the Reference Plane, once the hole has been milled.

### [X±5.5] Hole center coordinate along the abscissa axis.

It defines the hole center coordinate along the X axis. If not programmed, it will assume the current tool position on that axis.

### [ Y±5.5 ] Hole center coordinate along the ordinate axis

It defines the hole center coordinate along the Y axis. If not programmed, it will assume the current tool position on that axis.

### [ Z±5.5 ] Reference plane

Defines the reference plane coordinate. It may be programmed either in absolute or incremental coordinates, in which case it will be referred to the starting plane.

If not programmed, it assumes as reference plane the current position of the tool.

### [D5] Safety distance

Defines the distance between the reference plane and the surface of the part where the milling is to be done. If not programmed, it assumes 0.

### [ I±5.5 ] Machining depth

Defines the machining depth. It may be programmed either in absolute or incremental coordinates, in which case it will be referred to the reference plane.

If not programmed, the CNC issues the corresponding error.

### [J±5.5 ] Hole diameter

Defines the nominal diameter of the hole. The sign indicates the direction of the helical path associated with the machining of the hole (positive if clockwise and negative if counterclockwise).

If not programmed or programmed with a value smaller than the diameter of the active tool, the CNC issues the relevant error message.

### [ K5.5 ] Pre-drilling diameter

Starting with a hole previously drilled, this parameter defines the diameter of that hole. If not programmed or programmed with a 0 value, it means that no hole has been previously drilled.

**FAGOR**

FAGOR AUTOMATION

**CNC 8055
CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2x

·196·

The tool must meet the following conditions:

- The tool radius must be smaller than J/2.
- The tool radius must be equal to or larger than (J-K)/4.

If these two conditions are not met, the CNC issues the corresponding error.

**[ B±5.5 ] Penetration step**

It defines the penetration step when machining the hole.

- With a positive sign, it will mill the bottom of the hole.
- With a negative sign, it will not mill the bottom of the hole.

If not programmed or programmed with a 0 value, the CNC will display the corresponding error message.

### 9.16.1 Basic operation

1. Rapid movement to the center of the hole (X, Y).

2. Rapid movement to the reference plane (Z).

3. Rapid movement to the tangential entry coordinate along the longitudinal axis.

4. Tangential entry to the helical path of the drilling.

5. Helical movement, with the pitch given by parameter B and in the direction given by parameter J, down to the bottom of the hole.

6. Milling of the bottom of the hole (this step is only carried out if parameter B has a positive sign).

7. Tangential exit movement to the helical path of the drilling to the center of the hole.

8. Rapid movement to the reference plane (G99) or to the starting plane (G98).

## 9.17    G211. Inside thread milling cycle

This cycle may be used to make an inside thread through a helical movement of the tool.

Working in Cartesian coordinates, the basic structure of the block is as follows:

G211 G98/G99 X Y Z D I J K B C L A E Q

**[ G98/G99 ] Withdrawal plane**

G98    The tool withdraws to the Initial Plane, once the hole has been milled.

G99    The tool withdraws to the Reference Plane, once the hole has been milled.

**[X±5.5] Hole center coordinate along the abscissa axis.**

It defines the hole center coordinate along the X axis. If not programmed, it will assume the current tool position on that axis.

**[ Y±5.5 ] Hole center coordinate along the ordinate axis**

It defines the hole center coordinate along the Y axis. If not programmed, it will assume the current tool position on that axis.

**[ Z±5.5 ] Reference plane**

Defines the reference plane coordinate. It may be programmed either in absolute or incremental coordinates, in which case it will be referred to the starting plane.

If not programmed, it assumes as reference plane the current position of the tool.

**[D5] Safety distance**

Defines the distance between the reference plane and the surface of the part where the milling is to be done. If not programmed, it assumes 0.

**[ I±5.5 ] Machining depth**

Defines the threading depth. It may be programmed either in absolute or incremental coordinates, in which case it will be referred to the reference plane.

If not programmed, the CNC issues the corresponding error.

**[ J±5.5 ] Thread diameter**

Defines the nominal diameter of the thread The sign indicates the machining direction of the thread (positive if clockwise and negative if counterclockwise).

If not programmed, the CNC issues the corresponding error.

**FAGOR**

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2x

### [ K5.5 ] Thread depth

It defines the distance between the crest and the root of the thread. If not programmed, the CNC issues the corresponding error.

### [ B±5.5 ] Thread pitch

Defines the thread pitch.

- With a positive sign, the direction of the thread pitch is from the surface of the part to the bottom.
- With a negative sign, the direction of the thread pitch is from the bottom to the surface of the part.

If not programmed or programmed with a 0 value, the CNC will display the corresponding error message.

### [ C1 ] Type of tapping

Defines the type of tapping to be carried out. This parameter depends on the type of tool being used.

- When programming C= 0, the threading will be done in a single pass.
- When programming C=1, it will make one thread per each pass (single-edge cutter).
- When programming C=n (where n is the number of cutting edges of the cutter), it will make n threads per each pass.

If not programmed, a value of C=1 is assumed.



### [ L5.5 ] Finishing stock

It defines the finishing stock at the bottom of the thread. If not programmed, a value of 0 is assumed.

### [ A5.5 ] Maximum penetration step

Defines the maximum penetrating pass of the thread. If not programmed or programmed with a 0 value, it will run a single pass up to the finishing stock.

### [ E5.5 ] Approach distance

Approach distance to the thread entry. If not programmed, it will enter the thread from the center of the hole.

### [ Q±5.5 ] Thread entry angle

Angle (in degrees) of the segment formed by the center of the hole and the thread entry point with respect to the abscissa axis. If not programmed, a value of 0 is assumed.

### 9.17.1 Basic operation

1. Rapid movement to the center of the hole (X, Y).

2. Rapid movement to the reference plane (Z).

3. Rapid movement of the plane axes to the thread entry point (it only makes this movement if parameter E has been programmed).

4. Rapid movement to the thread entry point coordinate along the longitudinal axis.

5. Thread entry with a helical movement tangent to the first helical threading path.

6. Making the thread according to the value of parameter C.
   – If C=0:
      ·1· Helical movement, in the direction indicated in parameter J, to the bottom of the thread (the movement will only be one revolution).

      ·2· Helical thread exiting movement, tangent to the previous helical path. If parameter E has not been programmed, the exit point will correspond with the coordinates of the hole center.

      It must be borne in mind that in the exit tangent to the helical path, the exit point will exceed the coordinate of the bottom of the thread along the longitudinal axis.
   – If C=1:
      ·1· Helical movement, with the pitch and direction given in parameter J, to the bottom of the thread.

      ·2· Helical thread exiting movement, tangent to the previous helical path. If parameter E has not been programmed, the exit point will correspond with the coordinates of the hole center.

      It must be borne in mind that in the exit tangent to the helical path, the exit point will exceed the coordinate of the bottom of the thread along the longitudinal axis.
   – If C=n:
      ·1· Helical movement, with the pitch and direction given in parameter J, (the movement will be one revolution).

      ·2· Helical thread exiting movement, tangent to the previous helical path. If parameter E has not been programmed, the exit point will correspond with the coordinates of the hole center.

      ·3· Rapid movement to the thread entry point of the next threading path.

      ·4· Rapid movement to the Z coordinate of the thread entry point of the next threading path.

      ·5· Repetition of the previous 3 steps until reaching the bottom of the thread. It must be borne in mind that in the last helical exit, the exit point will exceed the coordinate of the bottom of the thread along the longitudinal axis.

7. Rapid movement to the center of the hole (X, Y).

8. Rapid movement to the thread entry coordinate along the longitudinal axis.

9. Repetition of steps 3 to 8 until reaching the depth of the finishing stock.

10. Repetition of steps 3 to 8 until reaching the bottom of the thread.

11. Rapid movement to the reference plane (G99) or to the starting plane (G98).

**9.**

**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

## 9.18 G212. Outside thread milling cycle

This cycle may be used to make an outside thread through a helical movement of the tool.

Working in Cartesian coordinates, the basic structure of the block is as follows:

G212 G98/G99 X Y Z D I J K B C L A E Q
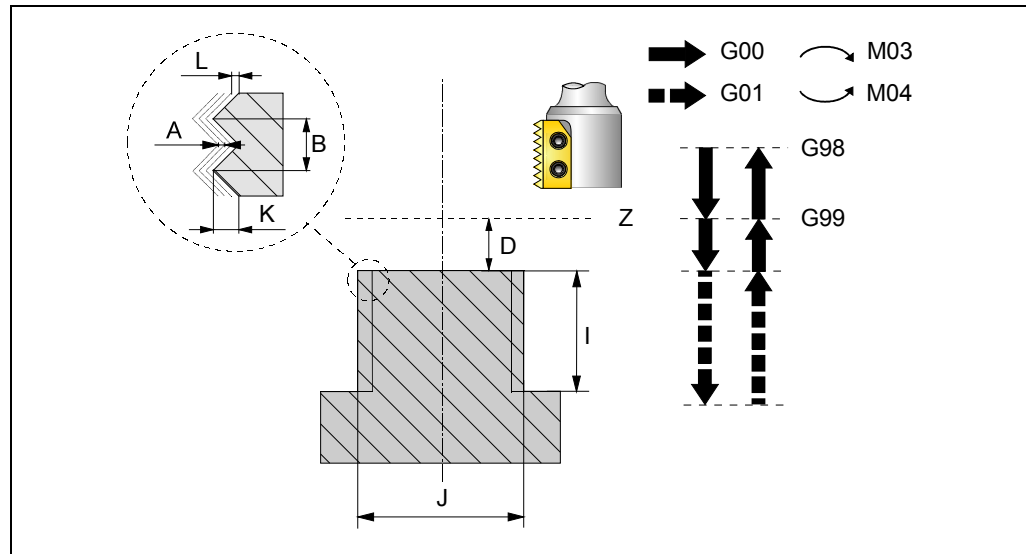
### [ G98/G99 ] Withdrawal plane

G98 The tool withdraws to the Initial Plane, once the hole has been milled.

G99 The tool withdraws to the Reference Plane, once the hole has been milled.

### [ X±5.5 ] Boss center coordinate along the abscissa axis.

It defines the boss center coordinate along the X axis. If not programmed, it will assume the current tool position on that axis.

### [ Y±5.5 ] Boss center coordinate along the ordinate axis.

It defines the boss center coordinate along the Y axis. If not programmed, it will assume the current tool position on that axis.

### [ Z±5.5 ] Reference plane

Defines the reference plane coordinate. It may be programmed either in absolute or incremental coordinates, in which case it will be referred to the starting plane.

If not programmed, it assumes as reference plane the current position of the tool.

### [D5] Safety distance

Defines the distance between the reference plane and the surface of the part where the milling is to be done. If not programmed, it assumes 0.

### [ I±5.5 ] Machining depth

Defines the threading depth. It may be programmed either in absolute or incremental coordinates, in which case it will be referred to the reference plane.

If not programmed, the CNC issues the corresponding error.

### [ J±5.5 ] Thread diameter

Defines the nominal diameter of the thread The sign indicates the machining direction of the thread (positive if clockwise and negative if counterclockwise).

If not programmed, the CNC issues the corresponding error.

### [ K5.5 ] Thread depth

It defines the distance between the crest and the root of the thread. If not programmed, the CNC issues the corresponding error.

**[ B±5.5 ] Thread pitch**

Defines the thread pitch.

- With a positive sign, the direction of the thread pitch is from the surface of the part to the bottom.

- With a negative sign, the direction of the thread pitch is from the bottom to the surface of the part.
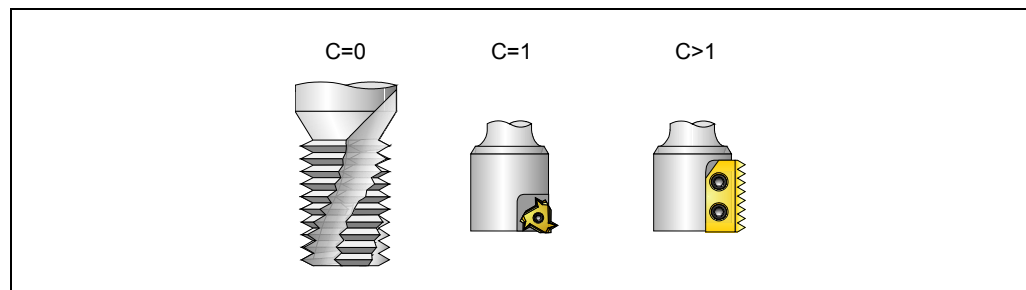
If not programmed or programmed with a 0 value, the CNC will display the corresponding error message.

**[ C1 ] Type of tapping**

Defines the type of tapping to be carried out. This parameter depends on the type of tool being used.

- When programming C= 0, the threading will be done in a single pass.

- When programming C=1, it will make one thread per each pass (single-edge cutter).

- When programming C=n (where n is the number of cutting edges of the cutter), it will make n threads per each pass.

If not programmed, a value of C=1 is assumed.



**[ L5.5 ] Finishing stock**

It defines the finishing stock at the bottom of the thread. If not programmed, a value of 0 is assumed.

**[ A5.5 ] Maximum penetration step**

Defines the maximum penetrating pass of the thread. If not programmed or programmed with a 0 value, it will run a single pass up to the finishing stock.

**[ E5.5 ] Approach distance**

Approach distance to the thread entry. If not programmed or programmed with a 0 value, the CNC will display the corresponding error message.

**[ Q±5.5 ] Thread entry angle**

Angle (in degrees) of the segment formed by the center of the hole and the thread entry point with respect to the abscissa axis. If not programmed, a value of 0 is assumed.

### 9.18.1 Basic operation

1. Rapid movement to the center of the hole (X, Y).

2. Rapid movement to the reference plane (Z).

3. Rapid movement of the plane axes to the thread entry point (it only makes this movement if parameter E has been programmed).

4. Rapid movement to the thread entry point coordinate along the longitudinal axis.

5. Rapid movement to the thread entry point (movement interpolated in 3 axes).

6. Thread entry with a helical movement tangent to the first helical threading path.

7. Making the thread according to the value of parameter C.

   – If C=0:

   ·1· Helical movement, in the direction indicated in parameter J, to the bottom of the thread (the movement will only be one revolution).

   ·2· Helical thread exiting movement, tangent to the previous helical path.

   It must be borne in mind that in the exit tangent to the helical path, the exit point will exceed the coordinate of the bottom of the thread along the longitudinal axis.

   – If C=1:

   ·1· Helical movement, with the pitch and direction given in parameter J, to the bottom of the thread.

   ·2· Helical thread exiting movement, tangent to the previous helical path.

   It must be borne in mind that in the exit tangent to the helical path, the exit point will exceed the coordinate of the bottom of the thread along the longitudinal axis.

   – If C=n:

   ·1· Helical movement, with the pitch and direction given in parameter J, (the movement will be one revolution).

   ·2· Helical thread exiting movement, tangent to the previous helical path to the thread entry point.

   ·3· Rapid movement to the Z coordinate of the thread entry point of the next threading path.

   ·4· Repetition of the previous 3 steps until reaching the bottom of the thread. It must be borne in mind that in the last helical exit, the exit point will exceed the coordinate of the bottom of the thread along the longitudinal axis.

8. Rapid movement to the reference plane (G99).

9. Repetition of steps 3 to 8 until reaching the depth of the finishing stock.

10. Repetition of steps 3 to 8 until reaching the bottom of the thread.

11. Rapid movement to the reference plane (G99) or to the starting plane (G98).

12. Rapid movement to the center of the hole (X, Y).

# MULTIPLE MACHINING

Multiple machining is defined as a series of functions which allow a machining operation to be repeated along a given path.

The programmer will select the type of machining, which can be a canned cycle or a subroutine (which must be programmed as a modal subroutine) defined by the user.

Machining paths are defined by the following functions:

    G60: Multiple machining in a straight line.

    G61: Multiple machining in rectangular pattern.

    G62: Multiple machining in a grid pattern.

    G63: Multiple machining in a circular pattern.

    G64: Multiple machining in an arc.

    G65: Machining programmed with an arc-chord.

These functions can be performed on any work plane and must be defined every time they are used, as they are not modal.

It is absolutely essential for the machining which it is required to repeat to be active. In other words, these functions will only make sense if they are under the influence of a canned cycle or under the influence of a modal subroutine.

To perform multiple machining, follow these steps:

**1.** Move the tool to the first point of the multiple machining operation.

**2.** Define the canned cycle or modal subroutine to be repeated at all the points.

**3.** Define the multiple operation to be performed.

All machining operations programmed with these functions will be done under the same working conditions (T,D,F,S) which were selected when defining the canned cycle or modal subroutine.

Once the multiple machining operation has been performed, the program will recover the history it had before starting this machining, even when the canned cycle or modal subroutine will remain active. Now feedrate F corresponds to the feedrate programmed for the canned cycle or modal subroutine.

Likewise, the tool will be positioned at the last point where the programmed machining operation was done.

If multiple machining of a modal subroutine is performed in the Single Block mode, this subroutine will be performed complete (not block by block) after each programmed movement.

A detailed explanation is given on the next page of multiple machining operations, assuming in each case, that the work plane is formed by X and Y axes.

## 10.1    G60: Multiple machining in a straight line

The programming format for this cycle is:

```
radius      X I      P Q R S T U V
            X K
            I K
```
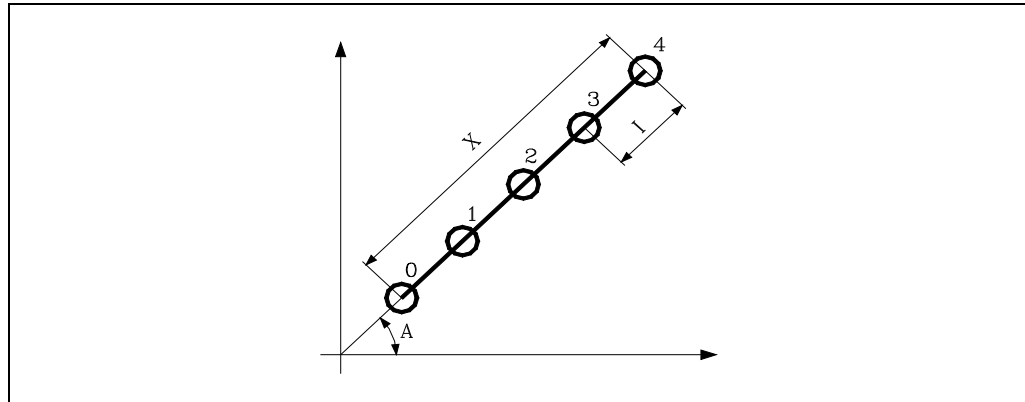


**[ A±5.5 ] Angle of the path**

Defines the angle that forms the machining path with the abscissa axis. It is expressed in degrees and if not programmed, the value A=0 will be taken.

**[ X5.5 ] Path length**

Defines the length of the machining path.

**[ I5.5 ] Step between machining operations.**

Defines the pass between machining operations.

**[ K5 ] Number of machining operations.**

Defines the number of total machining operations in the section, including the machining definition point.

Due to the fact that machining may be defined with any two points of the X I K group, the CNC allows the following definition combinations: XI, XK, IK.

Nevertheless, if format XI is defined, care should be taken to ensure that the number of machining operations is an integer number, otherwise the CNC will show the corresponding error code.

**[ P Q R S T U V ] Points where no drilling takes place**

These parameters are optional and are used to indicate at which points or between which of those programmed points it is not required to machine.

Thus, programming P7 indicates that it is not required to do machining at point 7, and programming Q10.013 indicates that machining is not required from point 10 to 13, or expressed in another way, that no machining is required at points 10, 11, 12 and 13.

When it is required to define a group of points (Q10.013), care should be taken to define the final point with three digits, as if Q10.13 is programmed, multiple machining understands Q10.130.

The programming order for these parameters is P Q R S T U V, it also being necessary to maintain the order in which the points assigned to these are numbered, i.e., the numbering order of the points assigned to Q must be greater than that assigned to P and less than that assigned to R.

Example:

| | |
|---|---|
| Proper programming | P5.006 Q12.015 R20.022 |
| Correct programming | P5.006 Q12.015 R20.022 |

If these parameters are not programmed, the CNC understands that it must perform machining at all the points along the programmed path.

### 10.1.1 Basic operation

1. Multiple machining calculates the next point of those programmed where it is wished to machine.

2. Rapid traverse (G00) to this point.

3. Multiple machining will perform the canned cycle or modal subroutine selected after this movement.

4. The CNC will repeat steps 1-2-3 until the programmed path has been completed.

   After completing multiple machining, the tool will be positioned at the last point along the programmed path where machining was performed.

Programming example assuming that the work plane is formed by the X and Y axes, that the Z axis is the longitudinal axis and that the starting point is X0 Y0 Z0:



```
; Canned cycle positioning and definition.
G81 G98 G00 G91 X200 Y300 Z-8 I-22 F100 S500
; Defines multiple machining.
G60 A30 X1200 I100 P2.003 Q6 R12
; Cancels the canned cycle.
G80
; Positioning.
G90 X0 Y0
; End of program.
M30
```

It is also possible to write the multiple machining definition block in the following ways:

```
G60 A30 X1200 K13 P2.003 Q6 R12
G60 A30 I100 K13 P2.003 Q6 R12
```

## 10.2    G61: Multiple machining in rectangular pattern

The programming format for this cycle is:

```
G61 A B  X I    Y J     P Q R S T U V
         X K    Y D
         I K    J D
```



**[ A±5.5 ] Angle of the path with respect to the abscissa axis**

Defines the angle that forms the machining path with the abscissa axis. It is expressed in degrees and if not programmed, the value A=0 will be taken.

**[ B±5.5 ] Angle between paths**

Defines the angle formed by the two machining paths. It is expressed in degrees and if not programmed, the value B=90 will be taken.

**[ X5.5 ] Length of the path along the abscissa axis**

Defines the length of the machining path according to the abscissa axis.

**[ I5.5 ] Machining pass along the abscissa axis**

Defines the pass between machining operations along the abscissa axis.

**[ K5 ] Number of machining operations along the abscissa axis**

Defines the total number of machining operations in the abscissa axis, including the machining definition point.

Due to the fact that machining may be defined according to the abscissa axis with any two points of the X I K group, the CNC allows the following definition combinations: XI, XK, IK.

Nevertheless, if format XI is defined, care should be taken to ensure that the number of machining operations is an integer number, otherwise the CNC will show the corresponding error code.

**[ Y5.5 ] Length of the path along the ordinate axis**

Defines the length of the machining path according to the ordinate axis.

**[ J5.5 ] Machining pass along the ordinate axis**

Defines the pass between machining operations according to the ordinate axis.

**[ D5 ] Number of machining operations along the ordinate axis**

Defines the total number of machining operations in the ordinate axis, including the machining definition point.

Due to the fact that machining may be defined according to the ordinate axis with any two points of the Y J D group, the CNC allows the following definition combinations: YJ, YD, JD.

Nevertheless, if format YJ is defined, care should be taken to ensure that the number of machining operations is an integer number, otherwise the CNC will show the corresponding error code.

**[ P Q R S T U V ] Points where no drilling takes place**

These parameters are optional and are used to indicate at which points or between which of those programmed points it is not required to machine.

Thus, programming P7 indicates that it is not required to do machining at point 7, and programming Q10.013 indicates that machining is not required from point 10 to 13, or expressed in another way, that no machining is required at points 10, 11, 12 and 13.

When it is required to define a group of points (Q10.013), care should be taken to define the final point with three digits, as if Q10.13 is programmed, multiple machining understands Q10.130.

The programming order for these parameters is P Q R S T U V, it also being necessary to maintain the order in which the points assigned to these are numbered, i.e., the numbering order of the points assigned to Q must be greater than that assigned to P and less than that assigned to R.

Example:

| | |
|---|---|
| Proper programming | P5.006 Q12.015 R20.022 |
| Correct programming | P5.006 Q12.015 R20.022 |

If these parameters are not programmed, the CNC understands that it must perform machining at all the points along the programmed path.

### 10.2.1 Basic operation

1. Multiple machining calculates the next point of those programmed where it is wished to machine.

2. Rapid traverse (G00) to this point.

3. Multiple machining will perform the canned cycle or modal subroutine selected after this movement.

4. The CNC will repeat steps 1-2-3 until the programmed path has been completed.

   After completing multiple machining, the tool will be positioned at the last point along the programmed path where machining was performed.

Programming example assuming that the work plane is formed by the X and Y axes, that the Z axis is the longitudinal axis and that the starting point is X0 Y0 Z0:



```
; Canned cycle positioning and definition.
G81 G98 G00 G91 X100 Y150 Z-8 I-22 F100 S500
; Defines multiple machining.
G61 X700 I100 Y180 J60 P2.005 Q9.011
; Cancels the canned cycle.
G80
; Positioning.
G90 X0 Y0
; End of program.
M30
```

It is also possible to write the multiple machining definition block in the following ways:

G61 X700 K8 J60 D4 P2.005 Q9.011
G61 I100 K8 Y180 D4 P2.005 Q9.011

## 10.3    G62: Multiple machining in grid pattern

The programming format for this cycle is:

```
G62 A B   X I     Y J      P Q R S T U V
          X K     Y D
          I K     J D
```

**[ A±5.5 ] Angle of the path with respect to the abscissa axis**

Defines the angle that forms the machining path with the abscissa axis. It is expressed in degrees and if not programmed, the value A=0 will be taken.

**[ B±5.5 ] Angle between paths**

Defines the angle formed by the two machining paths. It is expressed in degrees and if not programmed, the value B=90 will be taken.

**[ X5.5 ] Length of the path along the abscissa axis**

Defines the length of the machining path according to the abscissa axis.

**[ I5.5 ] Machining pass along the abscissa axis**

Defines the pass between machining operations along the abscissa axis.

**[ K5 ] Number of machining operations along the abscissa axis**

Defines the total number of machining operations in the abscissa axis, including the machining definition point.

Due to the fact that machining may be defined according to the abscissa axis with any two points of the X I K group, the CNC allows the following definition combinations: XI, XK, IK.

Nevertheless, if format XI is defined, care should be taken to ensure that the number of machining operations is an integer number, otherwise the CNC will show the corresponding error code.

**[ Y5.5 ] Length of the path along the ordinate axis**

Defines the length of the machining path according to the ordinate axis.

**[ J5.5 ] Machining pass along the ordinate axis**

Defines the pass between machining operations according to the ordinate axis.

**[ D5 ] Number of machining operations along the ordinate axis**

Defines the total number of machining operations in the ordinate axis, including the machining definition point.

Due to the fact that machining may be defined according to the ordinate axis with any two points of the Y J D group, the CNC allows the following definition combinations: YJ, YD, JD.

Nevertheless, if format YJ is defined, care should be taken to ensure that the number of machining operations is an integer number, otherwise the CNC will show the corresponding error code.

**[ P Q R S T U V ] Points where no drilling takes place**

These parameters are optional and are used to indicate at which points or between which of those programmed points it is not required to machine.

Thus, programming P7 indicates that it is not required to do machining at point 7, and programming Q10.013 indicates that machining is not required from point 10 to 13, or expressed in another way, that no machining is required at points 10, 11, 12 and 13.

When it is required to define a group of points (Q10.013), care should be taken to define the final point with three digits, as if Q10.13 is programmed, multiple machining understands Q10.130.

The programming order for these parameters is P Q R S T U V, it also being necessary to maintain the order in which the points assigned to these are numbered, i.e., the numbering order of the points assigned to Q must be greater than that assigned to P and less than that assigned to R.

Example:

Proper programming          P5.006 Q12.015 R20.022

Correct programming         P5.006 Q12.015 R20.022

If these parameters are not programmed, the CNC understands that it must perform machining at all the points along the programmed path.

### 10.3.1 Basic operation

1. Multiple machining calculates the next point of those programmed where it is wished to machine.

2. Rapid traverse (G00) to this point.

3. Multiple machining will perform the canned cycle or modal subroutine selected after this movement.

4. The CNC will repeat steps 1-2-3 until the programmed path has been completed.

   After completing multiple machining, the tool will be positioned at the last point along the programmed path where machining was performed.

Programming example assuming that the work plane is formed by the X and Y axes, that the Z axis is the longitudinal axis and that the starting point is X0 Y0 Z0:



```
; Canned cycle positioning and definition.
G81 G98 G00 G91 X100 Y150 Z-8 I-22 F100 S500
; Defines multiple machining.
G62 X700 I100 Y180 J60 P2.005 Q9.011 R15.019
; Cancels the canned cycle.
G80
; Positioning.
G90 X0 Y0
; End of program.
M30
```

It is also possible to write the multiple machining definition block in the following ways:

```
G62 X700 K8 J60 D4 P2.005 Q9.011 R15.019
G62 I100 K8 Y180 D4 P2.005 Q9.011 R15.019
```

## 10.4 G63: Multiple machining in a circular pattern

The programming format for this cycle is:

```
G63 X Y   I    C F P Q R S T U V
          K
```
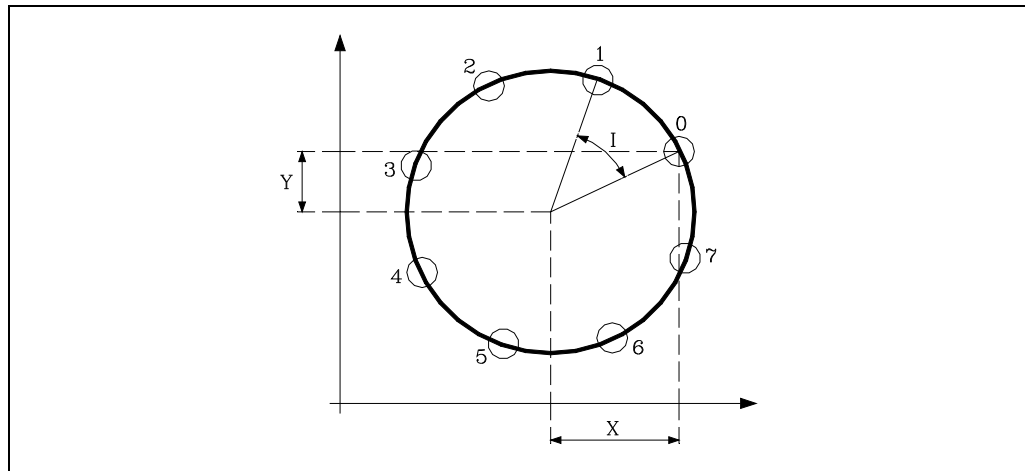
**[ X±5.5 ] Distance from the first machining point to the center along the abscissa axis**

Defines the distance from the starting point to the center along the abscissa axis.

**[ Y±5.5 ] Distance from the first machining point to the center along the ordinate axis**

Defines the distance from the starting point to the center along the ordinate axis.

With parameters X and Y the center of the circle is defined in the same way that I and J do this in circular interpolations (G02, G03).

**[ I±5.5 ] Angular pass between machining operations**

Defines the angular pass between machining operations. When moving from point to point in G00 or G01, the sign indicates the direction, "+" counterclockwise, "-" clockwise.

**[ K5 ] Total number of machining operations**

Defines the total number of machining operations along the circle, including the machining definition point.

It will be enough to program I or K in the multiple machining definition block. Nevertheless, if K is programmed in a multiple machining operation in which movement between points is made in G00 or G01, machining will be done counterclockwise.

**[ C 0/1/2/3 ] Type of move from point to point**

Indicates how movement is made between machining points. If not programmed, a value of C=0 is assumed.

| | |
|---|---|
| C=0: | Movement is made in rapid feedrate (G00). |
| C=1: | Movement is made in linear interpolation (G01). |
| C=2: | Movement is made in clockwise circular interpolation (G02) |
| C=3: | Movement is made in counterclockwise circular interpolation (G03) |

**[ F5.5 ] Feedrate to move from point to point**

Defines the feedrate that is used for moving from point to point. Obviously, it will only apply for "C" values other than zero. If it is not programmed, the value F0 will be taken, maximum feedrate selected by the "MAXFEED" axis machine parameter.

**[ P Q R S T U V ] Points where no drilling takes place**

These parameters are optional and are used to indicate at which points or between which of those programmed points it is not required to machine.

Thus, programming P7 indicates that it is not required to do machining at point 7, and programming Q10.013 indicates that machining is not required from point 10 to 13, or expressed in another way, that no machining is required at points 10, 11, 12 and 13.

When it is required to define a group of points (Q10.013), care should be taken to define the final point with three digits, as if Q10.13 is programmed, multiple machining understands Q10.130.

The programming order for these parameters is P Q R S T U V, it also being necessary to maintain the order in which the points assigned to these are numbered, i.e., the numbering order of the points assigned to Q must be greater than that assigned to P and less than that assigned to R.

Example:

| | |
|---|---|
| Proper programming | P5.006 Q12.015 R20.022 |
| Correct programming | P5.006 Q12.015 R20.022 |

If these parameters are not programmed, the CNC understands that it must perform machining at all the points along the programmed path.

### 10.4.1    Basic operation

1.  Multiple machining calculates the next point of those programmed where it is wished to machine.

2.  Movement at the feedrate programmed by "C" (G00, G01, G02 or G03) to this point.

3.  Multiple machining will perform the canned cycle or modal subroutine selected after this movement.

4.  The CNC will repeat steps 1-2-3 until the programmed path has been completed.

    After completing multiple machining, the tool will be positioned at the last point along the programmed path where machining was performed.

Programming example assuming that the work plane is formed by the X and Y axes, that the Z axis is the longitudinal axis and that the starting point is X0 Y0 Z0:



```
; Canned cycle positioning and definition.
G81 G98 G01 G91 X280 Y130 Z-8 I-22 F100 S500
; Defines multiple machining.
G63 X200 Y200 I30 C1 F200 P2.004 Q8
; Cancels the canned cycle.
G80
; Positioning.
G90 X0 Y0
; End of program.
M30
```

It is also possible to write the multiple machining definition block in the following ways:

G63 X200 Y200 K12 C1 F200 P2.004 Q8

## 10.5   G64: Multiple machining in an arc

The programming format for this cycle is:

G64 X Y B   I   C F P Q R S T U V
                K
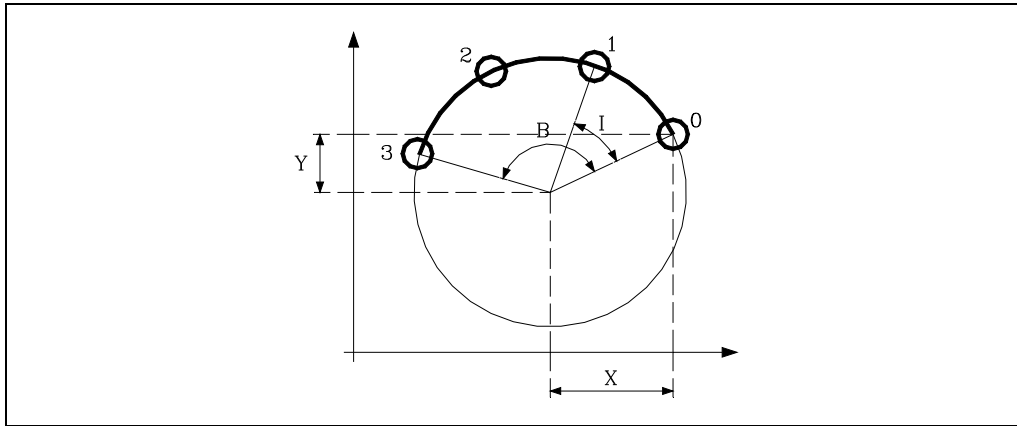


**[ X±5.5 ] Distance from the first machining point to the center along the abscissa axis**

Defines the distance from the starting point to the center along the abscissa axis.

**[ Y±5.5 ] Distance from the first machining point to the center along the ordinate axis**

Defines the distance from the starting point to the center along the ordinate axis.

With parameters X and Y the center of the circle is defined in the same way that I and J do this in circular interpolations (G02, G03).

**[ B5.5 ] Angular travel**

Defines the angular stroke of the machining path and is expressed in degrees.

**[ I±5.5 ] Angular pass between machining operations**

Defines the angular pass between machining operations. When moving from point to point in G00 or G01, the sign indicates the direction, "+" counterclockwise, "-" clockwise.

**[ K5 ] Total number of machining operations**

Defines the total number of machining operations along the circle, including the machining definition point.

It will be enough to program I or K in the multiple machining definition block. Nevertheless, if K is programmed in a multiple machining operation in which movement between points is made in G00 or G01, machining will be done counterclockwise.

**[ C 0/1/2/3 ] Type of move from point to point**

Indicates how movement is made between machining points. If not programmed, a value of C=0 is assumed.

| | |
|---|---|
| C=0: | Movement is made in rapid feedrate (G00). |
| C=1: | Movement is made in linear interpolation (G01). |
| C=2: | Movement is made in clockwise circular interpolation (G02) |
| C=3: | Movement is made in counterclockwise circular interpolation (G03) |

**[ F5.5 ] Feedrate to move from point to point**

Defines the feedrate that is used for moving from point to point. Obviously, it will only apply for "C" values other than zero. If it is not programmed, the value F0 will be taken, maximum feedrate selected by the "MAXFEED" axis machine parameter.

**[ P Q R S T U V ] Points where no drilling takes place**

These parameters are optional and are used to indicate at which points or between which of those programmed points it is not required to machine.

Thus, programming P7 indicates that it is not required to do machining at point 7, and programming Q10.013 indicates that machining is not required from point 10 to 13, or expressed in another way, that no machining is required at points 10, 11, 12 and 13.

When it is required to define a group of points (Q10.013), care should be taken to define the final point with three digits, as if Q10.13 is programmed, multiple machining understands Q10.130.

The programming order for these parameters is P Q R S T U V, it also being necessary to maintain the order in which the points assigned to these are numbered, i.e., the numbering order of the points assigned to Q must be greater than that assigned to P and less than that assigned to R.

Example:

| | |
|---|---|
| Proper programming | P5.006 Q12.015 R20.022 |
| Correct programming | P5.006 Q12.015 R20.022 |

If these parameters are not programmed, the CNC understands that it must perform machining at all the points along the programmed path.

**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

·**218**·

## 10.5.1 Basic operation

1. Multiple machining calculates the next point of those programmed where it is wished to machine.

2. Movement at the feedrate programmed by "C" (G00, G01, G02 or G03) to this point.

3. Multiple machining will perform the canned cycle or modal subroutine selected after this movement.

4. The CNC will repeat steps 1-2-3 until the programmed path has been completed.

   After completing multiple machining, the tool will be positioned at the last point along the programmed path where machining was performed.

Programming example assuming that the work plane is formed by the X and Y axes, that the Z axis is the longitudinal axis and that the starting point is X0 Y0 Z0:



```
; Canned cycle positioning and definition.
G81 G98 G01 G91 X280 Y130 Z-8 I-22 F100 S500
; Defines multiple machining.
G64 X200 Y200 B225 I45 C3 F200 P2
; Cancels the canned cycle.
G80
; Positioning.
G90 X0 Y0
; End of program.
M30
```

It is also possible to write the multiple machining definition block in the following ways:

G64 X200 Y200 B225 K6 C3 F200 P2

**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2x

## 10.6    G65: Machining programmed with an arc-chord

This function allows activated machining to be performed at a point programmed by means of an arc chord. Only one machining operation will be performed, its programming format being:

G65 X Y   A   C F
              I
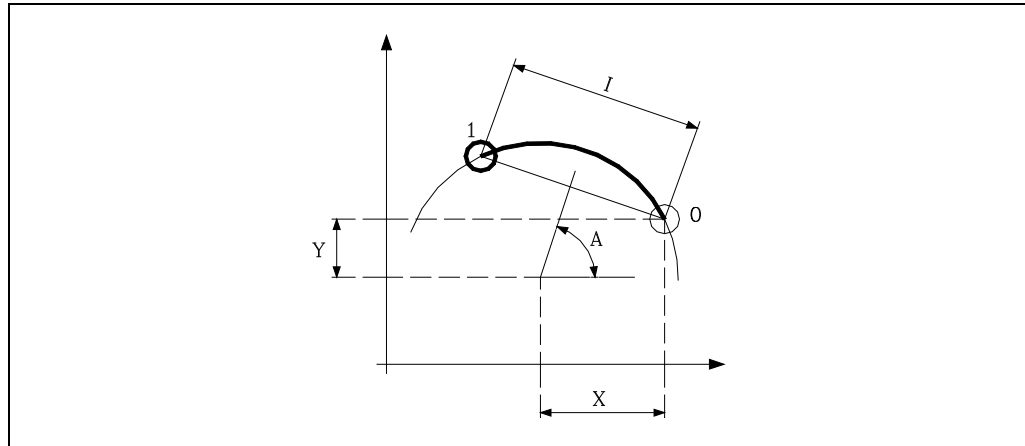


**[ X±5.5 ] Distance from the first machining point to the center along the abscissa axis**

Defines the distance from the starting point to the center along the abscissa axis.

**[ Y±5.5 ] Distance from the first machining point to the center along the ordinate axis**

Defines the distance from the starting point to the center along the ordinate axis.

With parameters X and Y the center of the circle is defined in the same way that I and J do this in circular interpolations (G02, G03).

**[ A±5.5 ] Angle of the chord**

Defines the angle formed by the perpendicular bisector of the chord with the abscissa axis and is expressed in degrees.

**[ I±5.5 ] Angular pass between machining operations**

Defines the chord length. When moving from point to point in G00 or G01, the sign indicates the direction, "+" counterclockwise, "-" clockwise.

**[ C0/1/2/3 ] Type of move from point to point**

Indicates how movement is made between machining points. If not programmed, a value of C=0 is assumed.

| | |
|---|---|
| C=0: | Movement is made in rapid feedrate (G00). |
| C=1: | Movement is made in linear interpolation (G01). |
| C=2: | Movement is made in clockwise circular interpolation (G02) |
| C=3: | Movement is made in counterclockwise circular interpolation (G03) |

**[ F5.5 ] Feedrate to move from point to point**

Defines the feedrate that is used for moving from point to point. Obviously, it will only apply for "C" values other than zero. If it is not programmed, the value F0 will be taken, maximum feedrate selected by the "MAXFEED" axis machine parameter.
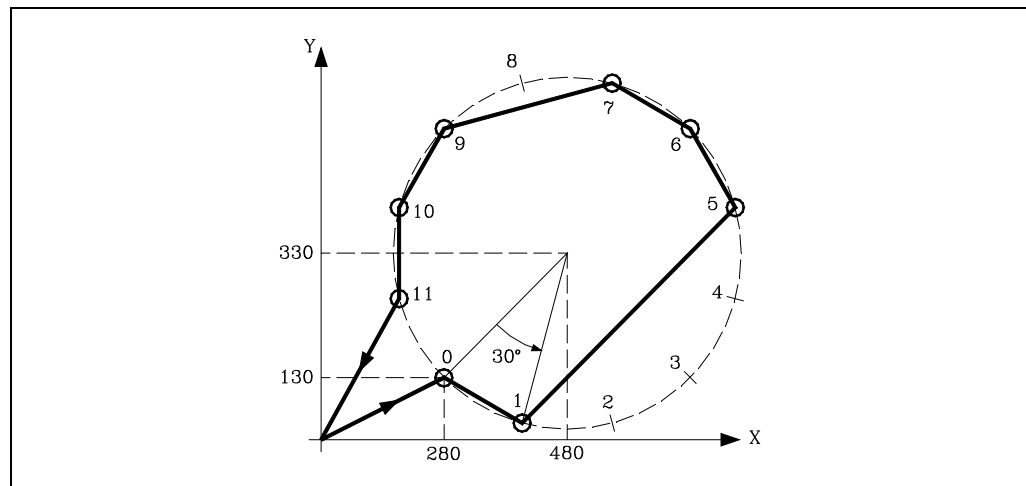
### 10.6.1    Basic operation

1. Multiple machining calculates the next point of those programmed where it is wished to machine.

2. Movement at the feedrate programmed by "C" (G00, G01, G02 or G03) to this point.

3. Multiple machining will perform the canned cycle or modal subroutine selected after this movement.

   After completing multiple machining, the tool will be positioned at the programmed point.

Programming example assuming that the work plane is formed by the X and Y axes, that the Z axis is the longitudinal axis and that the starting point is X0 Y0 Z0:
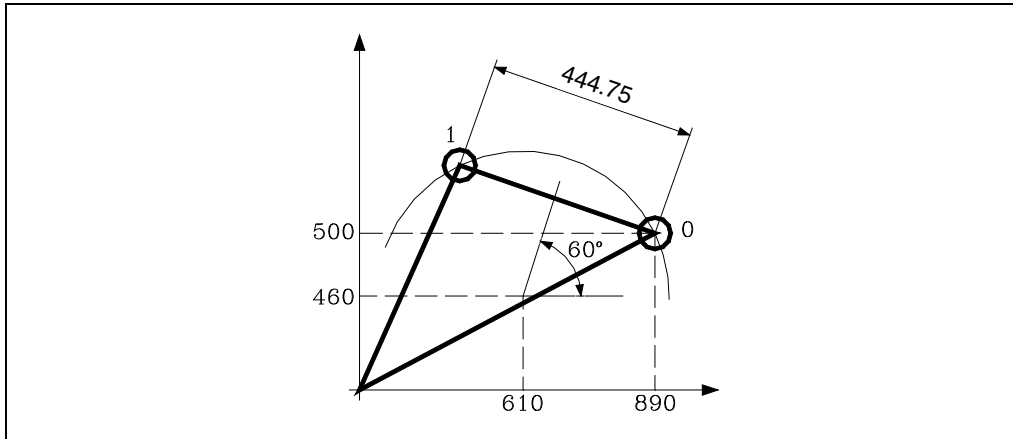


```
; Canned cycle positioning and definition.
G81 G98 G01 G91 X890 Y500 Z-8 I-22 F100 S500
; Defines multiple machining.
G65 X-280 Y-40 A60 C1 F200
; Cancels the canned cycle.
G80
; Positioning.
G90 X0 Y0
; End of program.
M30
```

It is also possible to write the multiple machining definition block in the following ways:

G65 X-280 Y-40 I444.75 C1 F200

**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

**10.**

FAGOR

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

# IRREGULAR POCKET CANNED CYCLE

## What is an irregular pocket with islands?

A pocket with islands is composed by an external contour or profile and a series of internal contours or profiles called islands.



(1) Outside contour or profile of the pocket.
(2) Inside contour or profile of the pocket.
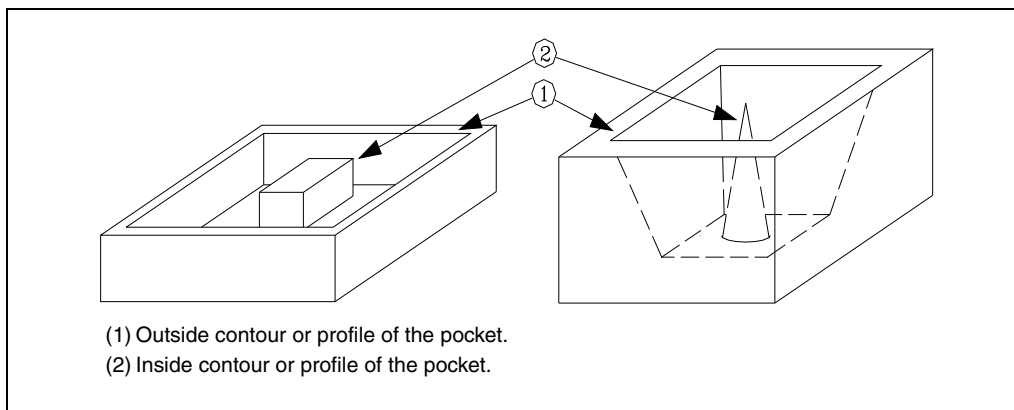
There are two types of pockets with islands, namely 2D and 3D pockets.

### *2D pocket:*

In a 2D pocket, all the walls of the outside profile and of the islands are vertical. To define the contours of a 2D pocket, the plane profile for all the contours must be defined.
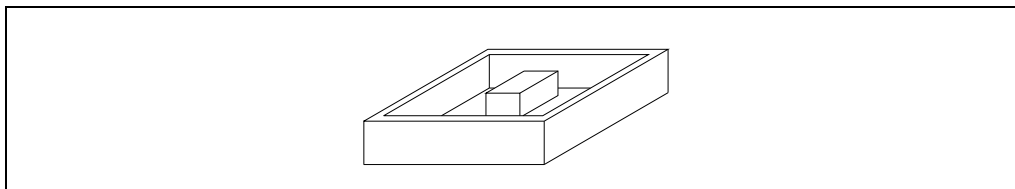


### *3D pocket:*

In a 3D pocket, one, several or all the walls of the outside pocket and/or of the islands are not vertical. To define the contours of a 2D pocket, the plane profile and the depth profile for all the contours must be defined (even if they are vertical).
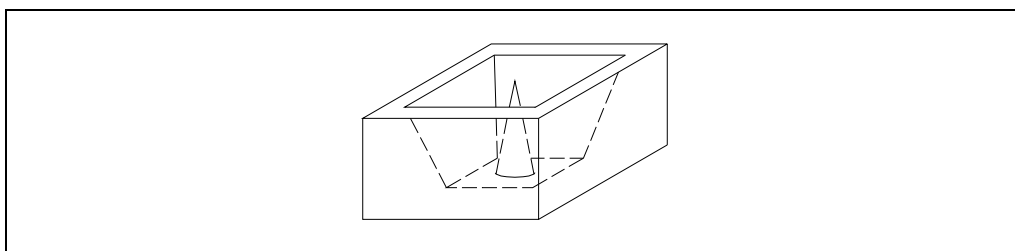
(A) Plane profile.                    (B) Depth profile.
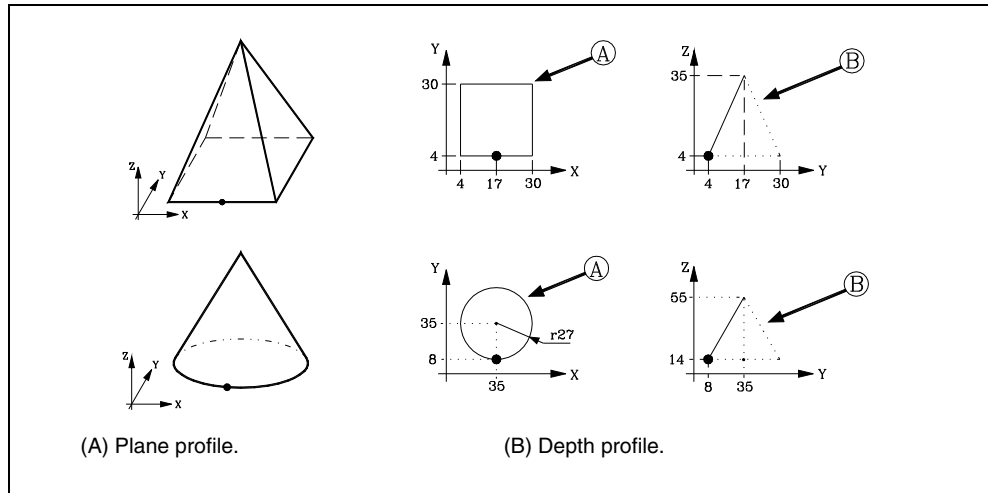
## Programming the irregular pocket canned cycle

The call function for a 2D or 3D irregular pocket canned cycle is G66. The machining of a pocket may consist of the following operations, each one is programmed with its relevant ·G· function.

| Function | Machining operation | Pocket |
|---|---|---|
| G69 G81 G82 G83 | Drilling operation, prior to machining. | 2D |
| G67 | Roughing operation. | 2D / 3D |
| G67 | Semi-finishing operation. | 3D |
| G68 | Finishing operation. | 2D / 3D |

G66 defines the operations that make up the machining of the pocket and where they are defined in the program. This function also indicates where the various profiles of the pocket are defined.

## 11.1    2D pockets

The G66 function is not modal, therefore it must be programmed whenever it is required to perform a 2D pocket. In a block defining an irregular pocket canned cycle, no other function can be programmed, its structure definition being:

G66 D H R I F K S E Q

**D (0-9999) / H (0-9999) Drilling operation**

Label number of the first block (D) and last block (H) defining the drilling operation.

- When not setting (H) only block (D) is executed.
- When not setting (D) there is no drilling operation.

**R (0-9999) / I (0-9999) Roughing operation**

Label number of the first block (R) and last block (I) defining the roughing operation.

- When not setting (I) only block (r) is executed.
- When not setting (R) there is no roughing operation.

**F (0-9999) / K (0-9999) Finishing operation**

Label number of the first block (F) and last block (K) defining the finishing operation.

- When not setting (K) only block (F) is executed.
- When not setting (F) there is no finishing operation.

**S (0-9999) / E (0-9999) Geometric description of the profiles**

Label number of the first block (S) and last block (E) defining the geometry of the profiles forming the pocket. Both parameters must be set.

**Q (0-999999) Program that contains the definition of the geometrical description of the profiles**

Number of the program containing the geometry definition, parameters S and E. If it is in the same program, (Q) need not be defined.

### *Programming example:*

```
; Initial positioning.
G00 G90 X100 Y200 Z50 F5000 T1 D2
M06
; Definition of irregular pocket canned cycle.
G66 D100 R200 I210 F300 S400 E500
; End of program.
M30

; Defines the drilling operation.
N100 G81...
; Roughing operation.
N200...
G67...
N210...
; Finishing operation.
N300 G68...
; Geometrical description.
N400 G0 G90 X300 Y50 Z3
...
...
N500 G2 G6 X300 Y50 I150 J0
```

## Basic operation

1. Drilling operation. Only if it has been programmed.

   After analyzing the geometry of the pocket with islands, the tool radius and the angle of the path programmed in the roughing operation, the CNC will calculate the coordinates of the point where the selected drilling operation must be performed.

2. Roughing operation. Only if it has been programmed.

   It consists of several surface milling passes, until the total depth programmed has been reached. On each surface milling pass, the steps below will be followed depending on the type of machining that has been programmed:

   **Case A:**

   When the machining paths are linear and maintain a certain angle with the abscissa axis.

   It first contours the external profile of the part. If the finishing operation has been selected on the cycle call, this contouring is performed leaving the finishing stock programmed for the finishing pass.



Next the milling operation, with the programmed feed and steps. If, while milling, an island is run into for the first time, it will be contoured.



After the contouring and the remaining times, the tool will pass over the island, withdrawing along the longitudinal axis, to the reference plane, and will continue machining once the island has been cleared.

**Case B:**

When the machining paths are concentric.

The roughing operation is carried out along paths concentric to the profile. The machining will be done as fast as possible avoiding (when possible) going over the islands.



3. Finishing operation. Only if it has been programmed.

This operation can be done on a single pass or on several, as well as following the profiles in the programmed direction or in the opposite.

The CNC will machine both the external profile and the islands, making tangential approaches and exits to these with a constant surface speed.

*Reference coordinates:*

In the pocket canned cycle with islands, there are four coordinates along the longitudinal axis (selected with G15), which, due to their importance, are discussed below:

1. Initial plane coordinate. This coordinate is given by the position which the tool occupies when the cycle is called.

2. Coordinate of the reference plane. This represents an approach coordinate to the part, and must be programmed in absolute coordinates.

3. Part surface coordinate. This is programmed in absolute coordinates and in the first profile definition block.

4. Machining depth coordinate. It must be programmed in absolute values.

## Conditions after finishing the cycle

Once the canned cycle has ended, the active feedrate will be the last one programmed, i.e. the one corresponding to the roughing operation or the finishing operation. The CNC will assume functions G00, G40 and G90.

### 11.1.1 Drilling operation

This operation is optional and in order to be executed it is necessary to also program a roughing operation.

It is mainly used when the tool programmed in the roughing operation does not machine along the longitudinal axis, allowing, by means of this operation, the access of this tool to the surface to be roughed off.

It will be programmed in a block which will need to bear a label number in order to indicate to the canned cycle the block where the drilling operation is defined.

```
; Definition of irregular pocket canned cycle.
G66 D100 R200 F300 S400 E500
; Defines the drilling operation.
N100 G81...
```

The drilling canned cycles that can be programmed are:

G69        Complex deep hole drilling

G81        Drilling canned cycle.

G82        Drilling cycle with dwell.

G83        Deep hole drilling canned cycle with constant peck (drilling step).

When defining the drilling operation, the corresponding definition parameters must be programmed together with the required function.

In a block of this type, only cycle definition parameters must be programmed, without defining XY positioning, as the canned cycle itself will calculate the coordinate of the point or points to be drilled according to the programmed profile and the roughing angle.

After the definition parameters, auxiliary F S T D M functions can be programmed, if so wished. No M function can be programmed if it has an associated subroutine.

It is possible to program the M06 function in this block (if it does not have an associated subroutine), to make the tool change. Otherwise, the CNC will show the corresponding error. If the M06 has an associated subroutine, the drilling tool "T" must be selected before calling the cycle.

```
N100 G69 G98 G91 Z-4 I-90 B1.5 C0.5 D2 H2 J4 K100 F500 S3000 M3
N120 G81 G99 G91 Z-5 I-30 F400 S2000 T3 D3 M3
N220 G82 G99 G91 Z-5 I-30 K100 F400 S2000 T2 D2 M6
N200 G83 G98 G91 Z-4 I-5 J6 T2 D4
```

### 11.1.2 Roughing operation

This is the main operation in the machining of an irregular pocket, and its programming is optional.

This operation will be performed keeping the square corner mode (G07) or round corner mode (G05) that is currently active. However, the canned cycle will assign the G07 format to the necessary movements.

It will be programmed in a block which will need to bear a label number in order to indicate to the canned cycle the block where the roughing operation is defined.

```
; Definition of irregular pocket canned cycle.
G66 D100 R200 F300 S400 E500
; Definition of the roughing operation.
N200 G67…
```

The function for the roughing operation is G67 and its programming format:

G67 A B C I R K V Q F S T D M

#### [ A±5.5 ] Angle of the path with respect to the abscissa axis

It defines the angle of the machining path with respect to the abscissa axis.



If parameter "A" is not programmed, the roughing operation is carried out following concentric paths. It will be machined as fast as possible since it does not have to go over the islands.

### [ B±5.5 ] Pass depth

Defines the machining pass along the longitudinal axis (depth of the roughing pass). It must be defined and it must have a value other than 0; otherwise, the roughing operation will be canceled.

- If programmed with a positive sign, all the roughing will be performed with the same machining pass, and the canned cycle calculates a pass equal to or smaller than the programmed pass.
- If programmed with a negative sign, all the roughing will be performed with the programmed pass, and the canned cycle will adjust the last pass to obtain the total programmed depth.



### [ C5.5 ] Milling pass

Defines the milling pass in roughing along the main plane, the entire pocket being performed with the given pass, and the canned cycle adjusts the last milling pass.



If not programmed or programmed with a 0 value, it assumes a value of 3/4 of the diameter of the selected tool. If programmed with a value greater than the tool diameter, the CNC issues the relevant error message.

### [ I±5.5 ] Pocket depth

Defines the total depth of the pocket and is programmed in absolute coordinates. It must be programmed.

### [ R±5.5 ] Reference plane

Defines the reference plane coordinate and is programmed in absolute coordinates. It must be programmed.



### [ K1 ] Type of profile intersection

Defines the type of profile intersection to be used.

K=0     Basic profile intersection.

K=1     Advanced profile intersection.

If not programmed, a value of 0 is assumed. Both types of intersection are described later on.

### [ V5.5 ] Penetration feedrate

Defines the tool penetrating feedrate.

If not programmed or programmed with a 0 value, it assumes 50% of the feedrate in the plane (F).

**[ Q5.5 ] Penetrating angle**

Optional. Tool penetration angle.
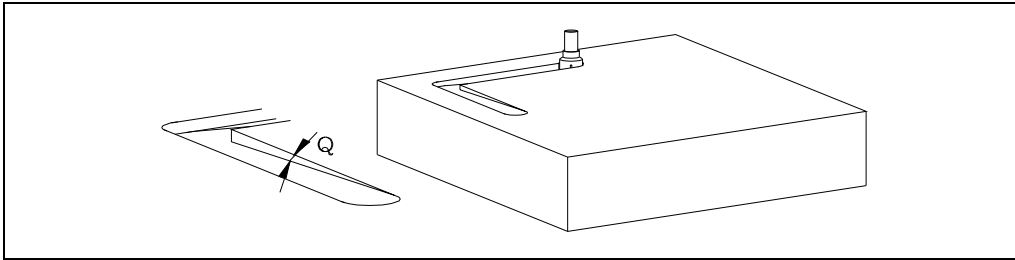


If not programmed or programming the value of 90, it means that the penetration is vertical. When programming a value lower than 0 or greater than 90, it issues the error "wrong parameter value in canned cycle".

**[ F5.5 ] Machining feedrate**

Optional. It sets the machining feedrate in the plane.

**[ S5.5 ] Spindle speed**

Optional. It sets the spindle speed.

**[ T4 ] Tool number**

Defines the tool used for the roughing operation. It must be programmed.

**[ D4 ] Tool offset**

Optional. Defines the tool offset number.

**[ M ] Auxiliary (miscellaneous) functions**

Optional. Up to 7 miscellaneous M functions can be programmed.

This operation allows M06 with an associated subroutine to be defined, and the tool change is performed before beginning the roughing operation.

### 11.1.3 Finishing operation

This operation is optional.

It will be programmed in a block that will need to bear a label number in order to indicate to the canned cycle the block where the finishing operation is defined.

```
; Definition of irregular pocket canned cycle.
G66 D100 R200 F300 S400 E500
; Defines the finishing operation.
N300 G68…
```

The function for the finishing operation is G68 and its programming format:

G68 B L Q I R K V F S T D M

**[ B±5.5 ] Pass depth**

Defines the machining pass along the longitudinal axis (depth of the finishing pass).

- If it is programmed with a value of 0, the CNC will perform a single finishing pass with the total depth of the pocket.

- If programmed with a positive sign, all the roughing will be performed with the same machining pass, and the canned cycle calculates a pass equal to or lower than the programmed pass.

- If programmed with a negative sign, all the roughing will be performed with the programmed pass, and the canned cycle will adjust the last pass to obtain the total programmed depth.
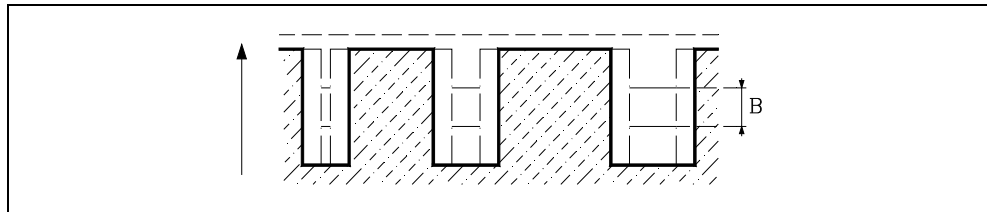


**[ L±5.5 ] Lateral finishing stock**

Defines the value of the finishing stock left on the side walls of the pocket before the finishing operation.
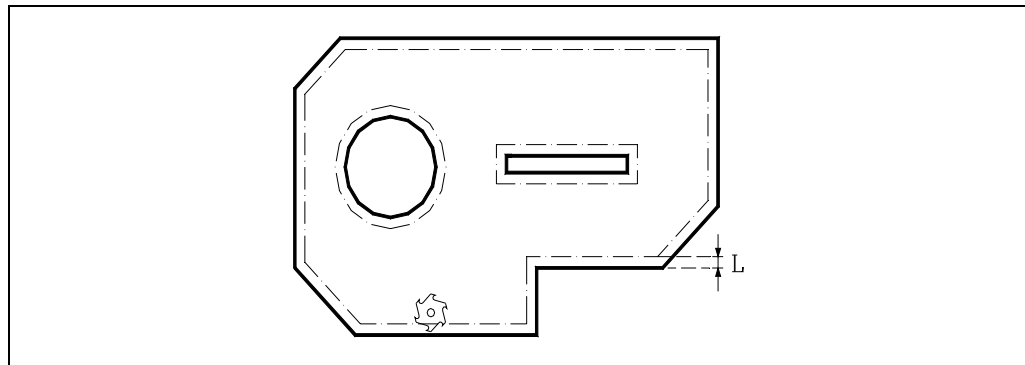


- If programmed with a positive value, the finishing pass will be carried out in square corner (G07).

- If a negative value is programmed, the finishing pass will be carried out in G5 (round corner).

- If not programmed or programmed with a 0 value, the cycle does not run the finishing pass.

**[ Q 0/1/2 ] Direction of the finishing pass**

Indicates the direction of the finishing pass on the outside profile. The finishing pass on the islands is always run in the opposite direction.

| | |
|---|---|
| Q = 0 | The finishing pass is carried out in the same direction as the outside profile was programmed. |
| Q = 1 | The finishing pass is carried out in the opposite direction to the one programmed. |
| Q = 2 | Reserved. |

Any other value will generate the corresponding error message. If parameter "Q" is not programmed, the cycle assumes Q0.

**[ I±5.5 ] Pocket depth**

Defines the total depth of the pocket and is programmed in absolute coordinates.

- If the island has a roughing operation, it is not necessary to define this parameter since it has been programmed in that operation. However, if programmed in both operations, the canned cycle will assume the particular depth indicated for each operation.

- If the island has no roughing operation, it is necessary to define this parameter.

**[ R±5.5 ] Reference plane**

Defines the reference plane coordinate and is programmed in absolute coordinates.

- If the island has a roughing operation, it is not necessary to define this parameter since it has been programmed in that operation. However, if programmed in both operations, the canned cycle will assume the particular depth indicated for each operation.

- If the island has no roughing operation, it is necessary to define this parameter.



**[ K1 ] Type of profile intersection**

Defines the type of profile intersection to be used.

    K=0          Basic profile intersection.

    K=1          Advanced profile intersection.

If the island has a roughing operation, it is not necessary to define this parameter since it has been programmed in that operation. However, if programmed in both operations, the canned cycle will assume the one defined for the roughing operation.

If no roughing operation has been defined and this parameter is not programmed, the canned cycle will assume a K0 value. Both types of intersection are described later on.

**[ V5.5 ] Penetration feedrate**

Defines the tool penetrating feedrate.

If not programmed or programmed with a 0 value, it assumes 50% of the feedrate in the plane (F).

**[ F5.5 ] Machining feedrate**

Optional. It sets the machining feedrate in the plane.

**[ S5.5 ] Spindle speed**

Optional. It sets the spindle speed.

**[ T4 ] Tool number**

Defines the tool used for the roughing operation. It must be programmed.

**[ D4 ] Tool offset**

Optional. Defines the tool offset number.

**[ M ] Auxiliary (miscellaneous) functions**

Optional. Up to 7 miscellaneous M functions can be programmed.

This operation allows M06 with an associated subroutine to be defined, and the tool change is performed before beginning the roughing operation.

### 11.1.4 Profile programming syntax

When outside and inside profiles of an irregular pocket are programmed the following programming rules must be followed: The canned cycle will verify all these geometry rules before beginning to make the pocket adapting the profile of the pocket to them and displaying the error message when necessary.

- All types of programmed profiles must be closed. The following examples cause a geometry error.
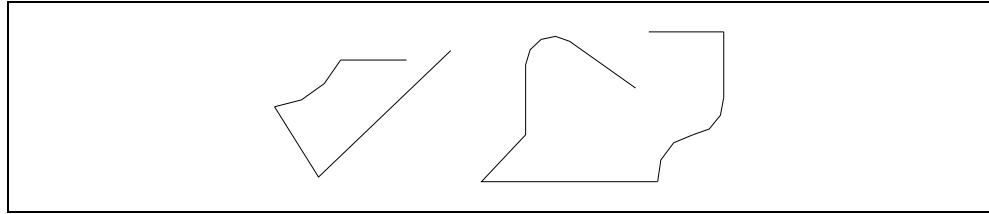


- No profile must intersect itself. The following examples cause a geometry error.
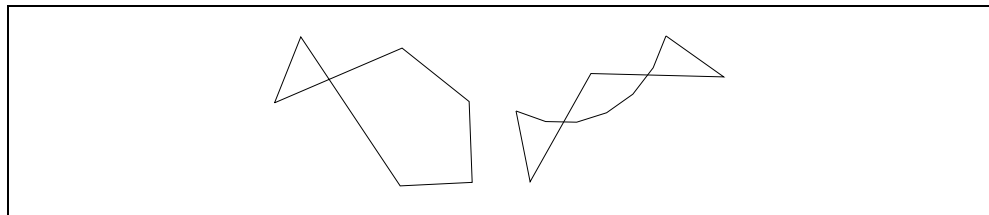


- When more than one outside profile has been programmed, the canned cycle assumes the one occupying the largest surface.
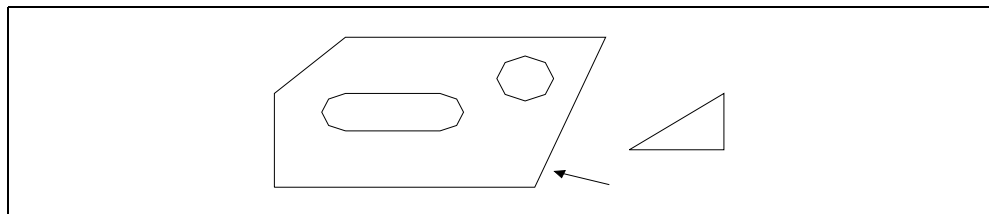


- It is not required to program inside profiles. Should these be programmed, they must be partially or totally internal with respect to the outside profile. Some examples are given below.
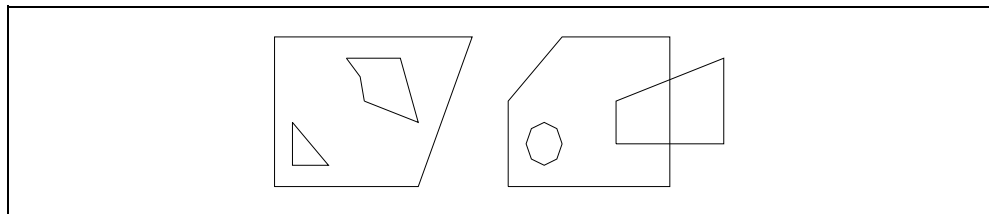


- An internal profile totally contained within another internal profile cannot be programmed. In this case, only the most external profile will be considered.
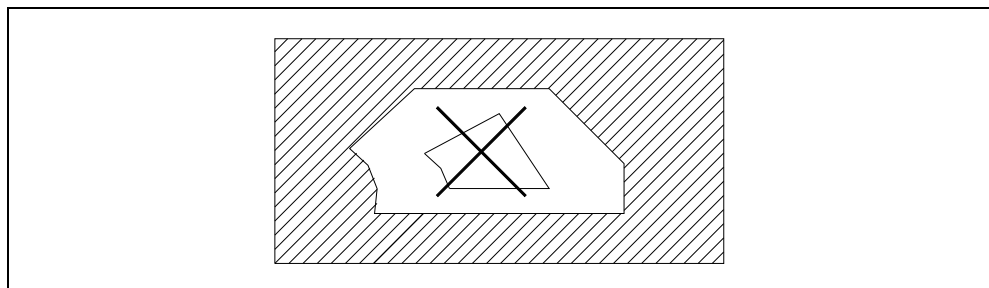
### 11.1.5   Profile intersection

In order to facilitate the programming of profiles, the canned cycle allows the profiles to intersect one another and the external profile.

The two available types of intersection can be selected by parameter "K"

## Basic profile intersection (K=0)

When selecting this type, the following profile intersecting rules are to be followed:

- The intersection of islands generates a new inside profile which is their Boolean union.
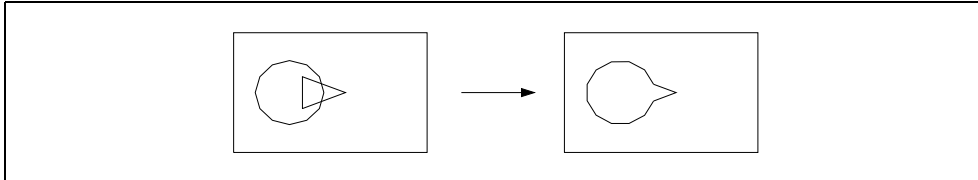
- The intersection between an internal and an external profile generates a new external profile as a result of the difference between the external and the internal profiles.
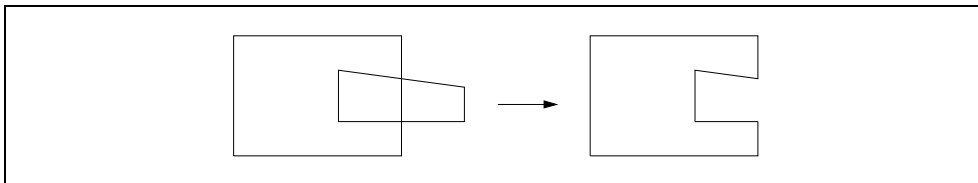
- If there is an inside profile which has an intersection with another inside profile and with the external profile, the canned cycle first makes the intersection between the inside profiles and then the intersection of these with the external profile.
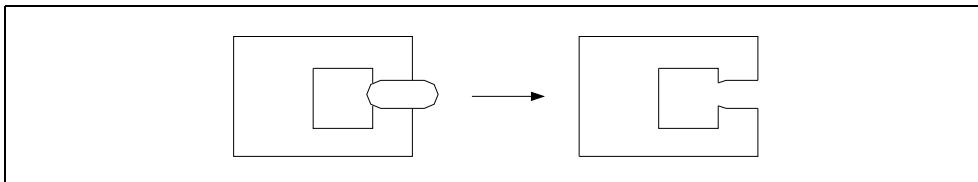
- As a result of the intersection of the inside profiles with the outside one, a single pocket will be obtained which corresponds to the outside profile having the largest surface. The rest will be ignored.
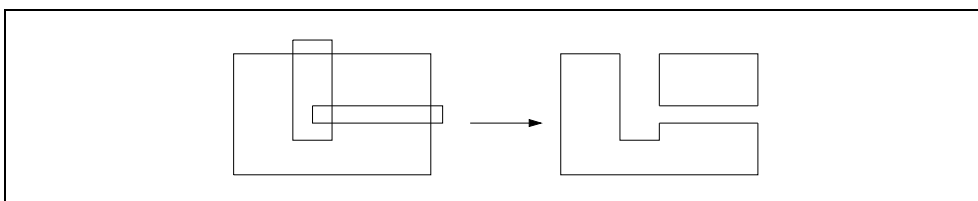
- If the finishing operation has been programmed, the profile of the resulting pocket must comply with all the tool compensation rules, since if a profile is programmed which cannot be machined by the programmed finishing tool, the CNC will show the corresponding error.

# Advanced profile intersection (K=1)

When selecting this type, the following profile intersecting rules are to be followed:

**1.** The initial point of each contour determines the section to be selected.

In a profile intersection, each contour is divided into several lines that could be grouped as:

- Lines external to the other contour.
- Lines internal to the other contour.

This type of profile intersection selects in each contour the group of lines where the profile defining point is included.

The following example shows the explained selection process. The solid lines indicate the lines external to the other contour and the dashes indicate the internal lines. The initial point of each contour is indicated with an "x".
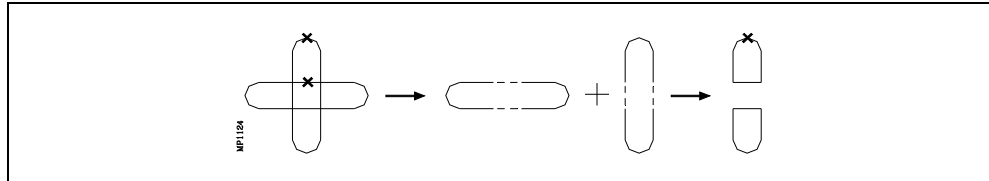


Examples of profile intersections:
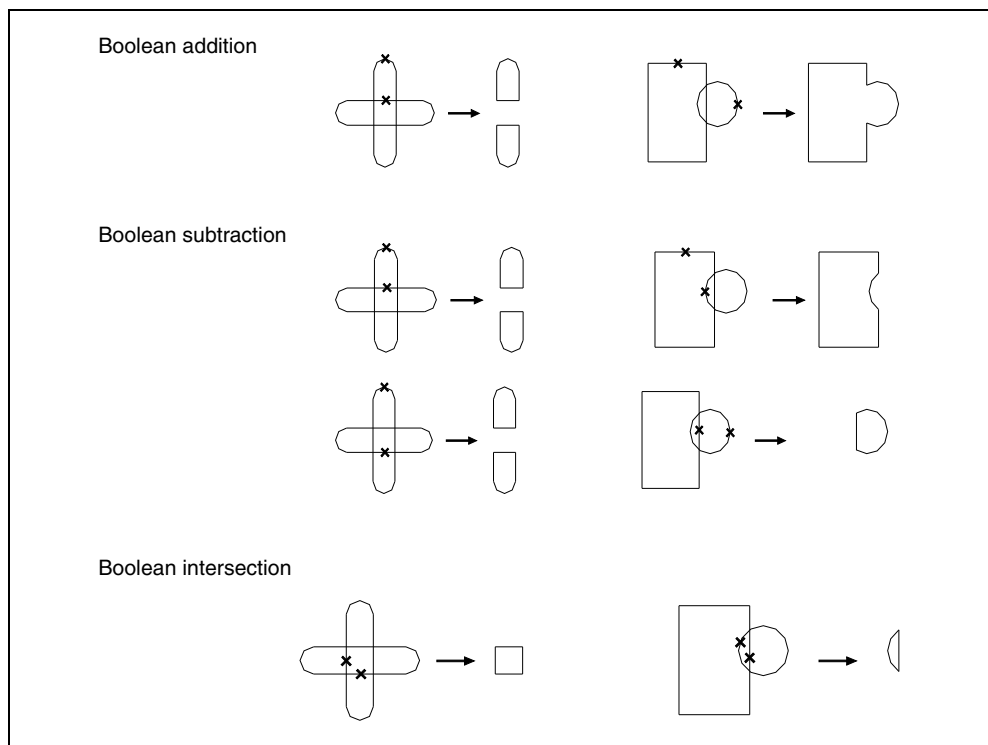
2. The programming sequence for the different profiles is determinant when having an intersection of more than 3 profiles.

The profile intersection process is performed according to the order in which the profiles have been programmed. This way, the result of the intersection between the first two will be intersected with the third one and so forth.

The initial point of the resulting profiles always coincides with the initial point that defined the first profile.
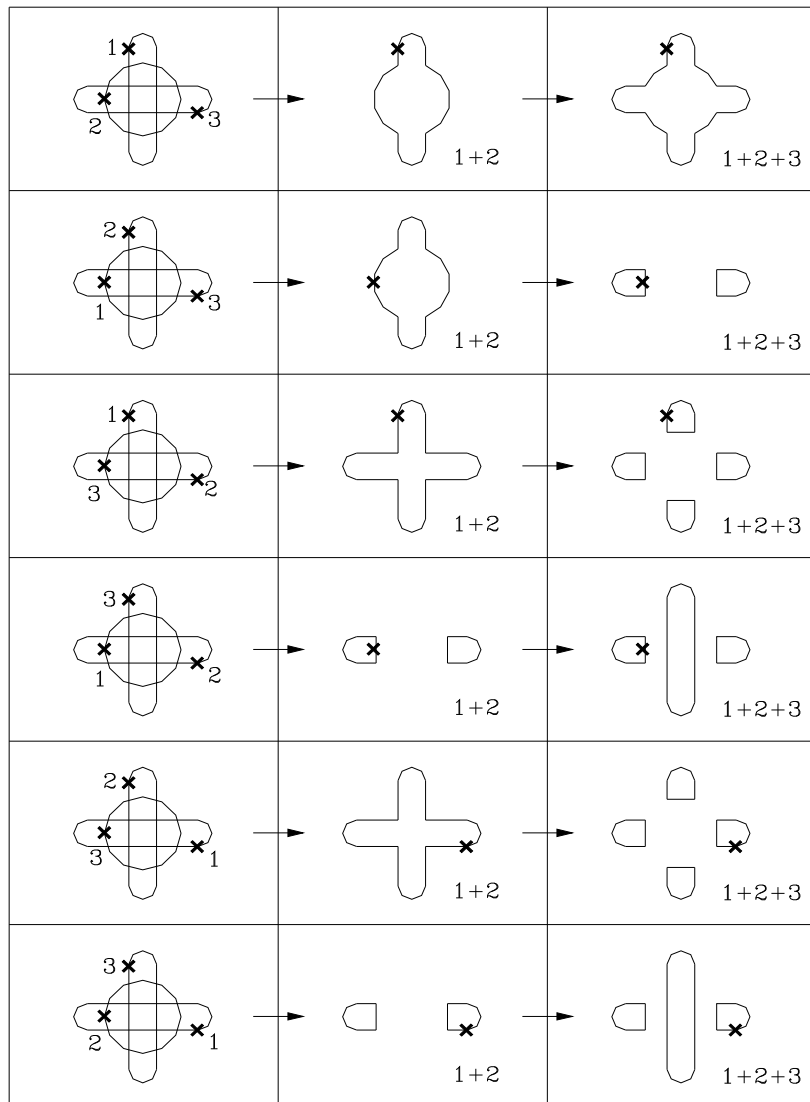


## Resulting profile

Once the profiles of the pocket and islands have been obtained, the canned cycle calculates the remaining profiles according to the radius of the roughing tool and the programmed finishing stock.

It may occur that in this process intersections are obtained which do not appear among the programmed profiles.
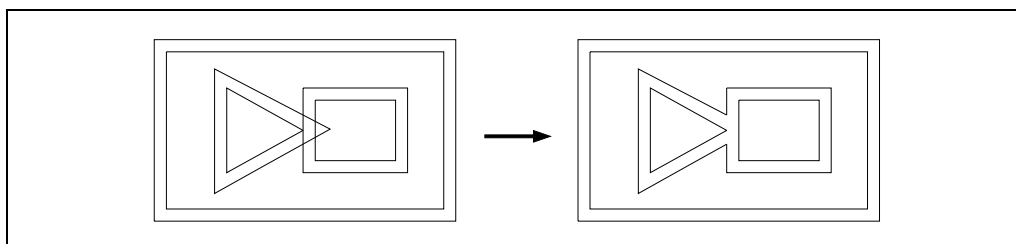


If there is an area in which the roughing tool cannot pass, when the intersection is made between the offset of the profiles, several pockets will be obtained as a result, all of which will be machined.

### 11.1.6 Profile programming syntax

The outside profile and the inside profiles or islands which are programmed must be defined by simple geometrical elements such as straight lines or arcs.

The first definition block (where the external profile starts) and the last (where the last profile defined ends) must be provided with the block label number. These label numbers will indicate the beginning and end of the profile geometry definition for the profiles that make up the pocket.

```
; Definition of irregular pocket canned cycle.
G66 D100 R200 F300 S400 E500
; Geometrical description.
N400 G0 G90 X300 Y50 Z3
...
N500 G2 G6 X300 Y50 I150 J0
```

The profile programming syntax must follow these rules:

- The external profile must begin in the first definition block of the geometric description of the part profiles. This block will be assigned a label number in order to indicate canned cycle G66 the beginning of the geometric description.

- The part surface coordinate will be programmed in this block.

- All the internal profiles which are required may be programmed, one after the other. Each of these must commence with a block containing the G00 function (indicating the beginning of the profile).

> **i** *Care must be taken to program G01, G02 or G03 in the block following the definition of the beginning, as G00 is modal, thus preventing the CNC from interpreting the following blocks as the beginnings of a new profile.*

- Once the definition of the profiles has been completed, a label number must be assigned to the last block programmed, in order to indicate the canned cycle G66 the end of the geometric description.

```
G0 G17 G90 X-350 Y0 Z50
; Definition of irregular pocket canned cycle.
G66 D100 R200 F300 S400 E500
G0 G90 X0 Y0 Z50
M30

; Defines the first profile.
N400 G0 G90 X-260 Y-190 Z4.5
--- --- --- ---
; Defines another profile.
G0 X230 Y170
G1 --- ---
--- --- --- ---
; Defines another profile.
G0 X-120 Y90
G2 --- ---
--- --- --- ---
; End of geometric description.
N500 G1 X-120 Y90
```

- Profiles are described as programmed paths, it being possible to include corner rounding, chamfers, etc., following the syntax rules defined for this purpose.

- The profile description must not contain: mirror images, scaling factor changes, pattern rotation, zero offsets, etc.

- It must not contain high level blocks such as jumps, calls to subroutines or parametric programming.

- It must not contain other canned cycles.

In addition to the G00 function, which has a special meaning, the irregular pocket canned cycle allows the use of the following functions for the definition of profiles.

| | |
|---|---|
| G01 | Linear interpolation. |
| G02 | Clockwise circular interpolation. |
| G03 | Counter-clockwise circular interpolation. |
| G06 | Circle center in absolute coordinates. |
| G08 | Arc tangent to previous path. |
| G09 | Arc defined by three points. |
| G36 | Corner rounding, radius blend. |
| G39 | Chamfer. |
| G53 | Programming with respect to machine zero. |
| G70 | Programming in inches. |
| G71 | Programming in millimeters. |
| G90 | Absolute programming. |
| G91 | Incremental programming. |
| G93 | Polar origin preset. |

### 11.1.7 Errors

The CNC will issue the following errors:

**ERROR 1023**          **G67. Tool radius too large.**

When selecting a wrong roughing tool.

**ERROR 1024**          **G68. Tool radius too large.**

When selecting a wrong finishing tool.

**ERROR 1025**          **A tool with no radius has been programmed**

When using a tool with "0" radius while machining a pocket.

**ERROR 1026**          **A step has been programmed that is larger than the tool diameter**

When parameter "C" of the roughing operation is greater than the diameter of the roughing tool.

**ERROR 1041**          **A parameter required in the canned cycle has not been programmed.**

It comes up in the following instances:

- When parameters "I" and "R" have not been programmed in the roughing operation.
- When not using a roughing operation and not programming the "I" and "R" parameters for the finishing operation.

**ERROR 1042**          **Wrong parameter value in the canned cycle**

It comes up in the following instances:

- When parameter "Q" of the finishing operation has the wrong value.
- When parameter "B" of the finishing operation has a "0" value.
- When parameter "J" of the finishing operation has been programmed with a value greater than the finishing tool radius.

**ERROR 1044**          **The plane profile intersects itself in an irregular pocket with islands.**

It comes up when any of the plane profiles of the programmed contours intersects itself.

**ERROR 1046**          **Wrong tool position prior to the canned cycle**

It comes up when calling the G66 cycle if the tool is positioned between the reference plane and the depth coordinate (bottom) of any of the operations.

**ERROR 1047**          **Open plane profile in an irregular pocket with islands**

It comes up when any of the programmed contours does not begin and end at the same point. It may be because G1 has not been programmed after the beginning, with G0, on any of the profiles.

**ERROR 1048**          **Part surface coordinate not programmed in pocket with islands**

It comes up when the first point of the geometry does not include the pocket top coordinate.

**ERROR 1049**          **Wrong reference plane coordinate for the canned cycle.**

It comes up when the coordinate of the reference plane is located between the part's "top" and "bottom" in any of the operations.

**ERROR 1084**          **Arc programmed wrong'**

It comes up when any of the paths programmed in the geometry definition of the pocket is wrong.

**11.**

**IRREGULAR POCKET CANNED CYCLE**
2D pockets

**FAGOR**

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2x

**ERROR 1227**          **Wrong profile intersection in a pocket with islands.**

It comes up in the following instances:

• When two plane profiles have a common section (drawing on the left).

• When the initial points of two profiles in the main plane coincide (drawing on the right).

### 11.1.8 Programming examples

*Programming example ·1·*

Programming example, without automatic tool changer
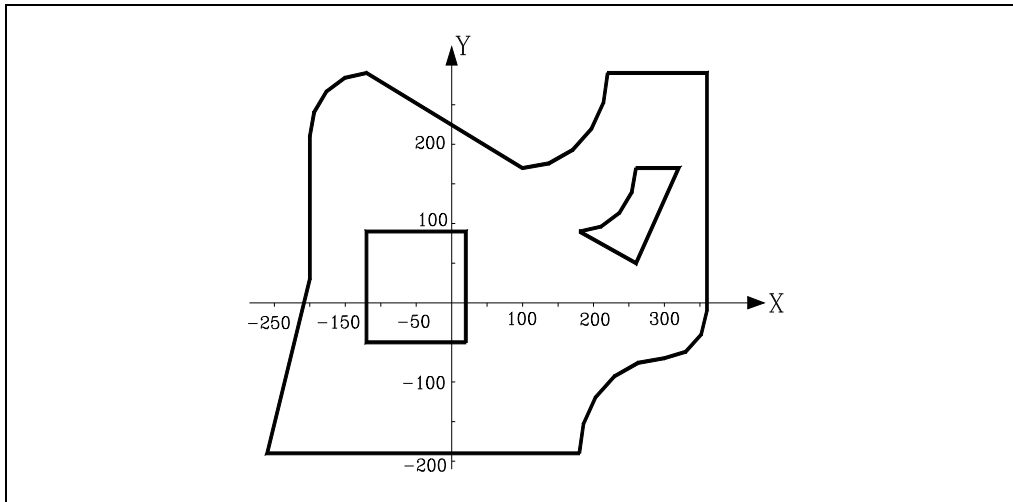


```
; Tool dimensions.
(TOR1=5, TOI1=0, TOL1=25, TOK1=0)
(TOR2=3, TOI2=0, TOL2=20, TOK2=0)
(TOR3=5, TOI3=0, TOL3=25, TOK3=0)

; Initial positioning and programming of a pocket with islands.
G0 G17 G43 G90 X0 Y0 Z25 S800
G66 D100 R200 F300 S400 E500
M30

; Definition of drilling operation.
N100 G81 Z5 I-40 T3 D3 M6

; Definition of roughing operation.
N200 G67 B20 C8 I-40 R5 K0 V100 F500 T1 D1 M6

; Definition of finishing operation.
N300 G68 B0 L0.5 Q0 V100 F300 T2 D2 M6

; Definition of pocket profiles.
N400 G0 G90 X-260 Y-190 Z0
; External contour.
G1 X-200 Y30
X-200 Y210
G2 G6 X-120 Y290 I-120 J210
G1 X100 Y170
G3 G6 X220 Y290 I100 J290
G1 X360 Y290
G1 X360 Y-10
G2 G6 X300 Y-70 I300 J-10
G3 G6 X180 Y-190 I300 J-190
G1 X-260 Y-190

; Contour of the first island.
G0 X230 Y170
G1 X290 Y170
G1 X230 Y50
G1 X150 Y90
G3 G6 X230 Y170 I150 J170

; Contour of the second island.
G0 X-120 Y90
G1 X20 Y90
G1 X20 Y-50
G1 X-120 Y-50

; End of contour definition.
N500 G1 X-120 Y90
```

**FAGOR**

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2x

**Programming example ·2·**

Programming example, with automatic tool changer. The "x" of the figure indicates the initial points of each profile:
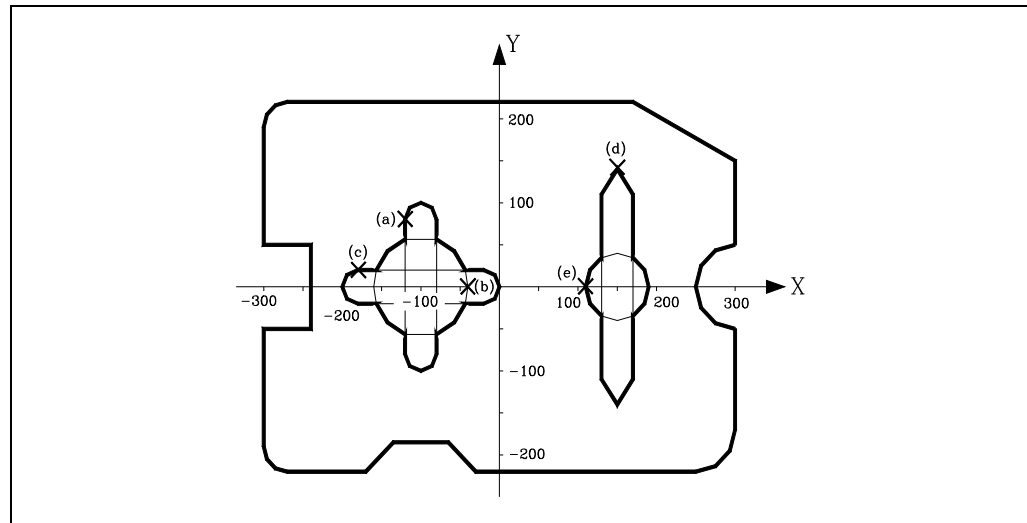
```
; Tool dimensions.
(TOR1=9, TOI1=0, TOL1=25, TOK1=0)
(TOR2=3.6, TOI2=0, TOL2=20, TOK2=0)
(TOR3=9, TOI3=0, TOL3=25, TOK3=0)

; Initial positioning and programming of a pocket with islands.
G0 G17 G43 G90 X0 Y0 Z25 S800
G66 D100 R200 F300 S400 E500
M30

; Definition of drilling operation.
N100 G81 Z5 I-40 T3 D3 M6

; Definition of roughing operation.
N200 G67 B10 C5 I-40 R5 K1 V100 F500 T1 D1 M6

; Definition of finishing operation.
N300 G68 B0 L0.5 Q1 V100 F300 T2 D2 M6

; Definition of pocket profiles.
N400 G0 G90 X-300 Y50 Z3

; External contour.
G1 Y190
G2 G6 X-270 Y220 I-270 J190
G1 X170
X300 Y150
Y50
G3 G6 X300 Y-50 I300 J0
G1 G36 R50 Y-220
X -30
G39 R50 X-100 Y-150
X-170 Y-220
X-270
G2 G6 X-300 Y-190 I-270 J-190
G1 Y-50
X-240
Y50
X-300
```

```
; Contour of the first island.
G0 X-120 Y80
G2 G6 X-80 Y80 I-100 J80; (Contour a)
G1 Y-80
G2 G6 X-120 Y-80 I-100 J-80
G1 Y80
G0 X-40 Y0; (Contour b)
G2 G6 X-40 Y0 I-100 J0
G0 X-180 Y20; (Contour c)
G1 X-20
G2 G6 X-20 Y-20 I-20 J0
G1 X-180
G2 G6 X-180 Y20 I-180 J0

; Contour of the second island.
G0 X150 Y140
G1 X170 Y110; (Contour d)
Y -110
X150 Y-140
X130 Y-110
Y110
X150 Y140
G0 X110 Y0; (Contour e)

; End of contour definition.
N500 G2 G6 X110 Y0 I150 J0
```

## 11.2    3D pockets

The cycle calling function G66 is not modal; therefore, it must be programmed every time a 3D pocket is to be executed.

In a block defining an irregular pocket canned cycle, no other function can be programmed, its structure definition being:

G66 R I C J F K S E

### R (0-9999) / I (0-9999) Roughing operation

Label number of the first block (R) and last block (I) defining the roughing operation.

- When not setting (I) only block (r) is executed.
- When not setting (R) there is no roughing operation.

### C (0-9999) / J (0-9999) Semi-finishing operation

Label number of the first block (C) and last block (J) defining the semi-finishing operation.

- When not setting "J" only block "c" is executed.
- When not setting (C) there is no semi-finishing operation.

### F (0-9999) / K (0-9999) Finishing operation

Label number of the first block (F) and last block (K) defining the finishing operation.

- When not setting (K) only block (F) is executed.
- When not setting (F) there is no finishing operation.

### S (0-9999) / E (0-9999) Geometric description of the profiles

Label number of the first block (S) and last block (E) defining the geometry of the profiles forming the pocket. Both parameters must be set.

### *Programming example:*

```
; Initial positioning.
G00 G90 X100 Y200 Z50 F5000 T1 D2
M06
; Definition of irregular pocket canned cycle.
G66 R100 C200 J210 F300 S400 E500
; End of program.
M30

; Roughing operation.
N100 G67...
; Semi-finishing operation.
N200...
G67...
N210...
; Finishing operation.
N300 G68...
; Geometrical description.
N400 G0 G90 X300 Y50 Z3
…
…
N500 G2 G6 X300 Y50 I150 J0
```

## Basic operation

1. Roughing operation. Only if it has been programmed.

   It consists of several surface milling passes, until the total depth programmed has been reached. On each surface milling pass, the steps below will be followed depending on the type of machining that has been programmed:

   **Case A:**

   When the machining paths are linear and maintain a certain angle with the abscissa axis.

   It first contours the external profile of the part. If the finishing operation has been selected on the cycle call, this contouring is performed leaving the finishing stock programmed for the finishing pass.



   Next the milling operation, with the programmed feed and steps. If, while milling, an island is run into for the first time, it will be contoured.



   After the contouring and the remaining times, the tool will pass over the island, withdrawing along the longitudinal axis, to the reference plane, and will continue machining once the island has been cleared.

**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2x

**Case B:**

When the machining paths are concentric.

The roughing operation is carried out along paths concentric to the profile. The machining will be done as fast as possible avoiding (when possible) going over the islands.
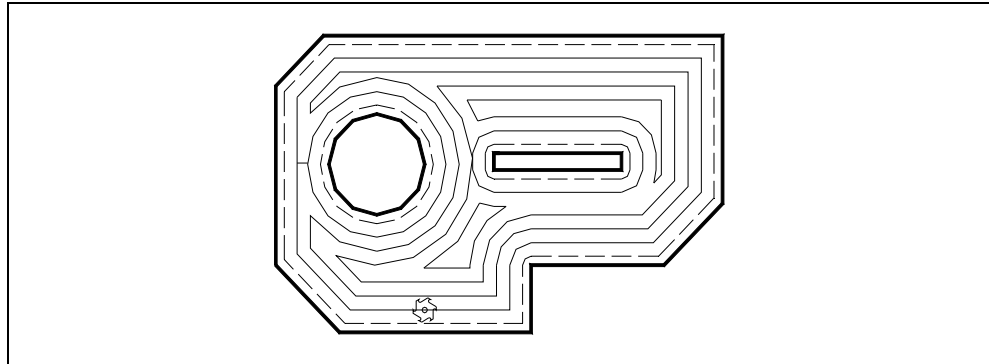


**2.** Semi-finishing operation. Only if it has been programmed.

After the roughing, some ridges appear on the external profile as well as on the islands themselves as shown in the illustration below:
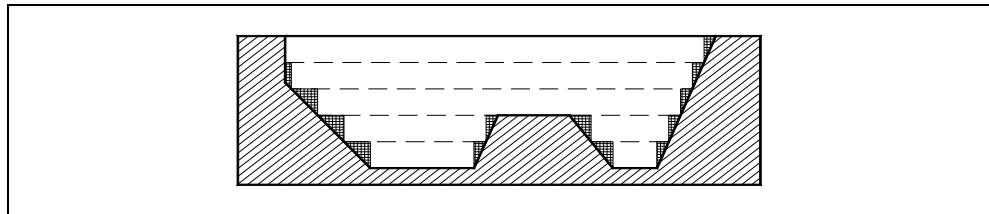


With the semi-finishing operation, it is possible to minimize these ridges by running several contouring passes at different depths.
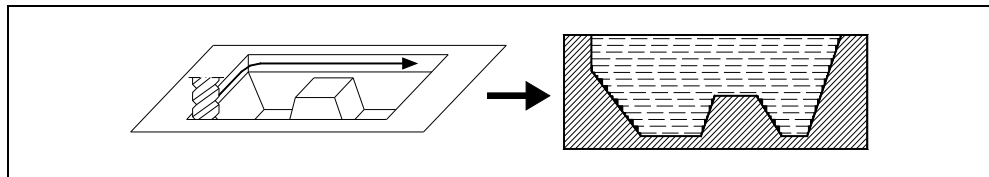


**3.** Finishing operation. Only if it has been programmed.

It runs consecutive finishing passes in 3D. Either inward or outward machining direction may be selected or both may be alternated.
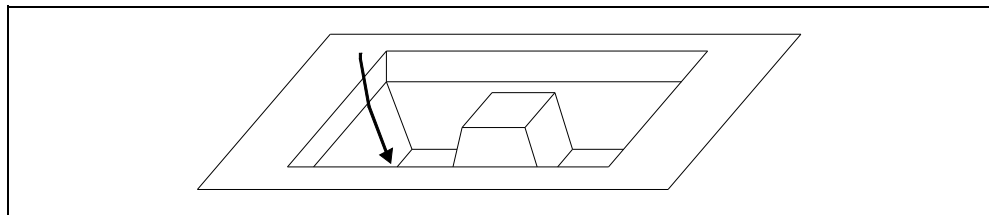


The CNC will machine both the external profile and the islands, making tangential approaches and exits to these with a constant surface speed.

### *Conditions after finishing the cycle:*

Once the canned cycle has ended, the active feedrate will be the last one programmed, i.e. the one corresponding to the roughing operation or the finishing operation. The CNC will assume functions G00, G40 and G90.

### *Reference coordinates:*

In the pocket canned cycle with islands, there are four coordinates along the longitudinal axis (selected with G15), which, due to their importance, are discussed below:

1. Initial plane coordinate. This coordinate is given by the position which the tool occupies when the cycle is called.
2. Coordinate of the reference plane. This represents an approach coordinate to the part, and must be programmed in absolute coordinates.
3. Part surface coordinate. This is programmed in absolute coordinates and in the first profile definition block.
4. Machining depth coordinate. It must be programmed in absolute values.

### 11.2.1    Roughing operation

This is the main operation in the machining of an irregular pocket, and its programming is optional.

It will be programmed in a block which will need to bear a label number in order to indicate to the canned cycle the block where the roughing operation is defined.

```
; Definition of irregular pocket canned cycle.
G66 R100 C200 F300 S400 E500
; Definition of the roughing operation.
N100 G67...
```

The function for the roughing operation is G67 and it cannot be executed independently from the G66.

Their programming formats are:

G67 A B C I R V F S T D M

**[ A±5.5 ] Angle of the path with respect to the abscissa axis**

It defines the angle of the machining path with respect to the abscissa axis.



If parameter "A" is not programmed, the roughing operation is carried out following concentric paths. It will be machined as fast as possible since it does not have to go over the islands.



**[ B±5.5 ] Pass depth**

Defines the machining pass along the longitudinal axis (depth of the roughing pass). It must be defined and it must have a value other than 0; otherwise, the roughing operation will be canceled.



- If programmed with a positive sign, all the roughing will be performed with the same machining pass, and the canned cycle calculates a pass equal to or smaller than the programmed pass.

- If programmed with a negative sign, all the roughing will be performed with the programmed pass, and the canned cycle will adjust the last pass to obtain the total programmed depth.

If done with B(+), the ridges will appear only on the pocket walls; but, if done with B(-), they could also show up above the islands.



B(+)                      B(−)

### [ C5.5 ] Milling pass

Defines the milling pass in roughing along the main plane, the entire pocket being performed with the given pass, and the canned cycle adjusts the last milling pass.



If not programmed or programmed with a 0 value, it assumes a value of 3/4 of the diameter of the selected tool.

If programmed with a value greater than the tool diameter, the CNC issues the relevant error message.

### [ I±5.5 ] Pocket depth

Defines the total depth of the pocket and is programmed in absolute coordinates. It must be programmed.

### [ R±5.5 ] Reference plane

Defines the reference plane coordinate and is programmed in absolute coordinates. It must be programmed.



### [ V5.5 ] Penetration feedrate

Defines the tool penetrating feedrate.

If not programmed or programmed with a 0 value, it assumes 50% of the feedrate in the plane (F).

### [ F5.5 ] Machining feedrate

Optional. It sets the machining feedrate in the plane.

### [ S5.5 ] Spindle speed

Optional. It sets the spindle speed.

**[ T4 ] Tool number**

Defines the tool used for the roughing operation. It must be programmed.

**[ D4 ] Tool offset**

Optional. Defines the tool offset number.

**[ M ] Auxiliary (miscellaneous) functions**

Optional. Up to 7 miscellaneous M functions can be programmed.

This operation allows M06 with an associated subroutine to be defined, and the tool change is performed before beginning the roughing operation.

### 11.2.2    Semi-finishing operation

This operation is optional.

It will be programmed in a block which will need to bear a label number in order to indicate to the canned cycle the block where the roughing operation is defined.

```
; Definition of irregular pocket canned cycle.
G66 R100 C200 F300 S400 E500
; Definition of the semi-finish operation.
N200 G67…
```

The function for the semi-finishing operation is G67 and it cannot be executed independently from the G66.

Both the roughing and the semi-finishing operations are defined with G67; but, in different blocks. It is function G66 who indicates which is which by means of parameters "R" and "C".

Their programming formats are:

G67 B I R V F S T D M

**[ B±5.5 ] Pass depth**

Defines the machining step along the longitudinal axis (semi-finishing pass). It must be programmed and with a value other than "0". Otherwise, the semi-finishing operation will be canceled.
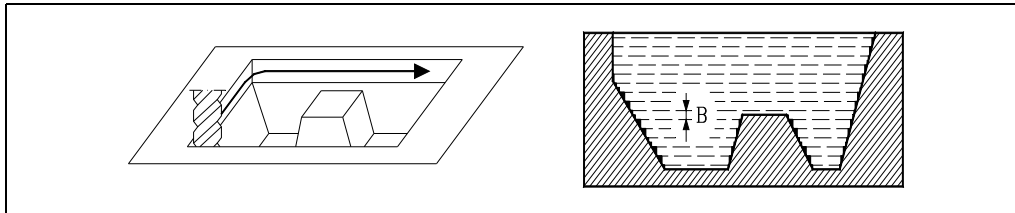


- If programmed with a positive sign, the whole semi-finish operation will be carried out with the same machining pass and the canned cycle will calculate a pass equal or smaller than the one programmed.

- If programmed with a negative sign, the whole semi-finish operation will be  run with the programmed pass. The canned cycle will adjust the last pass to obtain the total programmed depth.

**[ I±5.5 ] Pocket depth**

Defines the total depth of the pocket and is programmed in absolute coordinates.

If there is a roughing operation and it is not programmed, the CNC takes the value defined for the roughing operation.

If there is no roughing operation, it must be programmed.

**[ R±5.5 ] Reference plane**

Defines the reference plane coordinate and is programmed in absolute coordinates.

If there is a roughing operation and it is not programmed, the CNC takes the value defined for the roughing operation.

If there is no roughing operation, it must be programmed.

**[ V5.5 ] Penetration feedrate**

Defines the tool penetrating feedrate.

If not programmed or programmed with a 0 value, it assumes 50% of the feedrate in the plane (F).

**[ F5.5 ] Machining feedrate**

Optional. It sets the machining feedrate in the plane.

**[ S5.5 ] Spindle speed**

Optional. It sets the spindle speed.

**[ T4 ] Tool number.**

Defines the tool used for the semi-finishing operation. It must be programmed.

**[ D4 ] Tool offset**

Optional. Defines the tool offset number.

**[ M ] Auxiliary (miscellaneous) functions**

Optional. Up to 7 miscellaneous M functions can be programmed.

This operation allows M06 with an associated subroutine to be defined, and the tool change is performed before beginning the semi-finishing operation.

FAGOR

FAGOR AUTOMATION

CNC 8055
CNC 8055i

·M· & ·EN· MODELS
SOFT: V02.2X

·254·

### 11.2.3 Finishing operation

This operation is optional.

It will be programmed in a block that will need to bear a label number in order to indicate to the canned cycle the block where the finishing operation is defined.

```
; Definition of irregular pocket canned cycle.
G66 R100 C200 F300 S400 E500
; Defines the finishing operation.
N300 G68...
```

The function for the finishing operation is G68 and it cannot be executed independently from the G66.

Their programming formats are:

G68 B L Q J I R V F S T D M

#### [ B5.5 ] Drilling peck (step)

Defines the pass in the plane between two 3D paths of the finishing operation. It must be defined and with a value other than "0".
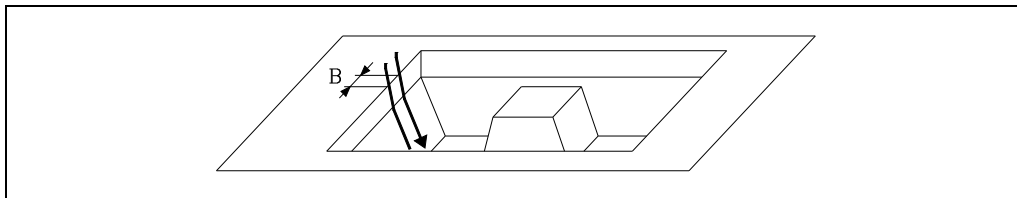


#### [ L±5.5 ] Lateral finishing stock

Defines the value of the finishing stock on the side walls of the pocket left by the roughing and semi-finishing operations. There is no finishing stock left on top of the islands nor on the bottom of the pocket.

If programmed with a positive value, the finishing pass will be carried out in square corner (G07). If a negative value is programmed, the finishing pass will be carried out in G5 (round corner). If not programmed, the cycle assumes "L0".
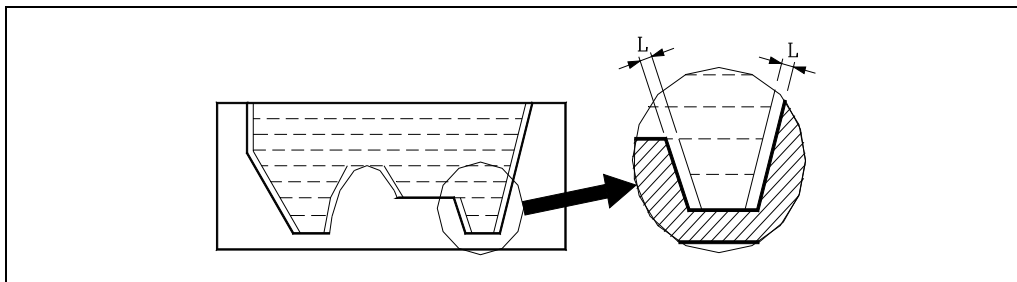


#### [ Q 0/1/2 ] Direction of the finishing pass

Indicates the direction of the finishing pass.

Q= 1:        All the passes will be inward from the top of the pocket to its bottom

Q= 2:        All the passes will be outward from the bottom of the pocket to the top.

Q=0:        Alternating direction for every 2 consecutive paths.

Any other value will generate the corresponding error message. If parameter "Q" is not programmed, the cycle assumes Q0.

**FAGOR** ➔

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2x

### [ J5.5 ] Tool tip radius

Indicates the tool tip radius and, therefore, the type of finishing tool being used.

Depending on the radius assigned to the tool in the tool offset table (of the CNC variables: "TOR" + "TOI") and the value of assigned to this parameter, three tool types may be defined.

FLAT           If J is not programmed or J = 0.

SPHERICAL  If J = R.

TORIC         If J other than 0 and J smaller than R.
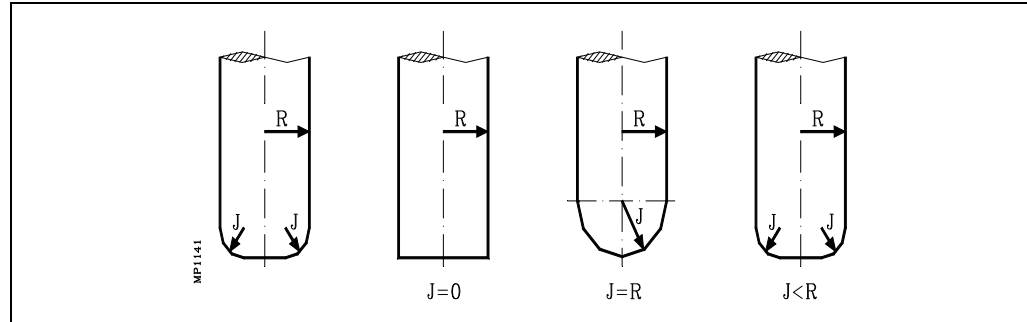


### [ I±5.5 ] Pocket depth

Defines the total depth of the pocket and is programmed in absolute coordinates.

- If defined, the cycle will take it into account during the finishing operation.
- If not defined and the pocket has a roughing operation, the cycle will assume the value defined for the roughing operation.
- If not defined and the pocket has no roughing operation, but it has a semi-finishing operation, the cycle will assume the one define in the semi-finishing operation.
- If the pocket has neither roughing nor semi-finishing operation, this parameter must be defined.

### [ R±5.5 ] Reference plane

Defines the reference plane coordinate and is programmed in absolute coordinates.

- If defined, the cycle will take it into account during the finishing operation.
- If not defined and the pocket has a roughing operation, the cycle will assume the value defined for the roughing operation.
- If not defined and the pocket has no roughing operation, but it has a semi-finishing operation, the cycle will assume the one define in the semi-finishing operation.
- If the pocket has neither roughing nor semi-finishing operation, this parameter must be defined.

### [ V5.5 ] Penetration feedrate

Defines the tool penetrating feedrate.

If not programmed or programmed with a 0 value, it assumes 50% of the feedrate in the plane (F).

### [ F5.5 ] Machining feedrate

Optional. It sets the machining feedrate in the plane.

### [ S5.5 ] Spindle speed

Optional. It sets the spindle speed.

### [ T4 ] Tool number

Defines the tool used for the finishing operation. It must be programmed.

### [ D4 ] Tool offset

Optional. Defines the tool offset number.

### [ M ] Auxiliary (miscellaneous) functions

Optional. Up to 7 miscellaneous M functions can be programmed.

This operation allows M06 with an associated subroutine to be defined, and the tool change is performed before beginning the finishing operation.

FAGOR

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· Models
Soft: V02.2x

·**257**·

### 11.2.4 Geometry of the contours or profiles

To define the contours of a 2D pocket, the plane profile (3) and the depth profile (4) for all the contours must be defined (even if they are vertical).
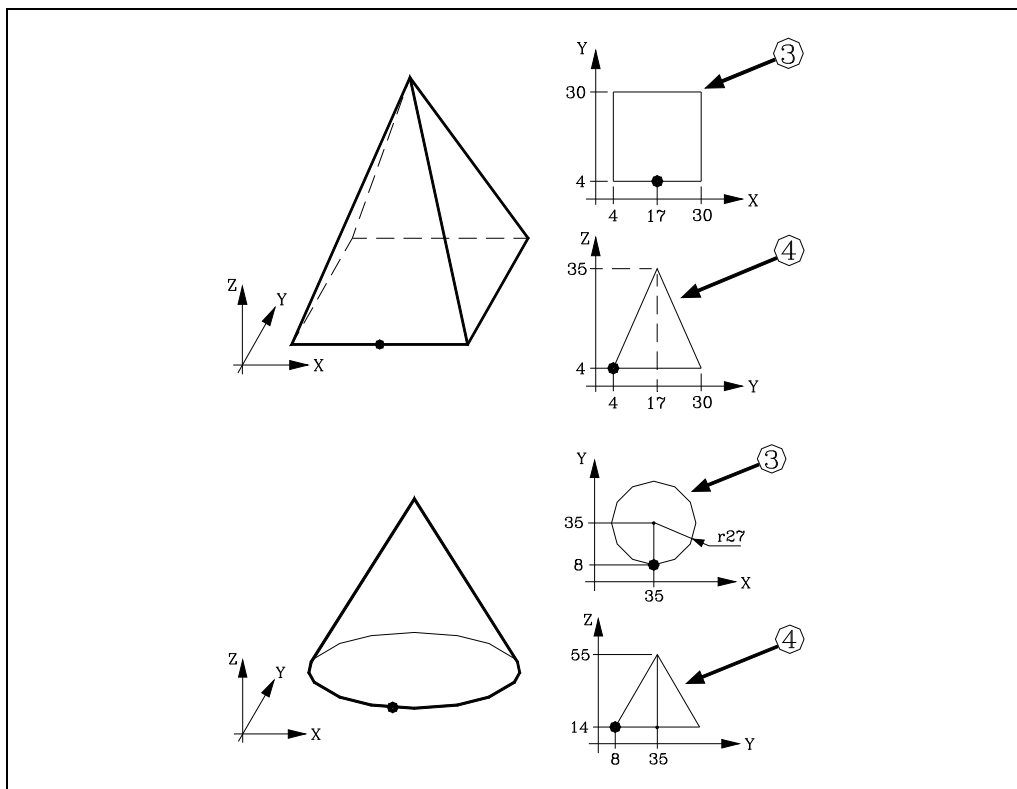


Since the canned cycle applies the same depth profile to the whole contour, the same start point must be used to define the plane profile as for the depth profile.
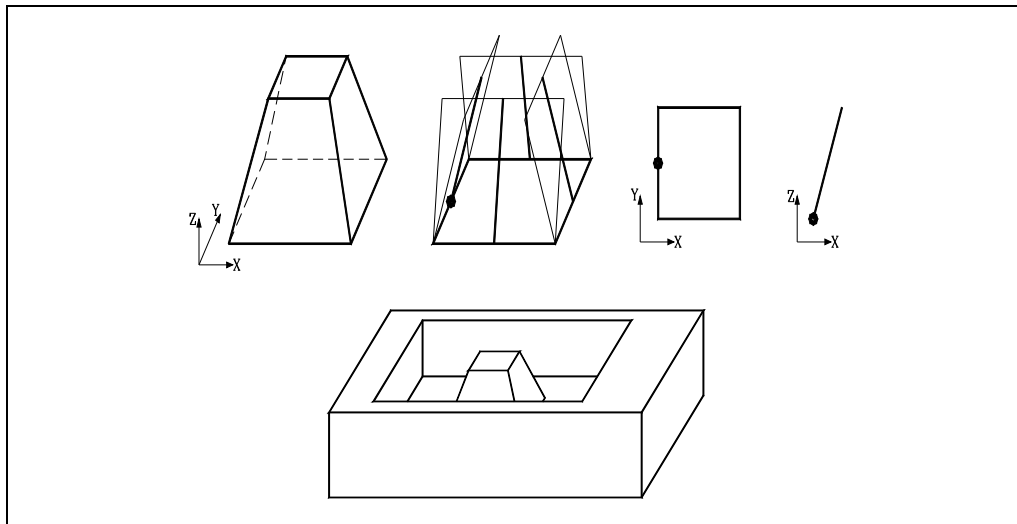


3D contours with more than one depth profile are also possible. These contours are called "composite 3D profiles" and will be described later on.

**FAGOR**

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

·258·
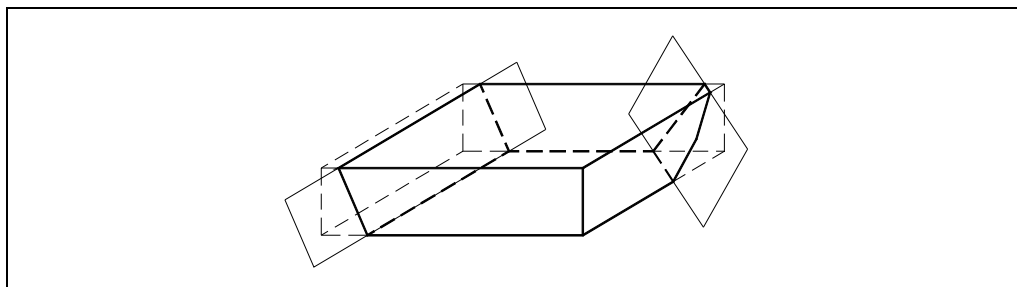
### 11.2.5    Profile programming syntax

When programming inside or outside contours of an irregular 3D pocket (with islands) , the following rules must be complied with:

**1.** The profile in the main plane indicates the shape of the contour.

Since a 3D contour has an infinite number of different profiles (1 per each depth coordinate), the following must be programmed:

− For the outside contour of the pocket, the one for the surface coordinate (1).

− For the inside contours: the one corresponding to the base or bottom (2).
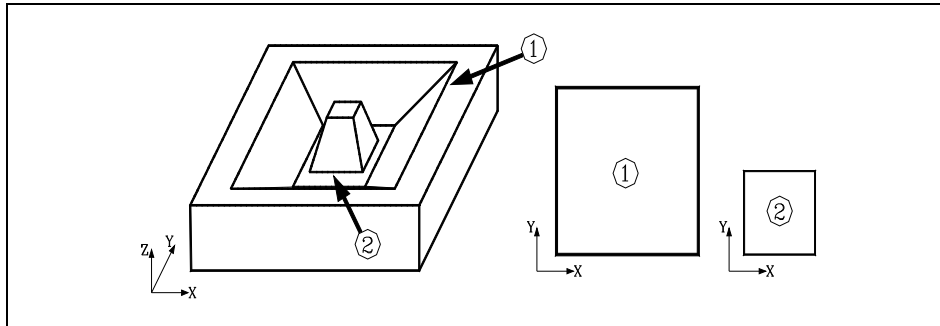


**2.** The profile in the plane must be closed (same starting and end points) and it must not intersect itself. Examples:
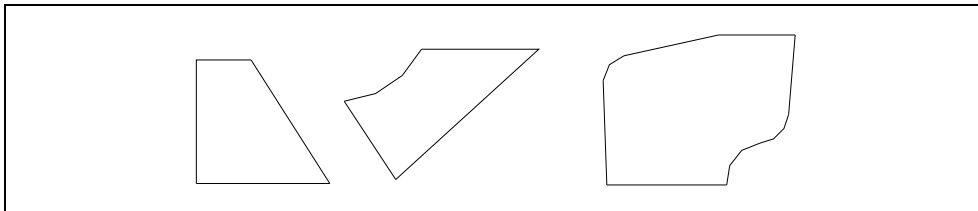


The following examples cause a geometry error.
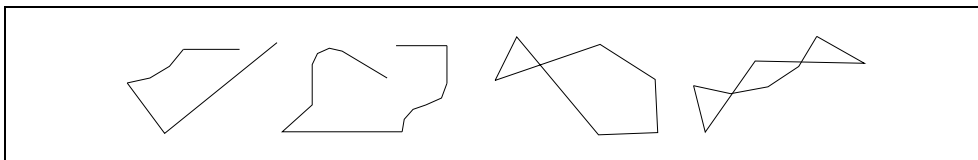


**3.** The depth profile (vertical cross section) must be programmed with any of the axes of the active plane. If the active plane is the XY and the perpendicular axis is the Z axis, one must program: G16XZ or G16YZ.

All profiles, plane and depth, must start with the definition of the plane containing it.

```
G16 XY ; Beginning of the outside profile definition.
; - - Plane profile definition - -
G16 XZ
; - - Depth profile definition - -

G16 XY ; Beginning of the island definition.
; - - Plane profile definition - -
G16 XZ
; - - Depth profile definition - -
```

**4.** The depth profile must be defined after having defined the plane profile.

The beginning points of the plane profile and depth profile must be the same one.

Nevertheless, the depth profile must be programmed:

– For the outside contour of the pocket starting from the top or surface coordinate (1).

– For the inside contours, islands, starting from the bottom or base coordinate (2).
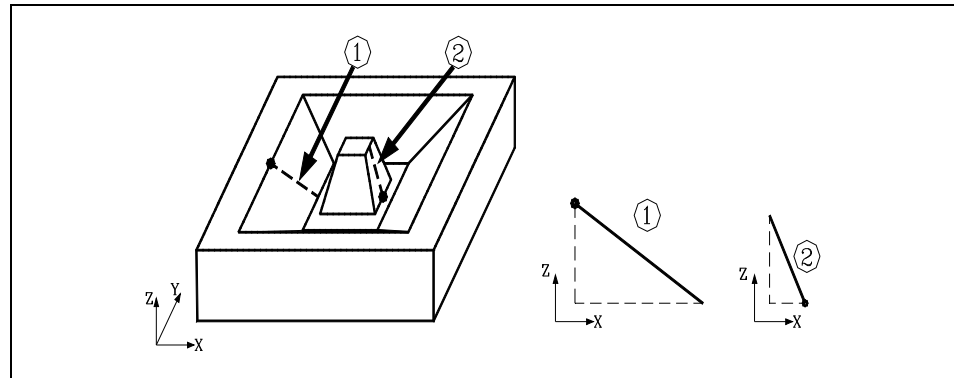


**5.** The depth profile must be open and without direction changes along its path. In other words, it cannot zig-zag.
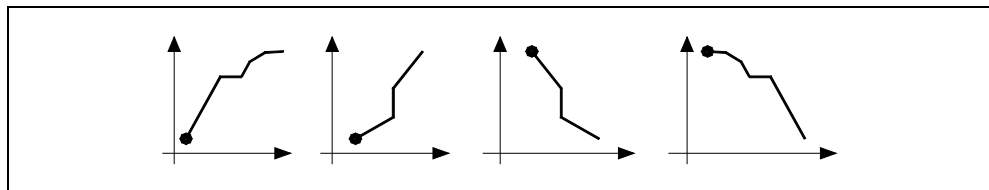


The following examples cause a geometry error.
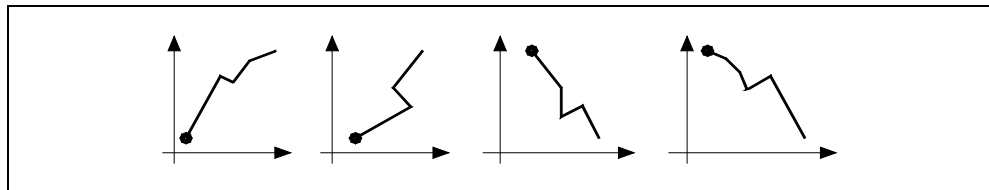
### Programming example. 3D pocket without islands.
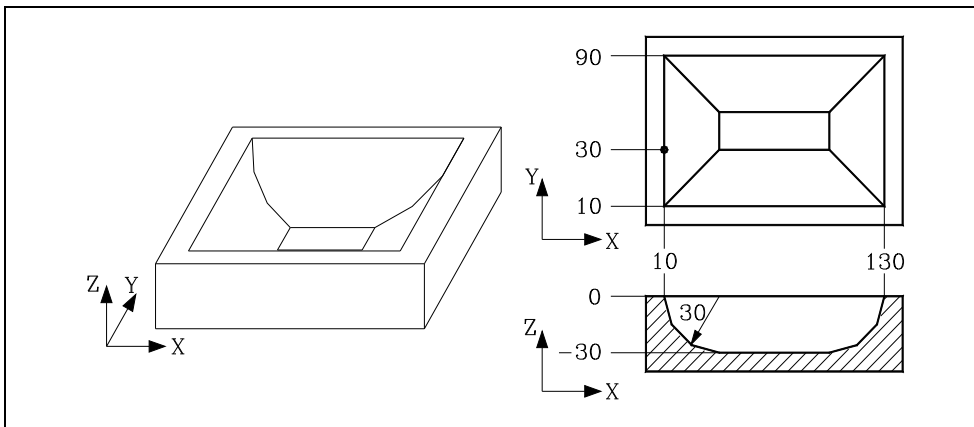


```
(TOR1=2.5,TOL1=20,TOI1=0,TOK1=0)
G17 G0 G43 G90 Z50 S1000 M4
G5
; Defines the 3D pocket.
G66 R200 C250 F300 S400 E500
M30

; Roughing operation.
N200 G67 B5 C4 I-30 R5 V100 F400 T1 D1 M6
; Semi-finishing operation.
N250 G67 B2 I-30 R5 V100 F550 T2 D1 M6
; Finishing operation.
N300 G68 B1.5 L0.75 Q0 I-30 R5 V80 F275 T3 D1 M6

; Definition of pocket geometry.
N400 G17
; Plane profile.
G90 G0 X10 Y30 Z0
G1 Y90
X130
Y10
X10
Y30
; Depth profile.
G16
G0 X10 Z0
N500 G3 X40 Z-30 I30 K0
```

*Programming examples. Profile definition.*

**Pyramid island**



```
; Plane profile
G17
G0 G90 X17 Y4
G1 X30
G1 Y30
G1 X4
G1 Y4
G1 X17

; Depth profile
G16 YZ
G0 G90 Y4 Z4
G1 Y17 Z35
```

**Conic island**



```
; Plane profile
G17
G0 G90 X35 Y8
G2 X35 Y8 I0 J27

; Depth profile
G16 YZ
G0 G90 Y8 Z14
G1 Y35 Z55
```
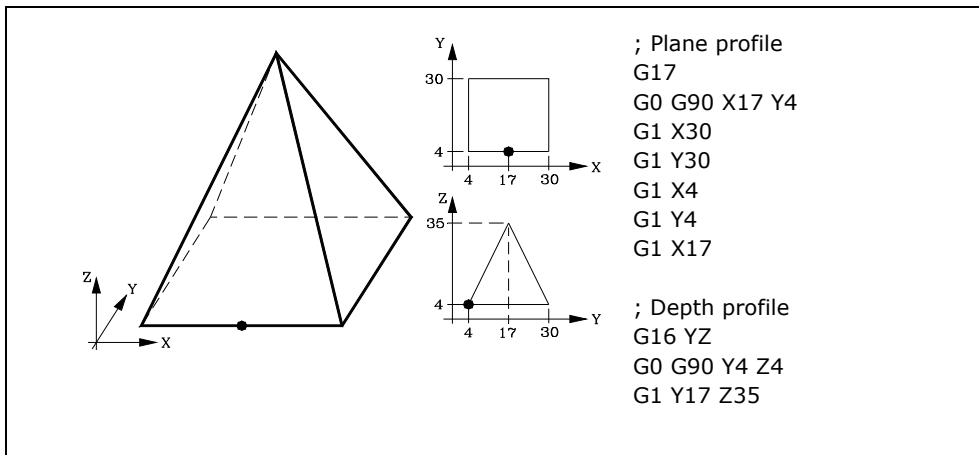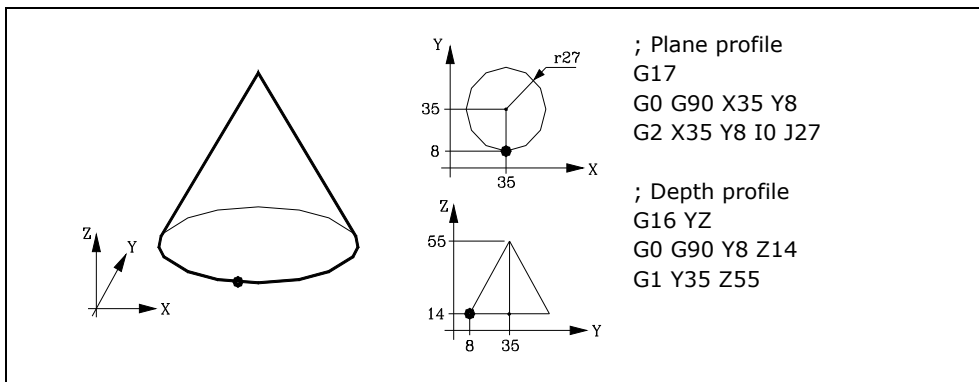
**Semi-spherical island**



```
; Plane profile
G17
G0 G90 X35 Y8
G2 X35 Y8 I0 J27

; Depth profile
G16 YZ
G0 G90 Y8 Z14
G2 Y35 Z41 R27
```

**Programming example. 3D pocket without islands.**
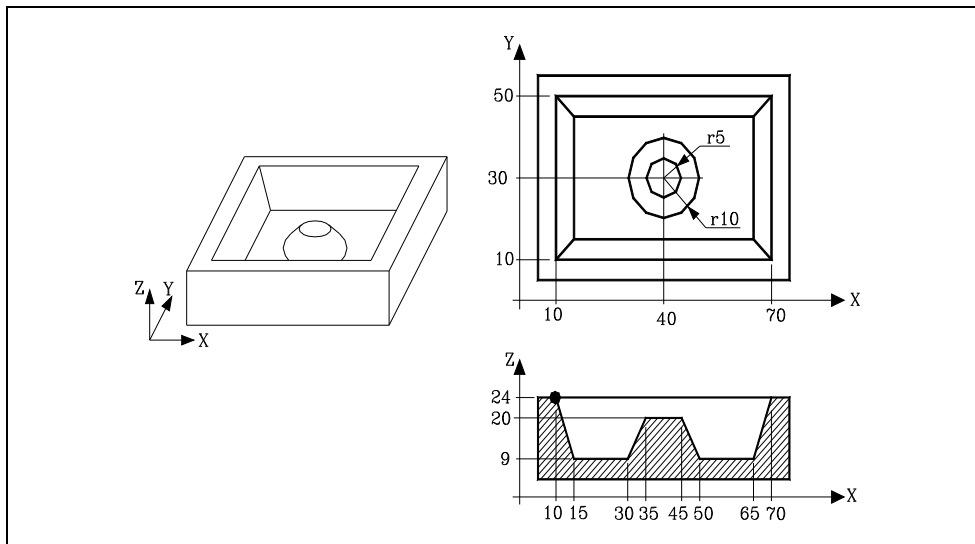


```
(TOR1=2.5,TOL1=20,TOI1=0,TOK1=0)
G17 G0 G43 G90 Z50 S1000 M4
G5
; Defines the 3D pocket.
G66 R200 C250 F300 S400 E500
M30
; Roughing operation.
N200 G67 B5 C4 I9 R25 V100 F400 T1D1 M6
; Semi-finishing operation.
N250 G67 B2 I9 R25 V100 F550 T2D1 M6
; Finishing operation.
N300 G68 B1.5 L0.75 Q0 I9 R25 V50 F275 T3D1 M6
; Definition of pocket geometry.
N400 G17
; External contour. Plane profile.
G90 G0 X10 Y30 Z24
G1 Y50
X70
Y10
X10
Y30
; Depth profile.
G16 XZ
G0 X10 Z24
G1 X15 Z9
; Island definition. Plane profile.
G17
G90 G0 X30 Y30
G2 X30 Y30 I10 K0
; Depth profile.
G16 XZ
G90 G0 X30 Z9
N500 G1 X35 Z20
```

### 11.2.6 Composite 3D profiles

A composite 3D profile is a 3D contour with more than one depth profile.
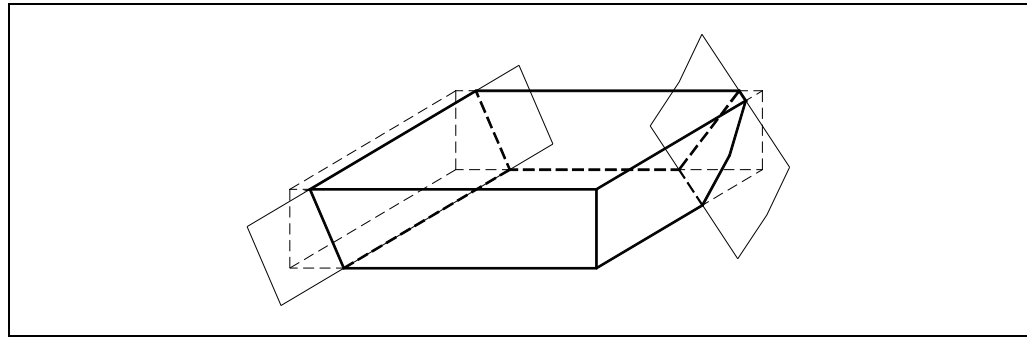
It is defined by means of the intersection of several contours with different depth profiles.

Each contour is defined by a profile on the plane and a depth profile, which requires the definition of an initiation point. All the contours must meet the following conditions:

- The plane profile must contain the corresponding sides completely.
- Only a depth profile per contour must be defined.
- The plane profile and the depth profile of the contour gathering several sides must start at the same point.

The resulting plane profile will be formed by the intersection of the plane profiles of each element or contour.



The initial point of the intersection contour must be included on the face.

Profile resulting from the intersections. Being the final pocket to be machined.

Pi Pf1 = Initial point of profile 1
Pi Pf2 = Initial point of profile 2
Pi Pf3 = Initial point of profile 3
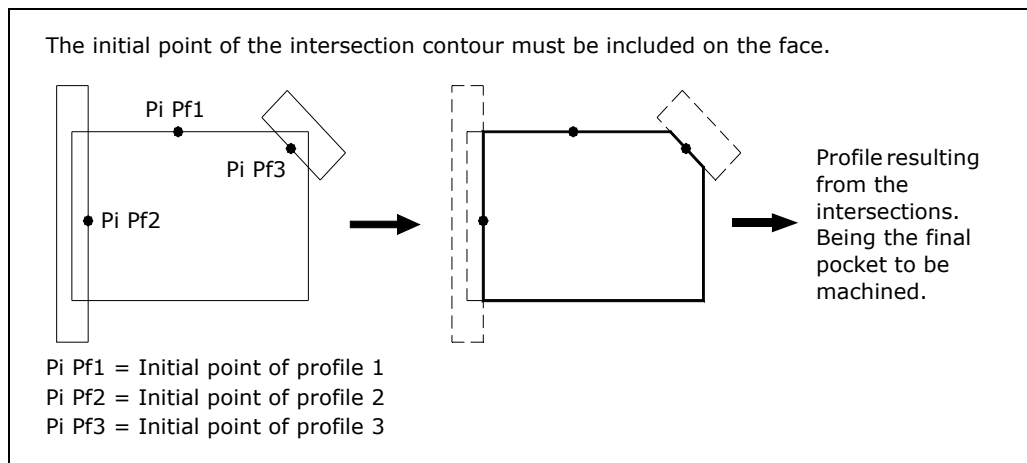
Each wall of the resulting profile will assume the corresponding depth profile.
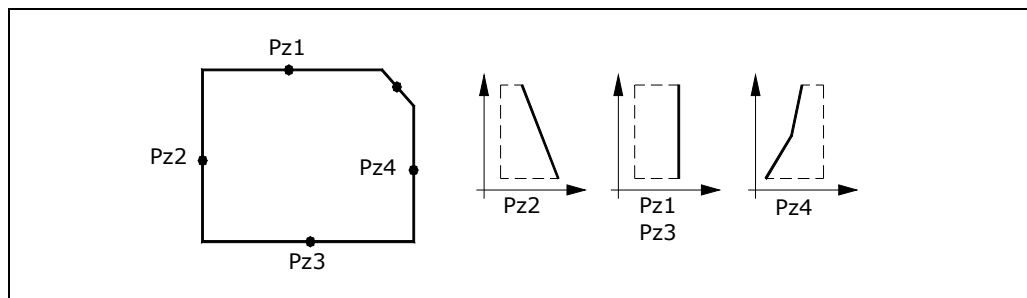


The CNC will interpret Pz1 and Pz3 as a straight line if the profiles are not defined; empty field.

## Profile intersection syntax

The plane profile intersecting rules are:

**1.** At a profile intersection, each contour is divided into several lines which could be grouped as:

    – Lines external to the other contour.

    – Lines internal to the other contour.

    The starting point of each contour (x) determines the group of lines to be selected.

    The following example shows the explained selection process. The solid lines indicate the lines external to the other contour and the dashes indicate the internal lines.



Examples of profile intersections:

**2.** The programming sequence for the different profiles is determinant when having an intersection of more than 3 profiles.

The profile intersection process is performed according to the order in which the profiles have been programmed. This way, the result of the intersection between the first two will be intersected with the third one and so forth.

The initial point of the resulting profiles always coincides with the initial point that defined the first profile.



**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

·266·

### 11.2.7 Profile stacking

When 2 or more profiles stack on top of each other, the following considerations must be taken into account.

For clarity sakes, refer to the drawing on the right that consists of 2 stacked profiles: 1 and 2.
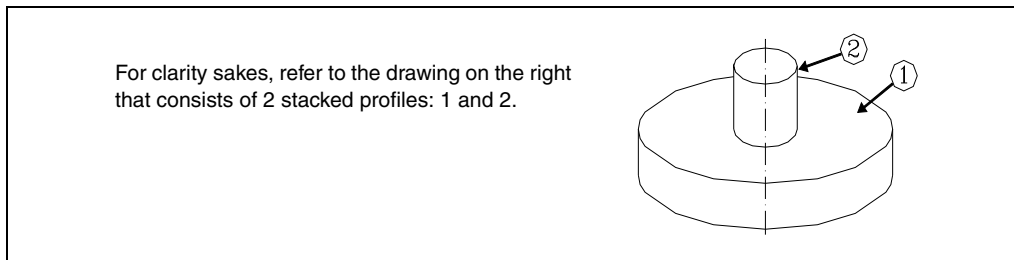
The base coordinate of the top profile (2) must coincide with the surface coordinate of the bottom profile (1).
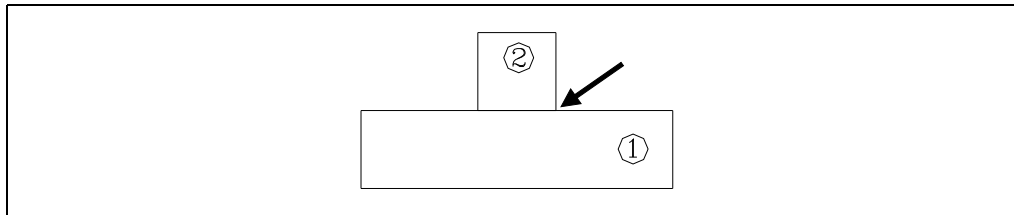
If there is a gap between them, the cycle will consider that they are 2 different profiles and it will eliminate the top profile when executing the bottom one.
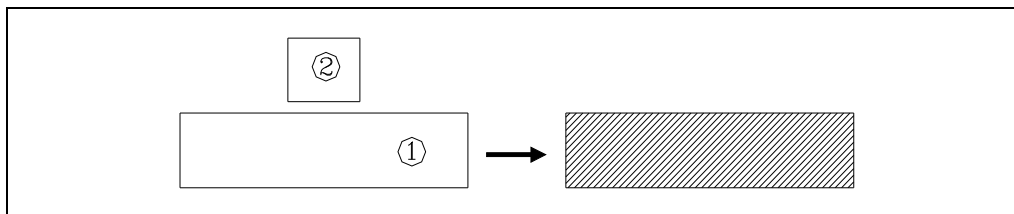
If the profiles mix, the canned cycle will make a groove around the top profile when running the finishing pass.

### 11.2.8 Profile programming syntax

The outside profile and the inside profiles or islands which are programmed must be defined by simple geometrical elements such as straight lines or arcs.

The first definition block (where the external profile starts) and the last (where the last profile defined ends) must be provided with the block label number. These label numbers will indicate the beginning and end of the profile geometry definition for the profiles that make up the pocket.

```
; Definition of irregular pocket canned cycle.
G66 R100 C200 F300 S400 E500
; Geometrical description.
N400 G17
…
N500 G2 G6 X300 Y50 I150 J0
```

The profile programming syntax must follow these rules:

- The first profile defining block must have a label number to indicate to the G66 canned cycle the beginning of the geometry description.

- First, the outside pocket contour must be defined and, then, the contour of each island.

  When a contour has more than one depth profile, the contours must be defined one by one indicating, on each one, the plane profile and, then, its depth profile.

- The first profile defining block of the plane profile as well as that of the depth profile must contain function G00 (indicative of the beginning of the profile).

  Care must be taken to program G01, G02 or G03 in the block following the definition of the beginning, as G00 is modal, thus preventing the CNC from interpreting the following blocks as the beginnings of a new profile.

- The last profile defining block must have a label number to indicate to the G66 canned cycle the end of the geometry description.

```
; Definition of irregular pocket canned cycle.
G66 R200 C250 F300 S400 E500

; Beginning of the definition of pocket geometry.
N400 G17
; External contour. Plane profile.
G0 G90 X5 Y-26 Z0
--- --- --- ---
; Depth profile.
G16 XZ
G0 --- ---
--- --- --- ---
; Island definition
G17
; Plane profile.
G0 X30 Y-6
--- --- --- ---
; Depth profile.
G16 XZ
G0
--- --- --- ---
; End of geometric description.
N500G3 Y-21 Z0 J-5 K0
```

- Profiles are described as programmed paths, it being possible to include corner rounding, chamfers, etc., following the syntax rules defined for this purpose.

- The profile description must not contain: mirror images, scaling factor changes, pattern rotation, zero offsets, etc.

- It must not contain high level blocks such as jumps, calls to subroutines or parametric programming.

- It must not contain other canned cycles.

In addition to the G00 function, which has a special meaning, the irregular pocket canned cycle allows the use of the following functions for the definition of profiles.

| | |
|---|---|
| G01 | Linear interpolation. |
| G02 | Clockwise circular interpolation. |
| G03 | Counter-clockwise circular interpolation. |
| G06 | Circle center in absolute coordinates. |
| G08 | Arc tangent to previous path. |
| G09 | Arc defined by three points. |
| G16 | Main plane selection by two addresses and longitudinal axis. |
| G17 | Main plane X-Y, and longitudinal axis Z. |
| G18 | Main plane Z-X, and longitudinal axis Y. |
| G19 | Main plane Y-Z, and longitudinal axis X. |
| G36 | Corner rounding, radius blend. |
| G39 | Chamfer. |
| G53 | Programming with respect to machine zero. |
| G70 | Programming in inches. |
| G71 | Programming in millimeters. |
| G90 | Absolute programming. |
| G91 | Incremental programming. |
| G93 | Polar origin preset. |

**FAGOR**

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

### 11.2.9   Programming examples
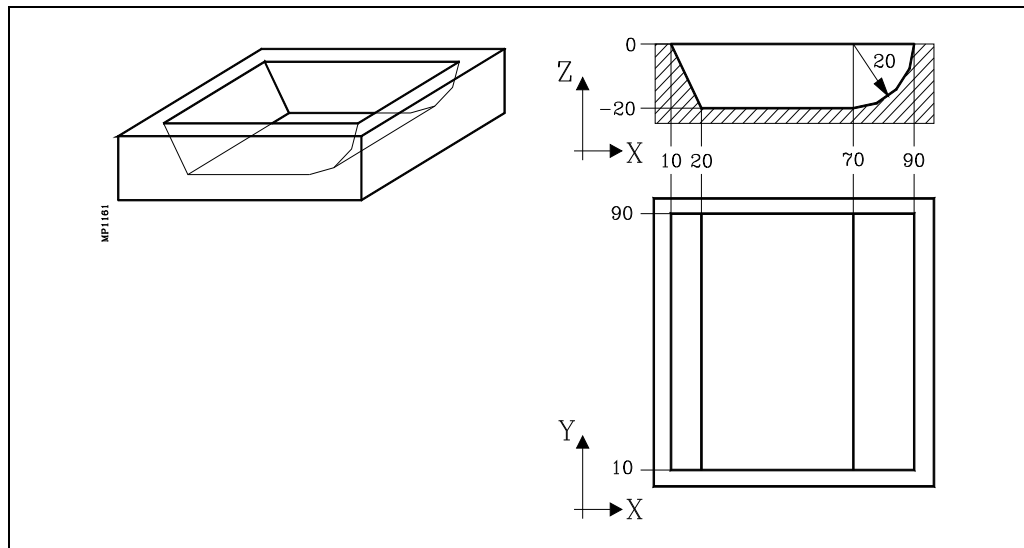
*Programming example ·1·*



The island of this example has 3 different depth profiles, type A, type B and type C. To define the island, 3 contours are used: contour A, contour B and contour C.
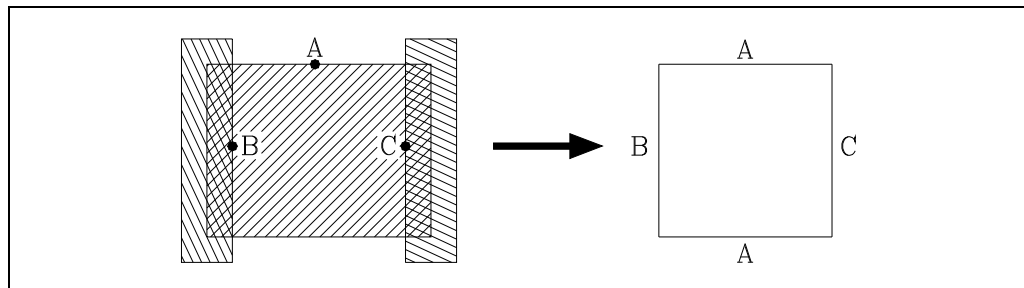


```
; Tool dimensions.
(TOR1=2.5,TOL1=20,TOI1=0,TOK1=0)

; Initial positioning and definition of the 3D pocket.
G17 G0 G43 G90 Z50 S1000 M4
G5
G66 R200 C250 F300 S400 E500
M30

; Definition of roughing operation.
N200 G67 B5 C4 I-20 R5 V100 F400 T1D1 M6

; Definition of the semi-finishing operation
N250 G67 B2 I-20 R5 V100 F550 T2D1 M6

; Definition of finishing operation.
N300 G68 B1.5 L0.75 Q0 I-20 R5 V80 F275 T3 D1 M6

; Definition of pocket geometry. Blocks N400 through N500.
N400 G17

; Definition of contour A. Profile on the plane.
G0 G90 X50 Y90 Z0
G1 X0
Y10
X100
Y90
X50
```

```
; Depth profile.
G16 YZ
G0 G90 Y90 Z0
G1 Z-20

; Definition of contour B. Profile on the plane.
G17
G0 G90 X10 Y50
G1 Y100
X-10
Y0
X10
Y50

; Depth profile.
G16 XZ
G0 G90 X10 Z0
G1 X20 Z-20

; Definition of contour C. Profile on the plane.
G17
G0 G90 X90 Y50
G1 Y100
X110
Y0
X90
Y50

; Depth profile.
G16 XZ
G0 G90 X90 Z0
N500 G2 X70 Z-20 I-20 K0
```

*Programming example ·2·*
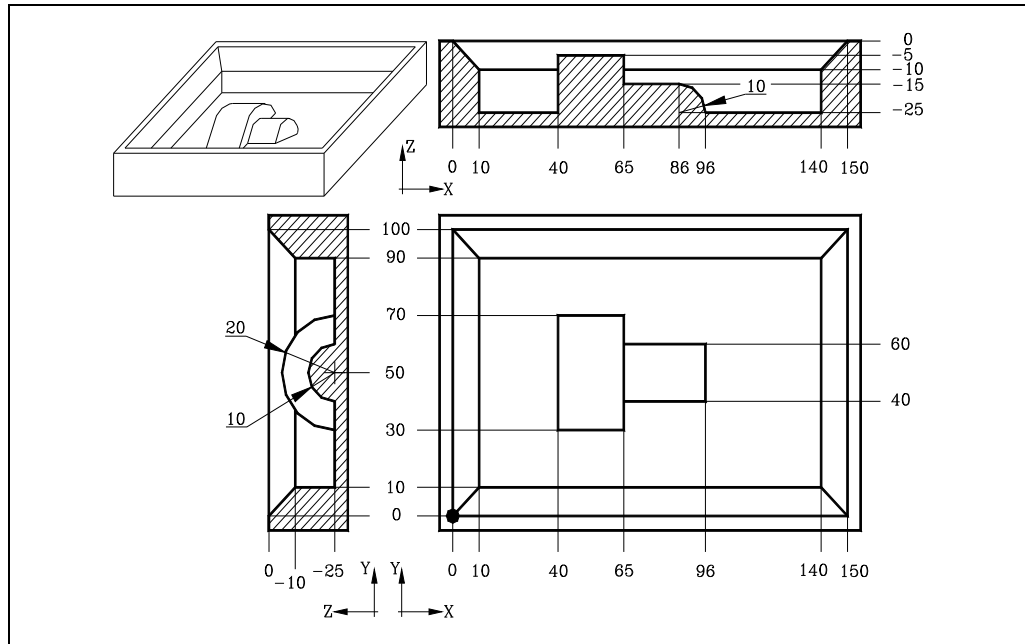


The island of this example has 3 different depth profiles, type A, type B and type C. To define the island, 3 contours are used: contour A, contour B and contour C.
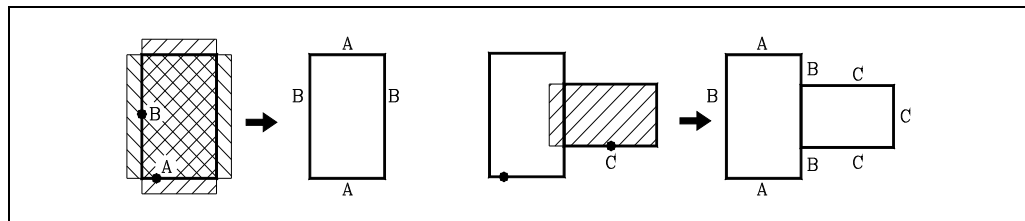


```
; Tool dimensions.
(TOR1=7.5,TOI1=0,TOR2=5,TOI2=0,TOR3=2.5,TOI3=0)

; Initial positioning and definition of the 3D pocket.
G17 G0 G43 G90 Z50 S1000 M4
G5
G66 R200 C250 F300 S400 E500
M30

; Definition of roughing operation.
N200 G67 B7 C14 I-25 R3 V100 F500 T1 D1 M6

; Definition of the semi-finishing operation
N250 G67 B3 I-25 R3 V100 F625 T2 D2 M6

; Definition of finishing operation.
N300 G68 B1 L1 Q0 J0 I-25 R3 V100 F350 T3 D3 M6

; Definition of pocket geometry. Blocks N400 through N500.
N400 G17

; Definition of outside contour. Plane profile.
G0 G90 X0 Y0 Z0
G1 X150
Y100
X0
Y0

; Depth profile.
G16 XZ
G0 G90 X0 Z0
G1 X10 Z-10
```

```
; Definition of contour A. Profile on the plane.
G17
G0 G90 X50 Y30
G1 X70
Y70
X35
Y30
X50

; Depth profile.
G16 YZ
G0 G90 Y30 Z-25
G2 Y50 Z-5 J20 K0

; Definition of contour B. Profile on the plane.
G17
G0 G90 X40 Y50
G1 Y25
X65
Y75
X40
Y50

; Depth profile.
G16 XZ
G0 G90 X40 Z-25
G1 Z-5

; Definition of contour C. Profile on the plane.
G17 G90 X80 Y40
G0 X96
G1 Y60
X60
Y40
X80

; Depth profile.
G16 YZ
G0 G90 Y40 Z-25
N500 G2 Y50 Z-15 J10 K0
```

## Programming example ·3·
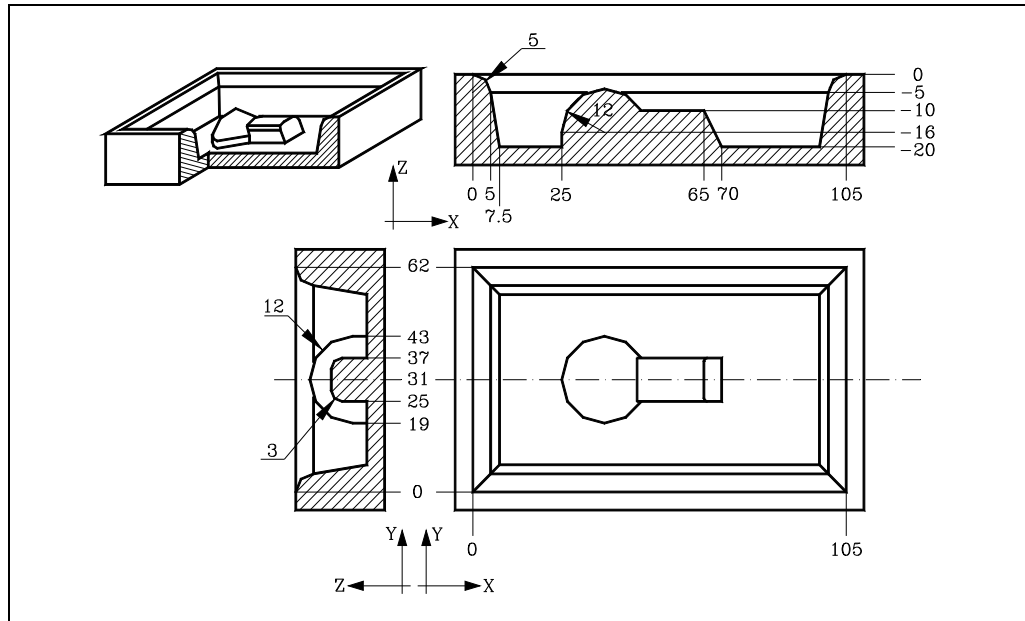


The island of this example has 3 different depth profiles, type A, type B and type C. To define the island, 3 contours are used: contour A, contour B and contour C.
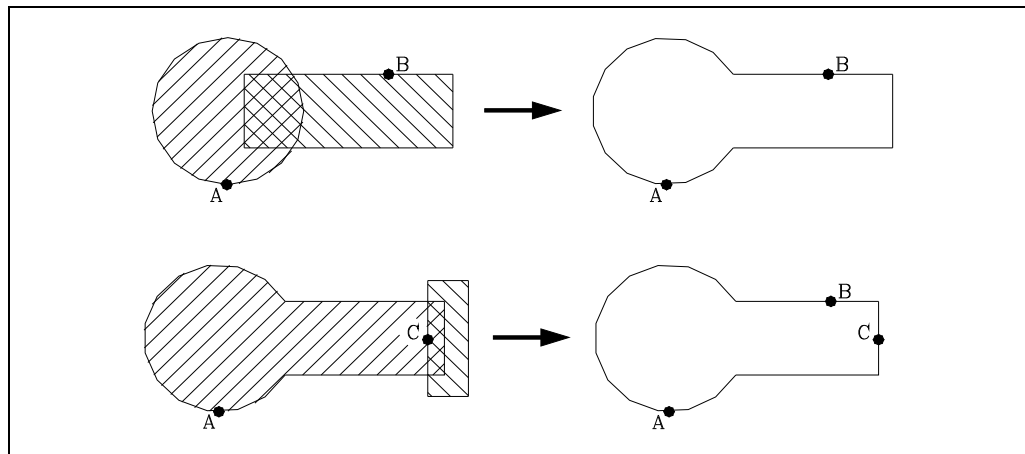


```
; Tool dimensions.
(TOR1=4,TOI1=0,TOR2=2.5,TOI2=0)

; Initial positioning and definition of the 3D pocket.
G17 G0 G43 G90 Z25 S1000 M3
G66 R200 C250 F300 S400 E500
M30

; Definition of roughing operation.
N200 G67 B5 C4 I-20 R5 V100 F700 T1 D1 M6

; Definition of the semi-finishing operation
N250 G67 B2 I-20 R5 V100 F850 T1 D1 M6

; Definition of finishing operation.
N300 G68 B1.5 L0.25 Q0 I-20 R5 V100 F500 T2 D2 M6

; Definition of pocket geometry. Blocks N400 through N500.
N400 G17
```

```
; Definition of outside contour. Plane profile.
G0 G90 X0 Y0 Z0
G1 X105
Y62
X0
Y0

; Depth profile.
G16 XZ
G0 X0 Z0
G2 X5 Z-5 I0 K-5
G1 X7.5 Z-20

; Definition of contour A. Profile on the plane.
G17
G90 G0 X37 Y19
G2 I0 J12

; Depth profile.
G16 YZ
G0 Y19 Z-20
G1 Z-16
G2 Y31 Z-4 R12

; Definition of contour B. Profile on the plane.
G17
G90 G0 X60 Y37
G1 X75
Y25
X40
Y37

; Depth profile.
G16 YZ
G0 Y37 Z-20
G1 Z-13
G3 Y34 Z-10 J-3 K0

; Definition of contour C. Profile on the plane.
G17
G0 X70 Y31
G1 Y40
X80
Y20
X70
Y31

; Depth profile.
G16 XZ
G0 X70 Z-20
N500 G1 X65 Z-10
```

*Programming example ·4·*
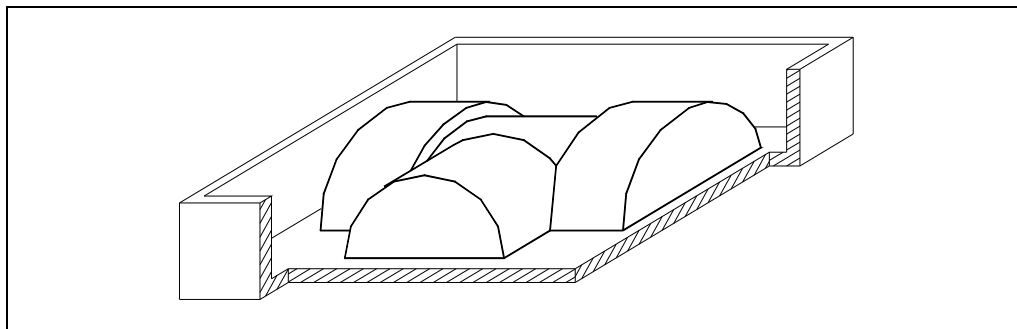


To define the island 10 contours are used as shown here:
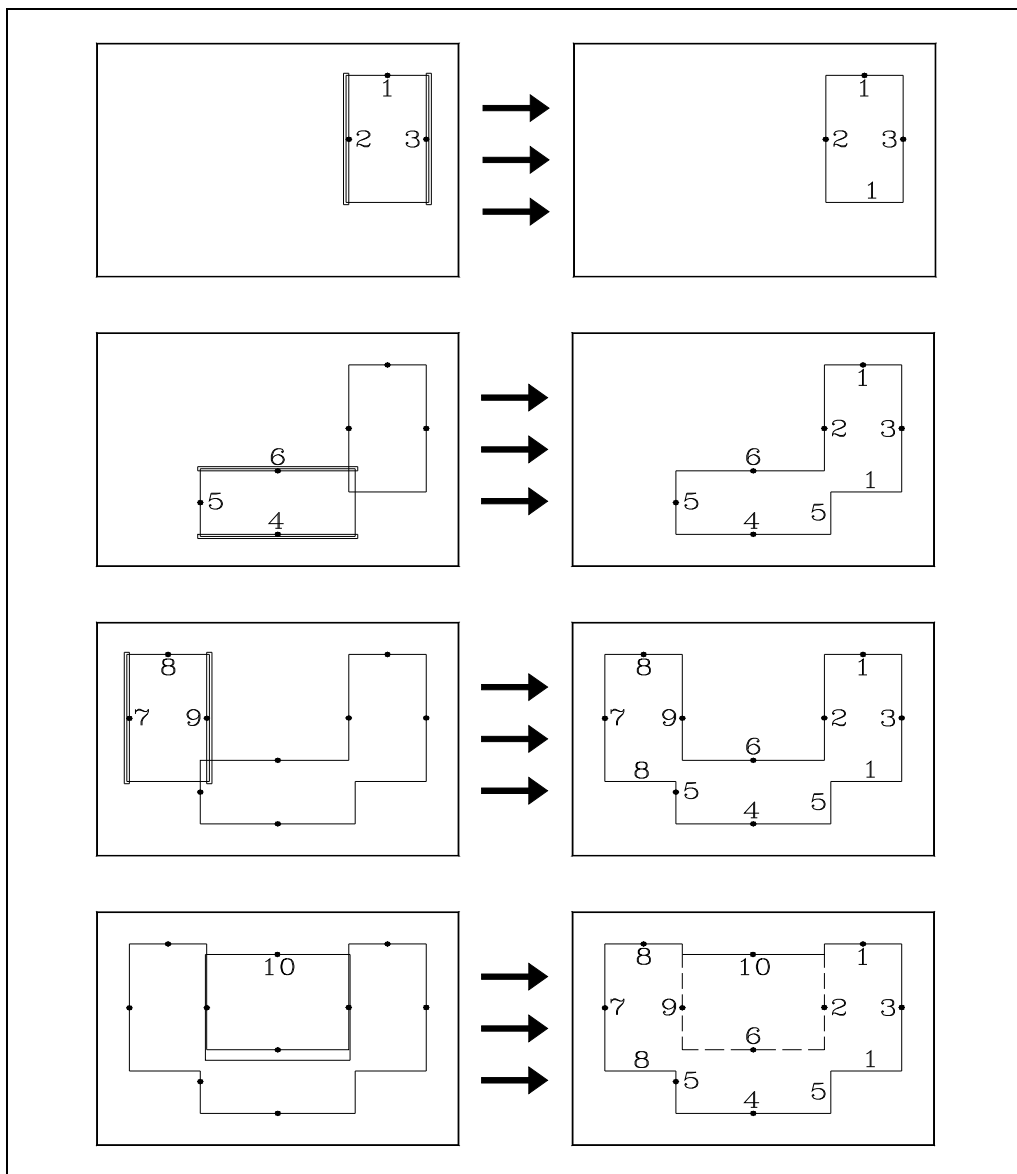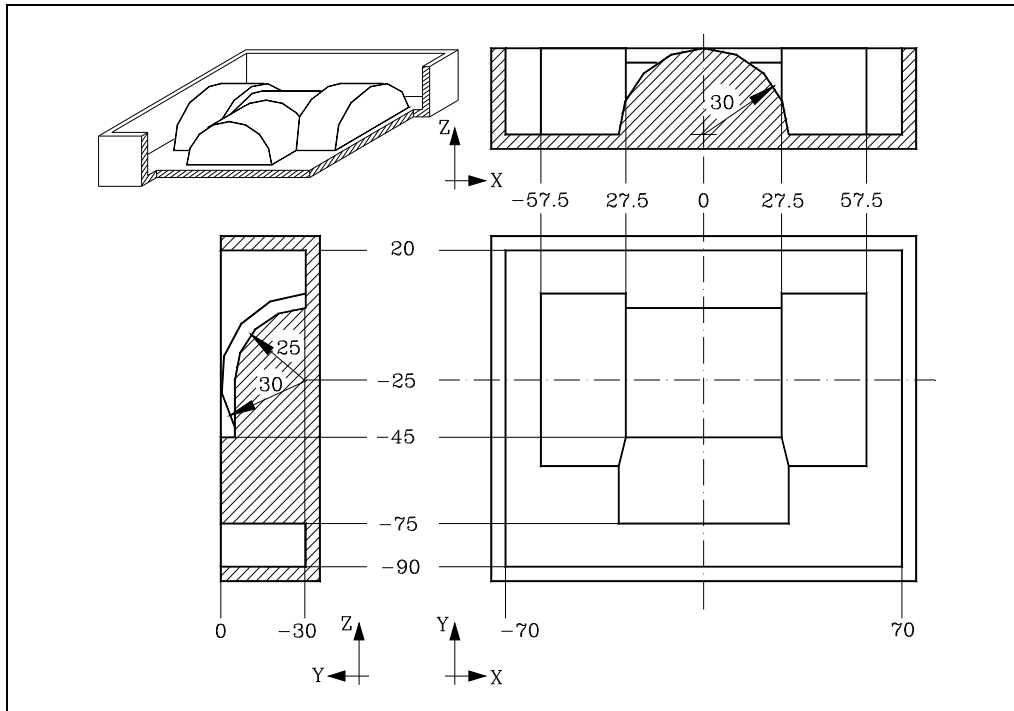
```
; Tool dimensions.
(TOR1=4,TOI1=0,TOR2=2.5,TOI2=0)

; Initial positioning and definition of the 3D pocket.
G17 G0 G43 G90 Z25 S1000 M3
G66 R200 C250 F300 S400 E500
M30

; Definition of roughing operation.
N200 G67 B5 C0 I-30 R5 V100 F700 T1 D1 M6

; Definition of the semi-finishing operation
N250 G67 B1.15 I-29 R5 V100 F850 T1 D1 M6

; Definition of finishing operation.
N300 G68 B1.5 L0.25 Q0 I-30 R5 V100 F500 T2 D2 M6

; Definition of pocket geometry. Blocks N400 through N500.
N400 G17

; Definition of outside contour. Plane profile.
G90 G0 X-70 Y20 Z0
G1 X70
Y-90
X-70
Y20

; Definition of contour 1. Plane profile.
G17
G90 G0 X42.5 Y5
G1 G91 X-16
Y-60
X32
Y60
X-16

; Depth profile.
G16 YZ
G0 G90 Y5 Z-30
G3 Y-25 Z0 J-30 K0
```

```
; Definition of contour 2.
G17
G0 X27.5 Y-25
G1 G91 Y31
G1 X-2
Y-62
X2
Y31

; Depth profile.
G16 XZ
G0 G90 X27.5 Z-30
G1 Z0

; Definition of contour 3.
G17
G0 X57.5 Y-25
G1 G91 Y-31
X2
Y62
X-2
Y-31

; Depth profile.
G16 XZ
G0 G90 X57.5 Z-30
G1 Z0

; Definition of contour 4.
G17
G0 X0 Y-75
G1 G91 X-31
Y-2
X62
Y2
X-31

; Depth profile.
G16 YZ
G0 G90 Y-75 Z-30
G1 Z0

; Definition of contour 5.
G17
G0 X-30 Y-60
G1 G91 Y-16
X60
Y32
X-60
Y-16

; Depth profile.
G16 XZ
G0 G90 X-30 Z-30
G2 X0 Z0 I30 K0

; Definition of contour 6.
G17
G0 X0 Y-45
G1 G91 X31
Y2
X-62
Y-2
X31
```

```
; Depth profile.
G16 YZ
G0 G90 Y-45 Z-30
G1 Z0

; Definition of contour 7.
G17
G0 X-57.5 Y-25
G1 G91 Y31
X-2
Y-62
X2
Y31

; Depth profile.
G16 XZ
G0 G90 X-57.5 Z-30
G1 Z0

; Definition of contour 8.
G17
G0 X-42.5 Y5
G1 G91 X-16
Y-60
X32
Y60
X-16

; Depth profile.
G16 YZ
G0 G90 Y5 Z-30
G3 Y-25 Z0 J-30 K0

; Definition of contour 9.
G17
G0 X-27.5 Y-25
G1 G91 Y-31
X2
Y62
X-2
Y-31

; Depth profile.
G16 XZ
G0 G90 X27.5 Z-30
G1 Z0

; Definition of contour 10.
G17
G0 X0 Y0
G1 X-28
Y-50
X28
Y0
X0

; Depth profile.
G16 YZ
G0 Y0 Z-30
N500 G3 Y-25 Z-5 J-25 K0
```

## Programming example ·5·
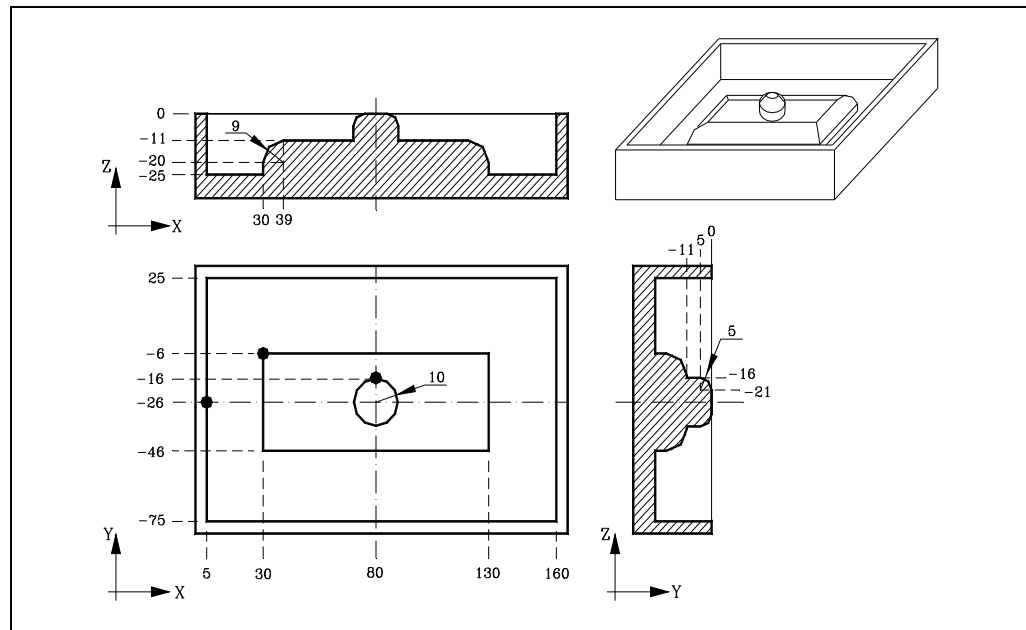


The island of this example has 2 different depth profiles, type A and type B. To define the island, 2 contours are used: contour A and contour B.
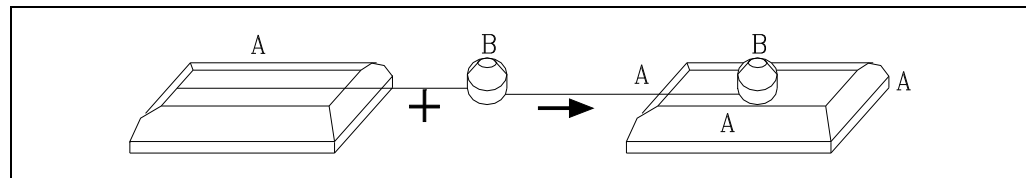


```
; Tool dimensions.
(TOR1=2.5,TOL1=20,TOI1=0,TOK1=0)

; Initial positioning and definition of the 3D pocket.
G17 G0 G43 G90 Z50 S1000 M4
G5
G66 R200 C250 F300 S400 E500
M30

; Definition of roughing operation.
N200 G67 B5 C4 I-25 R5 V100 F400 T1 D1 M6

; Definition of the semi-finishing operation
N250 G67 B2 I-25 R5 V100 F550 T2 D1 M6

; Definition of finishing operation.
N300 G68 B1.5 L0.75 Q0 I-25 R5 V100 F275 T3 D1 M6

; Definition of pocket geometry. Blocks N400 through N500.
N400 G17

; Definition of outside contour. Plane profile.
G90 G0 X5 Y-26 Z0
G1 Y25
X160
Y-75
X5
Y-26
```

**FAGOR**

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2x

·280·

```
; Definition of the low (A-type) contour. Plane profile.
G17
G90 G0 X30 Y-6
G1 Y-46
X130
Y-6
X30

; Depth profile.
G16 XZ
G0 X30 Z-25
G1 Z-20
G2 X39 Z-11 I9 K0

; Definition of the high (B-type) contour. Plane profile.
G17
G90 G0 X80 Y-16
G2 I0 J-10

; Depth profile.
G16 YZ
G0 Y-16 Z-11
G1 Y-16 Z-5
N500 G3 Y-21 Z0 J-5 K0
```

**FAGOR**

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·**M**· & ·**EN**· **MODELS**
**SOFT: V02.2x**

·**281**·

### 11.2.10 Errors

The CNC will issue the following errors:

**ERROR 1025**  **A tool with no radius has been programmed**

It comes up when using a tool with "0" radius while machining a 3D pocket.

**ERROR 1026**  **A step has been programmed that is larger than the tool diameter**

When parameter "C" of the roughing operation is greater than the diameter of the roughing tool.

**ERROR 1041**  **A parameter required in the canned cycle has not been programmed.**

It comes up in the following instances:

- When parameters "I" and "R" have not been programmed in the roughing operation.
- When not using a roughing operation and not programming the "I" and "R" parameters for the semi-finishing operation.
- When not using a semi-finishing operation and not programming the "I" and "R" parameters for the finishing operation.
- When parameter "B" has not been programmed in the finishing operation.

**ERROR 1042**  **Wrong parameter value in the canned cycle**

It comes up in the following instances:

- When parameter "Q" of the finishing operation has the wrong value.
- When parameter "B" of the finishing operation has a "0" value.
- When parameter "J" of the finishing operation has been programmed with a value greater than the finishing tool radius.

**ERROR 1043**  **Wrong depth profile in a pocket with islands.**

It comes up in the following instances:

- When the depth profiles of 2 sections of the same contour (simple or composite) cross each other
- When the finishing operation cannot be performed with the programmed tool. A typical case is a spherical mold with a non-spherical tool (parameter "J" not equal to the radius).

**ERROR 1044**  **The plane profile intersects itself in an irregular pocket with islands.**

It comes up when any of the plane profiles of the programmed contours intersects itself.

**ERROR 1046**  **Wrong tool position prior to the canned cycle**

It comes up when calling the G66 cycle if the tool is positioned between the reference plane and the depth coordinate (bottom) of any of the operations.

**ERROR 1047**  **Open plane profile in an irregular pocket with islands**

It comes up when any of the programmed contours does not begin and end at the same point. It may be because G1 has not been programmed after the beginning, with G0, on any of the profiles.

**ERROR 1048**  **Part surface coordinate not programmed in pocket with islands**

It comes up when the first point of the geometry does not include the pocket top coordinate.

**ERROR 1049**  **Wrong reference plane coordinate for the canned cycle.**

It comes up when the coordinate of the reference plane is located between the part's "top" and "bottom" in any of the operations.

**ERROR 1084**  **Arc programmed wrong'**

It comes up when any of the paths programmed in the geometry definition of the pocket is wrong.

**ERROR 1227**  **Wrong profile intersection in a pocket with islands.**

It comes up in the following instances:

- When two plane profiles have a common section (drawing on the left).

• When the initial points of two profiles in the main plane coincide (drawing on the right).

**FAGOR** ⊃

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

# PROBING
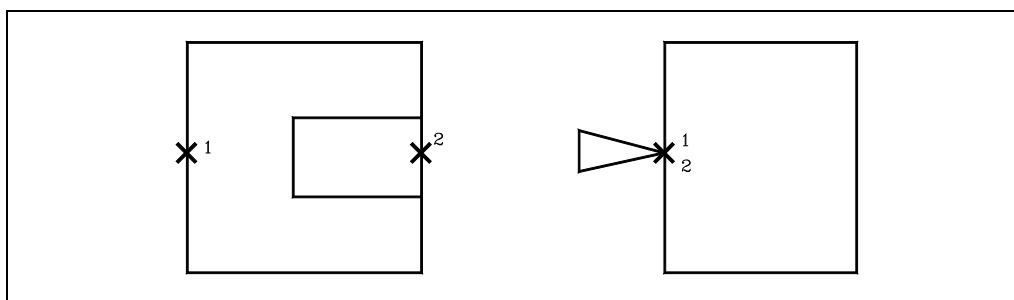
The CNC has two probe inputs, one for TTL-type 5Vdc signals and another for 24 Vdc signals.

The connection of the different types of probes to these inputs are explained in the appendix to the Installation manual.

This control allows the following operations to be performed, by using probes:

- Programming probing blocks with functions G75 and G76.
- Execution of the various tool calibration, part-measurement and part centering cycles by means of high-level language programming.

## 12.1 Probing (G75, G76)

The G75 function allows movements to be programmed that will end after the CNC receives the signal from the measuring probe used.

The G76 function allows movements to be programmed that will end after the CNC no longer receives the signal from the measuring probe used.

Their definition format is:

G75 X..C ±5.5
G76 X..C ±5.5

After G75 or G76, the required axis or axes will be programmed, as well as the coordinates of these axes that will define the end point of the programmed movement.

The machine will move according to the programmed path until it receives the signal from the probe (G75) or until it no longer receives the probe signal (G76). At this time, the CNC will consider the block finished, taking as the theoretical position of the axes the real position they have at that time.

If the axes reach the programmed position before receiving or while receiving the external signal from the probe, the CNC will stop the movement of the axes.

This type of movement with probing blocks are very useful when it is required to generate measurement or verification programs for tools and parts.

Functions G75 and G76 are not modal and, therefore, must be programmed whenever it is wished to probe.

Functions G75 and G76 are incompatible with each other and with G00, G02, G03, G33, G34, G41 and G42 functions. In addition, once this has been performed, the CNC will assume functions G01 and G40.

During G75 or G76 moves, the operation of the feedrate override switch depends on the setting of OEM machine parameter FOVRG75.

## 12.2    Probing canned cycles

The CNC offers the following probing canned cycles:

- Tool calibration canned cycle.
- Probe calibrating canned cycle.
- Surface measuring canned cycle.
- Outside corner measuring canned cycle.
- Inside corner measuring canned cycle.
- Angle measuring canned cycle.
- Corner and angle measuring canned cycle.
- Hole measuring cycle.
- Boss measuring cycle.
- Rectangular part centering canned cycle.
- Circular part centering canned cycle.
- Tabletop probe calibrating canned cycle.

All the movements of these probing canned cycles will be performed in the X, Y, Z axes and the work plane must be formed by 2 of these axes (XY, XZ, YZ, YX, ZX, ZY). The other axis must be perpendicular and it must be selected as the longitudinal axis.

Canned cycles will be programmed by means of the high level mnemonic, PROBE, which has the following programming format:

(PROBE (expression), (assignment instruction),…)

This instruction PROBE calls the probing cycle indicated by means of a number or any expression which results in a number. Besides, it allows the parameters of this cycle to be initialized with the values required to perform it, by means of assignment instructions.

### *Some general points to consider*

Probing canned cycles are not modal; therefore, they must be programmed every time any of them is to be executed.

The probes used in the performance of these cycles are:

- Probe placed on a fixed position on the machine, used for calibrating tools.
- Probe placed in the spindle, will be treated as a tool and will be used in the different measuring cycles.

The execution of a probing canned cycle does not alter the history of previous "G" functions, except for the radius compensation functions G41 and G42.

## 12.3    PROBE 1. Tool length calibrating canned cycle

This is used to calibrate the length and radius of the selected tool. The following operations are possible with this cycle.

- Calibrate the length of a tool.
- Calibrate the radius of a tool.
- To calibrate the tool radius and length.
- Measure the length wear of a tool.
- Measure the radius wear of a tool.
- Measure the radius wear and length wear of a tool.

To perform this cycle it is necessary to have a table-top probe, installed in a fixed position on the machine and with its faces parallel to axes X, Y, Z. Its position will be indicated in absolute coordinates with respect to machine zero by means of the general machine parameters:

PRBXMIN    Indicates the minimum coordinate occupied by the probe along the X axis.

PRBXMAX    Indicates the maximum coordinate occupied by the probe along the X axis.

PRBYMIN    Indicates the minimum coordinate occupied by the probe along the Y axis.

PRBYMAX    Indicates the maximum coordinate occupied by the probe along the Y axis.

PRBZMIN    Indicates the minimum coordinate occupied by the probe along the Z axis.

PRBZMAX    Indicates the maximum coordinate occupied by the probe along the Z axis.
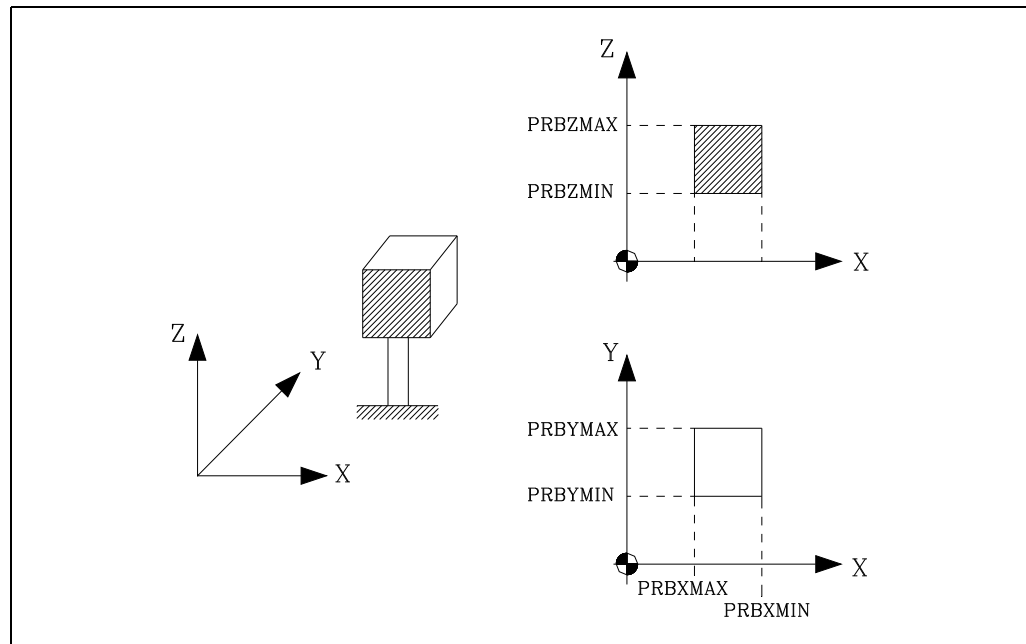


If it is the first time that the tool length has been calibrated, it is advisable to include an approximate value of its length (L) in the tool offset table.

## Programming format

The programming format for this cycle is.

(PROBE 1, B, I, F, J, K, L, C, D, E, S, M, C, N, X, U, Y, V, Z, W)

Certain parameters are only relevant in certain type of measurement. Later sections describe the various options of this cycle and the parameters to be set in each one.

### *Parameters X, U, Y, V, Z, W.*

Define the position of the probe. They are optional parameters that usually need not be defined. On certain machines, due to lack of repeatability in the mechanical positioning of the probe, the probe must be calibrated again before each tool calibration.

Instead of redefining machine parameters PRBXMIN, PRBXMAX, PRBYMIN, PRBYMAX, PRBZMAX, PRBZMIN every time the probe is calibrated, those coordinates may be indicated in parameters X, U, Y, V, Z, W, respectively.

The CNC does not modify the machine parameters. The CNC only takes into account the coordinates indicated in X, U, Y, V, Z, W during this calibration. If any of the X, U, Y, V, Z, W fields is left out, the CNC takes the value assigned to the corresponding machine parameter.

### 12.3.1 Calibrate the length or measure the length wear of a tool.

The type of operation (calibration or measurement) is selected when calling the cycle.

The calibration or measurement may be done on the tool shaft or on its tip. It is selected when calling the canned cycle.

The programming format depends on the operation to carry out:

- Tool length calibration along its shaft.

  (PROBE 1, B, I0, F, J0, X, U, Y, V, Z, W)

- Tool length calibration at its tip.

  (PROBE 1, B, I1, F, J0, D, S, N, X, U, Y, V, Z, W)

- Measurement of the tool length wear along its shaft.

  (PROBE 1, B, I0, F, J1, L, C, X, U, Y, V, Z, W)

- Measurement of the tool length wear at its tip.

  (PROBE 1, B, I1, F, J1, L, D, S, C, N, X, U, Y, V, Z, W)

**[ B5.5 ] Safety distance**

It must be programmed with a positive value greater than 0.

**[ I ] Type of calibration or wear measurement**

The calibration may be done on the tool shaft or on its tip.

I = 0          Calibration of the tool length or measurement of the tool length wear along its shaft.

I = 1          Calibration of the tool length or measurement of the tool length wear on its tip.

If not programmed, the cycle will assume the IO value.
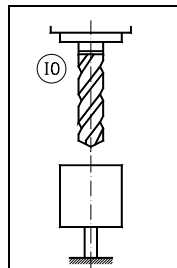


**I = 0. Tool calibration along its shaft.**

It is useful for drilling tools, ball end-mills, or tools whose diameter is smaller than the probe's probing surface.

This type of calibration is carried out with the spindle stopped.
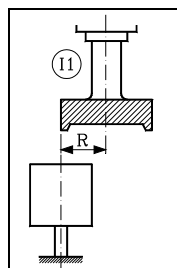


**I = 1. Tool calibration at its tip.**

It is useful for calibrating tools with several cutting edges (end-mills), or tools whose diameter is larger than the probe's probing surface.

This type of calibration may be carried out either with the spindle stopped or turning in the opposite direction to the cutting direction.

**[ F5.5 ] Probing feedrate**

It sets the probing feedrate. It must be programmed in mm/minute or inches/minute.

**[ J ] Type of operation**

The calibration may be done on the tool shaft or on its tip.

J = 0          Tool calibration.

J = 1          Tool wear measurement.

**[ L5.5 ] Maximum length wear allowed**

When set to zero, the tool will not be rejected due to length wear. When the measured wear is greater than the defined value, the tool will be rejected.

Only if J1 has been defined and tool life monitoring is being used. If not programmed, the canned cycle will assume the L0 value.

**[ D5.5 ] Distance from the tool shaft to the probing point**

It sets the radius or distance referred to the tool shaft being probed.

If not defined, probing is carried out on the tool tip.

**[ S±5.5 ] Speed and turning direction of the tool**

To probe with the spindle running, the tool must be turning in the opposite direction to the cutting direction.

• When set to 0, probing is carried out with the spindle stopped.

• When set with a positive value, the spindle starts in M3.

• When set with a negative value, the spindle starts in M4.

**[ C ] Behavior when exceeding the amount of wear allowed**

Only if "L" has been set to other than zero.

C = 0    It interrupts the execution for the user to select another tool.

C = 1    The cycle replaces the tool with another one of the same family.

If not programmed, the cycle will assume the C0 value.

**[ N ] Number of edges to measure**

When set to zero, it only takes one measurement. If not programmed, the cycle will assume the N0 value.

To measure each cutting edge when the spindle has feedback and s.m.p. M19TYPE (P43) =1.

**[ X U Y V Z W ] Probe position**

Optional parameters. See *"Programming format"* on page 289.

## Actions after finishing the cycle

### *Once the calibration cycle has ended*

It updates global arithmetic parameter P299 and assigns the measured length to the tool offset selected in the tool offset table.

P299        "Measured length" - "Previous length (L+K)".

L            Measured length.

K            0.

### *Once the wear measuring cycle has ended*

- When using tool live monitoring.

  In this case, it compares the measured value with the theoretical length assigned in the table. If the permitted maximum value is exceeds, it issues the message indicating that the tool has been rejected and acts as follows.

    C0      It interrupts the execution so the user can select another tool.

    C1      The cycle replaces the tool with another one from the same family. It sets the "rejected tool " indicator (status = R). It activates the general logic output PRTREJEC (M5564).

- If tool life monitoring is not available or the measuring difference does not exceed the maximum allowed.

  In this case, it updates global arithmetic parameter P299 and the length wear value of the tool offset selected in the tool offset table.

  P299        "Measured length" - "Theoretical length (L)".

  L            Theoretical length. It maintains the previous value.

  K            "Measured length" - "Theoretical length (L)". New wear value.

  If the dimension of each cutting edge was requested ("N" parameter) the measured values are assigned to global arithmetic parameters P271 and on.

### 12.3.2 Calibrate the radius or measure the radius wear of a tool

The type of operation (calibration or measurement) is selected when calling the cycle.

The programming format depends on the operation to carry out:

- Tool radius Calibration.

  (PROBE 1, B, I2, F, J0, K, E, S, N, X, U, Y, V, Z, W)

- Radius wear measurement.

  (PROBE 1, B, I2, F, J1, K, E, S, M, C, N, X, U, Y, V, Z, W)

**[ B5.5 ] Safety distance**

It must be programmed with a positive value greater than 0.

**[ I ] Type of calibration or wear measurement**

The calibration may be done on the tool shaft or on its tip.

I = 2       Calibrate the radius or measure the radius wear of a tool.

If not programmed, the cycle will assume the IO value.

**[ F5.5 ] Probing feedrate**

It sets the probing feedrate. It must be programmed in mm/minute or inches/minute.

**[ J ] Type of operation**

J = 0       Tool calibration.

J = 1       Tool wear measurement.

**[ K ] Probe side used**

It sets the probe side being used to probe the radius.

K = 0       Side X+.

K = 1       Side X-.

K = 2       Side Y+.

K = 3       Side Y-.

**[ E5.5 ] Distance from the tool tip to the probing point.**

Distance referred to the theoretical tool tip being probed.

This parameter may be very useful with cutters whose bottom is not horizontal.

If not programmed, the cycle will assume the E0 value.



**[ S±5.5 ] Speed and turning direction of the tool**

To probe with the spindle running, the tool must be turning in the opposite direction to the cutting direction.

- When set to 0, probing is carried out with the spindle stopped.
- When set with a positive value, the spindle starts in M3.
- When set with a negative value, the spindle starts in M4.

**[ M5.5 ] Maximum radius wear allowed**

When set to 0, the tool is not rejected due to radius wear. When the measured wear is greater than the defined value, the tool will be rejected.

Only if J1 has been defined and tool life monitoring is being used. If not programmed, a value of M0 will be assumed.

**[ C ] Behavior when exceeding the amount of wear allowed**

Only if "M" has been set to other than zero.

C = 0    It interrupts the execution for the user to select another tool.

C = 1    The cycle replaces the tool with another one of the same family.

If not programmed, the cycle will assume the C0 value.

**[ N ] Number of edges to measure**

When set to zero, it only takes one measurement. If not programmed, the cycle will assume the N0 value.

To measure each cutting edge when the spindle has feedback and s.m.p. M19TYPE (P43) =1.

**[ X U Y V Z W ] Probe position**

Optional parameters. See *"Programming format"* on page 289.

## Actions after finishing the cycle

### *Once the calibration cycle has ended*

It updates global arithmetic parameter P298 and assigns the measured radius to the tool offset selected in the tool offset table.

P298    "Measured radius" - "Previous radius (R+I)".

R    Measured radius.

I    0.

### *Once the wear measuring cycle has ended*

• When using tool live monitoring.

In this case, it compares the measured value with the theoretical radius assigned in the table. If the permitted maximum value is exceeds, it issues the message indicating that the tool has been rejected and acts as follows.

C0    It interrupts the execution so the user can select another tool.

C1    The cycle replaces the tool with another one from the same family. It sets the "rejected tool " indicator (status = R). It activates the general logic output PRTREJEC (M5564).

• If tool life monitoring is not available or the measuring difference does not exceed the maximum allowed.

In this case, it updates global arithmetic parameter P298 and the radius wear value of the tool offset selected in the tool offset table.

P298    "Measured radius" - "Theoretical radius (R)".

R    Theoretical radius. It maintains the previous value.

I    "Measured radius" - "Theoretical radius (R)". New wear value.

If the dimension of each cutting edge was requested ("N" parameter) the measured values are assigned to global arithmetic parameters P251 and on.

### 12.3.3 Measure or calibrate the tool radius wear and tool length wear.

The type of operation (calibration or measurement) is selected when calling the cycle.

The programming format depends on the operation to carry out:

- Tool radius Calibration.

  (PROBE 1, B, I3, F, J0, K, D, E, S, N, X, U, Y, V, Z, W)

- Radius wear measurement.

  (PROBE 1, B, I3, F, J1, K, L, D, E, S, M, C, N, X, U, Y, V, Z, W)

**[ B5.5 ] Safety distance**

It must be programmed with a positive value greater than 0.

**[ I ] Type of calibration or wear measurement**

The calibration may be done on the tool shaft or on its tip.

    I = 3          Measure or calibrate the tool radius wear and tool length wear.

If not programmed, the cycle will assume the IO value.

**[ F5.5 ] Probing feedrate**

It sets the probing feedrate. It must be programmed in mm/minute or inches/minute.

**[ J ] Type of operation**

    J = 0          Tool calibration.

    J = 1          Tool wear measurement.

**[ K ] Probe side used**

It sets the probe side being used to probe the radius.

    K = 0          Side X+.

    K = 1          Side X-.

    K = 2          Side Y+.

    K = 3          Side Y-.

**[ L5.5 ] Maximum length wear allowed**

When set to zero, the tool will not be rejected due to length wear. When the measured wear is greater than the defined value, the tool will be rejected.

Only if J1 has been defined and tool life monitoring is being used. If not programmed, the canned cycle will assume the L0 value.

**[ D5.5 ] Distance from the tool shaft to the probing point**

It sets the radius or distance referred to the tool shaft being probed.

If not defined, probing is carried out on the tool tip.

**[ E5.5 ] Distance from the tool tip to the probing point.**

Distance referred to the theoretical tool tip being probed.

This parameter may be very useful on tools with cutters whose bottom is not horizontal.

If not programmed, the cycle will assume the E0 value.
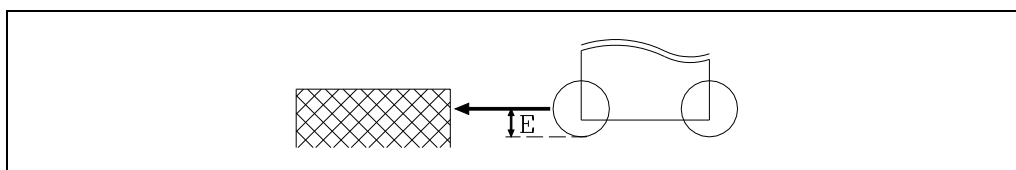
**FAGOR**

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2x

**[ S±5.5 ] Speed and turning direction of the tool**

To probe with the spindle running, the tool must be turning in the opposite direction to the cutting direction.

- When set to 0, probing is carried out with the spindle stopped.
- When set with a positive value, the spindle starts in M3.
- When set with a negative value, the spindle starts in M4.

**[ M5.5 ] Maximum radius wear allowed**

When set to 0, the tool is not rejected due to radius wear. When the measured wear is greater than the defined value, the tool will be rejected.

Only if J1 has been defined and tool life monitoring is being used. If not programmed, a value of M0 will be assumed.

**[ C ] Behavior when exceeding the amount of wear allowed**

Only if "M" or "L" has been set to other than zero.

| | |
|---|---|
| C = 0 | It interrupts the execution for the user to select another tool. |
| C = 1 | The cycle replaces the tool with another one of the same family. |

If not programmed, the cycle will assume the C0 value.

**[ N ] Number of edges to measure**

When set to zero, it only takes one measurement. If not programmed, the cycle will assume the N0 value.

To measure each cutting edge when the spindle has feedback and s.m.p. M19TYPE (P43) =1.

**[ X U Y V Z W ] Probe position**

Optional parameters. See

## Actions after finishing the cycle

### *Once the calibration cycle has ended*

It updates global arithmetic parameter P298 and assigns the measured radius to the tool offset selected in the tool offset table.

| | |
|---|---|
| P298 | "Measured radius" - "Previous radius (R+I)". |
| P299 | "Measured length" - "Previous length (L+K)". |
| R | Measured radius. |
| L | Measured length. |
| I | 0. |
| K | 0. |

### *Once the wear measuring cycle has ended*

- When using tool live monitoring.

  In this case, it compares the measured radius and length with the theoretical values in the table. If the permitted maximum value is exceeds, it issues the message indicating that the tool has been rejected and acts as follows.

  | | |
  |---|---|
  | C0 | It interrupts the execution so the user can select another tool. |
  | C1 | The cycle replaces the tool with another one from the same family. It sets the "rejected tool" indicator (status = R). It activates the general logic output PRTREJEC (M5564). |

- If tool life monitoring is not available or the measuring difference does not exceed the maximum allowed.

  In this case, it updates global arithmetic parameters P298 and P299 as well as the radius wear and length wear value of the tool offset selected in the tool offset table.

| | |
|---|---|
| P298 | "Measured radius" - "Theoretical radius (R)". |
| P299 | "Measured length" - "Theoretical length (L)". |
| R | Theoretical radius. It maintains the previous value. |
| I | "Measured radius" - "Theoretical radius (R)". New wear value. |
| L | Theoretical length. It maintains the previous value. |
| K | "Measured length" - "Theoretical length (L)". New wear value. |

If the dimension of each cutting edge was requested ("N" parameter) the lengths are assigned to global arithmetic parameters P271 and on; the radii are assigned to global arithmetic parameters P251 and on.

**12.**

PROBING

PROBE 1. Tool length calibrating canned cycle

**FAGOR** 

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

## 12.4    PROBE 2. Probe calibration canned cycle.

This is used to calibrate the probe located in the tool holding spindle. This probe which previously must be calibrated in length, will be the one used in probe measuring canned cycles.

The cycle measures the deviation which the probe ball axis has with respect to the tool holder axis, using a previously machined hole with known center and dimensions for its calibration.
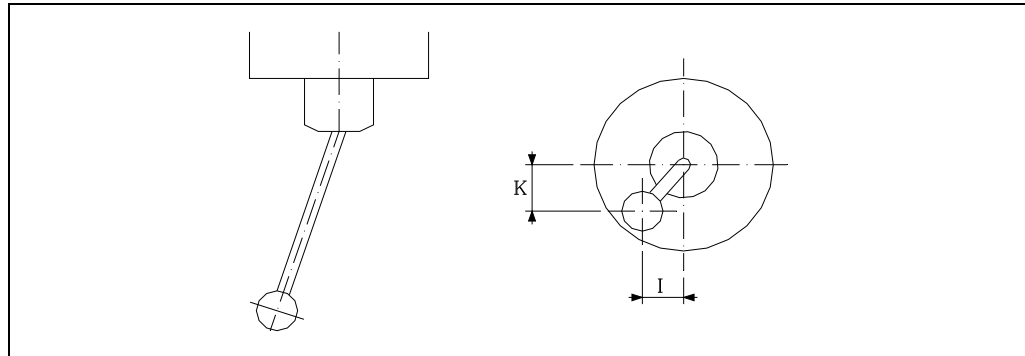
The CNC will treat each measuring probe used as just one more tool. The tool offset table fields corresponding to each probe will have the following meaning:

R        Radius of the sphere (ball) of the probe. This value will be loaded into the table manually.

L        Length of the probe. This value will be indicated by the tool length calibration cycle.

I        Deviation of the probe ball with respect to the tool-holder axis, along the abscissa axis. This value will be assigned by this cycle.

K        Deviation of the probe ball with respect to the tool-holder axis, along the ordinate axis. This value will be assigned by this cycle.

The following steps will be followed for its calibration:

1.  Once the characteristics of the probe have been consulted, the value for the sphere radius (R) will be entered manually in the corresponding tool offset.

2.  After selecting the corresponding tool number and tool offset the Tool Length Calibration Cycle will be performed, the value of (L) will be updated and the value of (K) will be initialized to 0.

3.  Execution of the probe calibration canned cycle, updating the "I" and "K" values.

The programming format for this cycle is:

(PROBE 2, X, Y, Z, B, J, E, H, F)

**[ X±5.5 ] Real coordinate, along the X axis, of the hole center.**

**[ Y±5.5 ] Real coordinate, along the Y axis, of the hole center.**

**[ Z±5.5 ] Real coordinate, along the Z axis, of the hole center.**

**[ B5.5 ] Safety distance**

Defines the safety distance. Must be programmed with a positive value and over 0.

**[ J5.5 ] Real diameter of the hole**

Defines the real diameter of the hole. Must be programmed with a positive value and over 0.

**[ E5.5 ] Withdrawal distance**

Defines the distance which the probe moves back after initial probing. Must be programmed with a positive value and over 0.

**[ H5.5 ] Initial probing feedrate**

Defines the feedrate for the initial probing movement. It must be programmed in mm/minute or inches/minute.

**[ F5.5 ] Probing feedrate**

It sets the probing feedrate. It must be programmed in mm/minute or inches/minute.

### 12.4.1 Basic operation



1. **Approach movement.**

   Probe's rapid movement (G00) from the cycle calling point to the center of the hole.

   The approaching movement is made in two stages:

   ·**1**· Movement in the main work plane.

   ·**2**· Movement along the longitudinal axis.

2. **Probing movement.**

   This movement consists of:

   ·**1**· Probe movement along the ordinate axis at the feedrate indicated by (H) until the probe signal is received.

   The maximum distance to be traveled in the probing movement is "B+(J/2)", if, after travelling that distance, the CNC does not receive the probe signal, it will display the corresponding error code and stop the movement of the axes.

   ·**2**· Return of the probe in rapid (G00) the distance indicated in (E).

   ·**3**· Probe movement along the ordinate axis at the feedrate indicated by (F) until the probe signal is received.

3. **Withdrawal movement.**

   Movement of the probe in rapid (G00) from the point where it probed to the real center of the hole.

4. **Second probing movement.**

   Same as above.

5. **Withdrawal movement.**

   Movement of the probe in rapid (G00) from the point where it probed to the real center of the hole along the ordinate axis.

6. **Third probing movement.**

   Same as above.

7. **Withdrawal movement.**

   Movement of the probe in rapid (G00) from the point where it probed to the real center of the hole.

8. **Fourth probing movement.**

   Same as above.

9. **Withdrawal movement.**

   This movement consists of:

   ·**1**· Movement of the probe in rapid (G00) from the point where it probed to the real center of the hole.

·**2**· Movement along the longitudinal axis to the coordinate of the point (along this axis) from where the cycle was called.

·**3**· Movement in the main work plane to the point where the cycle is called.

### *Correction of the tool offset.*

Once the cycle has been completed, the CNC will have updated the "I" and "K" values corresponding to the tool offset selected at the time on the tool offset table.

### *Arithmetic parameters modified by the cycle*

The cycle returns in arithmetic parameter P299 the best value to be assigned to general machine parameter PRODEL.

## 12.5 PROBE 3. Surface measuring canned cycle

A probe placed in the spindle will be used, which must be previously calibrated by means of canned cycles:

Tool length calibrating canned cycle.

Probe calibrating canned cycle.

This cycle allows correcting the value of the tool offset of the tool that has been used in the surface machining process. This correction will be used only when the measurement error exceeds a programmed value.

The programming format for this cycle is:

(PROBE 3, X, Y, Z, B, K, F, C, D, L)

**[ X±5.5 ] Theoretical coordinate, along the X axis, of the point over which it is required to measure.**

**[ Y±5.5 ] Theoretical coordinate, along the Y axis, of the point over which it is required to measure.**

**[ Z±5.5 ] Theoretical coordinate, along the Z axis, of the point over which it is required to measure.**

**[ B5.5 ] Safety distance**

Defines the safety distance. Must be programmed with a positive value and over 0.

The probe must be placed, with respect to the point to be measured, at a distance greater than this value when the cycle is called.

**[ K ] Probing axis**

Defines the axis with which it is required to measure the surface and will be defined by means of the following code:

K = 0          With the abscissa axis of the work plane.

K = 1          With the ordinate axis of the work plane.

K = 2          With the longitudinal axis of the work plane.

If this is not programmed, the canned cycle will take the value of K0.



**[ F5.5 ] Probing feedrate**

It sets the probing feedrate. It must be programmed in mm/minute or inches/minute.

**[ C ] Actions after finishing the probing operation**

Indicates where the probing cycle must finish.

C = 0          Will return to the same point where the call to the cycle was made.

C = 1          The cycle will finish over the measured point returning the longitudinal axis to the cycle calling point.

If not programmed, the canned cycle will take the value of "C0".

**[ D4 ] Tool offset**

Defines the number of the tool offset to be corrected, once the measurement cycle is completed. If this is not programmed or is programmed with a value of 0, the CNC will understand that it is not required to make this correction.

**[ L5.5 ] Error tolerance**

Defines the tolerance that will be applied to the error measured. It will be programmed with an absolute value and the tool offset will be corrected only when the error exceeds this value.

If this is not programmed, the canned cycle will take the value of 0.

**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2x

### 12.5.1 Basic operation
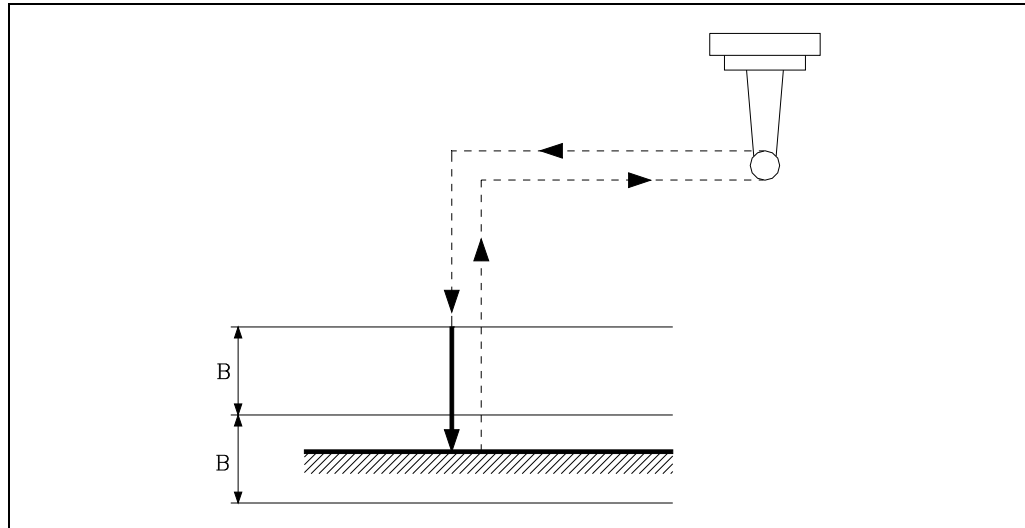


**1.** Approach movement.

Rapid probe movement (G00) from the cycle calling point to the approach point.

This point is located in front of the point to be measured, at a safety distance (B) from it and along the probing axis (K).

The approaching movement is made in two stages:

·1· Movement in the main work plane.

·2· Movement along the longitudinal axis.

**2.** Probing movement.

Movement of the probe along the selected axis (K) at the indicated feedrate (F), until the probe signal is received.

The maximum distance to be traveled in the probing movement is 2B; if, after travelling that distance, the CNC does not receive the probe signal, it will display the corresponding error code and stop the movement of the axes.

Once probing has been made, the CNC will assume as their theoretical position the real position of the axes when the probe signal is received .

**3.** Withdrawal movement.

Movement of the probe in rapid (G00) from the point where it probed to the point where the cycle was called.

The withdrawal movement is made in three stages:

·1· Movement along the probing axis to the approach point.

·2· Movement along the longitudinal axis to the coordinate of the point (along this axis) from where the cycle was called.

·3· When (C0) is programmed, movement is made in the main work plane to the point where the cycle is called.

#### *Arithmetic parameters modified by the cycle*

Once the cycle has been completed, the CNC will return the real values obtained after measurement, in the following global arithmetic parameters.

P298     Actual (real) surface coordinate.

P299     Detected error. Difference between the actual surface coordinate and the programmed theoretical coordinate.

### *Correction of the tool offset.*

If the Tool Offset Number (D) was selected, the CNC will modify the values of this tool offset, whenever the measurement error is equal to or greater than the tolerance (L).

Depending on the axis the measurement is made with (K), the correction will be made on the length or radius value.

- If the measurement is made with the axis longitudinal to the work plane, the length wear (K) of the indicated tool offset (D) will be modified.

- If the measurement is made with one of the axes that make up the work plane, the radius wear (I) of the indicated tool offset (D) will be modified.

## 12.6    PROBE 4. Outside corner measuring canned cycle

A probe placed in the spindle will be used, which must be previously calibrated by means of canned cycles:

Tool length calibrating canned cycle.

Probe calibrating canned cycle.

The programming format for this cycle is:

(PROBE 4, X, Y, Z, B, F)

**[ X±5.5 ] Theoretical coordinate, along the X axis, of the corner to be measured**

**[ Y±5.5 ] Theoretical coordinate, along the Y axis, of the corner to be measured**

**[ Z±5.5 ] Theoretical coordinate, along the Z axis, of the corner to be measured**

Depending on the corner of the part it is required to measure, the probe must be placed in the corresponding shaded area (see figure) before calling the cycle.



**[ B5.5 ] Safety distance**

Defines the safety distance. Must be programmed with a positive value and over 0.

The probe must be placed, with respect to the point to be measured, at a distance greater than this value when the cycle is called.

**[ F5.5 ] Probing feedrate**

It sets the probing feedrate. It must be programmed in mm/minute or inches/minute.

### 12.6.1    Basic operation



1. **Approach movement.**

   Movement of the probe in rapid (G00) from the point where the cycle is called to the first approach point, situated at a distance (B) from the first face to be probed.

   The approaching movement is made in two stages:

   ·1· Movement in the main work plane.

   ·2· Movement along the longitudinal axis.

2. **Probing movement.**

   Movement of the probe along the abscissa axis at the indicated feedrate (F), until the probe signal is received.

   The maximum distance to be traveled in the probing movement is 2B; if, after travelling that distance, the CNC does not receive the probe signal, it will display the corresponding error code and stop the movement of the axes.

3. **Withdrawal movement.**

   Rapid probe movement (G00) from the probing point to the first approach point.

4. **Second approach movement.**

   Rapid probe move (G00) from the first approach point to the second.

   The approaching movement is made in two stages:

   ·1· Movement along the ordinate plane.

   ·2· Movement along the abscissa axis.

5. **Second probing movement.**

   Movement of the probe along the abscissa axis at the indicated feedrate (F), until the probe signal is received.

   The maximum distance to be traveled in the probing movement is 2B; if, after travelling that distance, the CNC does not receive the probe signal, it will display the corresponding error code and stop the movement of the axes.

6. **Withdrawal movement.**

   Movement of the probe in rapid (G00) from the point where it probed for the second time to the point where the cycle was called.

   The withdrawal movement is made in three stages:

   ·1· Movement along the probing axis to the second approach point.

   ·2· Movement along the longitudinal axis to the coordinate of the point (along this axis) from where the cycle was called.

   ·3· Movement in the main work plane to the point where the cycle is called.

**Arithmetic parameters modified by the cycle**

Once the cycle has been completed, the CNC will return the real values obtained after measurement, in the following global arithmetic parameters.

P296    Real coordinate of the corner along the abscissa axis.

P297    Real coordinate of the corner along the ordinate axis.

P298    Error detected along the abscissa axis. Difference between the real coordinate of the corner and the theoretical programmed coordinate.

P299    Error detected along the ordinate axis. Difference between the real coordinate of the corner and the theoretical programmed coordinate.

## 12.7    PROBE 5. Inside corner measuring canned cycle.

A probe placed in the spindle will be used, which must be previously calibrated by means of canned cycles:

Tool length calibrating canned cycle.

Probe calibrating canned cycle.

The programming format for this cycle is:

(PROBE 5, X, Y, Z, B, F)

**[ X±5.5 ] Theoretical coordinate, along the X axis, of the corner to be measured**

**[ Y±5.5 ] Theoretical coordinate, along the Y axis, of the corner to be measured**

**[ Z±5.5 ] Theoretical coordinate, along the Z axis, of the corner to be measured**

The probe must be placed within the pocket before calling the cycle.



**[ B5.5 ] Safety distance**

Defines the safety distance. Must be programmed with a positive value and over 0.

The probe must be placed, with respect to the point to be measured, at a distance greater than this value when the cycle is called.

**[ F5.5 ] Probing feedrate**

It sets the probing feedrate. It must be programmed in mm/minute or inches/minute.

**FAGOR**

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

### 12.7.1 Basic operation



1. **Approach movement.**

   Movement of the probe in rapid (G00) from the point where the cycle is called to the first approach point, situated at a distance (B) from both faces to be probed.

   The approaching movement is made in two stages:

   ·1· Movement in the main work plane.

   ·2· Movement along the longitudinal axis.

2. **Probing movement.**

   Movement of the probe along the abscissa axis at the indicated feedrate (F), until the probe signal is received.

   The maximum distance to be traveled in the probing movement is 2B; if, after travelling that distance, the CNC does not receive the probe signal, it will display the corresponding error code and stop the movement of the axes.

3. **Withdrawal movement.**

   Rapid probe movement (G00) from the probing point to the approach point.

4. **Second probing movement.**

   Movement of the probe along the abscissa axis at the indicated feedrate (F), until the probe signal is received.

   The maximum distance to be traveled in the probing movement is 2B; if, after travelling that distance, the CNC does not receive the probe signal, it will display the corresponding error code and stop the movement of the axes.

5. **Withdrawal movement.**

   Movement of the probe in rapid (G00) from the point where it probed for the second time to the point where the cycle was called.

   The withdrawal movement is made in three stages:

   ·1· Movement along the probing axis to the approach point.

   ·2· Movement along the longitudinal axis to the coordinate of the point (along this axis) from where the cycle was called.

   ·3· Movement in the main work plane to the point where the cycle is called.

### *Arithmetic parameters modified by the cycle*

Once the cycle has been completed, the CNC will return the real values obtained after measurement, in the following global arithmetic parameters.

P296    Real coordinate of the corner along the abscissa axis.

P297    Real coordinate of the corner along the ordinate axis.

P298    Error detected along the abscissa axis. Difference between the real coordinate of the corner and the theoretical programmed coordinate.

P299    Error detected along the ordinate axis. Difference between the real coordinate of the corner and the theoretical programmed coordinate.

## 12.8   PROBE 6. Angle measuring canned cycle

A probe placed in the spindle will be used, which must be previously calibrated by means of canned cycles:

Tool length calibrating canned cycle.

Probe calibrating canned cycle.

The programming format for this cycle is:

(PROBE 6, X, Y, Z, B, F)

**[ X±5.5 ] Theoretical coordinate, along the X axis, of the vertex of the angle to be measured.**

**[ Y±5.5 ] Theoretical coordinate, along the Y axis, of the vertex of the angle to be measured**

**[ Z±5.5 ] Theoretical coordinate, along the Z axis, of the vertex of the angle to be measured.**

**[ B5.5 ] Safety distance**

Defines the safety distance. Must be programmed with a positive value and over 0.

The probe must be placed, with respect to the point to be measured, at a distance greater than double this value when the cycle is called.

**[ F5.5 ] Probing feedrate**

It sets the probing feedrate. It must be programmed in mm/minute or inches/minute.

## 12.8.1 Basic operation



1. **Approach movement.**

   Movement of the probe in rapid (G00) from the point where the cycle is called to the first approach point, situated at a distance (B) from the programmed vertex and at (2B) from the face to be probed.

   The approaching movement is made in two stages:

   ·1· Movement in the main work plane.

   ·2· Movement along the longitudinal axis.

2. **Probing movement.**

   Movement of the probe along the abscissa axis at the indicated feedrate (F), until the probe signal is received.

   The maximum distance to be traveled in the probing movement is 3B; if, after travelling that distance, the CNC does not receive the probe signal, it will display the corresponding error code and stop the movement of the axes.

3. **Withdrawal movement.**

   Rapid probe movement (G00) from the probing point to the first approach point.

4. **Second approach movement.**

   Rapid probe move (G00) from the first approach point to the second. It is at a distance (B) from the first one.

5. **Second probing movement.**

   Movement of the probe along the abscissa axis at the indicated feedrate (F), until the probe signal is received.

   The maximum distance to be traveled in the probing movement is 4B; if, after travelling that distance, the CNC does not receive the probe signal, it will display the corresponding error code and stop the movement of the axes.

6. **Withdrawal movement.**

   Movement of the probe in rapid (G00) from the point where it probed for the second time to the point where the cycle was called.

   The withdrawal movement is made in three stages:

   ·1· Movement along the ordinate axis to the second approach point.

   ·2· Movement along the longitudinal axis to the coordinate of the point (along this axis) from where the cycle was called.

   ·3· Movement in the main work plane to the point where the cycle is called.

### Arithmetic parameters modified by the cycle

Once the cycle has been completed, the CNC will return the real values obtained after measurement, in the following global arithmetic parameter.

P295    Inclination angle which the part has in relation to the abscissa axis.

### Considerations for the cycle

This cycle may be used to measure angles between ±45º.

- If the angle to be measured is > 45º, the CNC will display the corresponding error.
- If the angle to be measured is < -45º, the probe will collide with the part.

## 12.9    PROBE 7. Corner and angle measuring canned cycle.

A probe placed in the spindle will be used, which must be previously calibrated by means of canned cycles:

Tool length calibrating canned cycle.

Probe calibrating canned cycle.

The programming format for this cycle is:

(PROBE 7, K, X, Y, Z, B, F)

**[ X±5.5 ] Theoretical coordinate, along the X axis, of the corner to be measured**

**[ Y±5.5 ] Theoretical coordinate, along the Y axis, of the corner to be measured**

**[ Z±5.5 ] Theoretical coordinate, along the Z axis, of the corner to be measured**

If it is an outside corner, depending on the corner of the part it is required to measure, the probe must be placed in the corresponding shaded area (see figure) before calling the cycle.
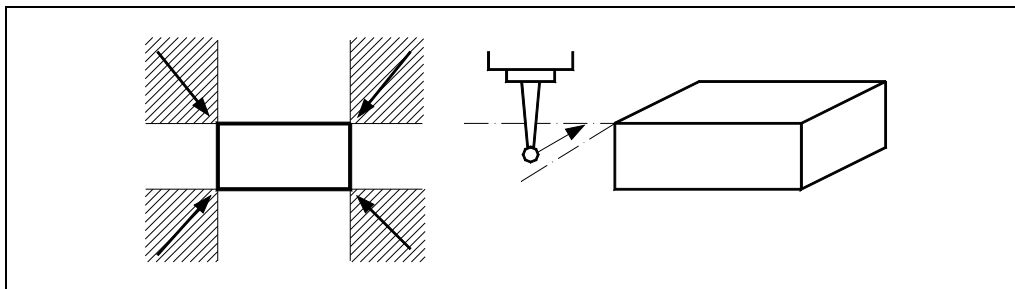


If it is an inside corner, the probe must be placed within the pocket before calling the cycle.
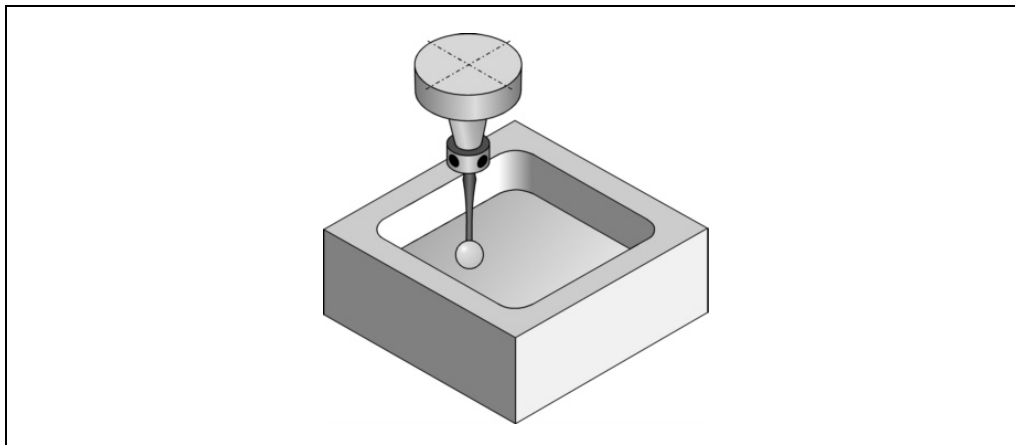


**[ K ] Type of corner**

Defines the type of corner to be measured:

K = 0:          Measuring an outside corner.

K = 1:          Measuring an inside corner.

**[ B5.5 ] Safety distance**

Defines the safety distance. Must be programmed with a positive value and over 0.

The probe must be placed, with respect to the point to be measured, at a distance greater than double this value when the cycle is called.
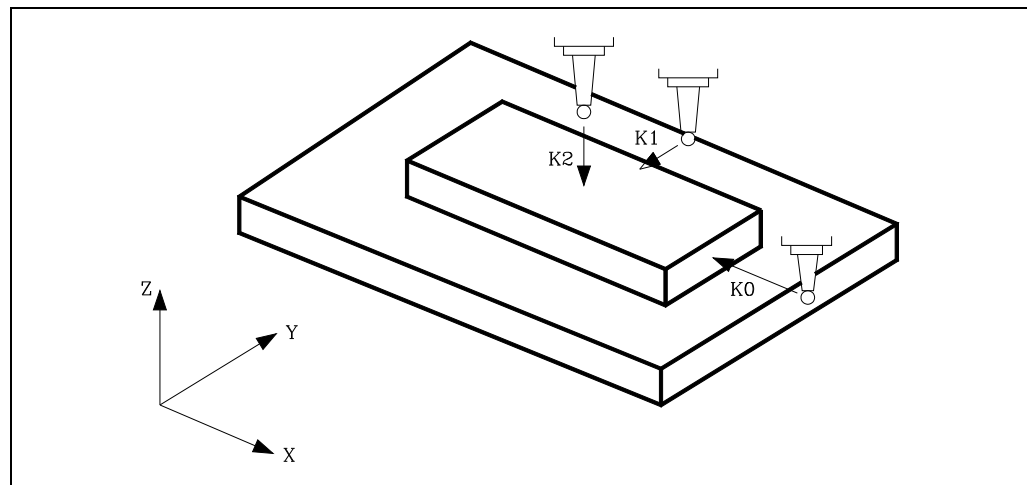
**[ F5.5 ] Probing feedrate**

It sets the probing feedrate. It must be programmed in mm/minute or inches/minute.

### 12.9.1 Basic operation (measuring an outside corner)

1. **Approach movement.**

   Movement of the probe in rapid (G00) from the point where the cycle is called to the first approach point, situated at a distance (2B) from the first side to be probed.

   The approaching movement is made in two stages:

   ·1· Movement in the main work plane.

   ·2· Movement along the longitudinal axis.

2. **Probing movement.**

   Movement of the probe along the abscissa axis at the indicated feedrate (F), until the probe signal is received.

   The maximum distance to be traveled in the probing movement is 3B; if, after travelling that distance, the CNC does not receive the probe signal, it will display the corresponding error code and stop the movement of the axes.

3. **Withdrawal movement.**

   Rapid probe movement (G00) from the probing point to the first approach point.

4. **Second approach movement.**

   Movement of the probe in rapid (G00) from the first approach point to the second, situated at a distance (2B) from the second side to be probed.

   The approaching movement is made in two stages:

   ·1· Movement along the ordinate plane.

   ·2· Movement along the abscissa axis.

5. **Second probing movement.**

   Movement of the probe along the abscissa axis at the indicated feedrate (F), until the probe signal is received.

   The maximum distance to be traveled in the probing movement is 3B; if, after travelling that distance, the CNC does not receive the probe signal, it will display the corresponding error code and stop the movement of the axes.

6. **Withdrawal movement.**

   Rapid probe movement (G00) from the probing point to the second approach point.

7. **Third approach movement.**

   Rapid probe move (G00) from the second approach point to the third. It is at a distance (B) from the previous one.

8. **Third probing movement.**

   Movement of the probe along the abscissa axis at the indicated feedrate (F), until the probe signal is received.

   The maximum distance to be traveled in the probing movement is 4B; if, after travelling that distance, the CNC does not receive the probe signal, it will display the corresponding error code and stop the movement of the axes.

**9.** Withdrawal movement.

Movement of the probe in rapid (G00) from the third probing point to the point where the cycle was called.

The withdrawal movement is made in three stages:

·1· Movement along probing axis to the third approach point.

·2· Movement along the longitudinal axis to the coordinate of the point (along this axis) from where the cycle was called.

·3· Movement in the main work plane to the point where the cycle is called.

### Arithmetic parameters modified by the cycle

Once the cycle has been completed, the CNC will return the real values obtained after measurement, in the following global arithmetic parameters.

| | |
|---|---|
| P295 | Inclination angle which the part has in relation to the abscissa axis. |
| P296 | Real coordinate of the corner along the abscissa axis. |
| P297 | Real coordinate of the corner along the ordinate axis. |
| P298 | Error detected along the abscissa axis. Difference between the real coordinate of the corner and the theoretical programmed coordinate. |
| P299 | Error detected along the ordinate axis. Difference between the real coordinate of the corner and the theoretical programmed coordinate. |

### Considerations for the cycle

This cycle may be used to measure angles between ±45º.

- If the angle to be measured is > 45º, the CNC will display the corresponding error.

- If the angle to be measured is < -45º, the probe will collide with the part.

## 12.9.2   Basic operation (measuring an inside corner)

1. **Approach movement.**
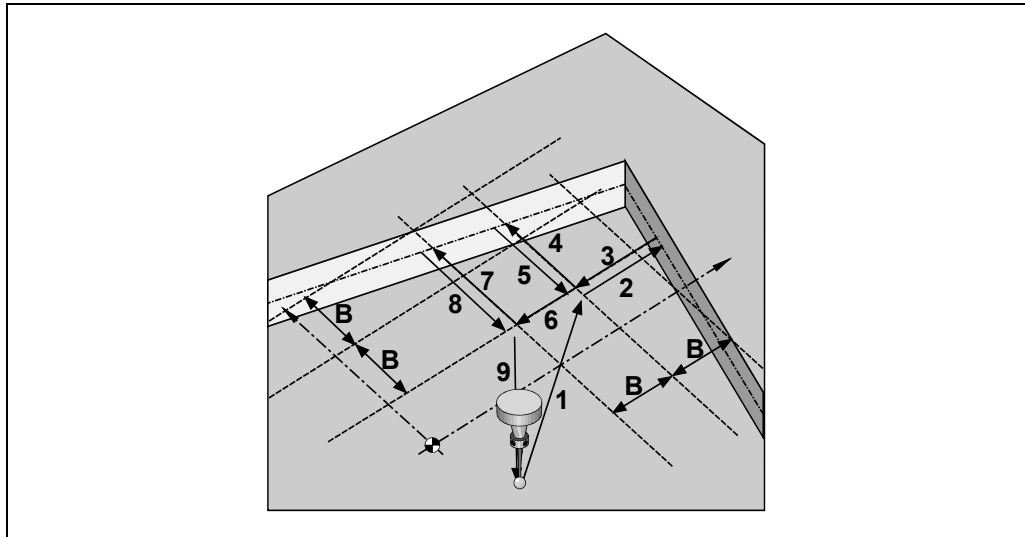
   Movement of the probe in rapid (G00) from the point where the cycle is called to the approach point, situated at a distance (B) from the first face to be probed.

   The approaching movement is made in two stages:

   ·1· Movement in the main work plane.

   ·2· Movement along the longitudinal axis.

2. **Probing movement.**

   Movement of the probe along the abscissa axis at the indicated feedrate (F), until the probe signal is received.

   The maximum probing distance is 2B. If the CNC does not receive the probe signal before reaching moving this probing distance, it displays the relevant error message and stops the axes.

3. **Withdrawal movement.**

   Rapid probe movement (G00) from the probing point to the first approach point.

4. **Second probing movement.**

   Movement of the probe along the abscissa axis at the indicated feedrate (F), until the probe signal is received.

   The maximum probing distance is 2B. If the CNC does not receive the probe signal before reaching moving this probing distance, it displays the relevant error message and stops the axes.

5. **Withdrawal movement.**

   Movement of the probe in rapid (G00) from the second probing point to the first approach point

6. **Second approach movement.**

   Rapid probe move (G00) from the first approach point to the second. It is at a distance (B) from the previous one.

7. **Third probing movement.**

   Movement of the probe along the abscissa axis at the indicated feedrate (F), until the probe signal is received.

   The maximum distance to be traveled in the probing movement is 3B; if, after travelling that distance, the CNC does not receive the probe signal, it will display the corresponding error code and stop the movement of the axes.

**8.** Withdrawal movement.

Movement of the probe in rapid (G00) from the third probing point to the point where the cycle was called.

The withdrawal movement is made in three stages:

·1· Movement along the probing axis to the second approach point.

·2· Movement along the longitudinal axis to the coordinate of the point (along this axis) from where the cycle was called.

·3· Movement in the main work plane to the point where the cycle is called.

### Arithmetic parameters modified by the cycle

Once the cycle has been completed, the CNC will return the real values obtained after measurement, in the following global arithmetic parameters.

| | |
|---|---|
| P295 | Inclination angle which the part has in relation to the abscissa axis. |
| P296 | Real coordinate of the corner along the abscissa axis. |
| P297 | Real coordinate of the corner along the ordinate axis. |
| P298 | Error detected along the abscissa axis. Difference between the real coordinate of the corner and the theoretical programmed coordinate. |
| P299 | Error detected along the ordinate axis. Difference between the real coordinate of the corner and the theoretical programmed coordinate. |

### Considerations for the cycle

This cycle may be used to measure angles between ±45º.

• If the angle to be measured is > 45º, the CNC will display the corresponding error.

• If the angle to be measured is < -45º, the probe will collide with the part.

## 12.10   PROBE 8. Hole measuring cycle

A probe placed in the spindle will be used, which must be previously calibrated by means of canned cycles:

Tool length calibrating canned cycle.

Probe calibrating canned cycle.
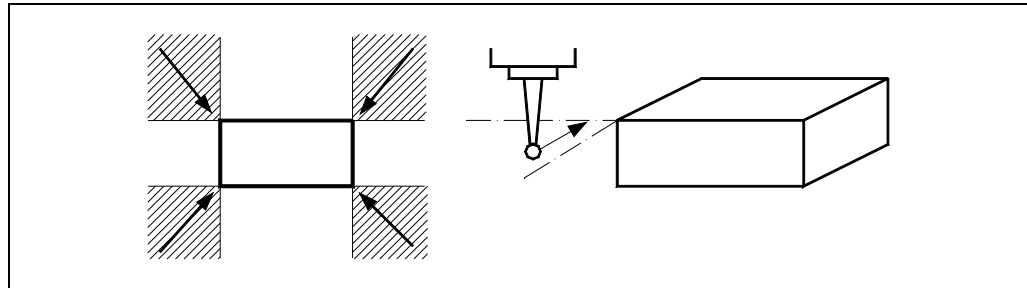
The programming format for this cycle is:

(PROBE 8, X, Y, Z, B, J, E, C, H, F)

**[ X±5.5 ] Theoretical coordinate, along the X axis, of the hole center**

**[ Y±5.5 ] Theoretical coordinate, along the Y axis, of the hole center**

**[ Z±5.5 ] Theoretical coordinate, along the Z axis, of the hole center**

**[ B5.5 ] Safety distance**

Defines the safety distance. Must be programmed with a positive value and over 0.

**[ J5.5 ] Theoretical diameter of the hole**

Defines the theoretical diameter of the hole. Must be programmed with a positive value and over 0.

This cycle allows holes to be measured with diameters of no more than (J+B).

**[ E5.5 ] Withdrawal distance**

Defines the distance which the probe moves back after initial probing. Must be programmed with a positive value and over 0.

**[ C ] Actions after finishing the probing operation**

Indicates where the probing cycle must finish.

C = 0          Will return to the same point where the call to the cycle was made.

C = 1          The cycle will finish at the real center of the hole.

If not programmed, the canned cycle will take the value of "C0".

**[ H5.5 ] Initial probing feedrate**

Defines the feedrate for the initial probing movement. It must be programmed in mm/minute or inches/minute.

**[ F5.5 ] Probing feedrate**

It sets the probing feedrate. It must be programmed in mm/minute or inches/minute.

### 12.10.1 Basic operation



1. **Approach movement.**

   Probe's rapid movement (G00) from the cycle calling point to the center of the hole.

   The approaching movement is made in two stages:

   ·1· Movement in the main work plane.

   ·2· Movement along the longitudinal axis.

2. **Probing movement.**

   This movement consists of:

   ·1· Movement of the probe along the ordinate axis at the indicated feedrate (H), until the probe signal is received.

   The maximum distance to be traveled in the probing movement is "B+(J/2), if, after travelling that distance, the CNC does not receive the probe signal, it will display the corresponding error code and stop the movement of the axes.

   ·2· Return of the probe in rapid (G00) the distance indicated in (E).

   ·3· Movement of the probe along the abscissa axis at the indicated feedrate (F), until the probe signal is received.

3. **Withdrawal movement.**

   Movement of the probe in rapid (G00) from the point where it probed to the theoretical center of the hole.

4. **Second probing movement.**

   Same as above.

5. **Withdrawal movement.**

   Movement of the probe in rapid (G00) from the point where it probed to the real center (calculated) of the hole along the ordinate axis.

6. **Third probing movement.**

   Same as above.

7. **Withdrawal movement.**

   Movement of the probe in rapid (G00) from the point where it probed to the theoretical center of the hole.

8. **Fourth probing movement.**

   Same as above.

**9.** Withdrawal movement.

This movement consists of:

·1· Movement of the probe in rapid (G00) from the point where it probed to the real center (calculated) of the hole.

·2· Should (C0) be programmed, the probe will be moved to the point where the cycle was called.

Movement along the longitudinal axis to the coordinate of the point (along this axis) from where the cycle was called.

Movement in the main work plane to the point where the cycle is called.

### *Arithmetic parameters modified by the cycle*

Once the cycle has been completed, the CNC will return the real values obtained after measurement, in the following global arithmetic parameters.

P294        Hole diameter.

P295        Hole diameter error. Difference between the real diameter and programmed diameter.

P296        Real coordinate of the center along the abscissa axis.

P297        Real center coordinate along the ordinate axis.

P298        Error detected along the abscissa axis. Difference between the real coordinate of the center and the programmed theoretical coordinate.

P299        Error detected along the ordinate axis. Difference between the real coordinate of the center and the programmed theoretical coordinate.

## 12.11   PROBE 9. Boss measuring cycle

A probe placed in the spindle will be used, which must be previously calibrated by means of canned cycles:

Tool length calibrating canned cycle.

Probe calibrating canned cycle.

The programming format for this cycle is:

(PROBE 9, X, Y, Z, B, J, E, C, H, F)

**[ X±5.5 ] Theoretical coordinate, along the X axis, of the boss center**

**[ Y±5.5 ] Theoretical coordinate, along the Y axis, of the boss center**

**[ Z±5.5 ] Theoretical coordinate, along the Z axis, of the boss center**

**[ B5.5 ] Safety distance**

Defines the safety distance. Must be programmed with a positive value and over 0.

**[ J5.5 ] Theoretical diameter of the boss.**

Defines the theoretical diameter of the boss. Must be programmed with a positive value and over 0.

This cycle allows bosses to be measured with diameters of no more than (J+B).

**[ E5.5 ] Withdrawal distance**

Defines the distance which the probe moves back after initial probing. Must be programmed with a positive value and over 0.

**[ C ] Actions after finishing the probing operation**

Indicates where the probing cycle must finish.

| | |
|---|---|
| C = 0 | Will return to the same point where the call to the cycle was made. |
| C = 1 | The cycle will finish by positioning the probe over the center of the boss, at a distance (B) from the programmed theoretical coordinate. |

If not programmed, the canned cycle will take the value of "C0".

**[ H5.5 ] Initial probing feedrate**

Defines the feedrate for the initial probing movement. It must be programmed in mm/minute or inches/minute.

**[ F5.5 ] Probing feedrate**

It sets the probing feedrate. It must be programmed in mm/minute or inches/minute.

### 12.11.1 Basic operation

1. Positioning over the center of the boss.

   Movement of the probe in rapid (G00) from the point where the cycle is called to the center of the boss.

   The approaching movement is made in two stages:

   ·1· Movement in the main work plane.

   ·2· Movement along the longitudinal axis up to a distance (B) from the programmed surface.

2. Movement to the first approach point.

   This movement of the probe that is made in rapid (G00) consists of:

   ·1· Movement along the ordinate plane.

   ·2· Movement of the longitudinal axis the distance (2B).

3. Probing movement.

   This movement consists of:

   ·1· Movement of the probe along the ordinate axis at the indicated feedrate (H), until the probe signal is received.

   The maximum distance to be traveled in the probing movement is "B+(J/2), if, after travelling that distance, the CNC does not receive the probe signal, it will display the corresponding error code and stop the movement of the axes.

   ·2· Return of the probe in rapid (G00) the distance indicated in (E).

   ·3· Movement of the probe along the abscissa axis at the indicated feedrate (F), until the probe signal is received.

4. Movement to second approach point.

   This movement of the probe that is made in rapid (G00) consists of:

   ·1· Withdrawal to the first approach point.

   ·2· Movement to a distance (B) above the boss, to the second approach point.

5. Second probing movement.

   Same as the first probing.

6. Third approach movement.

   Same as above.

7. Third probing movement.

   Same as above.

8. Fourth approach movement.

   Same as above.

**9.** Fourth probing movement.

Same as above.

**10.** Withdrawal movement.

This movement consists of:

·1· Withdrawal to the fourth approach point.

·2· Movement of the probe in rapid (G00) and at a distance (B) above the boss to the real center (calculated) of the boss.

·3· Should (C0) be programmed, the probe will be moved to the point where the cycle was called.

Movement along the longitudinal axis to the coordinate of the point (along this axis) from where the cycle was called.

Movement in the main work plane to the point where the cycle is called.

### *Arithmetic parameters modified by the cycle*

Once the cycle has been completed, the CNC will return the real values obtained after measurement, in the following global arithmetic parameters.

| | |
|---|---|
| P294 | Boss diameter. |
| P295 | Boss diameter error. Difference between the real diameter and programmed diameter. |
| P296 | Real coordinate of the center along the abscissa axis. |
| P297 | Real center coordinate along the ordinate axis. |
| P298 | Error detected along the abscissa axis. Difference between the real coordinate of the center and the programmed theoretical coordinate. |
| P299 | Error detected along the ordinate axis. Difference between the real coordinate of the center and the programmed theoretical coordinate. |

## 12.12 PROBE 10. Rectangular part centering canned cycle

Cycle that, with a digital probe, minimizes the preparation time of rectangular part calculating the real coordinates of the center, of the surface and of the part inclination.

(PROBE 10, I, J, X, Y, Z, K, L, B, D, E, H, F, Q)

### Initial conditions

- The probe must properly calibrated in radius and length.
- The probe position before the first probing movement must be as centered as possible in X and Y.

### Considerations for the cycle

- After the probing movements, the probe withdraws from the part in G0 before moving to the safety Z position.
- Depending on the PRBMOD variable, no error is issued in the following cases, even if machine parameter PROBERR=YES.
    - When a G75 probing move finishes before the probe has touched part.
    - When a G76 probing move finishes but the probe is still touching the part.

### Parameters

#### [ X±5.5 ] X coordinate where probing will begin

X coordinate of the probe position where the first probing move will start.

If not programmed, it will assume the current X position of the probe.

#### [ Y±5.5 ] Y coordinate where probing will begin

Y coordinate of the probe position where the first probing move will start.

If not programmed, it will assume the current Y position of the probe.

#### [ Z±5.5 ] Z coordinate where probing will begin

Z coordinate of the probe position where the first probing move will start.

If not programmed, it will assume the current Z position of the probe.

#### [ I5.5 ] X length of the rectangular part.

If not programmed or programmed with a 0 value, it generates the corresponding error message.

#### [ J5.5 ] Y length of the rectangular part.

If not programmed or programmed with a 0 value, it generates the corresponding error message.

#### [ K1 ] Axis and direction of the first probing movement.

The values are:
- For X+ : 0
- For X- : 1
- For Y+ : 2
- For Y- : 3

If not programmed, it assumes 0.

#### [ L1 ] It indicates whether the part surface is measured or not

- Value 0: It is not measured
- Value 1: It is measured
- If not programmed, a value of 0 is assumed

**[ B5.5 ] Approach distance**

Part approaching distance in each probing movement. If not programmed or programmed as 0, it assumes the approach distance value from the probe position to the part.

**[ D±5.5 ] Probe's up distance**

Distance for the probe to go up in Z for its movements over the part. If not programmed or programmed with a 0 value, it generates the corresponding error message.

**[ E±5.5 ] Withdrawal distance**

Distance the probe retracts after finding the part, to make the measurement. If not programmed or programmed with a 0 value, it generates the corresponding error message.

**[ H5 ] Probing feedrate for part search**

If not programmed or programmed with a 0 value, it generates the corresponding error message.

**[ F5 ] Probe's measuring feedrate**

If not programmed or programmed with a 0 value, it generates the corresponding error message.

**[ Q5] Probe feedrate when going to the approach points**

Probe feedrate when going to the approach points. If not programmed, it assumes rapid traverse (G0).

### 12.12.1 Basic operation

1. Approach movement (according to the value given in Q), first in the axes of the plane and then in the longitudinal axis, to the position of the first probing (only if X or Y or Z has been programmed).

2. Probing movement (at the feedrate given in H), in the given axis and direction until touching the first side.

3. Withdrawal (distance given in E) for the measuring probing movement.

4. Probing movement (at the feedrate given in F) until touching the same side again.

5. Withdrawal to the starting position.

6. Movement parallel to the probed side to touch a different point of the same side.

7. Probing movement (at the feedrate given in F), in the given axis and direction until touching the first side again. This calculates the part inclination angle with respect to the table and it is saved in parameter P296.

8. Rapid up movement in Z (distance given in D) up to the Safety Z coordinate.

9. Movement (according to the value given in Q) up to the approach point of the opposite side considering the length of the part, the calculated inclination angle and the value of parameter B.

10. Probing movement (at the feedrate given in H) to go down to the probing Z coordinate. If it touches the part, the probe goes back up to the safety Z position and moves the distance indicated by parameter B (in the same direction) until clearing the part.

11. Probing movement (at the feedrate given in H), considering the calculated inclination angle until touching that side.

12. Withdrawal (distance given in E) for the measuring probing movement.

13. Probing movement (at the feedrate given in F) until touching the same side again.

14. Rapid up movement to the safety Z coordinate.

15. Movement (according to the value given in Q) up to approach point in the middle of one of the remaining sides considering half the lengths and the calculated inclination angle.

16. Probing movement (at the feedrate given in H) to go down to the probing Z coordinate. If it touches the part, the probe goes back up to the safety Z position and moves the distance indicated by parameter B (in the same direction) until clearing the part.

17. Probing movement (at the feedrate given in H), considering the calculated inclination angle until touching that side.

18. Withdrawal (distance given in E) for the measuring probing movement.

19. Probing movement (at the feedrate given in F) until touching the same side again.

20. Rapid up movement to the safety Z coordinate.

21. If part surface measuring has not been programmed, it goes on to point 26; and if it has been programmed, movement (according to the value given in Q) up to the center of the part.

22. Probing movement (at the feedrate given in H) until touching the surface of the part.

23. Withdrawal (distance given in E) for the measuring probing movement.

24. Probing movement (at the feedrate given in F) until touching the surface of the part again. This measures the coordinate of the part surface and it is saved in parameter P297.

25. Rapid up movement to the safety Z coordinate.

26. Movement (according to the value given in Q) up to the approach point of the opposite side considering the length of the part, the calculated inclination angle.

27. Probing movement (at the feedrate given in H) to go down to the probing Z coordinate. If it touches the part, the probe goes back up to the safety Z position and moves the distance indicated by parameter B (in the same direction) until clearing the part.

28. Probing movement (at the feedrate given in H), considering the calculated inclination angle until touching that side.

29. Withdrawal (distance given in E) for the measuring probing movement.

30. Probing movement (at the feedrate given in F) until touching the same side again. This calculates the real center of the rectangular part and it is saved in parameters P298 and P299.

31. Rapid up movement to the safety Z coordinate.

32. Rapid movement up to the calculated center.

## 12.13 PROBE 11. Circular part centering canned cycle.

Cycle that, with a digital probe, minimizes the preparation time of circular part calculating the real coordinates of the center and of the surface of the part.

(PROBE 11, J, X, Y, Z, K, L, B, D, E, H, F, Q)

### *Initial conditions*

- The probe must properly calibrated in radius and length.
- The probe position before the first probing movement must be as centered as possible in X and Y.

### *Considerations for the cycle*

- After the probing movements, the probe withdraws from the part in G0 before moving to the safety Z position.
- Depending on the PRBMOD variable, no error is issued in the following cases, even if machine parameter PROBERR=YES.
  - When a G75 probing move finishes before the probe has touched part.
  - When a G76 probing move finishes but the probe is still touching the part.

### Parameters

**[ X±5.5 ] X coordinate where probing will begin**

X coordinate of the probe position where the first probing move will start.

If not programmed, it will assume the current X position of the probe.

**[ Y±5.5 ] Y coordinate where probing will begin**

Y coordinate of the probe position where the first probing move will start.

If not programmed, it will assume the current Y position of the probe.

**[ Z±5.5 ] Z coordinate where probing will begin**

Z coordinate of the probe position where the first probing move will start.

If not programmed, it will assume the current Z position of the probe.

**[ J5.5 ] Diameter of the circular part.**

If not programmed or programmed with a 0 value, it generates the corresponding error message.

**[ K1 ] Axis and direction of the first probing movement.**

The values are:

- For X+ : 0
- For X- : 1
- For Y+ : 2
- For Y- : 3

If not programmed, it assumes 0.

**[ L1 ] It indicates whether the part surface is measured or not**

- Value 0: It is not measured
- Value 1: It is measured
- If not programmed, a value of 0 is assumed

**[ B5.5 ] Part approaching distance in each probing movement.**

If not programmed or programmed as 0, it assumes the approach distance value from the probe starting position to the part.

**[ D±5.5 ] Probe's up distance along Z.**

Distance for the probe to go up in Z for its movements over the part. If not programmed or programmed with a 0 value, it generates the corresponding error message.

**[ E±5.5 ] Probe withdrawal distance**

Distance the probe retracts after finding the part, to make the measurement. If not programmed or programmed with a 0 value, it generates the corresponding error message.

**[ H5 ] Probing feedrate for searching the part.**

If not programmed or programmed with a 0 value, it generates the corresponding error message.

**[ F5 ] Probing feedrate for measuring.**

If not programmed or programmed with a 0 value, it generates the corresponding error message.

**[ Q5 ] Probe feedrate when going to the approach points.**

If not programmed, it assumes in G0.

### 12.13.1  Basic operation

1.  Approach movement (according to the value given in Q), first in the axes of the plane and then in the longitudinal axis, to the position of the first probing (only if X or Y or Z has been programmed).

2.  Probing movement (at the feedrate given in H), in the given axis and direction until touching the part.

3.  Withdrawal (distance given in E) for the measuring probing movement.

4.  Probing movement (at the feedrate given in F) until touching the same side again.

5.  Rapid up movement in Z (distance given in D) up to the Safety Z coordinate.

6.  Movement (according to the value given in Q) up to the opposite approach point considering the diameter of the part.

7.  Probing movement (at the feedrate given in H) to go down to the probing Z coordinate. If it touches the part, the probe goes back up to the safety Z position and moves the distance indicated by parameter B (in the same direction) until clearing the part.

8.  Probing movement (at the feedrate given in H) until touching the part.

9.  Withdrawal (distance given in E) for the measuring probing movement.

10. Probing movement (at the feedrate given in F) until touching the part again. This calculates one of the coordinates of the real center of the part.

11. Rapid up movement to the safety Z coordinate.

12. Movement (according to the value given in Q) up to the approach point of the remaining axis considering the calculated center coordinate.

13. Probing movement (at the feedrate given in H) to go down to the probing Z coordinate. If it touches the part, the probe goes back up to the safety Z position and moves the distance indicated by parameter B (in the same direction) until clearing the part.

14. Probing movement (at the feedrate given in H) until touching the part.

15. Withdrawal (distance given in E) for the measuring probing movement.

16. Probing movement (at the feedrate given in F) until touching the part again.

17. Rapid up movement to the safety Z coordinate.

18. If part surface measuring has not been programmed, it goes on to point 23; and if it has been programmed, movement (according to the value given in Q) up to the center of the part.

19. Probing movement (at the feedrate given in H) until touching the surface of the part.

20. Withdrawal (distance given in E) for the measuring probing movement.

21. Probing movement (at the feedrate given in F) until touching the surface of the part again. This measures the coordinate of the part surface and it is saved in parameter P297.

22. Rapid up movement to the safety Z coordinate.

23. Movement (according to the value given in Q) up to the opposite approach point considering the diameter of the part.

24. Probing movement (at the feedrate given in H) to go down to the probing Z coordinate. If it touches the part, the probe goes back up to the safety Z position and moves the distance indicated by parameter B (in the same direction) until clearing the part.

25. Probing movement (at the feedrate given in H) until touching the part.

26. Withdrawal (distance given in E) for the measuring probing movement.

27. Probing movement (at the feedrate given in F) until touching the part again. This calculates the real center of the circular part and it is saved in parameters P298 and P299.

28. Rapid up movement to the safety Z coordinate. Rapid movement up to the calculated center.

## 12.14 PROBE 12. Tabletop probe calibration

This cycle makes it easier to calibrate the probe reducing machine preparation time.

### Initial conditions

The tool used for calibration must properly calibrated in radius and length.

The values of the machine parameters of the probe must be close to their real values. These parameters are the following:

- g.m.p. PRBXMIN (P40).
- g.m.p. PRBXMAX (P41).
- g.m.p. PRBYMIN (P42).
- g.m.p. PRBYMAX (P43).
- g.m.p. PRBZMIN (P44).
- g.m.p. PRBZMAX (P45).

### Initial considerations

Measuring on the Z axis:

- If the Z coordinate of the tool before running the cycle is greater than or equal to the maximum Z coordinate of the probe, the Z side of the probe that will be measured will correspond with its maximum Z coordinate.
- If the Z coordinate of the tool before running the cycle is lower than or equal to the minimum Z coordinate of the probe, the Z side of the probe that will be measured will correspond with its minimum Z coordinate.
- If the Z coordinate of the tool before running the cycle is between the maximum and minimum Z coordinates of the probe, the CNC will issue the relevant error message.

Measuring on the X axis:

- If the X coordinate of the tool before running the cycle is greater than half the maximum and minimum X coordinates of the probe, the X side of the probe where the measuring will begin will correspond with its maximum X coordinate.
- If the X coordinate of the tool before running the cycle is lower than or equal to half the maximum and minimum X coordinates of the probe, the X side of the probe where the measuring will begin will correspond with its minimum X coordinate.

Initial positioning move:

- If the Z coordinate of the tool before running the cycle is located separated from the Z side of the probe to be measured by a distance smaller than the approach distance (B), the initial positioning move of the tool will be carried out first on the Z axis up to that approach distance and, then, in XY up to the approach distance from the X side of the corresponding probe.
- If the Z coordinate of the tool before running the cycle is located separated from the Z side of the probe to be measured by a distance greater than or equal to the approach distance (B), the initial positioning move of the tool will be carried out first in XY up to the approach distance from the X side of the corresponding probe and then in Z up to that approach distance from the Z side of the corresponding probe.

## Cycle programming format

Working in Cartesian coordinates, the basic structure of the block is as follows:

PROBE 12, B, E, H, F, I, X, U, Y, V, Z, W
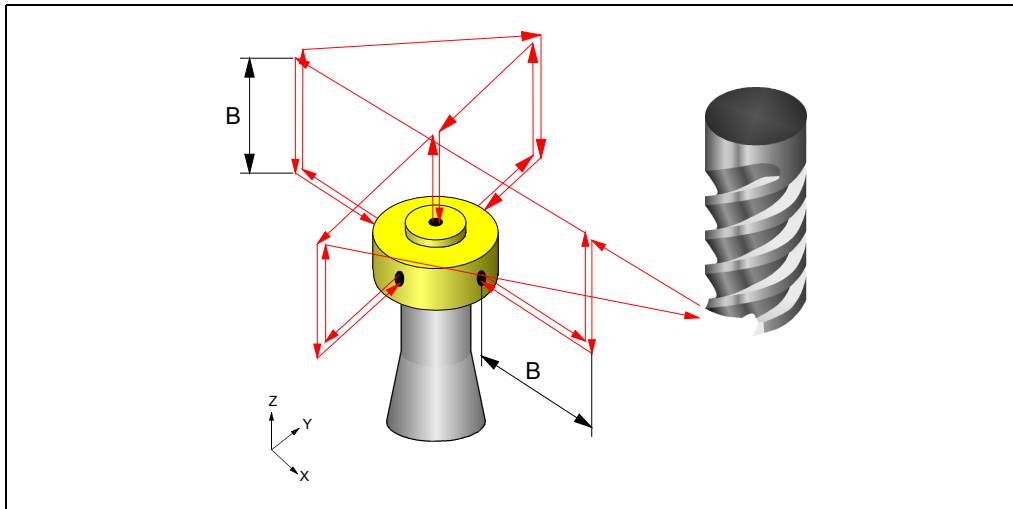
### [ B5.5 ] Approach distance

Probe approaching distance in each probing movement. If not programmed or programmed with a 0 value, the CNC will display the corresponding error message.

### [ E±5.5 ] Withdrawal distance

Distance the tool retracts after touching the probe to take the measurement. If not programmed or programmed with a 0 value, the CNC will display the corresponding error message.

### [ H5 ] Searching feedrate

Probe searching feedrate. If not programmed or programmed with a 0 value, the CNC will display the corresponding error message.

### [ F5 ] Measuring feedrate

Measuring feedrate. If not programmed or programmed with a 0 value, the CNC will display the corresponding error message.

### [ I1 ] Type of calibration

The type of calibration may be simple or double.

I=0    Single calibration: the calibration is carried out in the 4 quadrants of the probe with the spindle that holds the tool oriented at 0º.

I=1    Double calibration: the calibration is carried out twice in the 4 quadrants of the probe, one with the spindle positioned at 0º and the other with the spindle positioned at 180º. This prevents tool eccentricity errors.

If not programmed, the cycle will assume the I=0 value.

### [ X±5.5 ] Approximate X axis coordinate of the least positive side of the probe

Approximate coordinate of the least positive side of the probe, along the abscissa axis. If not programmed, it will assume the value of general machine parameter PRBXMIN (P40).

### [ U±5.5 ] Approximate X axis coordinate of the most positive side of the probe

Approximate coordinate of the most positive side of the probe, along the abscissa axis. If not programmed, it will assume the value of general machine parameter PRBXMAX (P41).

### [ Y±5.5 ] Approximate Y axis coordinate of the least positive side of the probe

Approximate coordinate of the least positive side of the probe, along the ordinate axis. If not programmed, it will assume the value of general machine parameter PRBYMIN (P42).

**[ V±5.5 ] Approximate Y axis coordinate of the most positive side of the probe**

Approximate coordinate of the most positive side of the probe, along the ordinate axis. If not programmed, it will assume the value of general machine parameter PRBYMAX (P43).

**[ Z±5.5 ] Approximate Z axis coordinate of the least positive side of the probe**

Approximate coordinate of the least positive side of the probe, along the Z axis. If not programmed, it will assume the value of general machine parameter PRBZMIN (P44).

**[ W±5.5 ] Approximate Z axis coordinate of the most positive side of the probe**

Approximate coordinate of the most positive side of the probe, along the Z axis. If not programmed, it will assume the value of general machine parameter PRBZMAX (P45).

## Operation

1. Spindle orientation at 0º (only for double calibration).
2. Initial positioning movement to the initial approach coordinates.
3. Probing movement (at the feedrate given in H) up to the Z probing coordinate (in the middle of the probe). If it touches the part, the CNC issues the corresponding error.
4. Probing movement (at the feedrate given in H), in the X axis and in the given direction until touching the first side.
5. Rapid withdrawal along the X axis (distance given in E) for the measuring probing move.
6. Probing movement (at the feedrate given in F) until touching the same side again.
7. Rapid withdrawal up to the X approach coordinate.
8. Rapid withdrawal in Z up to the Z approach coordinate.
9. Rapid X axis movement up to the approach point of the other side considering the theoretical length of the probe on the X axis and the value of parameter B.
10. Probing movement (at the feedrate given in H) to go down to the probing Z coordinate. If it touches the probe, it goes back up to the approach coordinate and moves the safety distance indicated in the same direction. This movement is repeated until clearing the probe.
11. Probing movement on X (at the feedrate given in H) until touching that side.
12. Rapid withdrawal along the X axis (distance given in E) for the measuring probing move.
13. Probing movement on X (at the feedrate given in F) until touching the same side again.
14. Rapid withdrawal up to the X approach coordinate.
15. Rapid movement along the Z axis up to the Z approach coordinate.
16. Rapid movement up to the approach coordinate to the minimum Y side of the probe (the X approach coordinate corresponds to the real center of the probe).
17. Probing movement (at the feedrate given in H) to go down to the probing Z coordinate. If it touches the probe, it goes back up to the approach coordinate and moves the safety distance indicated in the same direction. This movement is repeated until clearing the probe.
18. Probing movement on Y (at the feedrate given in H) until touching that side.
19. Rapid withdrawal along the Y axis (distance given in E) for the measuring probing move.
20. Probing movement on Y (at the feedrate given in F) until touching the same side again.
21. Rapid withdrawal up to the Y approach coordinate.
22. Rapid movement in Z up to the Z approach coordinate.
23. Rapid movement along the Y axis up to the theoretical center of the probe.
24. Probing movement on Z (at the feedrate given in H) until touching the Z side of the probe.
25. Rapid withdrawal along the Z axis (distance given in E) for the measuring probing move.
26. Probing movement on Z (at the feedrate given in F) until touching the same side again.
27. Rapid movement along the Z axis up to the Z approach coordinate.
28. Rapid Y axis movement up to the approach point of the other side considering the theoretical length of the probe on the Y axis and the value of parameter B.
29. Probing movement (at the feedrate given in H) to go down to the probing Z coordinate. If it touches the probe, it goes back up to the approach coordinate and moves the safety distance indicated in the same direction. This movement is repeated until clearing the probe.

**30.** Probing movement (at the feedrate given in H) until touching that side.

**31.** Rapid withdrawal (distance given in E) for the measuring probing movement.

**32.** Probing movement (at the feedrate given in F) until touching the same side again.

**33.** Rapid withdrawal on the Y axis up to the approach position.

**34.** Rapid withdrawal in Z up to the Z approach coordinate.

**35.** Rapid movement up to the starting XY point.

**36.** Rapid withdrawal in Z up to the Z starting coordinate.

**37.** If it is a double calibration, orient the spindle at 180º and repeat steps 2 through 36.

**38.** Setting the relevant parameters with the real values of the sides of the calibrated probe.

## Arithmetic parameters modified by the cycle

Once the cycle has been completed, the CNC will return the real values obtained after measurement, in the following global arithmetic parameters.

P295    Real coordinate of the least positive side of the probe, along the abscissa axis.

P296    Real coordinate of the most positive side of the probe, along the abscissa axis.

P297    Real coordinate of the least positive side of the probe, along the ordinate axis.

P298    Real coordinate of the most positive side of the probe, along the ordinate axis.

P299    Real coordinate of the measured probe side along the longitudinal axis.

**12.**

**FAGOR**

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· Models
Soft: V02.2x

**12.**

**FAGOR** ⱻ

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

## 13.1 Lexical description

All the words that make up the high-level language of the numerical control must be written in capital letters except for associated texts which may be written in upper and lower case letters.

The following elements are available for high-level programming:

- Reserved words.
- Numerical constants.
- Symbols.

### Reserved words

Reserved words are those that the CNC uses in high-level programming for naming system variables, operators, control instructions, etc.

All the letters of the alphabet A-Z are also reserved words, as they can make up a high-level language word when used alone.

### Numerical constants

Blocks programmed in high-level language admit numbers in decimal format and in hexadecimal format.

- The numbers in decimal format must not exceed the ±6.5 format (6 digits to the left of the decimal point and 5 decimals).
- The numbers in hexadecimal format must be preceded by the $ symbol and they must not have more than 8 digits.

A constant higher than the format ±6,5 must be assigned to a variable by means of arithmetic parameters by means of arithmetic expressions or by means of constants in hexadecimal format.

> To assign the value 100000000 to the variable "TIMER", It can be done in one of the following ways:
>     (TIMER = $5F5E100)
>     (TIMER = 10000 * 10000)
>     (P100 = 10000 * 10000)
>     (TIMER = P100)

If the CNC operates in the metric system (millimeters), the resolution is a tenth of a micron and the figures will be programmed in the ±5.4 format (positive or negative with 5 digits to the left of the decimal point and 4 decimals).

If the CNC operates in inches, the resolution is a hundred-thousandths of an inch (ten millionths of an inch) and the figures will be programmed in the ±4.5 format (positive or negative with 4 digits to the left of the decimal point and 5 decimals).

For the convenience of the programmer, this control always allows the format ±5.5 (positive or negative, with 5 integers and 5 decimals), adjusting each number appropriately to the working units every time they are used.

## Simbols

The symbols used in high-level language are:

( ) " = + - * / ,

## 13.2    Variables

The CNC has a number of internal variables that may be accessed from the user program, from the PLC program or via DNC. Depending on how they are used, these variables may be read-only or read-write.

These variables may be accessed from the user program using high-level commands. Each one of these variables is referred to by its mnemonic that must be written in upper-case (capital) letters.

- Mnemonics ending in *(X-C)* indicate a set of 9 elements formed by the corresponding root followed by X, Y, Z, U, V, W, A, B and C.

  ORG(X-C) ->  ORGX        ORGY        ORGZ

  ORGU        ORGV        ORGW

  ORGA        ORGB        ORGC

- Mnemonics ending in *n* indicate that the variables are grouped in tables. To access an element of any of these tables, indicate the field of the desired table using the relevant mnemonic followed by the desired element.

  TORn ->      TOR1        TOR3        TOR11

### The variables and block preparation

The variables that access the real values of the CNC interrupt block preparation. The CNC waits for that command to be executed before resuming block preparation. Thus, precaution must be taken when using this type of variable, because if they are inserted between machining blocks that are working with compensation, undesired profiles may be obtained.

**Example: Reading a variable that interrupts block preparation.**

The following program blocks are performed in a section with G41 compensation.

```
…
N10 X50 Y80
N15 (P100 = POSX); Assigns the value of the real coordinate in X to parameter P100
N20 X50 Y50
N30 X80 Y50
…
```
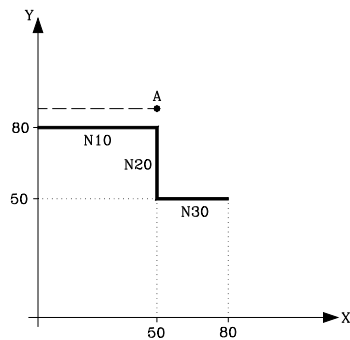


Block N15 interrupts block preparation and the execution of block N10 will finish at point A.

Once the execution of block N15 has ended, the CNC will resume block preparation from block N20 on.
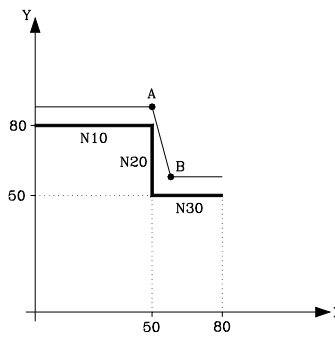
As the next point corresponding to the compensated path is point "B", the CNC will move the tool to this point, executing path "A-B".

As can be observed, the resulting path is not the desired one, and therefore it is recommended to avoid the use of this type of variable in sections having tool compensation active.

### 13.2.1 General purpose parameters or variables

General purpose variables are referred to with the letter "P" followed by an integer number. The CNC has four types of general purpose variables.

| Parameter type | Range |
|----------------|-------|
| Local parameters | P0-P25 |
| Global parameters | P100-P299 |
| User parameters | P1000-P1255 |
| OEM (manufacturer's) parameters | P2000-P2255 |

Blocks programmed in ISO code allow associating parameters to the G F S T D M fields and coordinates of the axes. The block label number must be defined with a numeric value. If parameters are used in blocks programmed in high-level language, they can be programmed within any expression.

Programmers may use general purpose variables when editing their own programs. Later on, during execution, the CNC will replace these variables with the values assigned to them at the time.

| When programming… | When executing… |
|-------------------|-----------------|
| GP0 XP1 Z100 | G1 X-12.5 Z100 |
| (IF (P100 * P101 EQ P102) GOTO N100) | (IF (2 * 5 EQ 12) GOTO N100) |

Using these general purpose variables will depend on the type of block in which they are programmed and the channel of execution. Programs that are executed in the user channel may contain any global, OEM or user parameter, but may not use local parameters.

## Types of arithmetic parameters

### Local parameters

Local parameters can only be accessed from the program or subroutine where they have been programmed. There are seven groups of parameters.

Local parameters used in high-level language may be defined either using the above format or by using the letter A-Z, except for Ñ, so that A is equal to P0 and Z to P25.

The following example shows these two methods of definition:

    (IF ((P0+P1)* P2/P3 EQ P4) GOTO N100)
    (IF ((A+B)* C/D EQ E) GOTO N100)

When using a parameter name (letter) for assigning a value to it (A instead of P0, for example), if the arithmetic expression is a constant, the instruction can be abbreviated as follows:

    (P0=13.7) ==> (A=13.7) ==> (A13.7)

Be careful when using parenthesis since M30 is not the same as (M30). The CNC interprets (M30) as a high level instruction meaning (P12 = 30) and not the execution of the miscellaneous M30 function.

### Global parameters

Global parameters can be accessed from any program and subroutine called from a program.

Global parameters may be used by the user, by the OEM or by the CNC cycles.

### User parameters

These parameters are an expansion of the global parameters but they are not used by the CNC cycles.

### OEM (manufacturer's) parameters

OEM parameters and subroutines with OEM parameters can only be used in OEM programs; those defined with the [O] attribute. Modifying one of these parameters in the tables requires an OEM password.

**13.**

HIGH-LEVEL LANGUAGE PROGRAMMING

Variables

FAGOR

FAGOR AUTOMATION

CNC 8055
CNC 8055i

·M· & ·EN· MODELS
SOFT: V02.2X

·340·

## Using arithmetic parameters by the cycles

Multiple machining cycles (G60 through G65) and the machining canned cycles (G69, G81 to G89) use the sixth nesting level of local parameters when they are active.

Machining canned cycles use the global parameter P299 for internal calculations and probing canned cycles use global parameters P294 to P299.

## Updating arithmetic parameter tables

The CNC will update the parameter table after processing the operations indicated in the block that is in preparation. This operation is always done before executing the block and for this reason, the values shown in the table do not necessarily have to correspond to the block being executed.

If the execution mode is abandoned after interrupting the execution of the program, the CNC will update the parameter tables with values corresponding to the block that was being executed.

When accessing the local parameter and global parameter table, the value assigned to each parameter may be expressed in decimal notation (4127.423) or in scientific notation (0.23476 E-3).

## Arithmetic parameters in the subroutines

The CNC offers high level instructions for defining and using subroutines that may be called upon from the main program (or a subroutine) and another subroutine from this one and so on, the CNC limits the number of these calls to a maximum of 15 nesting levels. The CNC limitation is 20 nesting levels.

Up to 26 (P0-P25) local parameters may be assigned to a subroutine. These parameters, that will be unknown for blocks outside the subroutine, may be referred to by the blocks that make up the subroutine.

Local parameters may be assigned to more than one subroutine up to 6 parameter nesting levels within the 15 subroutine nesting levels.

**13.**

HIGH-LEVEL LANGUAGE PROGRAMMING

Variables

**FAGOR** ⬤

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

### 13.2.2 Variables associated with tools.

These variables are associated with the tool offset table, tool table and tool magazine table, so the values which are assigned to or read from these fields will comply with the formats established for these tables.

#### Tool Offset table

The radius (R), length (L) and wear offset (I, K) values of the tool are given in the active units.

If G70, in inches (within ±3937.00787).

If G71, in millimeters (within ±99999.9999).

If rotary axis in degrees (within ±99999.9999).

#### Tool table

The tool offset number is an integer between 0 and 255. The maximum number of tool offsets is limited by g.m.p. NTOFFSET.

The family code is a number between 0 and 255.

0 to 199     if it is a normal tool.

200 to 255     if it is a special tool.

The nominal life is given either in minutes or in operations (0··65535).

The real (actual) life is given either in hundredths of a minute (0··9999999) or in operations (0··999999).

#### Tool magazine table

Each magazine position is represented as follows:

1··255     Tool number.

0     The magazine position is empty.

-1     The magazine position has been canceled.

The tool position in the magazine is represented as follows:

1.- Position number.

0     The tool is in the spindle.

-1     Tool not found

-2     The tool is in the change position.

## Read-only variables

**TOOL**

Returns the number of the active tool.

```
(P100=TOOL)
    Assigns the number of the active tool to parameter P100.
```

**TOD**

Returns the number of the active tool offset.

**NXTOOL**

Returns the next tool number, which is selected but is awaiting the execution of M06 to be active.

**NXTOD**

Returns the number of the tool offset corresponding to the next tool, which is selected but is awaiting the execution of M06 to be active.

**TMZPn**

Returns the position occupied in the tool magazine by the indicated tool (n).

**PTOOL**

Returns the magazine position to where the current tool is to be left. It matches the value that will be received later on in the register "T2BCD" (R559) with the M6, but the latter will be in BCD format.

This variable is only accessible via the CNC.

**PNXTOOL**

Returns the magazine position from where the next tool is to be picked up. It matches the value that will be received later on in the register "TBCD" (R558) with the M6, but the latter will be in BCD format.

This variable is only accessible via the CNC.

**HTOR**

The HTOR variable indicates the tool radius being used by the CNC to do the calculations.

Being a variable that can be read and written by the CNC and read-only from the PLC and DNC, its value may be different from the one assigned in the table (TOR).

On power-up, after a T function, after a RESET or after an M30 function, it assumes the value of the table (TOR).

### Application example

To machine a profile with a residual stock of 0.5 mm running 0.1mm-passes with a tool whose radius is 10 mm.

Assign to the tool radius the value of:

> 10.5 mm in the table and execute the profile.
>
> 10.4 mm in the table and execute the profile.
>
> 10.3 mm in the table and execute the profile.
>
> 10.2 mm in the table and execute the profile.
>
> 10.1 mm in the table and execute the profile.
>
> 10.0 mm in the table and execute the profile.

However, if while machining, the program is interrupted or a reset occurs, the table assumes the radius value assigned in that instant (e.g.: 10.2 mm). Its value has changed.

To avoid this, instead of modifying the tool radius in the (TOR) table, use the variable (HTOR) to change the tool radius value used by the CNC to calculate.

Now, if the program is interrupted, the tool radius value initially assigned in the (TOR) table will be correct because it has not changed.

## Read-and-write variables

**TORn**

This variable allows the value assigned to the radius of the indicated tool offset (n) in the tool offset table to be read or modified.

```
(P110=TOR3)
    Assigns the radius value of tool offset ·3· to parameter P110.
(TOR3=P111)
    Assigns the value indicated in parameter P111 to the radius of tool offset 3.
```

**TOLn**

This variable allows the value assigned to the length of the indicated tool offset (n) to be read or modified in the tool offset table.

**TOIn**

This variable allows the value assigned to the wear in radius (I) of the indicated tool offset (n) to be read or modified in the tool offset table.

**TOKn**

This variable allows the value assigned to the wear in length (K) of the indicated tool offset (n) to be read or modified in the tool offset table.

**TLFDn**

This variable allows the tool offset number of the indicated tool (n) to be read or modified in the tool table.

**TLFFn**

This variable allows the family code of the indicated tool (n) to be read or modified in the tool table.

**TLFNn**

This variable allows the value assigned as the nominal life of the indicated tool (n) to be read or modified in the tool table.

**TLFRn**

This variable allows the value corresponding to the real life of the indicated tool (n) to be read or modified in the tool table.

**TMZTn**

This variable allows the contents of the indicated position (n) to be read or modified in the tool magazine table.

### 13.2.3   Variables associated with zero offsets.

These variables are associated with the zero offsets and may correspond to the table values or to those currently preset either by means of function G92 or manually in the JOG mode.

The possible zero offsets in addition to the additive offset indicated by the PLC, are G54, G55, G56, G57, G58 and G59.

The values for each axis are given in the active units:

If G70, in inches (within ±3937.00787).

If G71, in millimeters (within ±99999.9999).

If rotary axis in degrees (within ±99999.9999).

Although there are variables which refer to each axis, the CNC only allows those referring to the axes selected at the CNC. Thus, if the CNC controls axes X, Y, Z, U and B, it only allows the variables ORGX, ORGY, ORGZ,. ORGU and ORGB in the case of  ORG(X-C).

## Read-only variables

**ORG(X-C)**

Returns the value of the active zero offset in the selected axis. The value of the additive offset indicated by the PLC or by the additive handwheel is not included in this value.

(P100=ORGX)
> It assigns to P100 the X value of the part zero active for the X axis. This value could have been set either manually, by means of function G92 or by the variable "ORG(X-C)n".

**PORGF**

Returns the abscissa value of the polar coordinate origin with respect to the Cartesian origin.

**PORGS**

Returns the ordinate value of the polar coordinate origin with respect to the Cartesian origin.

**ADIOF(X-C)**

It returns the value of the zero offset generated by the additive handwheel in the selected axis.

**ADDORG (X-C)**

Returns the value of the active incremental zero offset corresponding to the axis selected at the time. It is read-only variable that can be read from the CNC, PLC and DNC.

**EXTORG**

Returns the active absolute zero offset. The values returned by the variable are identical for both possible expressions of absolute zero offsets.

This read-only variable interrupts block preparation and may be read from the CNC, from the PLC and from DNC.

The values of the EXTORG variables that correspond to the absolute zero offsets are:

| EXTORG | Active zero offset | EXTORG | Active zero offset |
|--------|--------------------|--------|--------------------|
| 0 | G53 (there is no zero offset) | 11 | G159N11 |
| 1 | G54 or G159N1 | 12 | G159N12 |
| 2 | G55 or G159N2 | 13 | G159N13 |
| 3 | G56 or G159N3 | 14 | G159N14 |
| 4 | G57 or G159N4 | 15 | G159N15 |
| 5 | G159N5 | 16 | G159N16 |
| 6 | G159N6 | 17 | G159N17 |
| 7 | G159N7 | 18 | G159N18 |
| 8 | G159N8 | 19 | G159N19 |
| 9 | G159N9 | 20 | G159N20 |
| 10 | G159N10 | | |

**Considerations:**

- When programming only an incremental zero offset (G58 or G59), the value of the EXTORG variable will be 0.
- When programming an absolute and an incremental zero offset, the EXTORG variable will keep the value of the absolute zero offset.

    Example: If G54 + G58, EXTORG = 1 has been programmed.

### Read-and-write variables

**ORG(X-C)n**

This variable allows the value of the selected axis to be read or modified in the table corresponding to the indicated zero offset (n).

**If G54-G59 is used:**
(P110=ORGX 55)
    Loads parameter P100 with the X value of G55 in the zero offset table.
(ORGY 54=100.8)
    It assigns value 100.8 to the Y axis in the table for the G54 zero offset.
**If G159N1-N20 is used:**
(P110=ORGX 19)
    Loads parameter P110 with the X value of G159N19 in the zero offset table.
(ORGY 19=100.8)
    It assigns value 100.8 to the Y axis in the table for the G159N19 zero offset.

**PLCOF(X-C)**

This variable allows the value of the selected axis to be read or modified in the table of additive offsets indicated by the PLC.

Accessing any of the PLCOF(X-C) variables interrupts block preparation and the CNC waits for that command to be executed before resuming block preparation.

### 13.2.4   Variables associated with function G49

With function G49, it is possible to define a coordinate transformation or, in other words, the inclined plane resulting from that transformation.

The values for each axis are given in the active units:

If G70, in inches (within ±3937.00787).

If G71, in millimeters (within ±99999.9999).

## Read-only variables associated with the definition of function G49

**ORGROX**             **ORGROY**             **ORGROZ**

X coordinate of the new part zero with respect to home.

Y coordinate of the new part zero with respect to home.

Z coordinate of the new part zero with respect to home.

| **ORGROA** | **ORGROB** | **ORGROC** | **ORGROI** |
| **ORGROJ** | **ORGROK** | **ORGROQ** | **ORGROR** |
| **ORGROS** | | | |

Value assigned to parameter A.

Value assigned to parameter B.

Value assigned to parameter C.

Value assigned to parameter I.

Value assigned to parameter J.

Value assigned to parameter K.

Value assigned to parameter Q.

Value assigned to parameter R.

Value assigned to parameter S.

**GTRATY**

Type of G49 programmed.

| | |
|---|---|
| 0 = no G49 has been defined. | 3 = Type G49 T X Y Z S |
| 1 = Type G49 X Y Z A B C | 4 = Type G49 X Y Z I J K R S |
| 2 = Type G49 X Y Z Q R S | |

Every time G49 is programmed, the CNC updates the values of the parameters that have been defined.

For example when programming G49 XYZ ABC, the CNC updates the variables.

ORGROX, ORGROY, ORGROZ

ORGROA, ORGROB, ORGROC

The rest of variables keep their previous values.

# Read-write variables updated by the CNC once function G49 is executed

Accessing the variables TOOROF or TOOROS interrupts block preparation and the CNC waits for that command to be executed before resuming block preparation.

When having a swivel or angular spindle, general machine parameter XFORM (P93) with a value of 2 or 3, the CNC shows the following data:

### TOOROF

It indicates the position to be occupied by the spindle's main rotary axis in order to orient the spindle perpendicular to the indicated inclined plane.

### TOOROS

It indicates the position to be occupied by the spindle's secondary rotary axis in order to orient the spindle perpendicular to the indicated inclined plane.

### 13.2.5   Variables associated with machine parameters

These variables associated with machine parameters are read-only variables. These variables may be read and written when executed inside an OEM program or subroutine.

Refer to the installation and start-up manual to know the format of the values returned. The values of 1/0 correspond to the parameters that are set as YES/NO, +/- or ON/OFF.

The coordinate and feedrate values are given in the active units:

If G70, in inches (within ±3937.00787).

If G71, in millimeters (within ±99999.9999).

If rotary axis in degrees (within ±99999.9999).

#### *Modify the machine parameters from an OEM program or subroutine.*

These variables may be read and written when executed inside an OEM program or subroutine. In this case, these variables can be used to modify the value of certain machine parameters. Refer to the installation manual for the list of machine parameters that may be modified.

In order to be able to modify these parameters via PLC, an OEM subroutine containing the relevant variables must be executed using the CNCEX command.

## Read-only variables

**MPGn**

Returns the value assigned to general machine parameter (n).

```
(P110=MPG8)
    It assigns the value of the general machine parameter P8 "INCHES" to parameter P110,
    if millimeters P110=0 and if inches P110=1.
```

**MP(X-C)n**

Returns the value assigned to the machine parameter (n) of the indicated axis (X-C).

```
(P110=MPY 1)
    Assigns the value of Y axis machine parameter P1 "DFORMAT" to parameter P110.
```

**MPSn**

Returns the value assigned to the indicated machine parameter (n) of the main spindle.

**MPSSn**

Returns the value assigned to the indicated machine parameter (n) of the second spindle.

**MPASn**

Returns the value of the indicated machine parameter (n) of the auxiliary spindle.

**MPLCn**

Returns the value assigned to the indicated machine parameter (n) of the PLC.

### 13.2.6   Variables associated with work zones

Variables associated with work zones are read-only variables.

The values of the limits are given in the active units:

If G70, in inches (within ±3937.00787).

If G71, in millimeters (within ±99999.9999).

If rotary axis in degrees (within ±99999.9999).

The status of the work zones are defined according to the following code:

0 = Disabled.

1 = Enabled as no-entry zone.

2 = Enabled as no-exit zone.

## Read-only variables

**FZONE**

It returns the status of work zone 1.

**FZLO(X-C)**          **FZUP(X-C)**

Lower limit of zone 1 along the selected axis (X-C).

Upper limit of zone 1 along the selected axis (X-C).

```
(P100=FZONE)        ; It assigns to parameter P100 the status of work zone 1.
(P101=FZOLOX)       ; It assigns the lower limit of zone 1 to parameter P101.
(P102=FZUPZ)        ; It assigns the upper limit of zone 1 to parameter P102.
```

**SZONE**          **SZLO(X-C)**          **SZUP(X-C)**

Status of work zone 2.

Lower limit of zone 2 along the selected axis (X-C).

Upper limit of zone 2 along the selected axis (X-C).

**TZONE**          **TZLO(X-C)**          **TZUP(X-C)**

Status of work zone 3.

Lower limit of zone 3 along the selected axis (X-C).

Upper limit of zone 3 along the selected axis (X-C).

**FOZONE**          **FOZLO(X-C)**          **FOZUP(X-C)**

Status of work zone 4.

Lower limit of zone 4 along the selected axis (X-C).

Upper limit of zone 4 along the selected axis (X-C).

**FIZONE**          **FIZLO(X-C)**          **FIZUP(X-C)**

Status of work zone 5.

Lower limit of zone 5 along the selected axis (X-C).

Upper limit of zone 5 along the selected axis (X-C).

### 13.2.7   Variables associated with feedrates

## Read-only variables associated with the real (actual) feedrate

### FREAL

It returns the CNC's real feedrate. In mm/minute or inches/minute.

> (P100=FREAL)
>   It assigns the real feedrate value of the CNC to parameter P100.

### FREAL(X-C)

It returns the actual (real) CNC feedrate of the selected axis.

### FTEO(X-C)

It returns the theoretical CNC feedrate of the selected axis.

## Read-only variables associated with function G94

### FEED

It returns the feedrate selected at the CNC by function G94. In mm/minute or inches/minute.

This feedrate may be indicated by program, by PLC or by DNC; the CNC selects one of them, the one indicated by DNC has the highest priority and the one indicated by program has the lowest priority.

### DNCF

It returns the feedrate, in mm/minute or inches/minute selected by DNC. If it has a value of 0 it means that it is not selected.

### PLCF

It returns the feedrate, in mm/minute or inches/minute selected by PLC. If it has a value of 0 it means that it is not selected.

### PRGF

It returns the feedrate, in mm/minute or inches/minute selected by program.

## Read-only variables associated with function G95

### FPREV

It returns the feedrate selected at the CNC by function G95. In mm/turn or inches/turn.

This feedrate may be indicated by program, by PLC or by DNC; the CNC selects one of them, the one indicated by DNC has the highest priority and the one indicated by program has the lowest priority.

### DNCFPR

It returns the feedrate, in mm/turn or inches/turn selected by DNC. If it has a value of 0 it means that it is not selected.

### PLCFPR

It returns the feedrate, in mm/turn or inches/turn selected by PLC. If it has a value of 0 it means that it is not selected.

### PRGFPR

It returns the feedrate, in mm/turn or inches/turn selected by program.

### Read-only variables associated with function G32

**PRGFIN**

It returns the feedrate selected by program, in 1/min.

Likewise, the CNC variable FEED, associated with G94, indicates the resulting feedrate in mm/min or inches/min.

### Read-only variables associated with the override

**FRO**

It returns the feedrate override (%) currently selected at the CNC. It is given in integer values between 0 and "MAXFOVR" (maximum 255).

This feedrate percentage may be indicated by program, by PLC, by DNC or from the front panel; the CNC selects one of them, where the priority (from the highest to the lowest) is: by program, by DNC, by PLC and from the switch.

**DNCFRO**

It returns the feedrate override % currently selected by the DNC. If it has a value of 0 it means that it is not selected.

**PLCFRO**

It returns the feedrate override % currently selected by the PLC. If it has a value of 0 it means that it is not selected.

**CNCFRO**

It returns the feedrate override % currently selected by the switch.

**PLCCFR**

It returns the feedrate percentage currently selected by the PLC's execution channel.

### Read-write variables associated with the override

**PRGFRO**

This variable may be used to read or modify the feedrate override percentage currently selected by program. It is given in integer values between 0 and "MAXFOVR" (maximum 255). If it has a value of 0 it means that it is not selected.

```
(P110=PRGFRO)
    It assigns to P110 the % of feedrate override selected by program.
(PRGFRO=P111)
    It sets the feedrate override % selected by program to the value of P111.
```

**FAGOR**

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2x

·352·

### 13.2.8    Variables associated with coordinates

The coordinate values for each axis are given in the active units:

If G70, in inches (within ±3937.00787).

If G71, in millimeters (within ±99999.9999).

If rotary axis in degrees (within ±99999.9999).

## Read-only variables

Accessing any of the variables POS(X-C), TPOS(X-C), APOS(X-C), ATPOS(X-C), DPOS(X-C), FLWE(X-C), DEFLEX, DEFLEY or DEFLEZ interrupts block preparation and the CNC waits for that command to be executed before resuming block preparation.

#### PPOS(X-C)

Returns the programmed theoretical coordinate of the selected axis.

```
(P110=PPOSX)
    It assigns to P100 the programmed theoretical position of the X axis.
```

#### POS(X-C)

It returns the real tool base position value referred to machine reference zero (home).

On limit-less rotary axes, this variable takes into account the value of the active zero offset. The values of the variable are between the active zero offset and ±360º (ORG* ± 360º).

If ORG* = 20º          it displays between 20º and 380º / displays between -340º and 20º.

If ORG* = -60º         it displays between -60º and 300º / displays between -420 and -60º

#### TPOS(X-C)

It returns the theoretical position value (real coordinate + following error) of the tool base referred to machine reference zero (home).

On limit-less rotary axes, this variable takes into account the value of the active zero offset. The values of the variable are between the active zero offset and ±360º (ORG* ± 360º).

If ORG* = 20º          it displays between 20º and 380º / displays between -340º and 20º.

If ORG* = -60º         it displays between -60º and 300º / displays between -420 and -60º

#### APOS(X-C)

It returns the real tool base position value, referred to part zero, of the selected axis.

#### ATPOS(X-C)

It returns the theoretical position value (real coordinate + following error) of the tool base referred to part zero.

#### DPOS(X-C)

The CNC updates this variable whenever probing operations G75, G76 and probing cycles PROBE and DIGIT are carried out.

When the digital probe communicates with the CNC via infrared beams, there could be some delay (milliseconds) from the time the probe touches the part to the instant the CNC receives the probe signal.
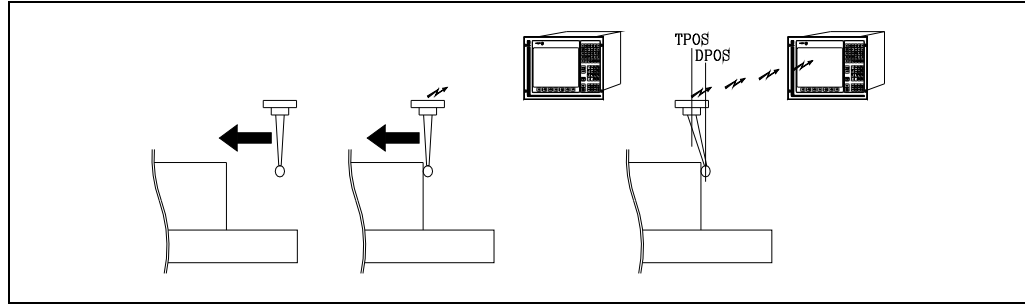
Although the probe keeps moving until the CNC receives the probing signal, the CNC takes into account the value assigned to general machine parameter PRODEL and provides the following information in the variables TPOS(X-C) and DPOS(X-C).

TPOS(X-C)    Actual position of the probe when the CNC receives the probe signal.

DPOS(X-C)    Theoretical position of the probe when the probe touched the part.

**FLWE(X-C)**

It returns the following error of the selected axis.

**DEFLEX                    DEFLEY                    DEFLEZ**

They return the amount of deflection obtained at the time by the Renishaw probe SP2 on each axis X, Y, Z.

**DPLY(X-C)**

It returns the position value (coordinate) shown on the screen for the selected axis.

**DRPO(X-C)**

It returns the position indicated by the Sercos drive of the selected axis (variable PV51 or PV53 of the drive).

**GPOS(X-C)n p**

Programmed coordinate for a particular axis in the indicated (n) block of the (p) program.

(P80=GPOSX N99 P100)
    It assigns to parameter P88 the value of the coordinate programmed for the X axis in the block having the label N99 and located in program P100.

Only programs located in the CNC's RAM memory may be consulted.

If the defined program or block does not exist, it shows the relevant error message. If the indicated block does not contain the requested axis, it returns the value 100000.0000.

**FAGOR**

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

·354·

### Read-and-write variables

#### DIST(X-C)

These variables may be used to read or modify the distance traveled by the selected axis. This value is accumulative and is very useful when it is required to perform an operation which depends on the distance traveled by the axes, their lubrication for example.

```
(P110=DISTX)
    It assigns to P110 the distance traveled by the X axis
(DISTX=P111)
    It presets the variable indicating the distance traveled by the Z axis with the value of
    arithmetic parameter P111.
```

Accessing any of the DIST(X-C) variables interrupts block preparation and the CNC waits for that command to be executed before resuming block preparation.

#### LIMPL(X-C)                    LIMMI(X-C)

With these variables, it is possible to set a second travel limit for each axis: LIMPL for the upper limit and LIMMI for the lower one.

Since the second limits are activated or deactivated from the PLC, through general logic input ACTLIM2 (M5052), besides setting the limits, an auxiliary M code must be executed to let it know.

It is also recommended to execute function G4 after the change so the CNC executes the following blocks with the new limits.

The second travel limit will be taken into account if the first one has been set using axis machine parameters LIMIT+ (P5) and LIMIT- (P6).

### 13.2.9 Variables associated with electronic handwheels

## Read-only variables

**HANPF**        **HANPS**        **HANPT**        **HANPFO**

They return the pulses of the first (HANPF), second (HANPS), third (HANPT) or fourth (HANPFO) handwheel received since the CNC was turned on. It is irrelevant to have the handwheel connected to the feedback inputs or to the PLC inputs.

**HANDSE**

For handwheels with axis selector button, it indicates whether that button has been pressed or not. A value of ·0· means that it has not been pressed.

**HANFCT**

It returns the multiplying factor set by PLC for each handwheel.

It must be used when using several electronic handwheels or when using a single handwheel but different multiplying factors (x1, x10, x100) are to be applied to each axis.

| C | | | B | | | A | | | W | | | V | | | U | | | Z | | | Y | | | X | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| c | b | a | c | b | a | c | b | a | c | b | a | c | b | a | c | b | a | c | b | a | c | b | a | c | b | a | lsb |

Once the switch has been turned to one of the handwheel positions, the CNC checks this variable and, depending on the values assigned to each axis bit (c, b, a) it applies the multiplying factor selected for each one of them.

| c | b | a | |
|---|---|---|---|
| 0 | 0 | 0 | The value indicated at the front panel or keyboard switch. |
| 0 | 0 | 1 | x1 factor |
| 0 | 1 | 0 | x10 factor |
| 1 | 0 | 0 | x100 factor |

If there are more than one bit set to "1" for an axis, the least significant bit will be considered. Thus:

| c | b | a | |
|---|---|---|---|
| 1 | 1 | 1 | x1 factor |
| 1 | 1 | 0 | x10 factor |

> **i** *The screen always shows the value selected at the switch.*

**HBEVAR**

It must be used when having a Fagor HBE handwheel.

It indicates whether the HBE handwheel is enabled or not, the axis to be jogged and the multiplying factor to be applied (x1, x10, x100).

| * | ^ | | | C | | | B | | | A | | | W | | | V | | | U | | | Z | | | Y | | | X | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| * | ^ | | | c | b | a | c | b | a | c | b | a | c | b | a | c | b | a | c | b | a | c | b | a | c | b | a | c | b | a | lsb |

(*) Indicates whether the HBE handwheel pulses will be taken into account or not in jog mode.

0 =      They are ignored.

1 =      They are taken into account.

(^) When the machine has a general handwheel and individual handwheels (associated with an axis), it indicates which handwheel has priority when both are turned at the same time.

0 =    The individual handwheel has priority. The relevant axis ignores the pulses from the general handwheel, the rest of the axes don't.

1 =The general handwheel has priority. It ignored the pulses from the individual handwheel.

(a, b, c) Indicate the axis to be moved and the selected multiplying factor.

| c | b | a | |
|---|---|---|---|
| 0 | 0 | 0 | The value indicated at the front panel or keyboard switch. |
| 0 | 0 | 1 | x1 factor |
| 0 | 1 | 0 | x10 factor |
| 1 | 0 | 0 | x100 factor |

If several axes are selected, the following order of priority is applied: X, Y, Z, U, V, W, A, B, C.

If there are more than one bit set to "1" for an axis, the least significant bit will be considered. Thus:

| c | b | a | |
|---|---|---|---|
| 1 | 1 | 1 | x1 factor |
| 1 | 1 | 0 | x10 factor |

The HBE handwheel has priority. That is, regardless of the mode selected at the CNC switch (continuous or incremental JOG, handwheel), HBEVAR is set to other than "0", the CNC goes into handwheel mode.

It shows the selected axis in reverse video and the multiplying factor selected by the PLC. When the HBEVAR variable is set to "0", it shows the mode selected by the switch again.

## Read-and-write variables

### MASLAN

It must be used when the path-handwheel or the path-jog is selected. Indicates the angle of the linear path.
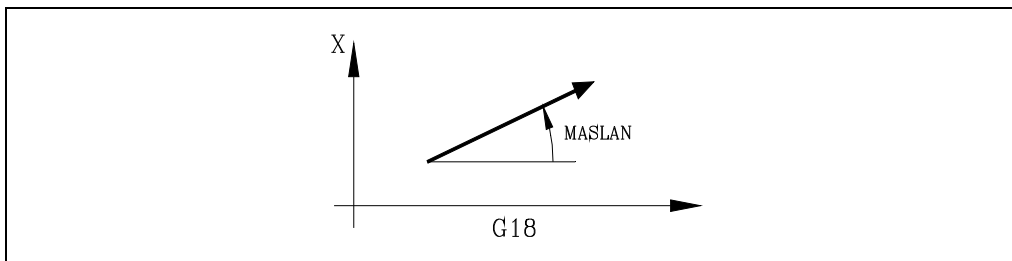


### MASCFI                MASCSE

They must be used when the path-handwheel or the path-jog is selected. On circular paths (arcs), they indicate the center coordinates.
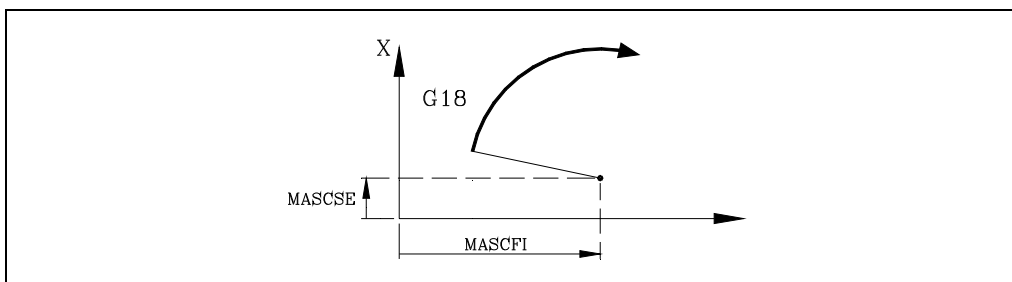
### 13.2.10  Variables associated with feedback

**ASIN(X-C)**

"A" signal of the CNC's sinusoidal feedback for the X-C axis.

**BSIN(X-C)**

"B" signal of the CNC's sinusoidal feedback for the X-C axis.

**ASINS**

"A" signal of the CNC's sinusoidal feedback for the spindle.

**BSINS**

"B" signal of the CNC's sinusoidal feedback for the spindle.

**SASINS**

"A" signal of the CNC sinusoidal feedback for the second spindle.

**SBSINS**

"B" signal of the CNC sinusoidal feedback for the second spindle.

### 13.2.11 Variables associated with the main spindle

In these variables associated with the spindle, their values are given in revolutions per minute and the main spindle override values are given in integers from 0 to 255.

Certain variables interrupt block preparation (it is indicated in each one) and the CNC waits for that command to be executed before resuming block preparation.

## Read-only variables

**SREAL**

Returns the real main spindle turning speed in revolutions per minute. It interrupts block preparation.

```
(P100=SREAL)
     It assigns to P100 the real turning speed of the main spindle.
```

**FTEOS**

It returns the theoretical turning speed of the main spindle.

**SPEED**

Returns, in revolutions per minute, the main spindle speed selected at the CNC.

This turning speed may be indicated by program, by PLC or by DNC; the CNC selects one of them, the one indicated by DNC has the highest priority and the one indicated by program has the lowest priority.

**DNCS**

Returns the turning speed in revolutions per minute, selected by DNC. If it has a value of 0 it means that it is not selected.

**PLCS**

Returns the turning speed in revolutions per minute selected by PLC. If it has a value of 0 it means that it is not selected.

**PRGS**

Returns the turning speed in revolutions per minute, selected by program.

**SSO**

It returns the turning speed override (%) of the main spindle currently selected at the CNC. It is given in integer values between 0 and "MAXFOVR" (maximum 255).

This turning speed percentage of the main spindle may be indicated by program, by PLC, by DNC or by the front panel; the CNC selects one of them and the priority (from the highest to the lowest) is: by program, by DNC, by PLC and from the panel frontal.

**DNCSSO**

It returns the turning speed override % of the main spindle currently selected via DNC. If it has a value of 0 it means that it is not selected.

**PLCSSO**

It returns the turning speed override % of the main spindle currently selected by PLC. If it has a value of 0 it means that it is not selected.

**CNCSSO**

It returns the turning speed override % of the main spindle currently selected from the front panel.

**SLIMIT**

It returns the value set in rpm at the CNC for the turning speed limit of the main spindle.

This limit may be indicated by program, by PLC or by DNC; the CNC selects one of them, the one indicated by DNC has the highest priority and the one indicated by program has the lowest priority.

**DNCSL**

It returns the speed limit of the main spindle in rpm currently selected via DNC. If it has a value of 0 it means that it is not selected.

**PLCSL**

It returns the speed limit of the main spindle in rpm currently selected by PLC. If it has a value of 0 it means that it is not selected.

**PRGSL**

It returns the speed limit of the main spindle in rpm currently selected by program.

**MDISL**

Maximum machining spindle speed. This variable is also updated (refreshed) when programming function G92 via MDI.

**POSS**

It returns the real position of the main spindle. Its value may be within ±99999.9999°. It interrupts block preparation.

**RPOSS**

It returns the real position of the main spindle. Its value is given in 0.0001 degree units (between -360º and 360º). It interrupts block preparation.

**TPOSS**

It returns the theoretical position of the main spindle (real position + lag). Its value may be within ±99999.9999º. It interrupts block preparation.

**RTPOSS**

It returns the theoretical position of the main spindle (real position + lag) in 360º module. Its value may be between 0 and 360º. It interrupts block preparation.

**DRPOS**

Position indicated by the Sercos drive of the main spindle.

**PRGSP**

Position programmed in M19 via program for the main spindle. This variable may be read from the CNC, from the PLC and from DNC.

**FLWES**

It returns the main spindle's following error in degrees (within ±99999.9999). It interrupts block preparation.

**SYNCER**

It returns, in degrees (within ±99999.9999), the following error of the second spindle with respect to the main spindle when they are synchronized in position.

## Read-and-write variables

**PRGSSO**

This variable may be used to read or modify the speed override percentage of the main spindle currently selected by program. It is given in integer values between 0 and "MAXFOVR" (maximum 255). If it has a value of 0 it means that it is not selected.

(P110=PRGSSO)
   It assigns to P110 the % of the main spindle speed selected by program.
(PRGSSO=P111)
   It sets the value indicating the main spindle speed % selected by program to the value of arithmetic parameter P111.

### 13.2.12 Variables associated with the second spindle

In these variables associated with the spindle, their values are given in revolutions per minute and the 2nd spindle override values are given in integers from 0 to 255.

## Read-only variables

**SSREAL**

Returns the real 2nd spindle turning speed in revolutions per minute.

```
(P100=SSREAL)
    It assigns to P100 the real turning speed of the second spindle.
```

If this variable is accessed, block preparation is interrupted and the CNC waits for this command to be executed to resume block preparation.

**SFTEOS**

It returns the theoretical turning speed of the second spindle.

**SSPEED**

Returns, in revolutions per minute, the 2nd spindle speed selected at the CNC.

This turning speed may be indicated by program, by PLC or by DNC; the CNC selects one of them, the one indicated by DNC has the highest priority and the one indicated by program has the lowest priority.

**SDNCS**

Returns the turning speed in revolutions per minute, selected by DNC. If it has a value of 0 it means that it is not selected.

**SPLCS**

Returns the turning speed in revolutions per minute selected by PLC. If it has a value of 0 it means that it is not selected.

**SPRGS**

Returns the turning speed in revolutions per minute, selected by program.

**SSSO**

It returns the turning speed override (%) of the second spindle currently selected at the CNC. It is given in integer values between 0 and "MAXFOVR" (maximum 255).

This turning speed percentage of the second spindle may be indicated by program, by PLC, by DNC or by the front panel; the CNC selects one of them and the priority (from the highest to the lowest) is: by program, by DNC, by PLC and from the panel frontal.

**SDNCSO**

It returns the turning speed override % of the second spindle currently selected via DNC. If it has a value of 0 it means that it is not selected.

**SPLCSO**

It returns the turning speed override % of the second spindle currently selected by PLC. If it has a value of 0 it means that it is not selected.

**SCNCSO**

It returns the turning speed override % of the second spindle currently selected from the front panel.

**SSLIMI**

It returns the value set in rpm at the CNC for the turning speed limit of the second spindle.

This limit may be indicated by program, by PLC or by DNC; the CNC selects one of them, the one indicated by DNC has the highest priority and the one indicated by program has the lowest priority.

**SDNCSL**

It returns the speed limit of the second spindle in rpm currently selected via DNC. If it has a value of 0 it means that it is not selected.

**SPLCSL**

It returns the speed limit of the second spindle in rpm currently selected by PLC. If it has a value of 0 it means that it is not selected.

**SPRGSL**

It returns the speed limit of the second spindle in rpm currently selected by program.

**SPOSS**

It returns the real position of the second spindle. Its value may be within ±99999.9999°.

**SRPOSS**

It returns the real position of the second spindle. Its value is given in 0.0001 degree units (between -360º and 360º).

**STPOSS**

It returns the theoretical position of the second spindle (real position + lag). Its value may be within ±99999.9999º.

**SRTPOS**

Returns the second spindle's theoretical position value (real coordinate+ following error) in a 360º module. Its value may be between 0 and 360º.

**SDRPOS**

Position indicated by the Sercos drive of the second spindle.

**SPRGSP**

Position programmed in M19 via program for the second spindle. This variable may be read from the CNC, from the PLC and from DNC.

**SFLWES**

It returns the second spindle's following error in degrees (within ±99999.9999).

When accessing one of these variables SPOSS, SRPOSS, STPOSS, SRTPOSS or SFLWES, block preparation is interrupted and the CNC waits for that command to be executed before resuming block preparation.

## Read-and-write variables

**SPRGSO**

This variable may be used to read or modify the speed override percentage of the second spindle currently selected by program. It is given in integer values between 0 and "MAXFOVR" (maximum 255). If it has a value of 0 it means that it is not selected.

(P110=SPRGSO)

It assigns to P110 the % of the second spindle speed selected by program.

(SPRGSO=P111)

It sets the value indicating the second spindle speed % selected by program to the value of arithmetic parameter P111.

### 13.2.13  Variables associated with the live tool

## Read-only variables

**ASPROG**

It must be used inside the subroutine associated with function M45.

Returns the revolutions per minute programmed in M45 S. If programmed only in M45, the value 0 will be used.

The ASPROG variable is updated just before executing the M45 function so it is already updated when executing its associated subroutine.

### 13.2.14 PLC related variables

It should be borne in mind that the PLC has the following resources:

| | |
|---|---|
| (I1 thru I512) | Inputs. |
| (O1 thru O512) | Outputs. |
| M1 thru M5957) | Marks. |
| (R1 thru R499) | 32-bit registers. |
| (T1 thru T512) | Timers with a timing count in 32 bits. |
| (C1 thru C256) | Counters with a count in 32 bits. |

If any variable is accessed which allows the status of a PLC variable to be read or modified (I,O,M,R,T,C), block preparation is interrupted and the CNC waits for this command to be executed in order to restart block preparation.

## Read-only variables

### PLCMSG

Returns the number of the active PLC message with the highest priority and will coincide with the number displayed on screen (1··128). If there is none, it returns 0.

```
(P110=PLCMSG)
    It assigns to P100 the number of the active PLC message with the highest priority.
```

## Read-and-write variables

### PLCIn

This variable allows 32 PLC inputs to be read or modified starting with the one indicated (n).

The value of the inputs which are used by the electrical cabinet cannot be modified as their values are determined by it. Nevertheless, the status of the remaining inputs can be modified.

### PLCOn

This variable allows 32 PLC outputs to be read or modified starting from the one indicated (n).

```
(P110=PLCO 22)
    It assigns to parameter P110 the value of outputs O22 through O53 (32 outputs) of the
    PLC.
(PLCO 22=$F)
    It sets outputs O22 through O25 to "1" and outputs O26 through O53 to "0".
```

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | ... | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | .... | 0 | 0 | 1 | 1 | 1 | 1 |
| Output | 53 | 52 | 51 | 50 | 49 | 48 | 47 | 46 | 45 | 44 | .... | 27 | 26 | 25 | 24 | 23 | 22 |

### PLCMn

This variable allows 32 PLC marks to be read or modified starting from the one indicated (n).

### PLCRn

This variable allows the status of 32 register bits to be read or modified starting from the one indicated (n).

### PLCTn

This variable allows the timer count to be read or modified starting from the one indicated (n).

### PLCCn

This variable allows the counter count to be read or modified starting from the one indicated (n).

**PLCMMn**

This variable permits reading or modifying the PLC mark (n).

(PLMM4=1)
    It sets mark M4 to ·1· and leaves the rest untouched.
(PLCM4=1)
    It sets mark M4 to ·1· and the following 31 marks (M5, through M35) to ·0·

**13.**

**FAGOR**

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2x

### 13.2.15 Variables associated with local parameters

The CNC allows 26 local parameters (P0-P25) to be assigned to a subroutine, by using mnemonics PCALL and MCALL. In addition to performing the required subroutine these mnemonics allow local parameters to be initialized.

## Read-only variables

**CALLP**

Allows us to know which local parameters have been defined and which have not, in the call to the subroutine by means of the PCALL or MCALL mnemonic.

The information will be given in the 26 least significant bits (bits 0··25), each of these corresponding to the local parameter of the same number; thus bit 12 corresponds to P12.

Each bit will indicate whether the corresponding local parameter has been defined (=1) or not (0).

| Bit | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | ... | 5 | 4 | 3 | 2 | 1 | 0 |
|-----|----|----|----|----|----|----|----|----|----|----|-----|---|---|---|---|---|---|
|     | 0  | 0  | 0  | 0  | 0  | 0  | *  | *  | *  | *  | ... | * | * | * | * | * | * |

**Example:**

```
;Call to subroutine 20.
(PCALL 20, P0=20, P2=3, P3=5)
…
…
;Beginning of subroutine 20.
(SUB 20)
(P100 = CALLP)
…
…
```

In parameter P100 the following will be obtained:

| 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 1101 | LSB |
|------|------|------|------|------|------|------|------|-----|

### 13.2.16 Sercos variables

They are used in the data exchange via Sercos between the CNC and the drives.

## Read-only variables

**TSVAR(X-C)**          **TSVARS**          **TSSVAR**

It returns the third attribute of the Sercos variable corresponding to the "identifier". The third attribute is used in particular software applications and its information is coded according to the Sercos standard.

TSVAR(X-C)   identifier   ... for the axes.

TSVARS       identifier   ... for the main spindle.

TSSVAR       identifier   ... for the second spindle.

> (P110=SVARX 40)
> It assigns to parameter P110 the third attribute of the Sercos variable of identifier 40 of the X axis which corresponds to "VelocityFeedback"

## Write variables

**SETGE(X-C)**          **SETGES**          **SSETGS**

The drive may have up to 8 gear ratios (0 through 7). Sercos identifier 218, GearRatioPreselection.

It may also have up to 8 parameter sets (0 through 7). Sercos identifier 217, ParameterSetPreselection.

With these variables the work range or gear ratio and the parameter set of each drive may be modified.

SETGE(X-C)  ... for the axes.

SETGES       ... for the main spindle.

SSETGS       ... for the second spindle

The 4 least significant bits of these variables indicate the gear ratio and the other 4 the parameter set to be selected.

## Read-and-write variables

**SVAR(X-C)**          **SVARS**          **SSVARS**

They permit reading or modifying the value of the Sercos variable corresponding to the axis identifier.

SVAR(X-C)    identifier   ... for the axes.

SVARS        identifier   ... for the main spindle.

SSVARS       identifier   ... for the second spindle.

### 13.2.17 Software & hardware configuration variables

## Read-only variables

**HARCON**

It indicates, with bits, the CNC's hardware configuration. The bit will be "1" when the relevant configuration is available.

**CNC8055 model:**

| Bit | Meaning | |
|---|---|---|
| 4,3,2,1 | 0000 | 8055 FL model. |
|  | 0010 | 8055 Power model. |
| 5 | Sercos integrated into CPU board. | |
| 6 | Sercos module on manager board. | |
| 7 | Axes module. | |
| 10,9,8 | 001 | One I/O module. |
|  | 010 | Two I/O modules. |
|  | 011 | Three I/O modules. |
|  | 100 | Four I/O modules. |
| 14 | It has analog video. | |
| 15 | It has CAN integrated into the CPU board. | |
| 18,17,16 | Keyboard type (technical service department). | |
| 20.19 | CPU type (technical service department). | |
| 23,22,21 | 1xx | CPU PPC5200. |
| 26,25,24 | 000 | Color LCD Monitor. |
|  | 001 | Monochrome LCD monitor. |
| 30 | Ethernet connector integrated in the CPU. | |
| 31 | Compact flash memory (KeyCF). | |

**CNC8055i model:**

| Bit | Meaning | |
|---|---|---|
| 4, 3, 2, 1 | 0100 | 8055i FL model. |
| | 0110 | 8055i Power model. |
| 5 | Sercos (digital model). | |
| 6 | Reserved. | |
| 9, 8, 7 | 000 | Expansion board missing. |
| | 001 | "Feedback + I/O" expansion board. |
| | 010 | Feedback-only expansion board. |
| | 011 | I/O-only expansion board. |
| | 101 | "Axes 2" board for expansion of "feedback + I/O". |
| | 110 | "Axes 2" board for expansion of feedback only. |
| | 111 | "Axes 2" board for expansion of I/O only. |
| 10 | Axis board with 12-bit (=0) or 16-bit (=1) Digital/Analog converter. | |
| 12, 11 | Reserved. | |
| 14, 13 | Reserved. | |
| 15 | It has CAN (digital module). | |
| 18,17,16 | Keyboard type (technical service department). | |
| 20.19 | CPU type (technical service department). | |
| 23,22,21 | 1xx | CPU PPC5200. |
| 26,25,24 | 000 | Color LCD Monitor. |
| | 001 | Monochrome LCD monitor. |
| 30 | Ethernet.. | |
| 31 | Compact flash memory (KeyCF). | |

**HARCOA**

It indicates, with bits, the CNC's hardware configuration. The bit will be "1" when the relevant configuration is available.

**CNC8055 model:**

| Bit | Meaning |
|---|---|
| 0 | Axis module 2. |
| 1 | It has a connector for compact flash. |
| 10 | The axis board is "Axis module SB"<br>Note: Bit 0 of HARCOA must be set to 0. |

Bit ·1· only indicates whether the hardware has a connector for the compact flash or not, it does not indicate whether the compact flash is inserted or not.

**CNC8055i model:**

| Bit | Meaning |
|---|---|
| 0 | "Axes 2" board. |
| 1 | It has a connector for compact flash. |
| 10 | The axis board is "Axis module SB"<br>Note: Bit 0 of HARCOA must be set to 0. |

Bit ·1· only indicates whether the hardware has a connector for the compact flash or not, it does not indicate whether the compact flash is inserted or not.

**13.**

HIGH-LEVEL LANGUAGE PROGRAMMING

Variables

**FAGOR**

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

**IDHARH          IDHARL**

They return, in BCD code, the hardware identification number corresponding to the KeyCF. It is the number appearing on the software diagnosis screen.

Since the identification number has 12 digits, the IDHARL variable shows the 8 least significant bits and the IDHARH the 4 most significant bits.

Example:
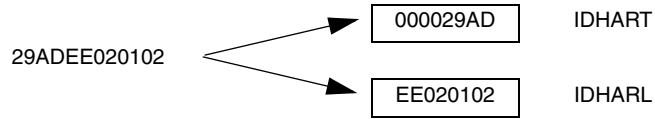


**SOFCON**

They return the software version numbers for the CNC and the hard disk.

Bits 15-0 return the CNC software version (4 digits)

Bits 31-16 return the software version of the hard disk (HD) (4 digits)
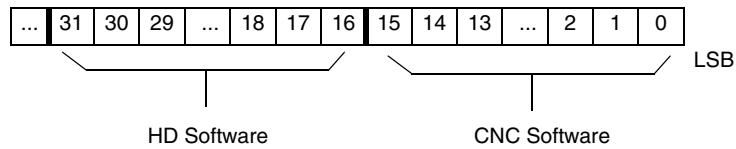


For example, SOFCON 01010311 indicates:

| | |
|---|---|
| Hard disk (HD) software version | 0101 |
| CNC software version | 0311 |

**HDMEGA**

It returns the size of the hard disk (in megabytes).

**KEYIDE**

Keyboard code, according to the auto-identification system.

| KEYIDE | CUSTOMY (P92) | Keyboard |
|---|---|---|
| 0 | - - - | Keyboard without auto-identification. |
| 130 | 254 | Mill model keyboard. |
| 131 | 254 | Lathe model keyboard. |
| 132 | 254 | Conversational mill model keyboard. |
| 133 | 254 | Conversational lathe model keyboard. |
| 134 | 254 | Educational model keyboard. |
| 135 | 252 | Operator panel: OP.8040/55.ALFA. |
| 136 | 0 | Operator panel OP.8040/55. MC. |
| 137 | 0 | Operator panel OP.8040/55. TC. |
| 138 | 0 | Operator panel OP.8040/55. MCO/TCO. |

### 13.2.18 Variables associated with telediagnosis

## Read-only variables

**HARSWA**        **HARSWB**

They return, in 4 bits, the central unit configuration, a value of "1" if it is present and "0" if not. Logic address (device select) set on each board with the dip-switches (see installation manual).

**HARSWA**

| Bits | Board |
|---|---|
| 31 - 28 | Large sercos |
| 27 - 24 | I/O 4 |
| 23 - 20 | I/O 3 |
| 19 - 16 | I/O 2 |
| 15 - 12 | I/O 1 |
| 11- 8 | Axes |
| 7 -4 | |
| 3 - 0 (LSB) | CPU |

**HARSWB**

| Bits | Board |
|---|---|
| 31 - 28 | |
| 27 - 24 | |
| 23 - 20 | Type of CAN at COM1. |
| 19 - 16 | |
| 15 - 12 | 0 - CAN board missing.<br>1 - CAN board in COM1<br>2 - CAN board in COM2<br>3 - Board in both COM |
| 11- 8 | Small sercos |
| 7 -4 | |
| 3 - 0 (LSB) | HD |

The CPU board must be present in all the configurations and set to "0". In the rest of cases, it returns a "0" if the board is missing.

There could be a large Sercos card (that takes up the whole module) or the small board that is installed in the CPU module.

There may be two types of CAN boards (value of ·0001· if it is an SJ1000 type and value of ·0010· if it is an OKI9225 type).

**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

### HARTST

It returns the result of the hardware test. The data comes at the least significant bits with a "1" if it failed and with a "0" if OK or if the relevant board is missing.

| Bits | |
|------|--|
| 14 | 24V test of IO4 module |
| 13 | Inside temperature |
| 12 | I/O 3                    (Board voltage) |
| 11 | I/O 2                    (Board voltage) |
| 10 | I/O 1                    (Board voltage) |
| 8 | Axes                    (Board voltage) |
| 7 | +3.3 V                  (Power supply) |
| 6 | 11                      (Power supply) |
| 5 | GNDA                    (Power supply) |
| 4 | - 15 V                  (Power supply) |
| 3 | + 15 V                  (Power supply) |
| 2 | Battery                 (Power supply) |
| 1 | - 5 V                   (Power supply) |
| 0 (LSB) | + 5 V               (Power supply) |

### MEMTST

It returns the result of the memory test. Each data uses 4 bits that will be at "1" if the test is OK and will have a value other than "1" if there is an error.

| Bits | Test |
|------|------|
| 30 | Test status |
| ... | ... |
| ... | ... |
| 19 - 16 | Caché |

| Bits | Test |
|------|------|
| 15 - 12 | Sdram |
| 11- 8 | HD |
| 7 -4 | Flash |
| 3 - 0 (LSB) | Ram |

Bit 30 stays at "1" during the test.

### NODE

It returns the number of the node used to configure the CNC in the Sercos ring.

### VCHECK

It returns the checksum of the code for the software version installed. It is the value appearing on the code test.

### IONODE

It returns in 16 bits the position of the "ADDRESS" switch of the CAN of the I/O. If it is not connected, it returns the value 0xFFFF.

### IOSLOC

They may be used to read the number of local digital I/O available.

| Bit | Meaning |
|------|---------|
| 0 - 15 | Number of inputs. |
| 16 - 31 | Number of outputs. |

**IOSREM**

They may be used to read the number of remote digital I/O available.

| Bit | Meaning |
| --- | --- |
| 0 - 15 | Number of inputs. |
| 16 - 31 | Number of outputs. |

### 13.2.19  Operating-mode related variables

## Read-only variables related to the standard mode

**OPMODE**

It returns the code corresponding to the selected operating mode.

0 = Main menu.

10 = Automatic execution.

11 = Single block execution.

12 = MDI in EXECUTION.

13 = Tool inspection.

14 = Repositioning.

15 = Block search executing G.

16 = Block search executing G, M, S, T.

20 = Theoretical path simulation.

21 = G function simulation.

22 = G, M, S and T function simulation.

23 = Simulation with movement in the main plane.

24 = Simulation with rapid movement.

25 = Rapid simulation with S=0.

30 = Normal editing.

31 = User editing.

32 = TEACH-IN editing.

33 = Interactive editor.

34 = Profile editor.

40 = Movement in continuous JOG.

41 = Movement in incremental JOG.

42 = Movement with electronic handwheel.

43 = HOME search in JOG.

44 = Position preset in JOG.

45 = Tool calibration.

46 = MDI in JOG.

47 = User JOG operation.

50 = Zero offset table.

51 = Tool offset table.

52 = Tool table.

53 = Tool magazine table.

54 = Global parameter table.

55 = Local parameter table.

56 = User parameter table.

57 = OEM parameter table.

60 = Utilities.

63 = Change of protections.

70 = DNC status.

71 = CNC status.


80 = PLC file editing.

81 = PLC program compilation.

82 = PLC monitoring.

83 = Active PLC messages.

84 = Active PLC pages.

85 = Save PLC program.

86 = Restore PLC program.

87 = PLC usage maps.

88 = PLC statistics.


90 = Customizing.


100 = General machine parameter table.

101 = Axis machine parameter tables.

102 = Spindle machine parameter table.

103 = Serial port machine parameter tables.

104 = PLC machine parameter table.

105 = M function table.

106 = Leadscrew error compensation tables and cross compensation tables.

107 = Machine parameter table for Ethernet.


110 = Diagnosis: configuration

111 = Diagnosis: hardware test.

112 = Diagnosis: RAM memory test.

113 = Diagnosis: Flash memory test.

114 = User diagnosis.

115 = Hard disk diagnosis (HD).

116 = Circle geometry test.

117 = Oscilloscope.


120 = DERGAIN auto-adjustment.


130 = Change of dates.

131 = Change of passwords.

## Read-only variables related to the conversational mode (MC, MCO) and configurable mode M, ([SHIFT]-[ESC]).

In these work modes, it is recommended to use variables OPMODA, OPMODB and OPMODC. The OPMODE variable is generic and contains different values to those of the standard mode.

### OPMODE

It returns the code corresponding to the selected operating mode.

0 = CNC starting up.

10 = In execution mode.

In execution or waiting for the [CYCLE START] key (drawing of the [CYCLE START] on top).

12 = Indicates one of the following situations:

- In MDI mode, when pressing the ISO key from the jog mode or tool inspection mode.

- One of the following fields of the main screen that admits the START key has been selected: Axes, T, F or S.

21 = In graphic simulation mode.

30 = Cycle editing.

40 = In Jog mode (standard screen).

43 = Doing a home search.

45 = In tool calibration mode.

60 = Managing parts. PPROG mode.

### OPMODA

Indicates the operating mode currently selected when working with the main channel.

Use the OPMODE variable to know at any time the selected operating mode (main channel, user channel, PLC channel).

This information is given at the least significant bits with a "1" when active and with a "0" when not active or when it is not available in the current version.

| | |
|---|---|
| Bit 0 | Program in execution. |
| Bit 1 | Program in simulation. |
| Bit 2 | Block in execution via MDI, JOG. |
| Bit 3 | Repositioning in progress. |
| Bit 4 | Program interrupted by CYCLE STOP. |
| Bit 5 | MDI, JOG block interrupted. |
| Bit6 | Repositioning interrupted. |
| Bit 7 | In tool inspection. |
| Bit 8 | Block in execution via CNCEX1. |
| Bit 9 | CNCEX1 block interrupted. |
| Bit 10 | CNC ready to accept JOG movements: manual, handwheel, teaching, inspection. |
| Bit 11 | CNC ready to accept running order (START): execution, simulation with movement and MDI modes. |
| Bit 12 | The CNC is not ready to execute anything involving axis or spindle movement. |
| Bit 13 | It identifies the block search. |

**OPMODB**

Indicates the type of simulation currently selected. This information is given at the least significant bits indicating with a "1" the one currently selected.

Bit 0          Theoretical path.

Bit 1          G functions.

Bit 2          G M S T functions.

Bit 3          Main plane.

Bit 4          Rapid.

Bit 5          Rapid (S=0).

**OPMODC**

Indicates the axes selected by handwheel. This information is given at the least significant bits indicating with a "1" the one currently selected.

Bit 0          Axis 1.

Bit 1          Axis 2.

Bit 2          Axis 3.

Bit 3          Axis 4.

Bit 4          Axis 5.

Bit 5          Axis 6.

Bit 6          Axis 7.

Bit 7

Bit 8

The axis name corresponds to the number according to the programming order for them.

Example: If the CNC controls axes X, Y, Z, U, B, C,  axis 1=X, axis2=Y, axis3=Z, axis4=U, axis5=B, axis6=C.

**FAGOR**

FAGOR AUTOMATION

### 13.2.20 Other variables

## Read-only variables

**NBTOOL**

Indicates the tool number being managed. This variable can only be used within the tool change subroutine.

Example: There is a manual tool changer. Tool T1 is currently selected and the operator requests tool T5.

The subroutine associated with the tools may contain the following instructions:

```
(P103 = NBTOOL)
(MSG "SELECT T?P103 AND PRESS CYCLE START")
```

Instruction (P103 = NBTOOL) assigns the number of the tool currently being managed to parameter P103. Therefore, P103=5.

The message displayed by the CNC will be ""SELECT T5 AND PRESS CYCLE START".

**PRGN**

Returns the program number being executed. If none is selected, a value of -1 is returned.

**BLKN**

It returns the label number of the last executed block.

**GGSA**

It returns the status of functions G00 through G24. The status of each one of the functions will be given in the 25 least significant bits and it will be indicated by a 1 when active and a 0 when not active or when not available in the current software version.

| G24 | G23 | G22 | G21 | G20 | ... | G04 | G03 | G02 | G01 | G00 |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|

```
CNCRD (GGSA, R110, M10)
    Loads register R110 with the status of functions G00 through G24.
```

**GGSB**

It returns the status of functions G25 through G49. The status of each one of the functions will be given in the 25 least significant bits and it will be indicated by a 1 when active and a 0 when not active or when not available in the current software version.

| G49 | G48 | G47 | G46 | G45 | ... | G29 | G28 | G27 | G26 | G25 |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|

**GGSC**

It returns the status of functions G50 through G24. The status of each one of the functions will be given in the 25 least significant bits and it will be indicated by a 1 when active and a 0 when not active or when not available in the current software version.

| G74 | G73 | G72 | G71 | G70 | ... | G54 | G53 | G52 | G51 | G50 |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|

**GGSD**

It returns the status of functions G5 through G99. The status of each one of the functions will be given in the 25 least significant bits and it will be indicated by a 1 when active and a 0 when not active or when not available in the current software version.

| G99 | G98 | G97 | G96 | G95 | ... | G79 | G78 | G77 | G76 | G75 |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|

**GGSE**

It returns the status of functions G100 through G124. The status of each one of the functions will be given in the 25 least significant bits and it will be indicated by a 1 when active and a 0 when not active or when not available in the current software version.

| G124 | G123 | G122 | G121 | G120 | ... | G104 | G103 | G102 | G101 | G100 |
|------|------|------|------|------|-----|------|------|------|------|------|

**GGSF**

It returns the status of functions G125 through G149. The status of each one of the functions will be given in the 25 least significant bits and it will be indicated by a 1 when active and a 0 when not active or when not available in the current software version.

| G149 | G148 | G147 | G146 | G145 | ... | G129 | G128 | G127 | G126 | G125 |
|------|------|------|------|------|-----|------|------|------|------|------|

**GGSG**

It returns the status of functions G150 through G174. The status of each one of the functions will be given in the 25 least significant bits and it will be indicated by a 1 when active and a 0 when not active or when not available in the current software version.

| G174 | G173 | G172 | G171 | G170 | ... | G154 | G153 | G152 | G151 | G150 |
|------|------|------|------|------|-----|------|------|------|------|------|

**GGSH**

It returns the status of functions G175 through G199. The status of each one of the functions will be given in the 25 least significant bits and it will be indicated by a 1 when active and a 0 when not active or when not available in the current software version.

| G199 | G198 | G197 | G196 | G195 | ... | G179 | G178 | G177 | G176 | G175 |
|------|------|------|------|------|-----|------|------|------|------|------|

**GGSI**

It returns the status of functions G200 through G224. The status of each one of the functions will be given in the 25 least significant bits and it will be indicated by a 1 when active and a 0 when not active or when not available in the current software version.

| G224 | G223 | G222 | G221 | G220 | ... | G204 | G203 | G202 | G201 | G200 |
|------|------|------|------|------|-----|------|------|------|------|------|

**GGSJ**

It returns the status of functions G225 through G249. The status of each one of the functions will be given in the 25 least significant bits and it will be indicated by a 1 when active and a 0 when not active or when not available in the current software version.

| G249 | G248 | G247 | G246 | G245 | ... | G229 | G228 | G227 | G226 | G225 |
|------|------|------|------|------|-----|------|------|------|------|------|

**GGSK**

It returns the status of functions G250 through G274. The status of each one of the functions will be given in the 25 least significant bits and it will be indicated by a 1 when active and a 0 when not active or when not available in the current software version.

| G274 | G273 | G272 | G271 | G270 | ... | G254 | G253 | G252 | G251 | G250 |
|------|------|------|------|------|-----|------|------|------|------|------|

**13.**

**FAGOR**

FAGOR AUTOMATION

**CNC 8055
CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

**GGSL**

It returns the status of functions G75 through G299. The status of each one of the functions will be given in the 25 least significant bits and it will be indicated by a 1 when active and a 0 when not active or when not available in the current software version.

| G299 | G298 | G297 | G296 | G295 | ... | G279 | G278 | G277 | G276 | G275 |
|------|------|------|------|------|-----|------|------|------|------|------|

**GGSM**

It returns the status of functions G300 through G324. The status of each one of the functions will be given in the 25 least significant bits and it will be indicated by a 1 when active and a 0 when not active or when not available in the current software version.

| G324 | G323 | G322 | G321 | G320 | ... | G304 | G303 | G302 | G301 | G300 |
|------|------|------|------|------|-----|------|------|------|------|------|

**GGSN**

It returns the status of functions G325 through G349. The status of each one of the functions will be given in the 25 least significant bits and it will be indicated by a 1 when active and a 0 when not active or when not available in the current software version.

| G349 | G348 | G347 | G346 | G345 | ... | G329 | G328 | G327 | G326 | G325 |
|------|------|------|------|------|-----|------|------|------|------|------|

**GGSO**

It returns the status of functions GG350 through G374. The status of each one of the functions will be given in the 25 least significant bits and it will be indicated by a 1 when active and a 0 when not active or when not available in the current software version.

| G374 | G373 | G372 | G371 | G370 | ... | G354 | G353 | G352 | G351 | G350 |
|------|------|------|------|------|-----|------|------|------|------|------|

**GGSP**

It returns the status of functions G375 through G399. The status of each one of the functions will be given in the 25 least significant bits and it will be indicated by a 1 when active and a 0 when not active or when not available in the current software version.

| G399 | G398 | G397 | G396 | G395 | ... | G379 | G378 | G377 | G376 | G375 |
|------|------|------|------|------|-----|------|------|------|------|------|

**GGSQ**

It returns the status of functions G400 through G424. The status of each one of the functions will be given in the 25 least significant bits and it will be indicated by a 1 when active and a 0 when not active or when not available in the current software version.

| G424 | G423 | G422 | G421 | G420 | ... | G404 | G403 | G402 | G401 | G400 |
|------|------|------|------|------|-----|------|------|------|------|------|

**GSn**

Returns the status of the G function indicated (n). 1 if it is active and 0 if not.

```
(P120=GS17)
    It assigns the value 1 to parameter P120 if the G17 function is active and 0 if not.
```

**MSn**

Returns the status of the M function indicated (n). 1 if it is active and 0 if not.

These functions are M00, M01, M02, M03, M04, M05, M06, M08, M09, M19, M30, M41, M42, M43, M44 and M45.

**PLANE**

Returns data on the abscissa axis (bits 4 to 7) and the ordinate axis (bits 0 to 3) of the active plane in 32 bits and in binary.

| ... | ... | ... | ... | ... | ... | 7654 | 3210 | lsb |
|-----|-----|-----|-----|-----|-----|------|------|-----|

Abscissa axis _____    _____ Ordinate axis
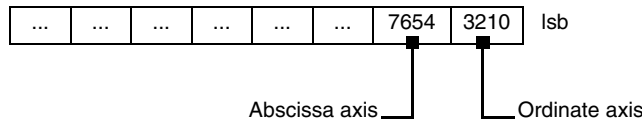
The axes are coded in 4 bits and indicate the axis number according to the programming order.

Example: If the CNC controls the X,Y,Z,U,B,C axes and the ZX plane (G18) is selected.

(P122 = GS17) assigns value $31 to parameter P122.

| 0000 | 0000 | 0000 | 0000 | 0000 | 0000 | 0011 | 0001 | LSB |
|------|------|------|------|------|------|------|------|-----|

Abscissa axis       = 3 (0011)       => Z axis

Ordinate axis       = 1 (0001)       => X axis

**LONGAX**

It returns the number according to the programming order corresponding to the longitudinal axis. This will be the one selected with the G15 function and by default the axis perpendicular to the active plane, if this is XY, ZX or YZ.

Example:

If the CNC controls the X, Y, Z, U, B, C axes and the U axis is selected.

(P122 = LONGAX) assigns the value 4 to parameter P122.

**MIRROR**

Returns in the least significant bits of the 32-bit group, the status of the mirror image of each axis, 1 in the case of being active and 0 if not.

| Bit 8 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | LSB |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-----|
|       |       | Axis 7 | Axis 6 | Axis 5 | Axis 4 | Axis 3 | Axis 2 | Axis 1 |     |

The axis name corresponds to the number according to the programming order for them.

Example: If the CNC controls axes X, Y, Z, U, B, C,  axis 1=X, axis2=Y, axis3=Z, axis4=U, axis5=B, axis6=C.

**SCALE**

It returns the general scaling factor being applied.

**SCALE(X-C)**

Returns the specific scaling factor of the indicated axis (X-C).

**ORGROT**

It returns the rotation angle of the coordinate system currently selected with G73. Its value is given in degrees (within ±99999.9999).

**ROTPF**

Returns the abscissa value of the rotation center with respect to the Cartesian coordinate origin. It is given in the active units:

If G70, in inches (within ±3937.00787).

If G71, in millimeters (within ±99999.9999).

**ROTPS**

Returns the ordinate value of the rotation center with respect to the Cartesian coordinate origin. It is given in the active units:

If G70, in inches (within ±3937.00787).

If G71, in millimeters (within ±99999.9999).

**PRBST**

Returns probe status.

0 = the probe is not touching the part.

1 = the probe is touching the part.

If this variable is accessed, block preparation is interrupted and the CNC waits for this command to be executed to resume block preparation.

**CLOCK**

Returns the time in seconds indicated by the system clock. Possible values 0..4294967295.

If this variable is accessed, block preparation is interrupted and the CNC waits for this command to be executed to resume block preparation.

**TIME**

Returns the time in hours-minutes-seconds format.

```
(P150=TIME)
    Loads P150 with hh-mm-ss. For example if the time is: 34sec. P150 = 182234.
```

If this variable is accessed, block preparation is interrupted and the CNC waits for this command to be executed to resume block preparation.

**DATE**

Returns the date in year-month-day format.

```
(P151=DATE)
    It assigns to P151 the year-month-day. For example if the date is April 25th 1992, P151
    = 920425.
```

If this variable is accessed, block preparation is interrupted and the CNC waits for this command to be executed to resume block preparation.

**CYTIME**

It returns the amount of time (in hundredths of a second) elapsed executing the part. It ignores the time the execution has been interrupted. Possible values 0..4294967295.

If this variable is accessed, block preparation is interrupted and the CNC waits for this command to be executed to resume block preparation.

**FIRST**

Indicates whether it is the first time that a program has been run or not. It returns a value of 1 if it is the first time and 0 if not.

A first-time execution is considered as being one which is done:

• After turning on the CNC.

• After pressing [SHIFT]+[RESET].

• Every time a new program is selected.

**ANAIn**

It returns the status of the indicated analog input (n). The value given in Volts and in ±1.4 format.

- At the –Axes– module, it is possible to select one of the 8 analog inputs (1··8) available. The values returned will be within the ±5 V range.

- At the –Vpp Axes– module, it is possible to select one of the 4 analog inputs (1··4) available. The returned values will be either within the ±5 V or ±10 V range depending on how the analog inputs have been set.

If this variable is accessed, block preparation is interrupted and the CNC waits for this command to be executed to resume block preparation.

**AXICOM**

It returns in the 3 least significant bits the axis pairs switched with function G28.

|  | Pair 3 | | Pair 2 | | Pair 1 | | |
|---|---|---|---|---|---|---|---|
|  | Axis 2 | Axis 1 | Axis 2 | Axis 1 | Axis 2 | Axis 1 | LSB |

The axes are coded in 4 bits and indicate the axis number (1 to 7) according to their programming order.

If the CNC controls the X, Y, Z, B, C axes and G28BC has been programmed, the AXICOM variable will show:

|  | Pair 3 | | Pair 2 | | Pair 1 | | |
|---|---|---|---|---|---|---|---|
|  |  |  |  |  | C | B |  |
|  | 0000 | 0000 | 0000 | 0000 | 0101 | 0100 | LSB |

**TANGAN**

Variable associated with the tangential control function, G45. It indicates the programmed angular position.

**TPIOUT(X-C)**

Output of the PI of the tandem master axis in rpm.

**TIMEG**

It shows the timing status of the timer programmed with G4 K in the CNC channel. This variable, returns the time remaining to end the timing block in hundredths of a second.

**TIPPRB**

It indicates the PROBE cycle being executed at the CNC.

If PROBE1 is running, the TPPRB variable takes the value of 1, if PROBE2 is running, it takes the value of 2, etc. if PROBE12 is running, it takes the value of 12.

**TIPDIG**

It indicates the DIGIT cycle being executed at the CNC.

**PANEDI**

WINDRAW55 application. Number of the screen created by the user or the manufacturer and is being consulted.

**DATEDI**

WINDRAW55 application. Number of the element (item) being consulted.

**RIP**

Linear theoretical feedrate resulting from the next loop (in mm/min).

The calculation of the resulting feedrate ignores the rotary axes, slave axes (gantry, coupled and synchronized) as well as DRO axes.

**TEMPIn**

It returns the temperature in tenths of a degree detected by the PT100. It is possible to select one of the 4 temperature inputs (1··4) available.

## Read-and-write variables

### TIMER

This variable allows reading or modifying the time, in seconds, indicated by the clock enabled by the PLC. Possible values 0..4294967295.

If this variable is accessed, block preparation is interrupted and the CNC waits for this command to be executed to resume block preparation.

### PARTC

The CNC has a part counter whose count increases, in all modes except simulation, every time M30 or M02 is executed and this variable allows its value to be read or modified. This value will be between 0 and 4294967295

If this variable is accessed, block preparation is interrupted and the CNC waits for this command to be executed to resume block preparation.

### KEY

Returns the code of the last key accepted.

This variable may be used as a write variable only inside a customizing program (user channel).

If this variable is accessed, block preparation is interrupted and the CNC waits for this command to be executed to resume block preparation.

### KEYSRC

This variable allows reading or modifying the source of keystrokes, possible values being:

    0 = Keyboard.

    1 = PLC.

    2 = DNC.

The CNC only allows modification of this variable if this is at 0.

### ANAOn

This variable allows the required analog output (n) to be read or modified. The value assigned will be expressed in volts and in the ±2.4 format (±10 V).

The analog outputs which are free among the eight (1 through 8) available at the CNC may be modified, the corresponding error being displayed if an attempt is made to write in one which is occupied.

If this variable is accessed, block preparation is interrupted and the CNC waits for this command to be executed to resume block preparation.

### SELPRO

When having two probe inputs, it allows selecting the active input.

On power-up, it assumes the value of ·1· thus selecting the first probe input. To select the second probe input, set it to a value of ·2·.

Accessing this variable from the CNC interrupts block preparation.

### DIAM

It changes the programming mode for X axis coordinates between radius and diameter. When changing the value of this variable, the CNC assumes the new way to program the following blocks.

When the variable is set to ·1·, the programmed coordinates are assumed in diameter; when is set to ·0·, the programmed coordinates are assumed in radius.

This variable affects the display of the real value of the X axis in the coordinate system of the part and the reading of variables PPOSX, TPOSX and POSX.

On power-up, after executing an M02 or M30 and after an emergency or a reset, the variable is initialized according to the value of the DFORMAT parameter of the X axis. If this parameter has a value equal to or greater than 4, the variable takes a value of 1; otherwise, it takes the value of ·0·.

### PRBMOD

It indicates whether a probing error is to issued or not in the following cases, even if general machine parameter PROBERR (P119) =YES.

- When a G75 probing move finishes before the probe has touched part.
- When a G76 probing move finishes but the probe is still touching the part.

The PRBMOD variable takes the following values.

| Value | Meaning |
|-------|---------|
| 0 | An error message is issued. |
| 1 | No error message is issued. |

Default value 0.

The PRBMOD variable can be read and written from the CNC and the PLC an read from the DNC.

### DISABMOD

This variable is used to disable some actions or modes by setting the corresponding bit value to 1. This variable may be written from the PLC and read from the PLC, DNC and CNC.

The following table shows the meaning of each bit:

| Bit | Meaning |
|-----|---------|
| 0 | If 1, the PLC program cannot be displayed. The PLC in ladder (contacts) diagram cannot be displayed either. |
| 1 | If 1, the date cannot be changed, although it displays the access softkey. It is valid for the explorer and for "UTILITIES". |
| 2 | If 1, the passwords cannot be changed. The passwords cannot be seen or changed although it displays the access softkey. It is valid for the explorer and for "UTILITIES". |

### CYCCHORDERR

This variable defines the chordal error of the canned cycles. It may be read and written from the part-program.

The CYCCHORDERR variable lets modify the chordal error of the cycles so the user can increase or decrease it for the parts as needed.

Using this variable is necessary, for example, on parts with curved areas using the 3D pocket cycle. On these parts, if the radius is very large, the segments are noticeable. The parts get better by decreasing the chordal error.

Using this variable, the user can decrease the chordal error on the part as needed. Decreasing the chordal error can increase machining time.

Once the value of this variable has been changed, it remains active until the CNC is turned off.

Default value of the CYCCHORDERR variable (250 tenths of a micron).

**13.**

*Programming example:*

```
(CYCCHORDERR = 25)
(PCALL 9986, P200=0)
M30
```

It is recommended to use a CYCCHORDERR value of 25 tenths of a micron. This value improves part finish and it does not increase machining time too much.

FAGOR

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

## 13.3 CONSTANTS

Constants are defined as being all those fixed values which cannot be altered by a program. The following are considered as constants:

- Numbers expressed in the decimal system.
- Hexadecimal numbers.
- PI constant.
- Read-only tables and variables as their value cannot be altered with a program.

## 13.4 Operators

An operator is a symbol that indicates the mathematical or logic operations to carry out. The CNC has arithmetic, relational, logic, binary, trigonometric operators and special operators.

### *Arithmetic operators.*

| | | | |
|---|---|---|---|
| + | add. | P1=3 + 4 | P1=7 |
| - | subtraction, also a negative value. | P2=5 - 2 | P2=3 |
| | | P3= -(2 * 3) | P3=-6 |
| * | multiplication. | P4=2 * 3 | P4=6 |
| / | division. | P5=9 / 2 | P5=4.5 |
| MOD | Module or remainder of the division. | P6=7 MOD 4 | P6=3 |
| EXP | exponential. | P7=2 EXP 3 | P7=8 |

### *Relational operators.*

| | |
|---|---|
| EQ | equal. |
| NE | different. |
| GT | greater than. |
| GE | greater than or equal to. |
| LT | Less than. |
| LE | Less than or equal to. |

### *Logic and binary operators.*

NOT, OR, AND, XOR: The act as logic operators between conditions and as binary operators between variables and constants.

```
IF (FIRST AND GS1 EQ 1) GOTO N100
P5 = (P1 AND (NOT P2 OR P3))
```

### *Trigonometric functions.*

| | | | |
|---|---|---|---|
| SIN | sine. | P1=SIN 30 | P1=0.5 |
| COS | cosine. | P2=COS 30 | P2=0.8660 |
| TAN | tangent. | P3=TAN 30 | P3=0.5773 |
| ASIN | arc sine. | P4=ASIN 1 | P4=90 |
| ACOS | arc cosine. | P5=ACOS 1 | P5=0 |
| ATAN | arc tangent. | P6=ATAN 1 | P6=45 |
| ARG | ARG(x,y) arctangent y/x. | P7=ARG(-1,-2) | P7=243.4349 |

There are two functions for calculating the arc tangent ATAN which returns the result between ±90° and ARG given between 0 and 360°.

**Other functions.**

| | | | |
|---|---|---|---|
| ABS | absolute value. | P1=ABS -8 | P1=8 |
| LOG | decimal logarithm. | P2=LOG 100 | P2=2 |
| SQRT | square root. | P3=SQRT 16 | P3=4 |
| ROUND | rounding up an integer number. | P4=ROUND 5.83 | P4=6 |
| FIX | Integer. | P5=FIX 5.423 | P5=5 |
| FUP | if integer takes integer.<br>if not, takes entire part + 1. | P6=FUP 7<br>P6=FUP 5,423 | P6=7<br>P6=6 |
| BCD | converts given number to BCD. | P7=BCD 234 | P7=564 |

| 0010 | 0011 | 0100 |
|---|---|---|

| | | | |
|---|---|---|---|
| BIN | converts given number to binary. | P8=BIN $AB | P8=171 |

| 1010 | 1011 |
|---|---|

Conversions to binary and BCD are made in 32 bits, it being possible to represent the number 156 in the following formats:

| | |
|---|---|
| Decimal | 156 |
| Hexadecimal | 9C |
| Binary | 0000 0000 0000 0000 0000 0000 1001 1100 |
| BCD | 0000 0000 0000 0000 0000 0001 0101 0110 |

## 13.5 Expressions

An expression is any valid combination of operators, constants, parameters and variables.

All expressions must be placed between brackets, but if the expression is reduced to an integer, the brackets can be removed.

### 13.5.1 Arithmetic expressions

These are formed by combining functions and arithmetic, binary and trigonometric operators with the constants and variables of the language.

The priorities of the operators and the way they can be associated determine how these expressions are calculated:

| Priority from highest to lowest | To be associated |
| --- | --- |
| NOT, functions, - (negative) | from right to left. |
| EXP, MOD | from left to right. |
| * , / | from left to right. |
| +,- (add, subtract) | from left to right. |
| Relational operators | from left to right. |
| AND, XOR | from left to right. |
| OR | from left to right. |

Brackets should be used in order to clarify the order in which the expression is to be evaluated.

```
(P3 = P4/P5 - P6 * P7 - P8/P9 )
(P3 = (P4/P5)-(P6 * P7)-(P8/P9))
```

Using redundant or additional brackets will neither cause errors nor slow down the execution.

In functions, brackets must be used except when these are applied to a numerical constant, in which case they are optional.

(SIN 45) (SIN (45))    they're both valid and equivalent.

(SIN 10+5)            the same as ((SIN 10)+5).

Expressions can be used also to reference parameters and tables:

```
(P100 = P9)
(P100 = P(P7))
(P100 = P(P8 + SIN(P8 * 20)))
(P100 = ORGX 55)
(P100 = ORGX (12+P9))
(PLCM5008 = PLCM5008 OR 1)
    ; Selects single block execution mode (M5008=1)
(PLCM5010 = PLCM5010 AND $FFFFFFFE)
    ; Frees feedrate override (M5010=0)
```

### 13.5.2 Relational expressions

These are arithmetic expressions joined by relational operators.

 (IF (P8 EQ 12.8)
  ; It checks if the value of P8 is equal to 12.8.
 (IF (ABS(SIN(P24)) GT SPEED)
  ;Analyzes if the sine is greater than the spindle speed.
 (IF (CLOCK LT (P9 * 10.99))
  ; Analyzes if the clock count is less than (P9 * 10.99)

At the same time these conditions can be joined by means of logic operators.

 (IF ((P8 EQ 12.8) OR (ABS(SIN(P24)) GT SPEED)) AND (CLOCK LT (P9 * 10.99)) …

The result of these expressions is either true or false.

**13.**

**HIGH-LEVEL LANGUAGE PROGRAMMING**

Expressions

FAGOR

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

# PROGRAM CONTROL
# INSTRUCTIONS

# 14

The control instructions available to high-level programming can be grouped as follows:

- Assignment instructions.
- Display instructions.
- Enable-disable instructions.
- Flow control instructions.
- Subroutine instructions.
- Probe related instructions.
- Interruption-subroutine instructions.
- Program instructions.
- Kinematics related instructions.
- Screen customizing instructions.

Only one instruction can be programmed in each block, and no other additional information may be programmed in this block.

## 14.1    Assignment instructions

This is the simplest type of instruction and can be defined as:

(target = arithmetic expression)

A local or global parameter or a read-write variable may be selected as target. The arithmetic expression may be as complex as required or a simple numerical constant.

(P102 = FZLOY)
(ORGY 55 = (ORGY 54 + P100))

In the specific case of designating a local parameter using its name (A instead of P0, for example) and the arithmetic expression being a numerical constant, the instruction can be abbreviated as follows:

(P0=13.7) ==> (A=13.7) ==> (A13.7)

Within a single block, up to 26 assignments can be made to different targets, a single assignment being interpreted as the set of assignments made to the same target.

(P1=P1+P2, P1=P1+P3, P1=P1*P4, P1=P1/P5)

It is the same as saying:

(P1=(P1+P2+P3)*P4/P5).

The different assignments which are made in the same block will be separated by commas "**,**".

FAGOR

FAGOR AUTOMATION

CNC 8055
CNC 8055i

·M· & ·EN· MODELS
SOFT: V02.2X

·396·

## 14.2    Display instructions

### *(ERROR integer, "error text")*

This instruction stops the execution of the program and displays the indicated error, it being possible to select this error in the following ways:

(ERROR integer)

This will display the error number indicated and the text  associated to this number according to the CNC error code (should there be one).

(ERROR integer, "error text")

This will display the number and the error text indicated, it being necessary to write the text between quote marks "".

(ERROR "error text").

This will display the error text only.

The error number may be defined by means of a numerical constant or an arithmetic parameter. When using a local parameter, its numeric format must be used (P0 thru P25 instead of A thru Z).

Programming examples:

(ERROR 5)

(ERROR P100)

(ERROR "User error")

(ERROR 3, "User error")

(ERROR P120, "User error)

### *(MSG "message")*

This instruction will display the message indicated between quote marks.

The CNC screen is provided with an area for displaying DNC or user program messages, and always displays the last message received irrespective of where it has come from.

Example: (MSG "Check tool")

### *(DGWZ expression 1, expression 2, expression 3, expression 4, expression 5, expression 6)*

The DGWZ instruction (Define Graphic Work Zone) defines the graphics area.

Each expression forming the instruction syntax correspond to one of the limits and they must be defined in  millimeters or inches.

| | |
|---|---|
| expression 1 | X minimum |
| expression 2 | X maximum |
| expression 3 | Y minimum |
| expression 4 | Y maximum |
| expression 5 | Z minimum |
| expression 6 | Z maximum |

## 14.3 Enable-disable instructions

### (ESBLK and DSBLK)

After executing the mnemonic ESBLK, the CNC executes all the blocks that come after as if it were dealing with a single block.

This single block treatment is kept active until it is cancelled by executing the mnemonic DSBLK.

In this way, should the program be executed in the SINGLE BLOCK operating mode, the group of blocks which are found between the mnemonics ESBLK and DSBLK will be executed in a continuous cycle, i.e., execution will not be stopped at the end of a block but will continue by executing the following one.

```
G01 X10 Y10 F8000 T1 D1
(ESBLK)                               ; Start of single block
G02 X20 Y20 I20 J-10
G01 X40 Y20
G01 X40 Y40 F10000
G01 X20 Y40 F8000
(DSBLK)                               ; Cancellation of single block
G01 X10 Y10
M30
```

### (ESTOP and DSTOP)

After executing the mnemonic DSTOP, the CNC enables the Stop key, as well as the Stop signal from the PLC.

It will remain disabled until it is enabled once again by means of the mnemonic ESTOP.

### (EFHOLD and DFHOLD)

After executing the mnemonic DFHOLD, the CNC enables the Feed-Hold input from the PLC.

It will remain disabled until it is enabled once again by means of the mnemonic EFHOLD.

## 14.4    **Flow control instructions**

The GOTO and RPT instructions cannot be used in programs that are executed from a PC connected through the serial line.

### *( GOTO N(expression) )*

The mnemonic GOTO causes a jump within the same program, to the block defined by the label N(expression). The execution of the program will continue after the jump, from the indicated block.

The jump label can be addressed by means of a number or by any expression which results in a number.

```
        G00 X0 Y0 Z0 T2 D4
        X10
        (GOTO N22)                  ; Jump instruction
        X15 Y20                     ; It is not executed
        Y22 Z50                     ; It is not executed
N22     G01 X30 Y40 Z40 F1000       ; Continues execution in this block
        G02 X20 Y40 I-5 J-5
        …
```

### *( RPT N(expression), N(expression), P(expression) )*

The mnemonic RPT executes the part of the program between the blocks defined by means of the labels N(expression). The blocks to be executed may be in the execution program or in a RAM memory program.

The label P(expression) indicates the number of the program containing the blocks to be executed. If not defined, the CNC interprets that the portion to be repeated is located in the same program.

All the labels can be indicated by means of a number or by any expression which results in a number. The part of the program selected by means of the two labels must belong to the same program, by first defining the initial block and then the final block.

The execution of the program will continue in the block following the one in which the mnemonic RPT was programmed, once the selected part of the program has been executed.

```
N10     G00 X10
        Z20
        G01 X5
        G00 Z0
N20     X0
N30     (RPT N10, N20) N3
N40     G01 X20
        M30
        When reaching block N30, the program will execute section N10-N20 three times.
        Once this has been completed, the program will continue execution in block N40.
```

ℹ️  *Since the RPT instruction does not interrupt block preparation or tool compensation, it may be used when using the EXEC instruction and while needing to maintain tool compensation active.*

### *(IF condition <action1> ELSE <action2>)*

This instruction analyzes the given condition that must be a relational expression. If the condition is true (result equal to 1), <action1> will be executed, otherwise (result equal to 0) <action2> will be executed.

Example:

```
(IF (P8 EQ 12.8) CALL 3 ELSE PCALL 5, A2, B5, D8)
If P8 = 12.8 executes the mnemonic (CALL3)
If P8<>12.8 executes the mnemonic (PCALL 5, A2, B5, D8)
```

The instruction can lack the ELSE part, i.e., it will be enough to program IF condition <action1>.

Example:

```
(IF (P8 EQ 12.8) CALL 3
```

FAGOR ➔

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2x

·**399**·

Both <action1> and <action2> can be expressions or instructions, except for mnemonics IF and SUB.

Due to the fact that in a high level block local parameters can be named by means of letters, expressions of this type can be obtained:

> (IF (E EQ 10) M10)

If the condition of parameter P5 (E) having a value of 10 is met, the miscellaneous function M10 will not be executed, since a high level block cannot have ISO code commands. In this case M10 represents the assignment of value 10 to parameter P12, i.e., one can program either:

> (IF (E EQ 10) M10) ó (IF (P5 EQ 10) P12=10)

## 14.5    Subroutine instructions

A subroutine is a part of a program which, being properly identified, can be called from any position of a program to be executed.

A subroutine can be kept in the memory of the CNC as an independent part of a program and be called one or several times, from different positions of a program or different programs.

Only subroutines stored in the CNC's RAM memory can be executed. Therefore, to execute a subroutine stored in the hard disk (KeyCF) or in a PC connected through the serial line, copy it first into the CNC's RAM memory.

If the subroutine is too large to be copied into RAM, it must be converted into a program and then the EXEC instruction must be used.

### *(SUB integer)*

The SUB instruction defines as subroutine the set of program blocks programmed next until reaching the RET subroutine. The subroutine is identified with an integer which also defines the type of subroutine; either general or OEM.

| | |
|---|---|
| Range of general subroutines | SUB 0000 - SUB 9999 |
| Range of OEM (manufacturer's) subroutines | SUB 10000 - SUB 20000 |

The OEM subroutines are treated like the general ones, but with the following restrictions:

- They can only be defined in OEM programs, having the [O] attribute. Otherwise, the CNC will display the corresponding error.

    Error 63: Program subroutine number 1 thru 9999.

- To execute an OEM subroutine using CALL, PCALL or MCALL, it must be inside an OEM program. Otherwise, the CNC will display the corresponding error.

    Error 1255: Subroutine restricted to an OEM program.

There can not be two subroutines with the same identification number in the CNC memory, even when they belong to different programs.

### *( RET )*

The mnemonic RET indicates that the subroutine which was defined by the mnemonic SUB, finishes in this block.

```
(SUB 12)                              ; Definition of subroutine 12
G91 G01 XP0 F5000
YP1
X-P0
Y-P1
(RET)                                 ; End of subroutine
```

**14.**

**( CALL (expression) )**

The mnemonic CALL makes a call to the subroutine indicated by means of a number or by means of any expression that results in a number.

As a subroutine may be called from a main program, or a subroutine, from this subroutine to a second one, from the second to a third, etc..., the CNC limits these calls to a maximum of 15 nesting levels, it being possible to repeat each of the levels 9999 times.
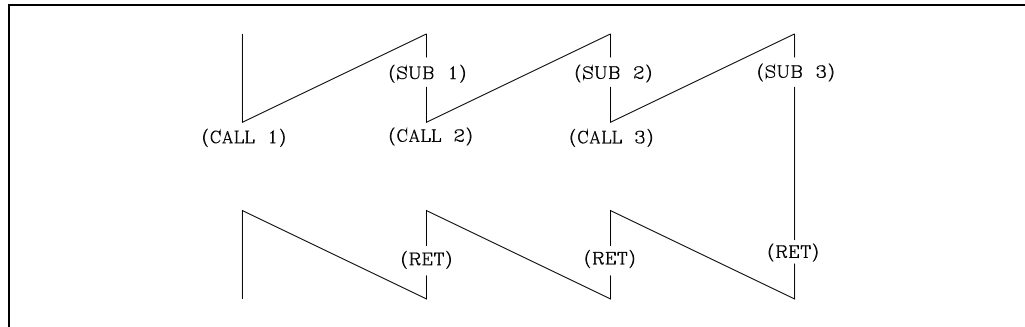


**Programming example.**
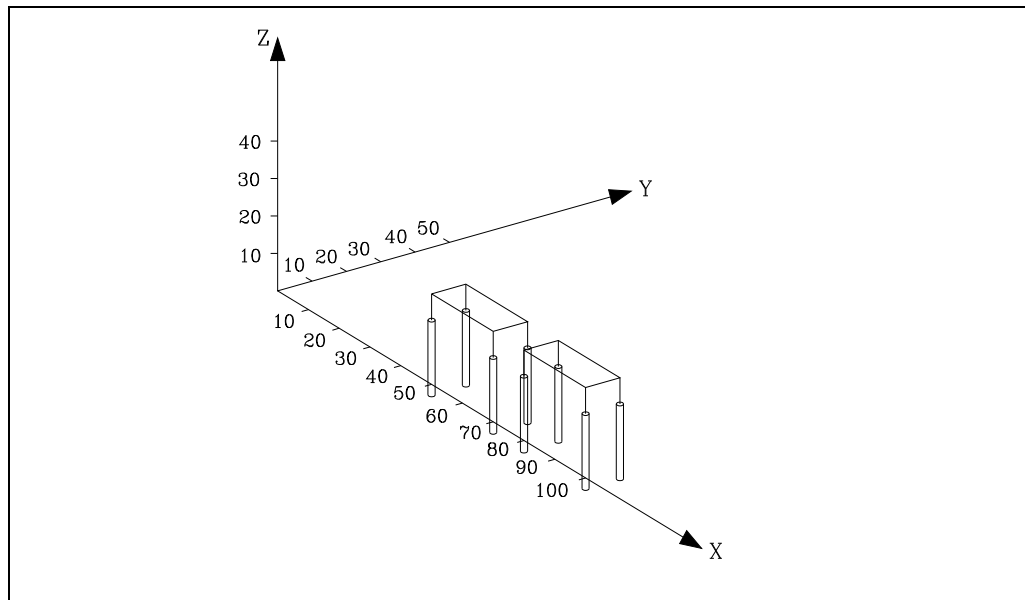


```
G90 G00 X30 Y20 Z10
(CALL 10)
G90 G00 X60 Y20 Z10
(CALL 10)
M30

(SUB 10)
G91 G01 X20 F5000
(CALL 11)                                      ; Drilling and threading
G91 G01 Y10
(CALL 11)                                      ; Drilling and threading
G91 G01 X-20
(CALL 11)                                      ; Drilling and threading
G91 G01 Y-10
(CALL 11)                                      ; Drilling and threading
(RET)

(SUB 11)
G81 G98 G91 Z-8 I-22 F1000 S5000 T1 D1         ; Drilling canned cycle
G84 Z-8 I-22 K15 F500 S2000 T2 D2              ; Threading canned cycle
G80
(RET)
```

**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

·M· & ·EN· Models
Soft: V02.2x

**(PCALL (expression), (assignment instruction), (assignment instruction),...) )**

The mnemonic PCALL calls the subroutine indicated by means of a number or any expression that results in a number. In addition, it allows up to a maximum of 26 local parameters of this subroutine to be initialized.

These local parameters are initialized by means of assignment instructions.

Example: (PCALL 52, A3, B5, C4, P10=20)

In this case, in addition to generating a new subroutine nesting level, a new local parameter nesting level will be generated, there being a maximum of 6 levels of local parameter nesting, within the 15 levels of subroutine nesting.

Both the main program and each subroutine that is found on a parameter nesting level, will have 26 local parameters (P0-P25).

**Programming example.**
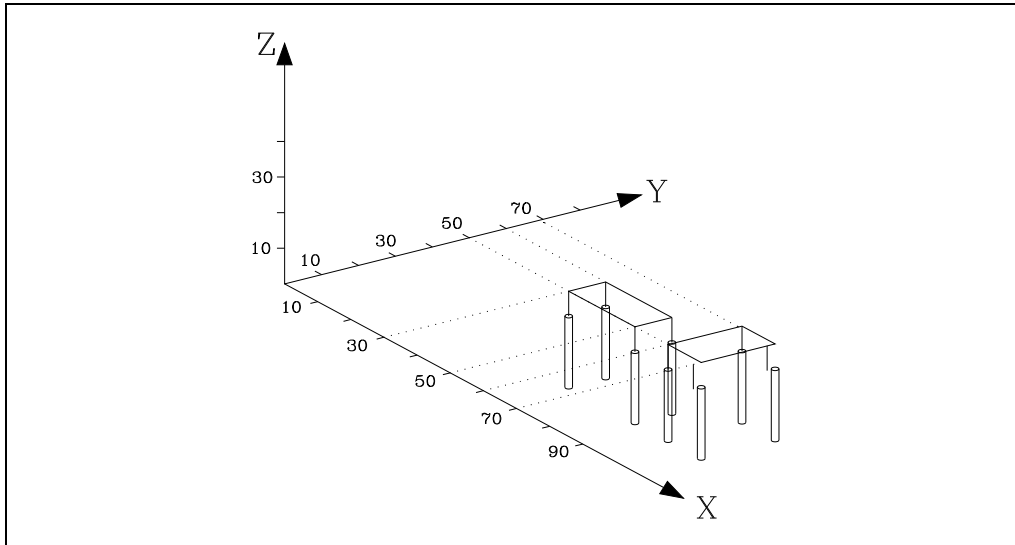


```
G90 G00 X30 Y50 Z0
(PCALL 10, P0=20, P1=10)          ; Also (PCALL 10, A20, B10)
G90 G00 X60 Y50 Z0
(PCALL 10, P0=10, P1=20)          ; Also (PCALL 10, A10, B20)
M30
(SUB 10)
G91 G01 XP0 F5000
(CALL 11)
G91 G01 YP1
(CALL 11)
G91 G01 X-P0
(CALL 11)
G91 G01 Y-P1
(CALL 11)
(RET)
(SUB 11)
G81 G98 G91 Z-8 I-22 F1000 S5000 T1 D1      ; Drilling canned cycle
G84 Z-8 I-22 K15 F500 S2000 T2 D2           ; Threading canned cycle
G80
(RET)
```

**(MCALL (expression), (assignment instruction), (assignment instruction),...) )**

By means of the mnemonic MCALL, any user-defined subroutine (SUB integer) acquires the category of canned cycle.

The execution of this mnemonic is the same as the mnemonic PCALL, but the call is modal, i.e., if another block with axis movement is programmed at the end of this block, after this movement, the subroutine indicated will be executed and with the same call parameters.

If, when a modal subroutine is selected, a movement block with a number of repetitions is executed, for example X10 N3, the CNC will execute the movement only once (X10) and after the modal subroutine, as many times as the number of repetitions indicates.

Should block repetitions be chosen, the first execution of the modal subroutine will be made with updated call parameters, but not for the remaining times, which will be executed with the values which these parameters have at that time.

If, when a subroutine is selected as modal, a block containing the MCALL mnemonic is executed, the present subroutine will lose its modal quality and the new subroutine selected will be changed to modal.

### ( MDOFF )

The MDOFF instruction indicates that the mode assumed by a subroutine with the MCALL instruction or a part-program with MEXEC ends in that block.

The use of modal subroutines simplifies programming.

**Programming example.**
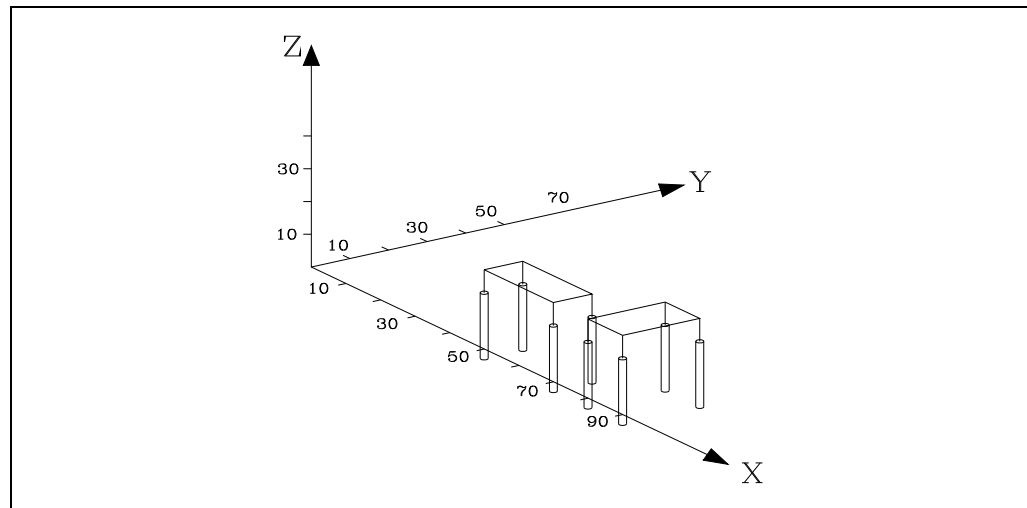


```
G90 G00 X30 Y50 Z0
(PCALL 10, P0=20, P1=10)
G90 G00 X60 Y50 Z0
(PCALL 10, P0=10, P1=20)
M30

(SUB 10)
G91 G01 XP0 F5000
(MCALL 11)
G91 G01 YP1
G91 G01 X-P0
G91 G01 Y-P1
(MDOFF)
(RET)
(SUB 11)
G81 G98 G91 Z-8 I-22 F1000 S5000 T1 D1
G84 Z-8 I-22 K15 F500 S2000 T2 D2
G80
(RET)
```

### 14.5.1 Calls to subroutines using G functions

Subroutine calls are made using the CALL and PCALL instructions. In addition to using these statements, it is also possible to make subroutine calls using specific G functions. This way, the calls to subroutines are more similar to machine tool language.

Functions G180-G189 and G380-G399 can call the associated OEM and user subroutine as long as they are global subroutines. G functions cannot be used to call local subroutines.

Up to 30 subroutines may be defined and associated with functions G180-G189 and G380-G399, and it is also possible to initialize local parameters for each subroutine.

When executing one of these functions, its associated subroutine will be executed.

Functions G180-G189 and G380-G399 are not modal.

## Programming format

The programming format is the following:

G180 <P0..Pn>
<P0..Pn> Optional. Initializing parameters.

**Example:**

G183 P1=12.3 P2=6
G187 A12.3 B45.3 P10=6

### *Defining local parameters.*

The values of the parameters are defined after the call function and they may be defined using the name of the parameter (P0-P25) or using letters (A-Z) so "A" is the same as P0 and "Z" is the same as P25.

Plus, parameters may also be programmed as follows:

• S=P100

• SP100

In either case, local parameter P18(S) would assume the established global parameter value P100.

The definitions described here may be combined in the same block.

## Nesting levels

If the functions initialize local parameters, this instruction generates a new nesting level .

The maximum parameter nesting level is 6, within the 15 nesting level of the subroutines, just like PCALL instructions.

## Identification via PLC

All the G functions are identified through GGS* read-only variables . GGSH and GGSP read-only variables are used to identify the new G functions via PLC; these variables return the status of the G functions.

## Executing a call

Each function G180-G189 and G380-G399, has its corresponding subroutine associated with it. Calling a G function means calling the subroutine of the same name.

FAGOR

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

## 14.6 Probe related instructions

### *(PROBE (expression), (assignment instruction), (assignment instruction),...) )*

The mnemonic PROBE calls the probe cycle indicated by means of a number or any expression that results in a number. In addition, it allows the local parameters of this subroutine to be initialized by means of assignment instructions.

This instruction also generates a new subroutine nesting level.

## 14.7    Interruption-subroutine instructions

Whenever one of the general interruption logic input is activated, "INT1" (M5024), "INT2" (M5025), "INT3" (M5026) or "INT4 (M5027), the CNC temporarily interrupts the execution of the program in progress and starts executing the interruption subroutine whose number is indicated by the corresponding general parameter.

With INT1 (M5024) the one indicated by machine parameter INT1SUB (P35)

With INT2 (M5025) the one indicated by machine parameter INT2SUB (P36)

With INT3 (M5026) the one indicated by machine parameter INT3SUB (P37)

With INT4 (M5027) the one indicated by machine parameter INT4SUB (P38)

The interruption subroutines are defined like any other subroutine by using the instructions: "(SUB integer)" and "(RET)".

These interruption subroutines do not change the nesting level of local parameters, thus only global parameters must be used in them.

Within an interruption subroutine, it is possible to use the "(REPOS X, Y, Z, ...)" instruction described next.

Once the execution of the subroutine is over, the CNC resumes the execution of the program which was interrupted.

### ( REPOS X, Y, Z, ... )

The REPOS instruction must always be used inside an interruption subroutine and facilitates the repositioning of the machine axes to the point of interruption.

When executing this instruction, the CNC moves the axes to the point where the program was interrupted.

Inside the REPOS instruction, indicate the order the axes must move to the point where the program was interrupted.

• The axes move one by one.

• It is not necessary to define all the axes, only those to be repositioned.

• The axes that make up the main plane of the machine move together. Both axes need not be defined because the CNC moves the first one. The movement is not repeated when defining the second one, it is ignored.

Example:

The main plane is formed by the X and Y axes, the Z axis is the longitudinal (perpendicular) axis and the machine uses the C and W axes as auxiliary axes. It is desired to first move the C axis, then the X and Y axes and finally the Z axis.

This repositioning move may be defined in any of the following ways:

(REPOS C, X, Y, Z)(REPOS C, X, Z)(REPOS C, Y, Z)

If the REPOS instruction is detected while executing a subroutine not activated by an interruption input, the CNC will issue the corresponding error message.

## 14.8 Program instructions

With this CNC, from a program in execution, it is possible to:

- Execute another program. Instruction (EXEC P.....)
- Execute another program in modal mode. Instruction (MEXEC P.....)
- Generate a new program. Instruction (OPEN P.....)
- Add blocks to an existing program. Instruction (WRITE P.....)

### ( EXEC P(expression), (directory) )

The EXEC P instruction executes the part-program of the indicated directory

The part-program may be defined by a number or any expression resulting in a number.

By default, the CNC interprets that the part-program is in the CNC's RAM memory. If it is in another device, it must be indicated in (directory).

| | |
|---|---|
| HD | in the Hard Disk (KeyCF). |
| DNC2 | in a PC connected through the serial line. |
| DNCE | in a PC connected through Ethernet. |

### ( MEXEC P(expression), (directory) )

The MEXEC instruction executes the part-program of the indicated directory and it also becomes modal; i.e. if after this block, another one is programmed with axis movement; after this movement, it will execute the indicated program again.

The part-program may be defined with a number or with an expression whose result is a number.

By default, the CNC interprets that the part-program is in the CNC's RAM memory. If it is in another device, it must be so indicated in (directory):

| | |
|---|---|
| HD | in the Hard Disk (KeyCF). |
| DNC2 | in a PC connected through the serial line. |
| DNCE | in a PC connected through Ethernet. |

If while the modal part-program is selected, a motion block is executed with a number of repetitions (for example X10 N3), the CNC ignores the number of repetitions and executes the movement and the modal part-program only once.

If while a part-program is selected as modal, a block containing the MEXEC instruction is executed from the main program, the current part-program stops being modal and the part-program called upon with MEXEC will then become modal.

If within the modal part-program, an attempt is made to execute a block using the MEXEC instruction, it will issue the relevant error message.

1064: The program cannot be executed.

### ( MDOFF )

The MDOFF instruction indicates that the mode assumed by a subroutine with the MCALL instruction or a part-program with MEXEC ends in that block.

### ( OPEN P(expression), (destination directory), A/D, "program comment" )

The OPEN instruction begins editing a part-program. The part-program number may be indicated by a number or any expression resulting in a number.

By default, the new part-program edited will be stored in the CNC's RAM memory. To store it another device, it must be indicated in (destination directory).

| | |
|---|---|
| HD | in the Hard Disk (KeyCF). |
| DNC2 | in a PC connected through the serial line. |
| DNCE | in a PC connected through Ethernet. |

Parameter A/D is used when the program to be edited already exists.

A      The CNC appends the new blocks after the ones already existing.

D      The CNC deletes the existing program and starts editing a new one.

A program comment may also be associated with it; this comment will later be displayed next to it in the program directory.

To edit blocks, the WRITE instruction must be used as described next.

Notes:

If the program to be edited already exists and the A/D parameters are not defined, the CNC will display an error message when executing the block.

The program opened with the OPEN instruction is closed when executing an M30, or another OPEN instruction and after an Emergency or Reset.

From a PC, only programs stored in the CNC'S RAM memory, or in the hard disk (KeyCF) can be opened

### ( WRITE <block text> )

The mnemonic WRITE adds, after the last block of the program which began to be edited by means of the mnemonic OPEN P, the information contained in <block text> as a new program block.

When it is an ISO coded parametric block, all the parameters (global and local) are replaced by the numeric value they have at the time.

(WRITE G1 XP100 YP101 F100) => G1 X10 Y20 F100

When it is a parametric block edited in high level, use the "?" character to indicate that the parameter is supposed to be replaced by the numeric value it has at the time.

```
(WRITE (SUB P102))          =>      (SUB P102)
(WRITE (SUB ?P102))         =>      (SUB 55)

(WRITE (ORGX54=P103))       =>      (ORGX54=P103)
(WRITE (ORGX54=?P103))      =>      (ORGX54=222)

(WRITE (PCALL P104))        =>      (PCALL P104)
(WRITE (PCALL ?P104))       =>      (PCALL 25)
```

If the mnemonic WRITE is programmed without having programmed the mnemonic OPEN previously, the CNC will display the corresponding error, except when editing a user customized program, in which case a new block is added to the program being edited.

**Using the character "$" in the WRITE instruction to write the number of a parameter:**

While using the character "$" in the WRITE instruction, it will be possible to write the parameter number directly. For that, use the "$" character followed by "P" as long is it is preceded by an axis.

For example, programming (WRITE X$P100) the result will be: XP100.

To show something in $, the value must be programmed after the $ sign. But, to take the value from a parameter, put a space between the "$" sign and the parameter.

In short, these are the options:

- When programmed $P, it will output $P.
- When programmed $[space]P, it will output $[space] and the content of P.
- When programmed $[number], it will output $[number].

**Example:**

Being parameter P100=22.

| Program | Result |
|---|---|
| (WRITE XP100) | X22 |
| (WRITE X$P100) | XP100 |
| (WRITE $ P100) | $ 22 |
| (WRITE $3000) | $3000 |

**Example of the creation of a program which contains several points of a cardioid:**



$$| R = B \cos (Q/2) |$$
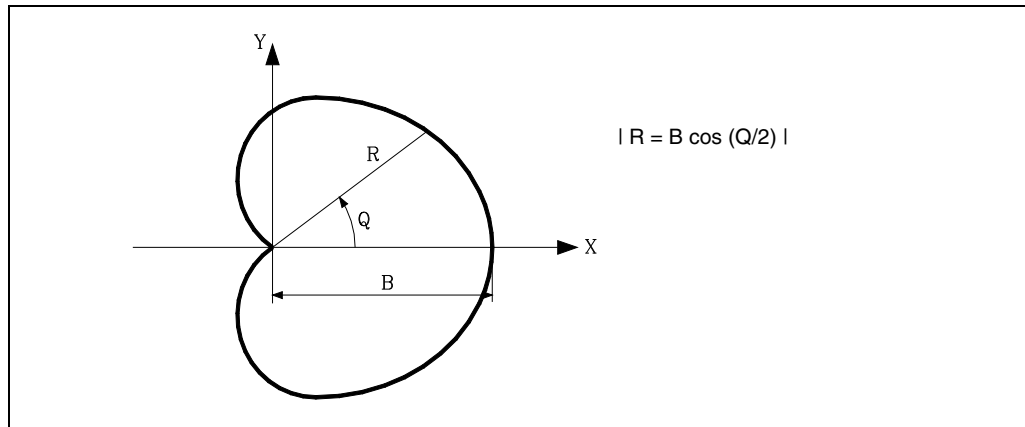
Subroutine number 2 is used, its parameters having the following meaning:

A or P0      Value of angle Q.

B or P1      Value of B.

C or P2      Angular increment for calculation.

D or P3      The maximum feedrate of the axes.

A way to use this example could be:

```
G00 X0 Y0
G93
(PCALL 2, A0, B30, C5, D500)
M30
```

Program generating subroutine.

```
      (SUB 2)
      (OPEN P12345)                  ; Starts editing of program P12345
      (WRITE FP3)                    ; Selects machining feedrate
N100  (P10=P1*(ABS(COS(P0/2))))      ; Calculates R
      (WRITE G01 G05 RP10 QP0)       ; Movement block
      (P0=P0+P2)                     ; New angle
      (IF (P0 LT 365) GOTO N100)     ; If angle smaller than 365º, calculates new
                                     ; point
      (WRITE M30)                    ; End of program block
      (RET)                          ; End of subroutine
```

## 14.9    Kinematics related instructions

Changing the kinematics requires modifying the general machine parameters associated with them and validating them. The machine parameters may be modified from the OEM subroutines and then the values validated using the INIPAR instruction.

An OEM program may have different subroutines that will set the machine parameters for each kinematics. Then, the spindle change subroutine will call the subroutine with the parameters to be activated and, then, will execute the INIPAR instruction.

### ( INIPAR )

This instruction validates the machine parameters modified from an OEM subroutine that defines the kinematics. This instruction is only valid when used in subroutines that are located inside OEM programs.

To validate the machine parameters associated with a kinematics, G48 and G49 must NOT be active. Otherwise, the CNC will display the corresponding error.

    1074: INIPAR cannot be executed.

## 14.10   Screen customizing instructions

Customizing instructions may be used only when customizing programs made by the user.

These customizing programs must be stored in the CNC's RAM memory and they may utilize the "Programming Instructions" and they will be executed in the special channel designed for this use; the program selected in each case will be indicated in the following general machine parameters.

In "USERDPLY" the program to be executed in the Execution Mode will be indicated.

In "USEREDIT" the program to be executed in the Editing Mode will be indicated.

In "USERMAN" the program to be executed in the Manual (JOG) Mode will be indicated.

In "USERDIAG" the program to be executed in the Diagnosis Mode will be indicated.

The customizing programs may have up to five nesting levels besides their current one. Also, the customizing instructions do not allow local parameters, nevertheless all global parameters may be used to define them.

### ( PAGE (expression) )

The mnemonic PAGE displays the page number indicated by means of a number or by means of any expression resulting in a number.

From V02.03 and following versions, the JPG/JPEG formats are supported. This way, if there is an "n.jpg", "n.jpeg" or "n.pan", the screen will display this file. When having several files, the order of priorities will be the following:

1. "n.jpg".

2. "n.jpeg".

3. "n.pan".

The format of the JPG/JPEG files must be a 3 digit number. For example "001.jpg" for page 1. The size of the page must be 638x335.

User-defined pages will be from page 0 to page 255 and will be defined from the CNC keyboard in the Graphic Editor mode and as indicated in the Operating Manual.

System pages will be defined by a number greater than 1000. See the corresponding appendix.

### ( SYMBOL (expression 1), (expression 2), (expression 3) )

The mnemonic SYMBOL displays the symbol whose number is indicated by means of the value of expression 1 once this has been evaluated.

Its position on screen is also defined by expression 2 (column) and by expression 3 (row).

Expression 1, expression 2 and expression 3 may contain a number or any expression resulting in a number.

From V02.03 and following versions, the PNG format is supported. This way, if there is an "n.png" file, it will be displayed in the position indicated by the expressions 2 and 3. If it is missing, it will display the "n.sim" file. The format of the PNG files must be a 3 digit number.

The CNC allows displaying any user-defined symbol (0-255) defined at the CNC keyboard in the Graphic Editor mode such as is indicated in the Operating Manual.

In order to position it within the display area its pixels must be defined, 0-639 for columns (expression 2) and 0-335 for rows (expression 3).

### (IB (expression) = INPUT "text", format)

The CNC has 26 data entry variables (IB0-1B25)

The IB mnemonic displays the text indicated in the data input window and stores the data input by the user in the entry variable indicated by means of a number or by means of any expression resulting in a number.

The wait for data entry will only occur when programming the format of the requested data. This format may have a sign, integer part and decimal part.

If it bears the "-" sign, it will allow positive and negative values, and if it does not have a sign, it will only allow positive values.

The integer part indicates the maximum number of digits (0-6) desired to the left of the decimal point.

The decimal part indicates the maximum number of digits (0-5) desired to the right of the decimal point.

If the numerical format is not programmed; for example: (IB1 =INPUT "text"), the mnemonic will only display the indicated text without waiting for the data to be entered.

**14.**

### ( ODW (expression 1), (expression 2), (expression 3) )

The mnemonic ODW defines and draws a white window on the screen with fixed dimensions (1 row and 14 columns).

Each mnemonic has an associated number which is indicated by the value of expression 1 once this has been evaluated.

Likewise, its position on screen is defined by expression 2 (row) and by expression 3 (column).

Expression 1, expression 2 and expression 3 may contain a number or any expression resulting in a number.

The CNC allows 26 windows (0-25) to be defined and their positioning within the display area, providing 21 rows (0-20) and 80 columns (0-79).

### ( DW(expression 1) = (expression 2), DW (expression 3) = (expression 4), ... )

The instruction DW displays in the window indicated by the value for expression 1, expression 3, .. once they have been evaluated, the numerical data indicated by expression 2, expression 4, ...

Expression 1, expression 2, expression 3, .... may contain a number or any expression which may result in a number.

The following example shows a dynamic variable display:

```
        (ODW 1, 6, 33)
            ; Defines data window 1
        (ODW 2, 14, 33)
            ; Defines data window 2
N10     (DW1=DATE, DW2=TIME)
            ; Displays the date in window 1 and the time in 2
        (GOTO N10)
```

The CNC allows displaying the data in decimal, hexadecimal and binary format; the following instructions are available:

(DW1 = 100)

Decimal format. Value "100" displayed in window 1.

(DWH2 = 100)

Hexadecimal format. Value "64" displayed in window 2.

(DWB3 = 100)

Binary format. Value "01100100" displayed in window 3.

When using the binary format, the display is limited to 8 digits in such a way that a value of "11111111" will be displayed for values greater than 255 and the value of "10000000" for values more negative than -127.

Besides, the CNC allows the number stored in one of the 26 data input variables (IB0-IB25) to be displayed in the requested window.

The following example shows a request and later displays the axis feedrate:

```
(ODW 3, 4, 60)
    ; Defines data window 3.
(IB1=INPUT "Axis feed: ", 5.4)
    ; Axis feedrate request.
(DW3=IB1)
    ; Displays feedrate in window 3.
```

### (SK (expression 1) = "text1" (expression 2) = "text 2", .... )

The instruction SK defines and displays the new softkey menu indicated.

Each of the expressions will indicate the softkey number which it is required to modify (1-7, starting from the left) and the texts which it is required to write in them.

Expression 1, expression 2, expression 3, .... may contain a number or any expression which may result in a number.

Each text will allow a maximum of 20 characters that will be shown in two lines of 10 characters each. If the text selected has less than 10 characters, the CNC will center it on the top line, but if it has more than 10 characters the programmer will center it.

Examples:

(SK 1="HELP", SK 2="MAXIMUN POINT")

| HELP | MAXIMUN POINT |
|------|---------------|

(SK 1="FEED", SK 2=" _ _MAXIMUN_ _ _POINT")

| FEED | MAXIMUN POINT |
|------|---------------|

⚠️ *If while a standard CNC softkey menu is active, one or more softkeys are selected via high level language instruction: "SK", the CNC will clear all existing softkeys and it will only show the selected ones.*

*If while a user softkey menu is active, one or more softkeys are selected via high level language instruction "SK", the CNC will only replace the selected softkeys leaving the others intact.*

### ( WKEY )

The mnemonic WKEY stops execution of the program until the key is pressed.

The pressed key will be recorded in the KEY variable.

```
...
(WKEY)                              ; Wait for key
(IF KEY EQ $FC00 GOTO N1000)        ; If key F1 has been pressed, continue in N1000
...
```

### ( WBUF "text", (expression) )

The WBUF instruction can only be used when editing a program in the user channel.

This instruction may be programmed in two ways:

- ( WBUF "text", (expression) )

  It adds the text and value of the expression, once it has been evaluated, to the block that is being edited and within the data entry window.

  (Expression) may contain a number or any expression resulting in a number.

  It will be optional to program the expression, but it will be required to define the text. If no text is required, "" must be programmed.

  Examples for P100=10:

```
(WBUF "X", P100)              =>      X10
(WBUF "X P100")               =>      X P100
```

**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

- ( WBUF )

Enters into memory, adding to the program being edited and after the cursor position, the block being edited by means of (WBUF "text", (expression)). It also clears the editing buffer in order to edit a new block.

This allows the user to edit a complete program without having to quit the user editing mode after each block and press ENTER to "enter" it into memory.

```
(WBUF "(PCALL 25, ")
    ; Adds "(PCALL 25, " to the block being edited.
(IB1=INPUT "Parameter A:",-5.4)
    ; Request of Parameter A.
(WBUF "A=", IB1)
    ; Adds "A=(value entered)" to the block being edited.
(IB2=INPUT "Parameter B: ", -5.4)
    ; Request of Parameter B.
(WBUF ", B=", IB2)
    ; Adds "B=(value entered)" to the block being edited.
(WBUF ")")
    ; Adds ")" to the block being edited.
(WBUF )
    ; Enters the edited block into memory.
...
```

After executing this program the block being edited contains:

(PCALL 25, A=23.5, B=-2.25)

### ( SYSTEM )

The mnemonic SYSTEM stops execution of the user customized program and returns to the corresponding standard menu of the CNC.

### Customizing program example:

The following customizing program must be selected as user program associated to the Editing Mode.

After selecting the Editing Mode and pressing the USER softkey, this program starts executing and it allows assisted editing of 2 user cycles. This editing process is carried out a cycle at a time and as often as desired.

**Displays the initial editing page (screen)**

```
N0    (PAGE 10 )
```

**Sets the softkeys to access the various modes and requests a choice**

```
      (SK 1="CYCLE 1",SK 2="CYCLE 2",SK 7="EXIT")
N5    (WKEY )                                    ; Request a key
      (IF KEY EQ $FC00 GOTO N10)                 ; Cycle 1
      (IF KEY EQ $FC01 GOTO N20)                 ; Cycle 2
      (IF KEY EQ $FC06 SYSTEM ELSE GOTO N5)      ; Quit or request a key
```

**CICLO 1**

```
        ; Displays page 11 and defines 2 data entry windows
N10     (PAGE 11)
        (ODW 1,10,60)
        (ODW 2,15,60)

        ;Editing
        (WBUF "( PCALL 1,")            ; Adds "(PCALL 1," to the block being edited.

        (IB 1=INPUT "X:",-6.5)         ; Requests the value of X.
        (DW 1=IB1)                     ; Data window 1 shows the entered value.
        (WBUF "X",IB1)                 ; Adds X (entered value) to the block being edited.

        (WBUF ",")                     ; Adds "," to the block being edited.

        (IB 2=INPUT "Y:",-6.5)         ; Requests the value of Y.
        (DW 2=IB2)                     ; Data window 2 shows the entered value.
        (WBUF "Y",IB2)                 ; Adds Y (entered value) to the block being edited.

        (WBUF ")")                     ; Adds ")" to the block being edited.
        (WBUF )                        ; Enters the edited block into memory.
                                       ; For example : (PCALL 1, X2, Y3)

        (GOTO N0)
```

**CICLO 2**

```
        ; Displays page 12 and defines 3 data entry windows
N20     (PAGE 12)
        (ODW 1,10,60)
        (ODW 2,13,60)
        (ODW 3,16,60)

        ; Editing
        (WBUF "( PCALL 2,")            ; Adds "(PCALL 2," to the block being edited.

        (IB 1=INPUT "A:",-6.5)         ; Requests the value of A.
        (DW 1=IB1)                     ; Data window 1 shows the entered value.
        (WBUF "A",IB1)                 ; Adds A (entered value) to the block being edited.

        (WBUF ",")                     ; Adds "," to the block being edited.

        (IB 2=INPUT "B:",-6.5)         ; Requests the value of B.
        (DW 2=IB2)                     ; Data window 2 shows the entered value.
        (WBUF "B",IB2)                 ; Adds B (entered value) to the block being edited.

        (WBUF ",")                     ; Adds "," to the block being edited.
        (IB 3=INPUT "C:",-6.5)         ; Requests the value of C.
        (DW 3=IB3)                     ; Data window 3 shows the entered value.
        (WBUF "C",IB3)                 ; Adds C (entered value) to the block being edited.

        (WBUF ")")                     ; Adds ")" to the block being edited.

        (WBUF )                        ; Enters the edited block into memory.
                                           For example: (PCALL 2, A3, B1, C3).

        (GOTO N0)
```

**14.**

**PROGRAM CONTROL INSTRUCTIONS**

Screen customizing instructions

**FAGOR**

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

# COORDINATE TRANSFORMATION

The description of the general coordinate transformation is divided into three basic functions:

- Movement in an inclined plane (G49).
- Tool movement according to the tool coordinate system (G47).
- TCP transformation, Tool Center Point (G48).

For a better understanding of coordinate transformation, three machine coordinate systems will be considered in the following examples.

- Machine coordinate system. X Y Z in the figures.
- Part coordinate system. X' Y' Z' in the figures.
- Tool coordinate system. X" Y" Z" in the figures.

When no transformation has been made and the spindle is in the starting position, the three coordinate systems coincide. Figure on the left.
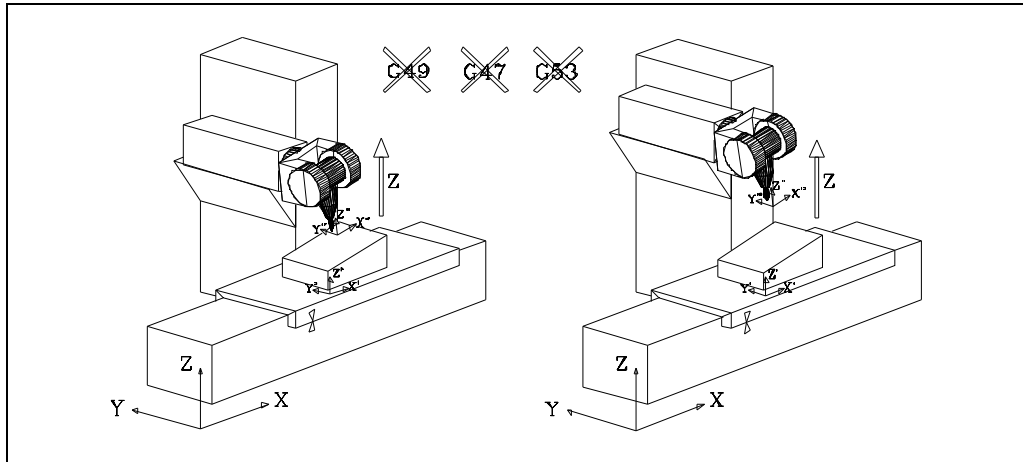
When turning the spindle, the tool coordinate system (X" Y" Z") changes. Figure on the right.

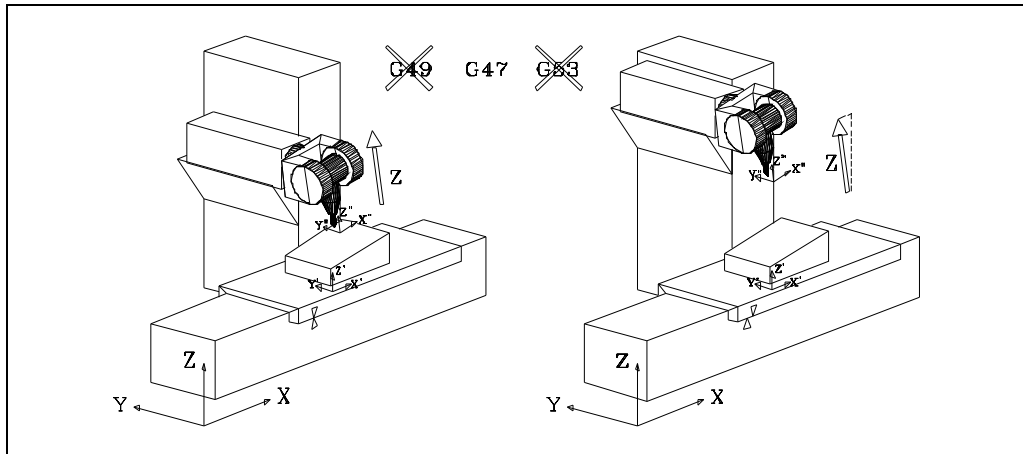If, also, an inclined plane is selected (G49), the part coordinate system also changes (X', Y', Z'). Bottom figure.
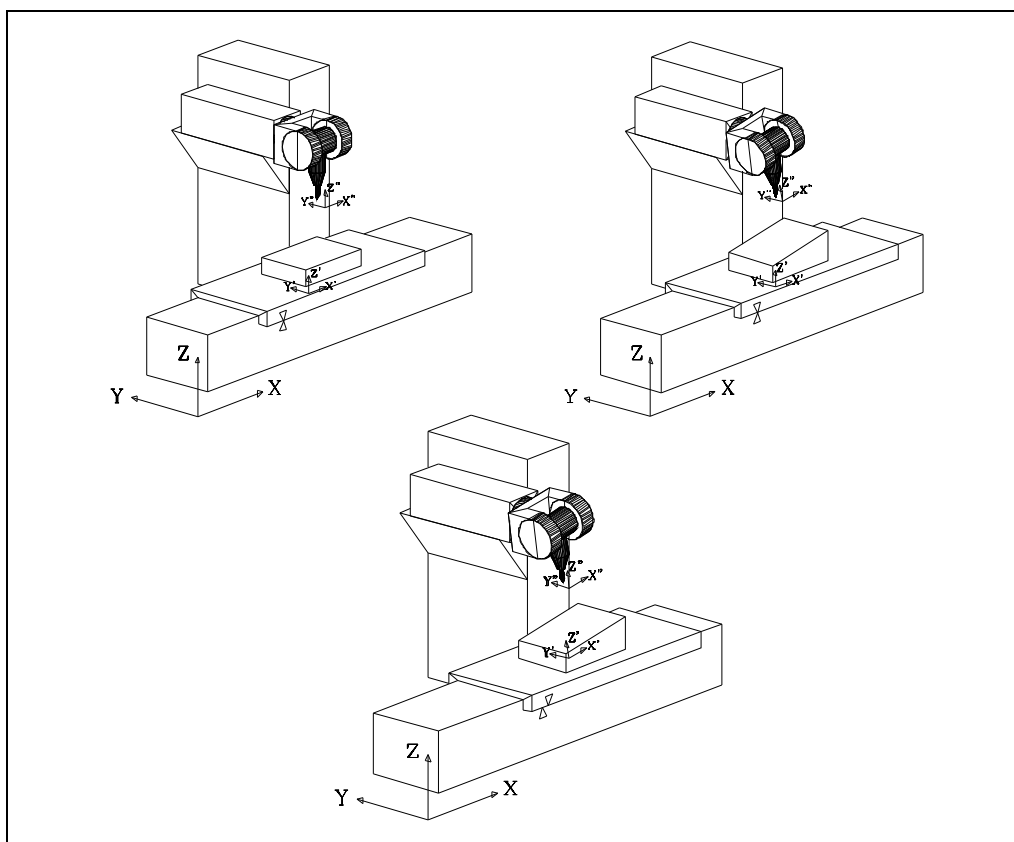
## Case –A–
## No transformation has been done and the spindle is spinning.

If a Z axis movement is programmed (G01 Z), this axis will move according to the part coordinate system which, in this case, coincides with machine coordinates.



Now, to move the tool according to the tool coordinate system, function G47 must be used when programming the movement of the Z axis (G01 G47 Z).



In this type of movements, when the tool coordinate system does not coincide with the machine coordinate system, the CNC moves several axes in order to move the tool according to the part coordinates. In the example, the X and Z axes move.

Function G47 is not modal and only affects the programmed movement.

In order for the jog movements, to be carried out according to the tool coordinate system, the CNC general logic input "TOOLMOVE (M5021" must be activated at the PLC.

## Case –B–
## An inclined plane has been selected (G49) and the spindle is perpendicular to it.

If a Z axis movement is programmed (G01 Z), this axis will move according to the part coordinate system.
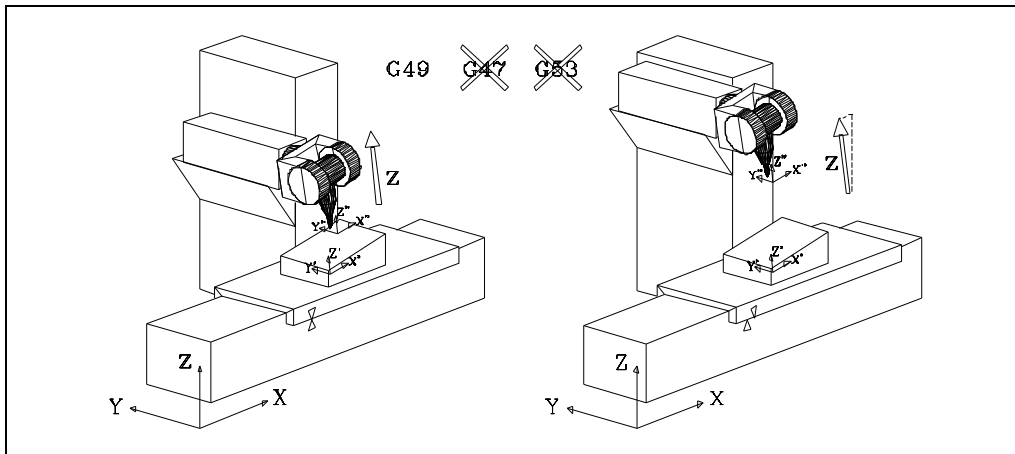


In this type of movements, when the part coordinate system does not coincide with the machine coordinate system, the CNC moves several axes in order to move the tool according to the part coordinates. In the example, the X and Z axes move.

To move the tool according to the machine coordinate system, function G53 (programming with respect to home) must be used when programming the movement of the Z axis (G01 G53 Z).



Function G53 is not modal and only affects the programmed movement.

In order for the jog movements to be carried out according to the machine coordinate system, CNC general logic input "MACHMOVE (M5012)" must be activated at the PLC.
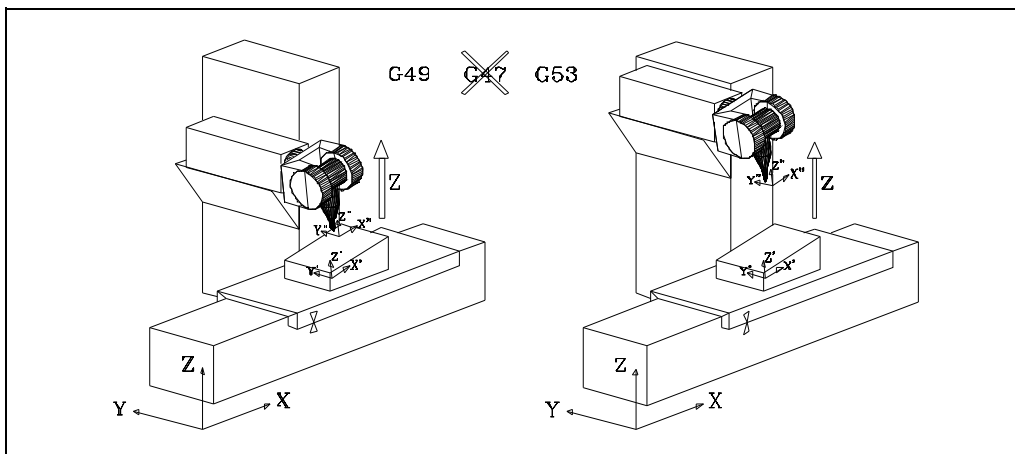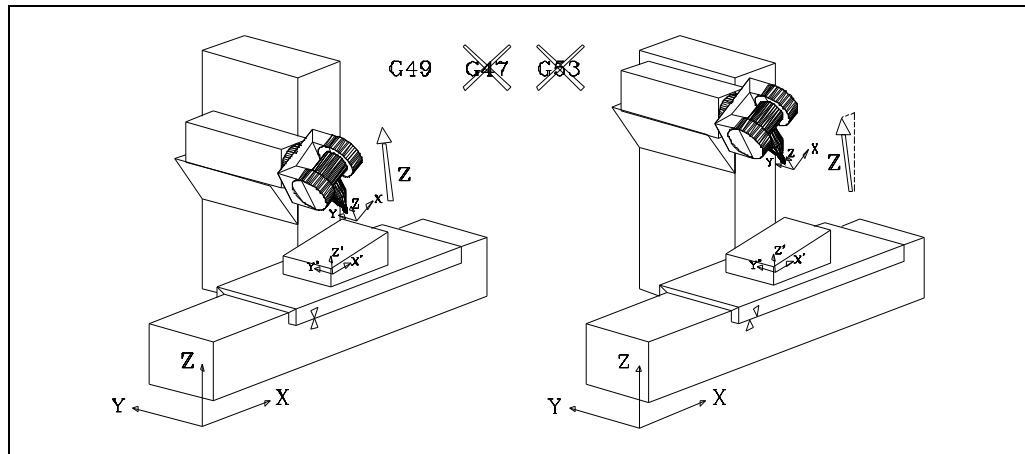
**15.**

## Case –C–
## An inclined plane has been selected (G49) and the spindle is not perpendicular to it.

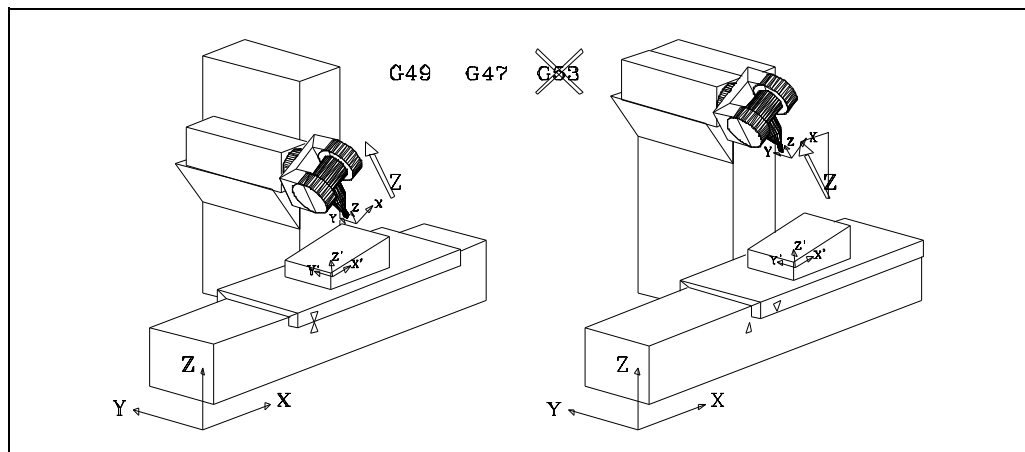If a Z axis movement is programmed (G01 Z), this axis will move according to the part coordinate system.



In this type of movements, when the part coordinate system does not coincide with the machine coordinate system, the CNC moves several axes in order to move the tool according to the part coordinates. In the example, the X and Z axes move.

Now, to move the tool according to the tool coordinate system, function G47 must be used when programming the movement of the Z axis (G01 G47 Z).



In this type of movements, when the tool coordinate system does not coincide with the machine coordinate system, the CNC moves several axes in order to move the tool according to the part coordinates. In the example, the X and Z axes move.

Function G47 is not modal and only affects the programmed movement.

In order for the jog movements, to be carried out according to the tool coordinate system, the CNC general logic input "TOOLMOVE (M5021" must be activated at the PLC.
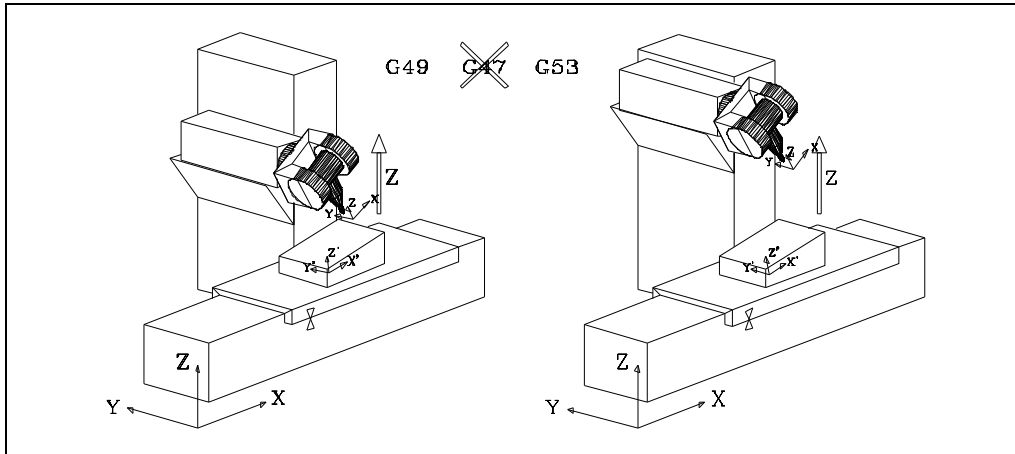
To move the tool according to the machine coordinate system, function G53 (programming with respect to home) must be used when programming the movement of the Z axis (G01 G53 Z).



Function G53 is not modal and only affects the programmed movement.

In order for the jog movements to be carried out according to the machine coordinate system, CNC general logic input "MACHMOVE (M5012)" must be activated at the PLC.

## Case –D–
## It is working with TCP, Tool Center Point

When working with TCP transformation, function G48 active, the CNC allows changing the tool orientation without changing the position of its tool tip (part coordinates).

Obviously, the CNC must move several axes in order to maintain the tool tip position at all times.
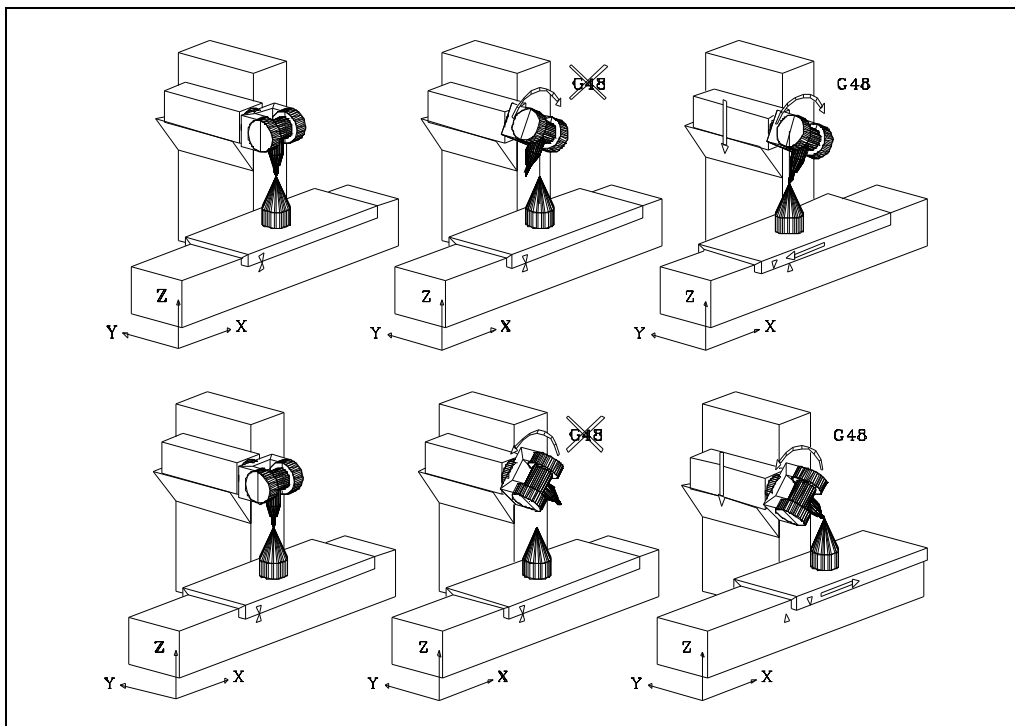


Function G48, as described later on, is modal and it indicates when the TCP transformation becomes active and when it is canceled.

Function G48, TCP transformation, may be used together with function G49 (movement in the inclined plane) and G47 (movement along the tool axes)

## 15.1 Movement in an inclined plane
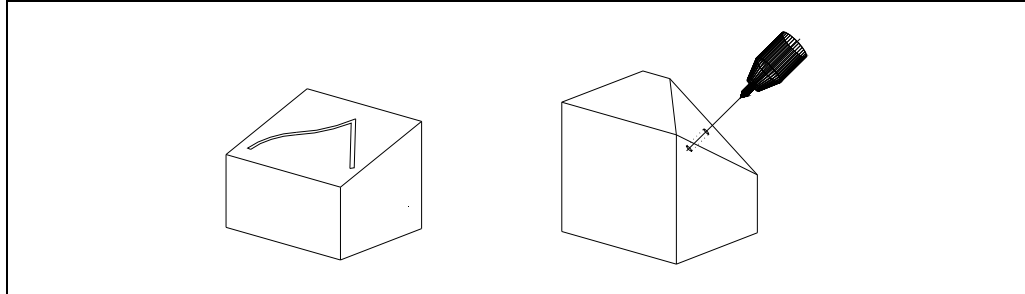
An inclined plane is any plane in space resulting from the coordinate transformation of the X, Y, Z axes.

Any plane in space may be selected to carry out machining operations in it.

The coordinates are programmed as if it were a regular XY plane, but the program will be executed in the indicated inclined plane.



To work with inclined planes, always proceed as follows:

1. Define, with G49, the inclined plane corresponding to the machining operation. G49 is described later on in this chapter.

2. The CNC variables TOOROF, TOOROS and parameters P297, P298 show the position to be occupied by the spindle rotary axes (main and secondary spindle respectively) in order to orient the tool perpendicular to the indicated inclined plane.

3. To work with the tool perpendicular to the inclined plane, rotate the spindle rotary axes to the indicated position.

From this moment on, the X, Y movements are carried out along the selected plane and those of the Z axis will be perpendicular to it.
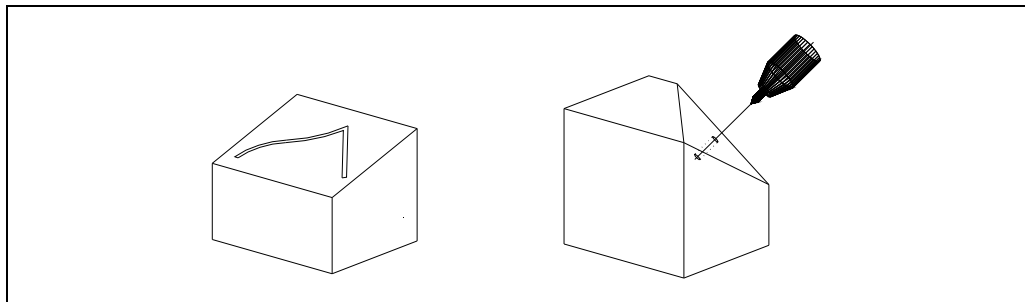
### 15.1.1 Definition of the inclined plane (G49)

With function G49, it is possible to define a coordinate transformation or, in other words, the inclined plane resulting from that transformation. There are several ways to define G49.

## G49 X Y Z A B C

Defines the inclined plane resulting from rotating around the X axis first and around the Z axis last the amounts indicated in A, B, C respectively.

### X Y Z

Define the coordinate origin of the inclined plane.

Indicate the X, Y, Z coordinates with respect to the current coordinate origin.
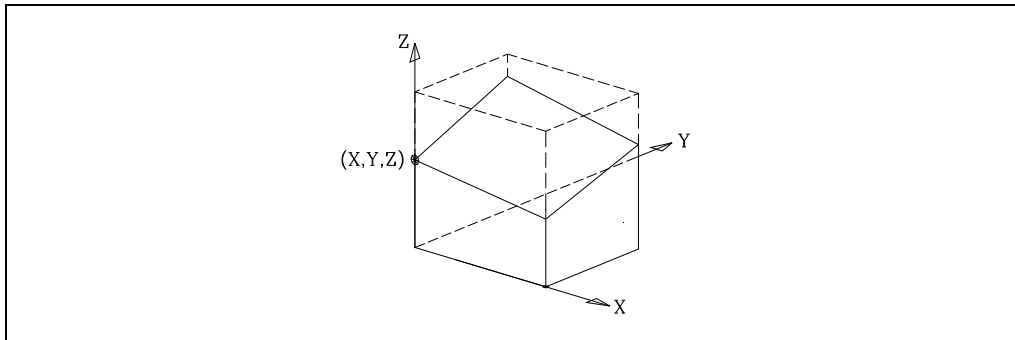


### A B C

They define the inclined plane resulting from:

Having rotated around the X axis first the amount indicated by A.



The new coordinate system resulting from this transformation is called X Y' Z' because the Y, Z axes have been rotated.

Then, it must be rotated around the Y' axis the B amount.
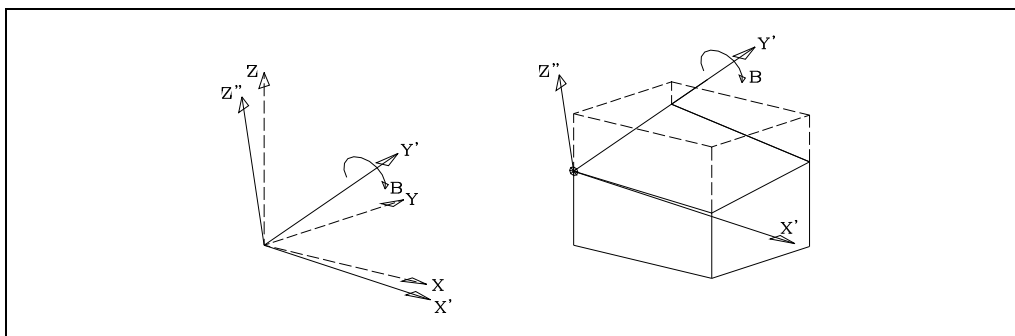


In the figure, the new coordinate system resulting from this transformation is called X'' Y' Z'' because the X, Z axes have been rotated.

And last, rotate around the Z'' axis the amount indicated by C.
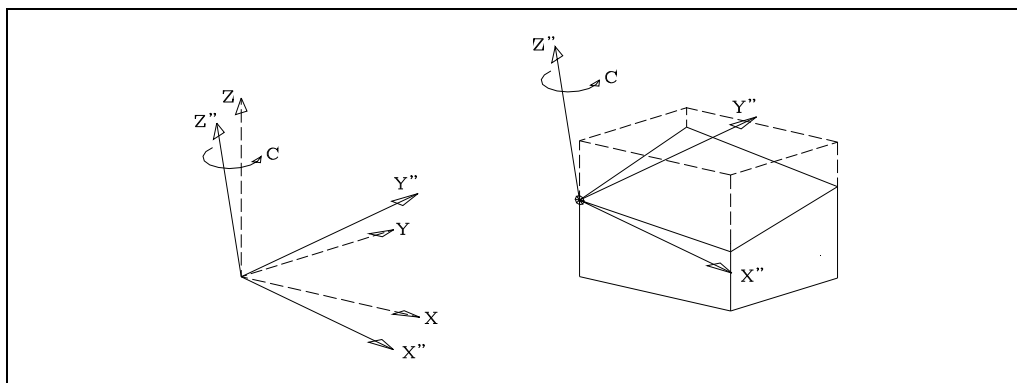
## G49 X Y Z Q R S

Spherical coordinates. Defines the inclined plane resulting from rotating around the Z axis first, then around the Y axis and again around the Z axis the amounts indicated by Q, R, S respectively.

### X Y Z

Define the coordinate origin of the inclined plane.

Indicate the X, Y, Z coordinates with respect to the current coordinate origin.
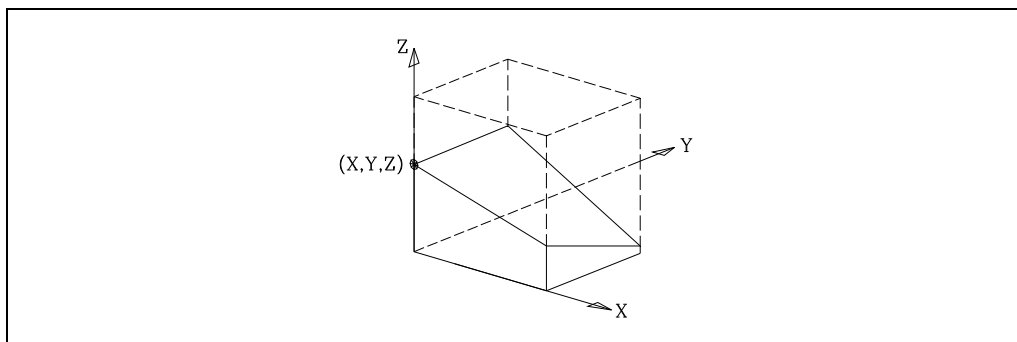


### Q R S

They define the inclined plane resulting from:

Having rotated around the Z axis first the amount indicated by Q.
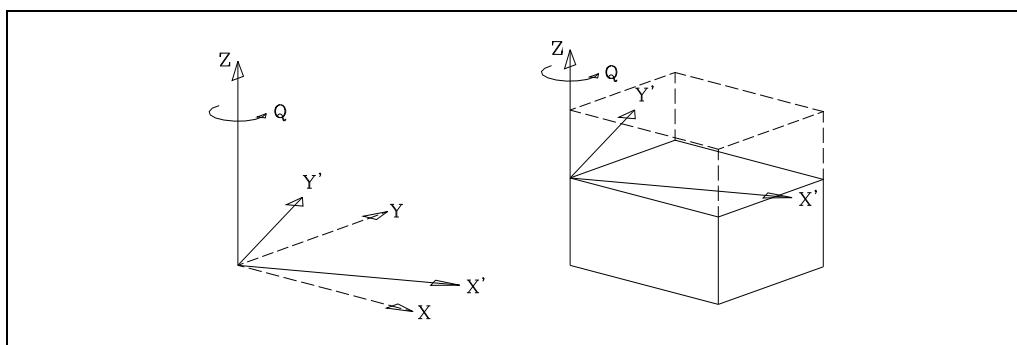


In the figure, the new coordinate system resulting from this transformation is called X' Y' Z because the X, Y axes have been rotated.

FAGOR

FAGOR AUTOMATION

CNC 8055
CNC 8055i

·M· & ·EN· MODELS
SOFT: V02.2X

·426·

Then, it must be rotated around the Y' axis the R amount.
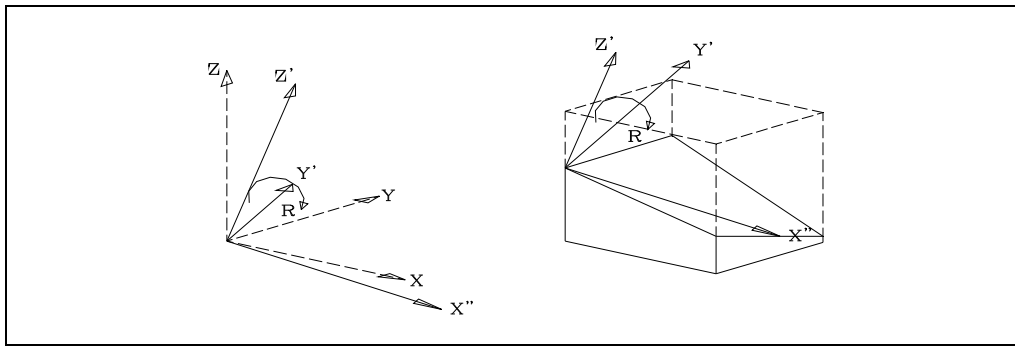


In the figure, the new coordinate system resulting from this transformation is called X'' Y' Z' because the X, Z axes have been rotated.

And last, rotate around the Z' axis the amount indicated by S.
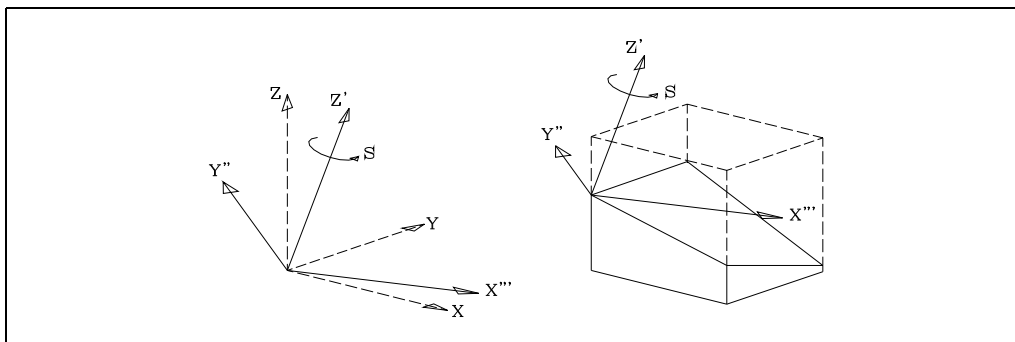


## G49 X Y Z I J K R S

Defines the inclined plane specifying the angles forming the new inclined plane with the X Y and Z axes of the machine's coordinate system.
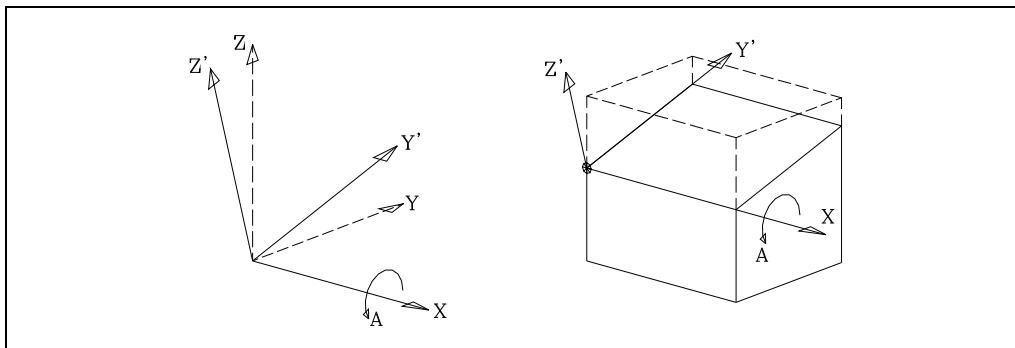
### X Y Z

Define the coordinate origin of the inclined plane.

Indicate the X, Y, Z coordinates with respect to the current coordinate origin.
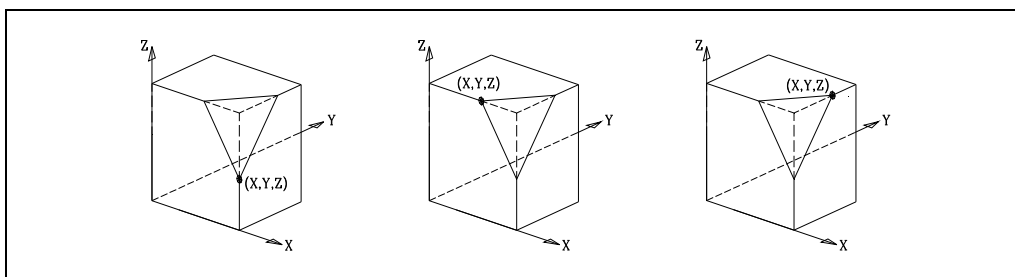


### I J K

They define the angles forming the new inclined plane with the X Y and Z axes of the machine's coordinate system. Only two of these three angles are programmed.

**FAGOR**

FAGOR AUTOMATION

CNC 8055
CNC 8055i

·M· & ·EN· MODELS
SOFT: V02.2x
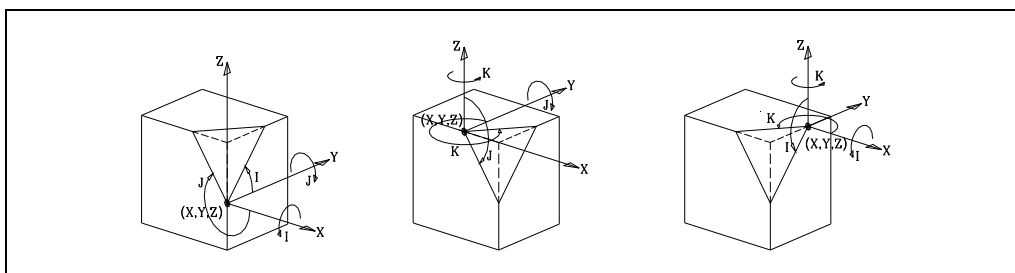
**R**

Defines which of the axes of the new Cartesian plane (X' Y' ) is aligned with the edge. If R0, the X' axis is lined up and If R1, the Y' axis is lined up. If not programmed, a value of R0 is assumed.

**S**
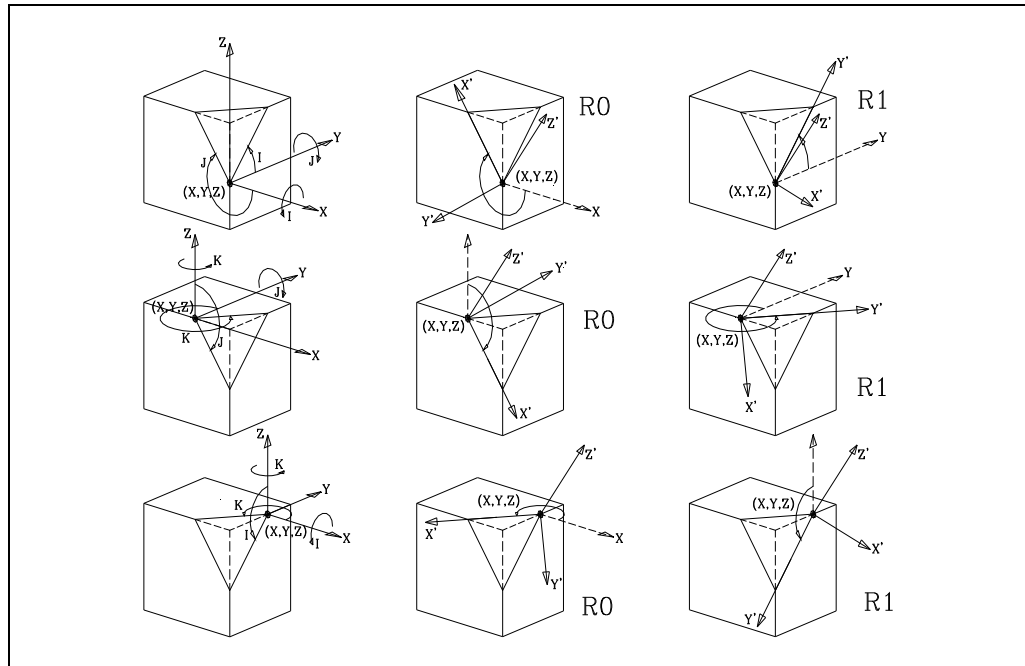
Lets rotate the coordinates in the new Cartesian plane.

## G49 T X Y Z S

It defines a new work plane perpendicular to the direction of the tool.

Is it a good idea to have a swivel or angled spindle (machine parameter "XFORM (P93) set to 2 or 3) when using this type of definition.

**T**

Indicates that one wishes to select a work plane perpendicular to the orientation of the tool.

**X Y Z**

Define the coordinate origin of the inclined plane.

Indicate the X, Y, Z coordinates with respect to the current coordinate origin.

**S**

Lets rotate the coordinates around the new Z' corresponding to the new work plane.
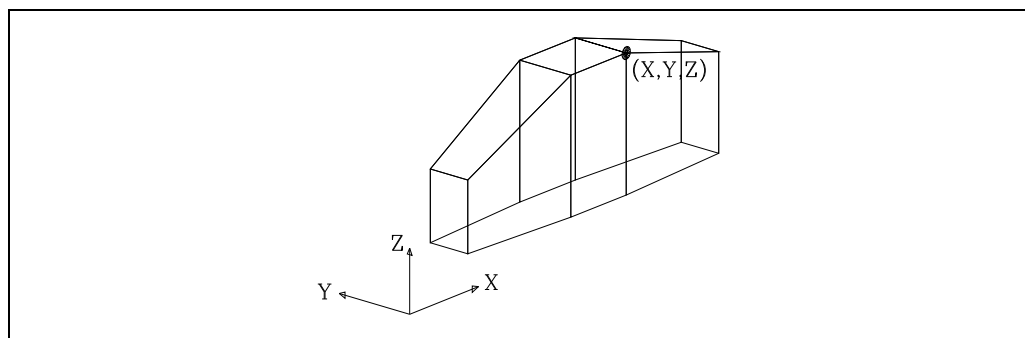
The new work plane will be perpendicular to the orientation of the tool.

The Z axis keeps the same orientation as the tool.

The orientation of the X, Y axes in the new work plane depends on the spindle type and on how its rotary axes are oriented.

When setting the machine up, it must be set as the spindle's resting position, when the tool is parallel to the Z axis of the machine.

Later, every time the spindle is rotated, the relative tool coordinates will also rotate.
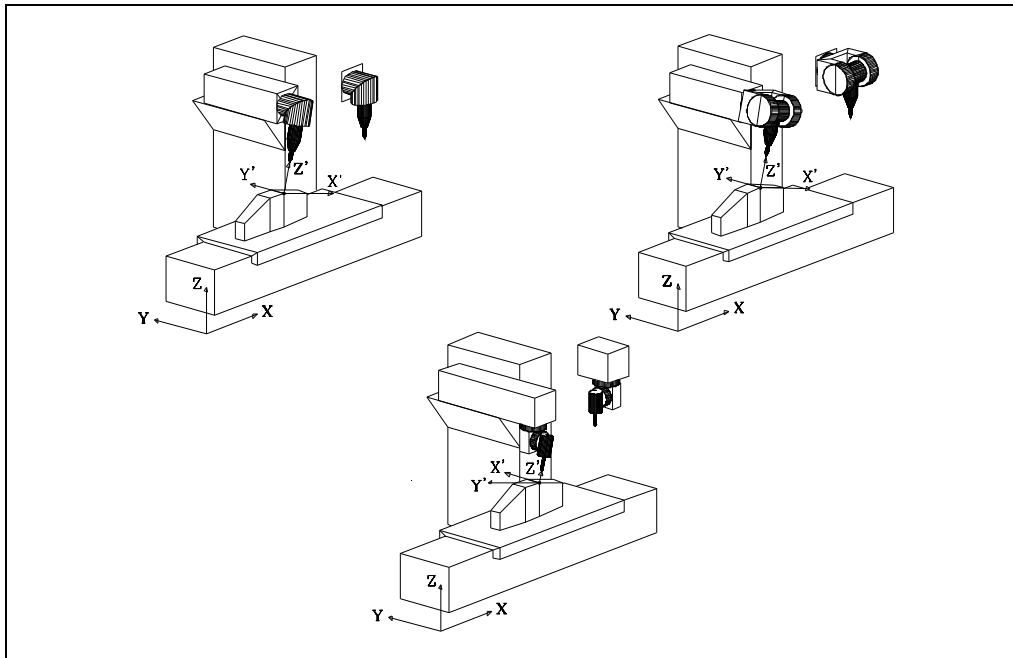


On the two machines on the left, only the main rotary axis has rotated.

On the contrary, in the machine on the right, to achieve the same tool orientation, both the main and secondary rotary axes have rotated.

If in the machine on the right, we wanted to orient the X', Y' axes like in the other two cases, we would have to program the following:

```
G49 T XYZ S-90
```

Programming S-90 means rotating -90º around the new Z' corresponding to the new work plane and, this way, compensate for the rotation of the main rotary axis.

### 15.1.2   G49 in swinging spindles

For the function G49, inclined plane definition, the W parameter W has been added. This indicates that it is an oscillating spindle and that it must be defined at the end: G49 ****** W.

G49 X Y Z A B C W          G49 X Y Z Q R S W

G49 X Y Z I J K R S W      G49 T X Y Z S W

The spindle orients to the new plane and later movements are carried out in X, Y, W.

To observe the graphics in the new plane, set general machine parameter GRAPHICS (P16)=1.

There are two choices to execute canned cycles in the inclined plane:

- After defining the plane (G49 **** W) select the W axis as longitudinal (G15W) so the movements programmed on Z are carried out on the W axis.

- Switch the Z-W axes (G28 ZW) before defining the plane (G49 **** W) so the movements programmed in Z are carried out on the W axis.

To cancel the inclined plane, program G49 alone.

### 15.1.3 G49 on Huron type spindles

When defining a new inclined plane, the CNC provides the position that each rotary axis must occupy to orient the tool perpendicular to the new plane.

That position is indicated in variables TOOROF and TOOROS and arithmetic parameters P297 and P298.

Since there are two possible solutions on Huron type spindles (45º spindles), the L parameter has been added to function G49, definition of inclined plane, that indicates which one of these two solution is to be used. It is optional and it is defined at the end. G49 ****** L.

    G49 X Y Z A B C L        G49 X Y Z Q R S L

    G49 X Y Z I J K R S L     G49 T X Y Z S L

If "L" is not defined or it is defined as "L0", it provides the solution where the main rotation (articulation closest to the ram or furthest away from the tool) is closer to 0º

If "L1" is defined, it provides the other solution (the articulation furthest away from the ram).

If "L" is programmed on the rest of the spindles, it issues the error "Option not available".

### 15.1.4    Considerations about the G49 function

G49 cannot be programmed in the following instances:

- At the GP model CNC
- From the PLC channel (although it can be programmed from the user channel).
- Within a profile definition for pockets or other cycles.

In order to work with coordinate transformation (G49), the X, Y, Z axes must be defined, they must form the active trihedron and be linear. The X, Y and Z axes may have GANTRY axes, coupled or synchronized via PLC, associated with them.

When working with coordinate transformation and performing rigid tapping in inclined planes, all axes gains (not only for the Z axis) must be adjusted by using the second gains and accelerations.

The parameters associated with G49 are optional. When programming G49 without parameters, the active coordinate transformation is canceled.

G49 is modal and no more G codes may be programmed alone in the block.

Coordinate transformation is kept active even after turning the CNC off and back on.

To cancel it, program the following:

| | |
|---|---|
| G49 E1 | it cancels it, but it keeps the part zero defined in the transformation. |
| G49 E0 ó G49 | It cancels it and it recovers the part zero that was active before the G49 was activated. |
| G74 | It cancels it, it activates the home search and recovers the part zero that was active before the G49 was activated. |

Zero offsets G54-G59, pattern rotation (G73) and presets (G92, G93) are possible while coordinate transformation is active.

But the following cannot be done:

- Program a new coordinate transformation without previously canceling the previous one.
- Probing (G75).
- Movement against hardstop (G52).

**15.**

**COORDINATE TRANSFORMATION**
Movement in an inclined plane

**FAGOR**

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

·432·

### 15.1.5 Variables associated with function G49

**Read-only variables associated with the definition of function G49**

**ORGROX**          **ORGROY**          **ORGROZ**

New part zero coordinates with respect to home.

**ORGROA**          **ORGROB**          **ORGROC**

Values assigned to parameters A, B, C.

**ORGROI**          **ORGROJ**          **ORGROK**

Values assigned to parameters I, J, K.

**ORGROQ**          **ORGROR**          **ORGROS**

Values assigned to parameters Q, R, S.

**GTRATY**

Type of G49 programmed.

0 = no G49 has been defined.
1 = Type G49 X Y Z A B C
2 = Type G49 X Y Z Q R S
3 = Type G49 T X Y Z S
4 = Type G49 X Y Z I J K R S

Every time G49 is programmed, the CNC updates the values of the parameters that have been defined.

For example when programming G49 XYZ ABC, the CNC updates the variables ORGROX, ORGROY, ORGROZ, ORGROA, ORGROB, ORGROC. The rest of variables keep their previous values.

**Read-write variables updated by the CNC once function G49 is executed.**

When having a swivel or angular spindle, general machine parameter XFORM (P93) with a value of 2 or 3, the CNC shows the following data:

**TOOROF**

It indicates the position to be occupied by the spindle's main rotary axis in order to orient the spindle perpendicular to the indicated inclined plane.

**TOOROS**

It indicates the position to be occupied by the spindle's secondary rotary axis in order to orient the spindle perpendicular to the indicated inclined plane.

Accessing the variables TOOROF or TOOROS interrupts block preparation and the CNC waits for that command to be executed before resuming block preparation.

**15.**

**COORDINATE TRANSFORMATION**
Movement in an inclined plane

FAGOR

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2x

### 15.1.6   Parameters associated with function G49

Once G49 has been executed, the CNC updates global parameters P297 and P298:

P297   It indicates the position to be occupied by the spindle's main rotary axis in order to orient the spindle perpendicular to the indicated inclined plane. It is the same value as shown by the TOOROF variable.

P298   It indicates the position to be occupied by the spindle's secondary rotary axis in order to orient the spindle perpendicular to the indicated inclined plane. It is the same value as shown by the TOOROS variable.

These parameters are global. Therefore, they can be modified by the user or even by probing cycles of the CNC itself.

They should be used after executing G49. Otherwise, variables TOOROF and TOOROS should be used.

### 15.1.7 Programming example
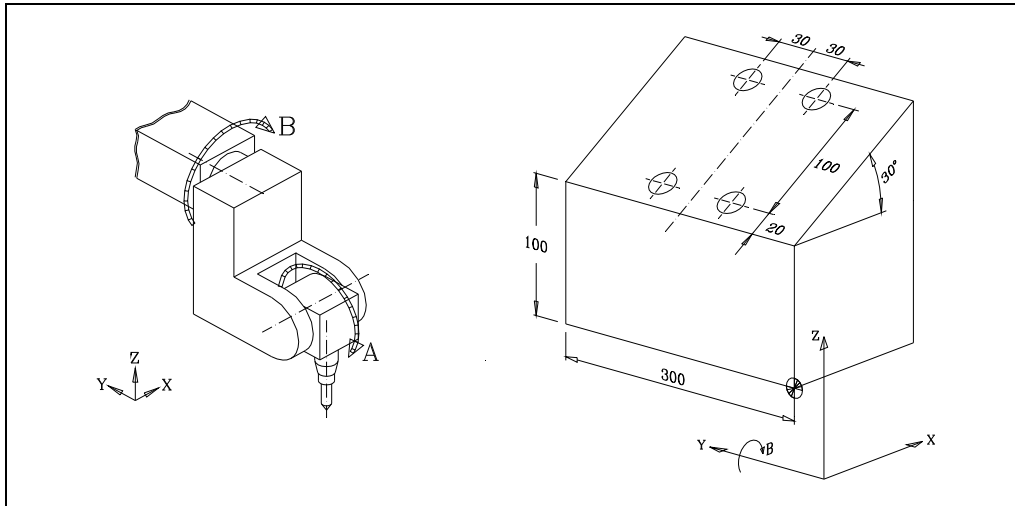


| | |
|---|---|
| G49 X0 Y0 Z100 B-30 | Defines the inclined plane. |
| G01 AP298 BP297 | Orients the main axis (B) and the secondary axis (A) so the tool is perpendicular to the plane. |
| | The programming sequence is ABC, regardless of which one is the main axis or the secondary. |
| G90 G01 Z5 | Tool approach to the work plane. |
| G90 G01 X20 Y120 | Positioning at the 1st point. |
| G?? | Machining at the 1st point. |
| G91 G01 Y60 | Positioning at the 2nd point. |
| G?? | Machining at the 2nd point. |
| G91 G01 X100 | Positioning at the 3rd point. |
| G?? | Machining at the 3rd point. |
| G91 G01 Y-60 | Positioning at the last point. |
| G?? | Machining at the last point. |
| G90 G01 Z 20 | Withdrawal the tool. |
| G49 | Cancel the inclined plane. |

## 15.2    Movement according to the tool coordinate system (G47)

When using this function, a swivel or angled spindle should be utilized (general machine parameter "XFORM (P93)" set to 2 or 3).

When not using function G47, the tool moves according to the part coordinate system
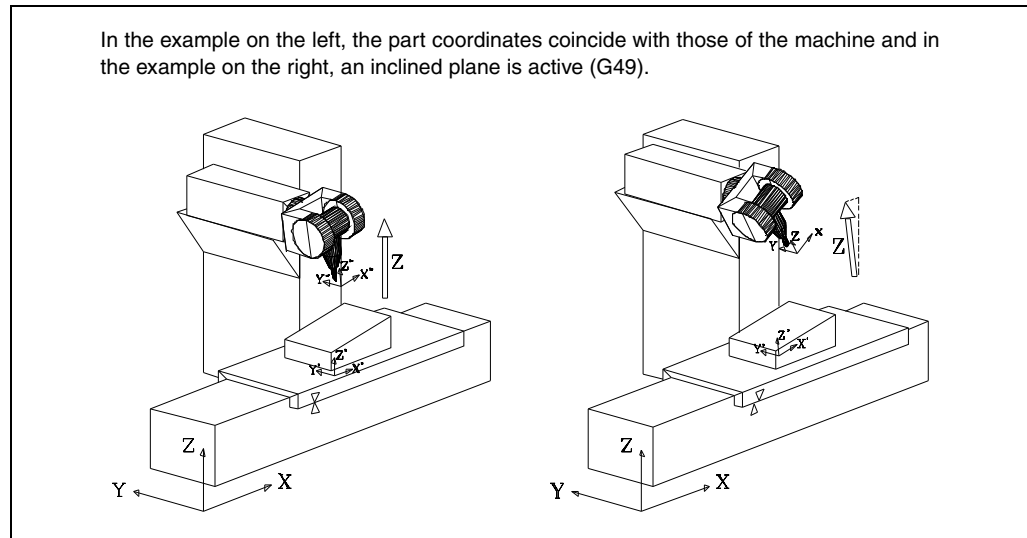


In the example on the left, the part coordinates coincide with those of the machine and in the example on the right, an inclined plane is active (G49).

Now, to move the tool according to the tool coordinate system, function G47 must be used when programming the movement of the Z axis (G01 G47 Z).
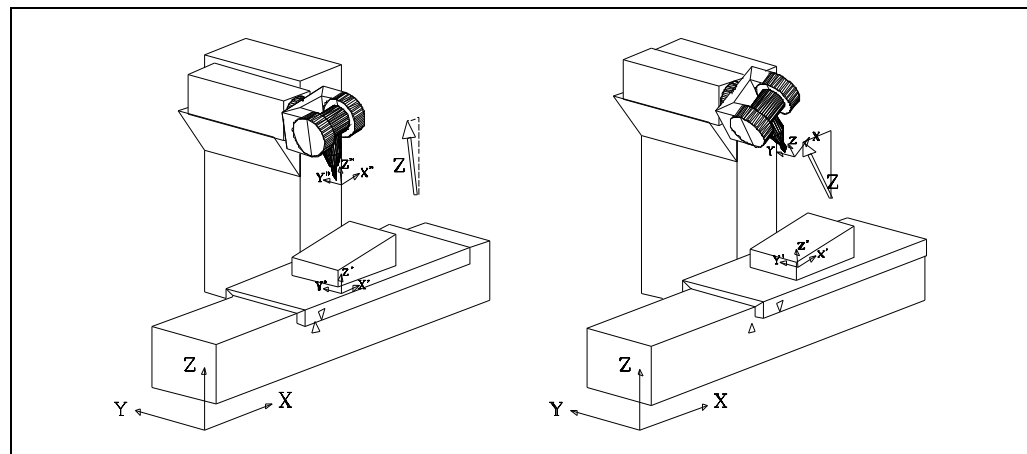


The movements programmed with G47 are always incremental.

Function G47 is not modal and it only acts within the block (linear path) where it has been programmed.

G47 can also be programmed while G48 and G49 are active.

## 15.3 TCP Transformation (G48)

In order to use this feature, the spindle articulations must have encoders and they must be controlled by the CNC.

When working with TCP transformation, Tool Center Point, the tool orientation may be modified without changing the position of its tip (part coordinates).

Obviously, the spindle must be swivel or angled and general machine parameter "XFORM (P93)" set to a value other than "0".

To orient the tool without changing its tip position, the CNC must move several axes of the machine.
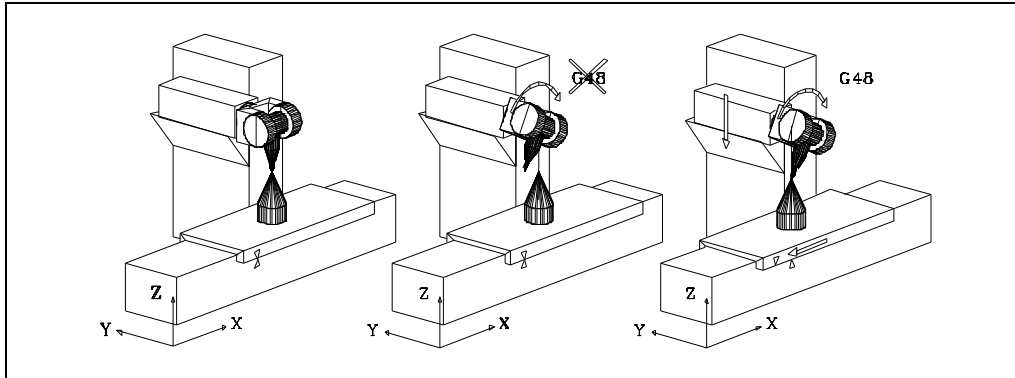


TCP transformation is activated and deactivated by function G48:

    G48 S1         Turn TCP transformation on

    G48 S0         Turn TCP transformation off

TCP transformation is also turned off by programming G48 without parameters.

Function G48 is modal and it must be programmed alone in the block.

Once TCP transformation is active, spindle positioning may be combined with linear and circular interpolations.

To orient the spindle, one must program the target angular position for the main rotary axis and for the secondary axis of the spindle.

The following examples use a double swivel rectangular spindle head:
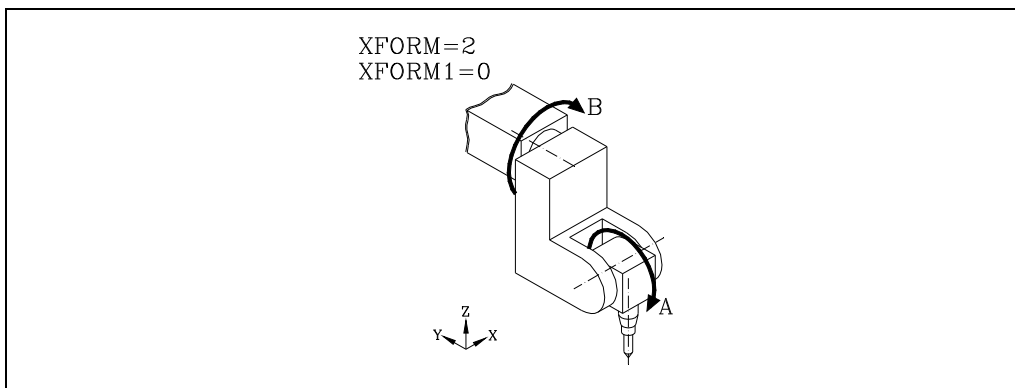
### Example –A–
### Circular interpolation keeping tool orientation fixed.



```
N20 G18 G90 G01 X30 Z90
N21 G48 S1
N22 G01 X100 Z20 B-60
N23 G03 X170 Z90 I70 K0
N23 G01 X170 Z120 B0
N25 G48 S0
```
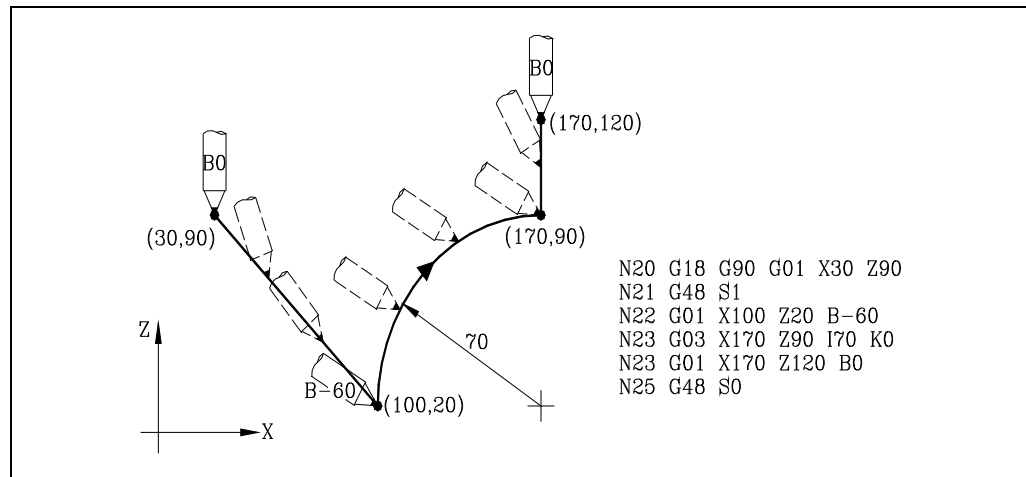
- Block N20 selects the ZX plane (G18) and positions the tool at the starting point (30,90).

- Block N21 turns TCP transformation on.

- Block N22 positions the tool at (100,20) orienting it to -60º.

  The CNC interpolates the XZB axes executing the programmed linear interpolation while rotating the tool from the starting position (0º) to the programmed final orient position (-60º).

- Block N23 makes a circular interpolation to point (170,90) maintaining the same tool orientation along the whole path.

- Block N24 positions the tool at (170,120) orienting it to 0º.

  The CNC interpolates the XZB axes executing the programmed linear interpolation while rotating the tool from the current position (-60º) to the programmed final orient position (0º).

- Block N25 turns TCP transformation off.

### Example –B–
### Circular interpolation with the tool perpendicular to the path.



```
N30 G18 G90 G01 X30 Z90
N31 G48 S1
N32 G01 X100 Z20 B-90
N33 G03 X170 Z90 I70 K0 B0
N34 G01 X170 Z120
N35 G48 S0
```
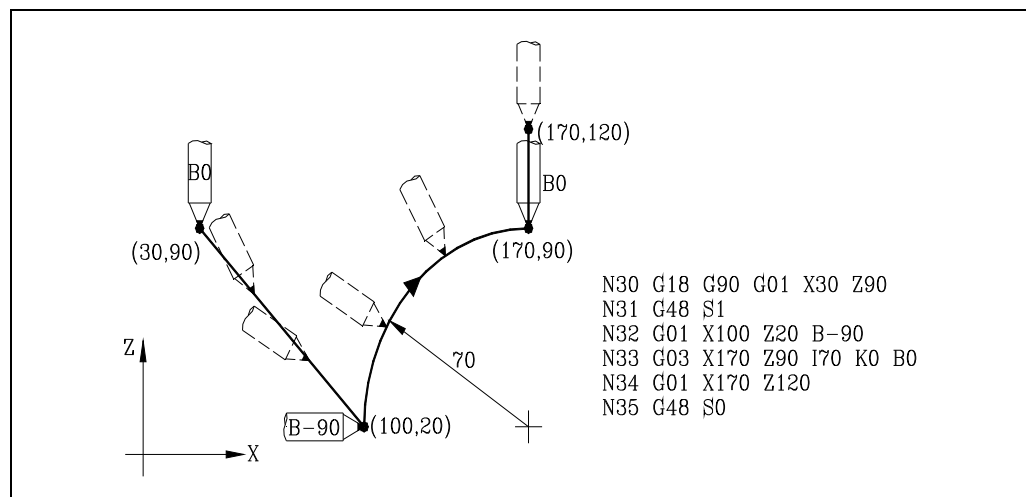
- Block N30 selects the ZX plane (G18) and positions the tool at the starting point (30,90).

- Block N31 turns TCP transformation on.

- Block N32 positions the tool at (100,20) orienting it to -90º.

  The CNC interpolates the XZB axes executing the programmed linear interpolation while rotating the tool from the starting position (0º) to the programmed final orient position (-90º).

- Block N33 defines a circular interpolation up to point (170,90) setting the final tool orientation to (0º).

  The CNC interpolates the XZB axes executing the programmed circular interpolation while rotating the tool from the current position (-90º) to the programmed final orient position (0º).

  Since both orientations are radial, the tool stays radially oriented at all times. In other words, perpendicular to the path.

- Block N34 positions the tool at (170,120).

- Block N35 cancels the TCP transformation..

### Example –C–
### Machining of a profile
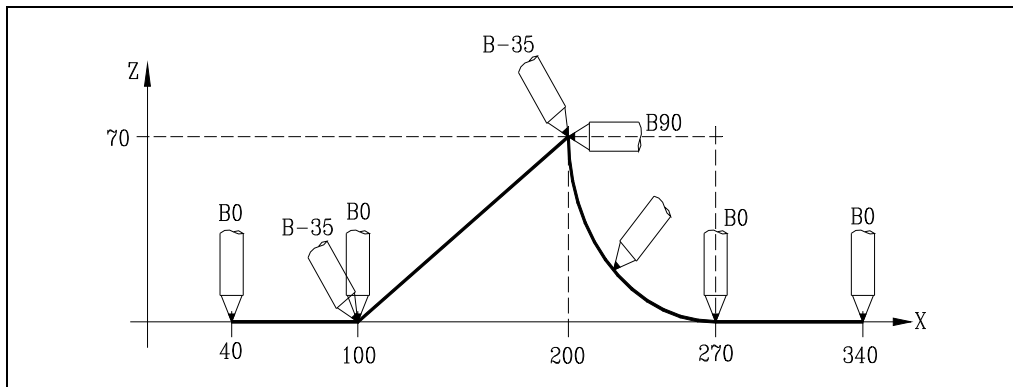


| G18 G90 | Selects the ZX plane (G18) |
|---|---|
| G48 S1 | It activates the TCP transformation. |
| G01 X40 Z0 B0 | Positions the tool at (40,0) oriented to (0º) |
| X100 | Movement to (100,0) with tool oriented to (0º) |
| B-35 | Orients the tool to (-35º) |
| X200 Z70 | Movement to (200,70) with tool oriented to (-35º) |
| B90 | Orients the tool to (90º) |
| G02 X270 Z0 R70 B0 | Circular interpolation to (270,0) maintaining the tool perpendicular to the path. |
| G01 X340 | Movement to (340,0) with tool oriented to (0º) |
| G48 S0 | It cancels the TCP transformation. |

### 15.3.1 Considerations about the G48 function

G49 cannot be programmed in the following instances:

- At the GP model CNC
- From the PLC channel (although it can be programmed from the user channel).

In order to work with TCP transformation (G48), the X, Y, Z axes must be defined, they must form the acive trihedron and be linear. The X, Y and Z axes may have GANTRY axes, coupled or synchronized via PLC, associated with them.

When working with TCP transformation and performing rigid tapping in inclined planes, all axes gains (not only for the Z axis) must be adjusted by using the second gains and accelerations.

TCP transformation is kept active even after turning the CNC off and back on.

G48 can be programmed while G49 is active and vice versa.

To cancel TCP, program "G48 S0" or G48 without parameters. It is also canceled after a home search (G74).

While TCP is on, the following operations are possible:

- Zero offsets G54-G59
- Pattern rotation (G73)
- Presets (G92, G93).
- Movements in continuous / incremental jog and handwheel

But the following cannot be done:

- Probing (G75).
- Do corner rounding or chamfering because in these instances tool orientation has to be maintained.
- Compensate for tool length (G43) because TCP already implies a particular length compensation. CAD/CAM programs usually program the coordinates of the spindle base.

Special care must be taken when turning G48 on and off.

- When G48 is on, the CNC shows the coordinates of the tool tip.
- When G48 is off, the CNC shows the coordinates of the tool base or theoretical tip (unturned tool).
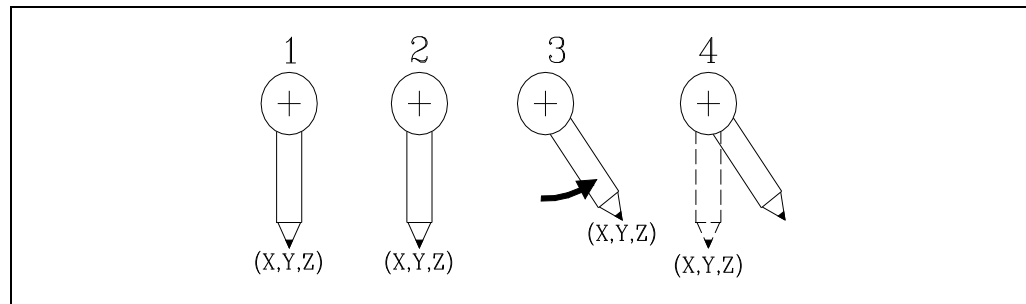


**1.** G48 off. The CNC shows the coordinates of the tool tip.

**2.** G48 is turned on. The CNC still shows the coordinates of the tool tip.

**3.** The tool is turned. Since G48 is already on, the CNC still shows the coordinates of the tool tip.

**4.** G48 is turned off. The CNC shows the coordinates of the theoretical tip (unturned tool).

**15.**

**COORDINATE TRANSFORMATION**
TCP Transformation (G48)

**FAGOR**

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

·440·

When working with inclined planes and TCP transformation, it is recommended to follow this programming order (sequence):

| | |
|---|---|
| G48 S1 | Turn TCP transformation on. |
| G49 ... | Define the inclined plane. |
| G01 AP298 BP297 | Position the tool perpendicular to the plane. |
| G... | Start machining. |
| ... | End of the machining operation. |
| G49 | Cancel the inclined plane. |
| G48 S0 | Turn TCP transformation off. |
| M30 | End of part program. |

RTCP should be turned on first because it allows orienting the tool without modifying the tool tip position, thus avoiding undesired collisions.

**15.**

**COORDINATE TRANSFORMATION**

TCP Transformation (G48)

**15.**

FAGOR

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

# ANGULAR TRANSFORMATION OF AN INCLINE AXIS

With the angular transformation of an incline axis, it is possible to make movements along an axis that is not perpendicular to another. The movements are programmed in the Cartesian system and to make the movements, they are transformed into movements on the real axes.

On certain machines, the axes are configured in a Cartesian way, they are not perpendicular to each other. A typical case is the X axis of a lathe that for sturdiness reasons is not perpendicular to the Z axis.



| | |
|---|---|
| X | Cartesian axis. |
| X' | Angular axis. |
| Z | Orthogonal axis. |
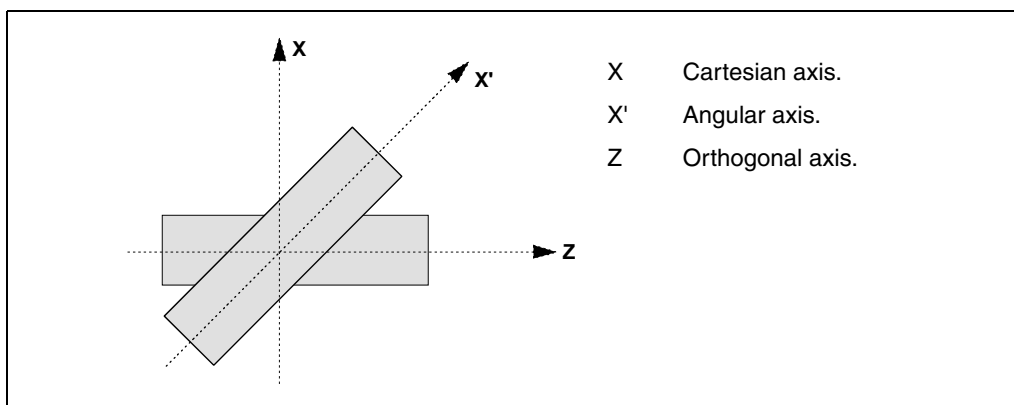
Programming in the Cartesian system (Z-X) requires activating an angular transformation of an inclined plane that converts the movements of the real (non-perpendicular) axes (Z-X'). This way, a movement programmed on the X axis is transformed into movements on the Z-X' axes; i.e. it then moves along the Z axis and the angular X' axis.

### Turning angular transformation on and off.

The CNC assumes no transformation on power-up; the angular transformations are activated via part-program using the instruction G46.

The angular transformations are turned off via part-program using function G46. Optionally, a transformation may be "frozen" (suspended) to move angular axis by programming in Cartesian coordinates.

### Influence of the reset, turning the CNC off and of the M30.

The angular transformation of an incline axis stays active after a RESET, M30 and even after turning the CNC off and back on.

*Considerations for the angular transformation of an incline axis.*

The axes involved in an angular transformation must be linear. Both axes may have Gantry axes associated with them, the may be slaved (coupled) or synchronized by PLC.

If the angular transformation is active, the coordinates displayed will be those of the Cartesian system. Otherwise, it will display the coordinates of the real axes.

The following operations are possible while the transformation is active:

- Zero offsets.
- Coordinate preset.
- Movements in continuous / incremental jog and handwheels.

The following operations are not possible while the transformation is active:

- Movement until making contact (against a hard stop).
- Coordinate rotation.
- Surface feedrate in milling.

**Machine reference zero (home) search.**

Function G46 is canceled when homing an axis that is involved in the angular transformation (machine parameters ANGAXNA and ORTAXNA). When homing the axes that are not involved in the angular axis transformation, function G46 stays active.

While searching home, only the real axes move.

*Jogging and handwheel movements.*

Either the real or the Cartesian axes may be jogged depending on how they've been set by the manufacturer. It is selected via PLC (MACHMOVE) and it may be available, for example, from a user key.

## 16.1 Turning angular transformation on and off

### Turn angular transformation on

When the transformation is on, the movements are programmed in the Cartesian system and to make the movements, the CNC transforms them into movements on the real axes. The coordinates displayed on the screen will be those of the Cartesian system.

The angular transformation is turned on using function G46 whose programming format is:

G46 S1

This instruction turns a "frozen" (suspended) transformation on again. See *"16.2 Freezing the angular transformation"* on page 446.

### Turning the angular transformation off

If the transformation is off, the movements are programmed and executed in the system of the real axes. The coordinates displayed on the screen will be those of the real axes.

The angular transformation is turned off using function G46 whose programming format is:

G46 S0
G46

The angular transformation of an incline axis stays active after a RESET, M30 and even after turning the CNC off and back on.

**FAGOR** ⬤

FAGOR AUTOMATION

CNC 8055
CNC 8055i

·M· & ·EN· MODELS
SOFT: V02.2x

## 16.2    Freezing the angular transformation

Freezing the angular transformation is a special way to make movements along the angular axis, but programming it in the Cartesian system. The angular transformation cannot be "frozen" (suspended) while jogging.

The angular transformation is "frozen" (suspended) using function G46 whose programming format is:

G46 S2

### *Programming movements after "freezing" the angular transformation.*

If an angular transformation is "frozen" (suspended), only the coordinate of the angular axis must be programmed in the motion block. If the coordinate of the orthogonal axis is programmed, the movement is carried out according to the normal angular transformation.

### *Canceling the freezing of a transformation.*

The "freezing" of an angular transformation is canceled after a reset or an M30. Turning the transformation on (G46 S1) also cancels the "freezing".

**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

# APPENDIX

**FAGOR**

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

SOFT: V02.2x

# ISO CODE PROGRAMMING

| Function | M | D | V | Meaning | Section |
|----------|---|---|---|---------|---------|
| G00 | * | ? | * | Rapid traverse | 6.1 |
| G01 | * | ? | * | Linear interpolation | 6.2 |
| G02 | * | | * | Clockwise circular (helical) interpolation | 6.3 / 6.7 |
| G03 | * | | * | Counterclockwise circular (helical) interpolation | 6.3 / 6.7 |
| G04 | | | | Dwell/interruption of block preparation | 7.1 / 7.2 |
| G05 | * | ? | * | Round corner | 7.3.2 |
| G06 | | | * | Circle center in absolute coordinates | 6.4 |
| G07 | * | ? | | Square corner | 7.3.1 |
| G08 | | | * | Arc tangent to previous path | 6.5 |
| G09 | | | * | Arc defined by three points | 6.6 |
| G10 | * | * | | Mirror image cancellation | 7.5 |
| G11 | * | | * | Mirror image on X axis | 7.5 |
| G12 | * | | * | Mirror image on Y axis | 7.5 |
| G13 | * | | * | Mirror image on Z axis | 7.5 |
| G14 | * | | * | Mirror image in the programmed directions | 7.5 |
| G15 | * | | * | Longitudinal axis selection | 8.2 |
| G16 | * | | * | Main plane selection by two addresses and longitudinal axis | 3.2 |
| G17 | * | ? | * | Main plane X-Y and longitudinal Z | 3.2 |
| G18 | * | ? | * | Main plane Z-X and longitudinal Y | 3.2 |
| G19 | * | | * | Main plane Y-Z and longitudinal X | 3.2 |
| G20 | | | | Definition of lower work zone limits | 3.7.1 |
| G21 | | | | Definition of upper work zone limits. | 3.7.1 |
| G22 | | | * | Enable/disable work zones. | 3.7.2 |
| G28 | * | | * | Second spindle selection | 5.4 |
| G29 | * | * | | Main spindle selection | 5.4 |
| G28-G29 | | | * | Axis toggle | 7.5 |
| G30 | * | | * | Spindle synchronization (offset) | 5.5 |
| G32 | * | | * | Feedrate "F" as an inverted function of time. | 6.15 |
| G33 | * | | * | Electronic threading | 6.12 |
| G34 | | | | Variable-pitch threading | 6.13 |
| G36 | | | * | Corner rounding | 6.10 |
| G37 | | | * | Tangential entry | 6.8 |
| G38 | | | * | Tangential exit | 6.9 |
| G39 | | | * | Chamfer | 6.11 |
| G40 | * | * | | Cancellation of tool radius compensation | 8.1 |
| G41 | * | | * | Left-hand tool radius compensation | 8.1 |
| G41 N | * | | * | Collision detection | 8.3 |
| G42 | * | | * | Right-hand tool radius compensation | 8.1 |
| G42 N | * | | * | Collision detection | 8.3 |
| G43 | * | ? | * | Tool length compensation | 8.2 |
| G44 | * | ? | | Cancellation of tool length compensation | 8.2 |
| G45 | * | | * | Tangential control (G45) | 6.16 |
| G47 | | | * | Tool movement according to the tool coordinate system. | 15.2 |
| G48 | * | | * | TCP transformation | 15.3 |
| G49 | * | | * | Inclined plane definition | 15.1 |
| G50 | * | | * | Controlled corner rounding | 7.3.3 |
| G51 | * | | * | Look-Ahead | 7.4 |
| G52 | | | * | Movement until making contact | 6.14 |
| G53 | | | * | Programming with respect to machine zero | 4.3 |
| G54 | * | | * | Absolute zero offset 1 | 4.4.2 |
| G55 | * | | * | Absolute zero offset 2 | 4.4.2 |
| G56 | * | | * | Absolute zero offset 3 | 4.4.2 |
| G57 | * | | * | Absolute zero offset 4 | 4.4.2 |
| G58 | * | | * | Additive zero offset 1 | 4.4.2 |
| G59 | * | | * | Additive zero offset 2 | 4.4.2 |
| G60 | | | * | Multiple machining in a straight line | 10.1 |

| Function | M | D | V | Meaning | Section |
|----------|---|---|---|---------|---------|
| G61 | | | * | Multiple machining in rectangular pattern | 10.2 |
| I* | | | * | Grid pattern canned cycle | 10.3 |
| G63 | | | * | Multiple machining in a circular pattern | 10.4 |
| G64 | | | * | Multiple machining in an arc | 10.5 |
| G65 | | | * | Machining programmed with an arc-chord | 10.6 |
| G66 | | | * | Irregular pocket canned cycle | 11.1 / 11.2 |
| G67 | | | * | Irregular pocket roughing | 11.1.2 |
| G68 | | | * | Irregular pocket finishing | 11.1.3 |
| G69 | * | | * | Drilling canned cycle with variable peck | 9.6 |
| G70 | * | ? | * | Programming in inches | 3.3 |
| G71 | * | ? | | Programming in millimeters | 3.3 |
| G72 | * | | * | General and specific scaling factor | 7.6 |
| G73 | * | | * | Rotation of the coordinate system | 7.7 |
| G74 | | | * | Machine reference (home) search | 4.2 |
| G75 | | | * | Probing move until touching | 12.1 |
| G76 | | | * | Probing move while touching | 12.1 |
| G77 | * | | * | Axis coupling (slaving) | 7.8.1 |
| G77S | * | | * | Spindle synchronization | 5.5 |
| G78 | * | * | | Cancellation of axis coupling (slaving) | 7.8.2 |
| G78S | * | * | | Cancellation of spindle synchronization | 5.5 |
| G79 | | | | Canned cycle parameter modification | 9.2.1 |
| G80 | * | * | | Canned cycle cancellation | 9.3 |
| G81 | * | | * | Drilling canned cycle | 9.7 |
| G82 | * | | * | Drilling canned cycle with dwell | 9.8 |
| G83 | * | | * | Deep-hole drilling canned cycle with constant peck | 9.9 |
| G84 | * | | * | Tapping canned cycle | 9.10 |
| G85 | * | | * | Reaming canned cycle | 9.11 |
| G86 | * | | * | Boring canned cycle with withdrawal in G00 | 9.12 |
| G87 | * | | * | Rectangular pocket canned cycle. | 9.13 |
| G88 | * | | * | Circular pocket canned cycle | 9.14 |
| G89 | * | | * | Boring canned cycle with withdrawal in G01 | 9.15 |
| G90 | * | ? | | Absolute programming: | 3.4 |
| G91 | * | ? | * | Incremental programming | 3.4 |
| G92 | | | | Coordinate preset / spindle speed limit | 4.4.1 |
| G93 | | | | Polar origin preset | 4.5 |
| G94 | * | ? | | Feedrate in millimeters (inches) per minute | 5.2.1 |
| G95 | * | ? | * | Feedrate in millimeters (inches) per revolution. | 5.2.2 |
| G96 | * | | * | Constant cutting point speed | 5.2.3 |
| G97 | * | * | | Constant tool center speed | 5.2.4 |
| G98 | * | * | | Withdrawal to the starting plane at the end of the canned cycle | 9.5 |
| G99 | * | | * | Withdrawal to the reference plane at the end of the canned cycle | 9.5 |
| G145 | * | | * | Temporary cancellation of tangential control | 6.17 |

M means modal, i.e. the G function, once programmed, remains active until another incompatible G function is programmed or until an M02, M30, EMERGENCY or RESET is executed or the CNC is turned off and back on.

D means BY DEFAULT, i.e. they will be assumed by the CNC when it is powered on, after executing M02, M30 or after EMERGENCY or RESET.

In those cases indicated by ?, it should be understood that the DEFAULT of these G functions depends on the setting of the general machine parameters of the CNC.

The letter V means that the G code is displayed next to the current machining conditions in the execution and simulation modes.

# PROGRAM CONTROL INSTRUCTIONS

### Display instructions.

| ( *section 14.2* ) |
|---|

| **(ERROR integer, "error text")** |
|---|
| Stops the execution of the program and displays the indicated error. |

| **(MSG "message")** |
|---|
| Displays the indicated message. |

| **(DGWZ expression 1, ..... expression 6)** |
|---|
| Define the graphics area. |

### Enabling and disabling instructions.

| ( *section 14.3* ) |
|---|

| **(ESBLK and DSBLK)** |
|---|
| The CNC executes all the blocks between ESBLK and DSBLK as if it were a single block. |

| **(ESTOP and DSTOP)** |
|---|
| Enabling (ESTOP) and disabling (DSTOP) of the Stop key and the external Stop signal (PLC). |

| **(EFHOLD and DFHOLD)** |
|---|
| Enabling (EFHOLD) and disabling (DFHOLD) of the Feed-hold input (PLC). |

### Flow control instructions.

| ( *section 14.4* ) |
|---|

| **( GOTO N(expression) )** |
|---|
| It causes a jump within the same program, to the block defined by the label N(expression). |

| **( RPT N(expression), N(expression), P(expression) )** |
|---|
| It repeats the execution of the portion of the program between the blocks defined by means of the labels N(expression). |

| **(IF condition <action1> ELSE <action2>)** |
|---|
| It analyzes the given condition that must be a relational expression. If the condition is true (result equal to 1), <action1> will be executed, otherwise (result equal to 0) <action2> will be executed. |

### Subroutine instructions.

| ( *section 14.5* ) |
|---|

| **(SUB integer)** |
|---|
| Subroutine definition. |

| **( RET )** |
|---|
| End of subroutine. |

| **( CALL (expression) )** |
|---|
| Call to a subroutine. |

| **(PCALL (expression), (assignment instruction), (assignment instruction),...) )** |
|---|
| Call to a subroutine. In addition, using assignment instructions, it is possible to initialize up to a maximum of 26 local parameters of this subroutine. |

| **(MCALL (expression), (assignment instruction), (assignment instruction),...) )** |
|---|
| Same as the PCALL instruction, but making the indicated subroutine modal. |

| **( MDOFF )** |
|---|
| Cancellation of modal subroutine. |

### *Probe related instructions.*

|  |
|---|
| ( *section  14.6* ) |

| |
|---|
| **(PROBE (expression), (assignment instruction), (assignment instruction),...) )** |
| It executes a probing canned cycle initializing its parameters by means of assignment instructions. |

### *Interruption-subroutine instructions.*

|  |
|---|
| ( *section  14.7* ) |

| |
|---|
| **( REPOS X, Y, Z, .... )** |
| It must always be used inside an interruption subroutine and facilitates the repositioning of the machine axes to the point of interruption. |

### *Program instructions.*

|  |
|---|
| ( *section  14.8* ) |

| |
|---|
| **( EXEC P(expression), (directory) )** |
| Starts program execution. |

| |
|---|
| **( MEXEC P(expression), (directory) )** |
| Starts program execution in modal mode. |

| |
|---|
| **( OPEN P(expression), (destination directory), A/D, "program comment" )** |
| It begins editing a new program being possible to associate a comment with the program. |

| |
|---|
| **( WRITE <block text> )** |
| It adds, after the last block of the program which began to be edited by means of the mnemonic OPEN P, the information contained in <block text> as a new program block. |

### *Kinematics related instructions.*

|  |
|---|
| ( *section  14.9* ) |

| |
|---|
| **( INIPAR )** |
| Validates the machine parameters modified from the OEM subroutine that defines the kinematics. |

### *Screen customizing instructions.*

|  |
|---|
| ( *section  14.10* ) |

| |
|---|
| **( PAGE (expression) )** |
| The screen displays the indicated user page number (0-255) or system page (1000). |

| |
|---|
| **( SYMBOL (expression 1), (expression 2), (expression 3) )** |
| The screen displays the symbol (0-255) indicated by the expression 1. |
| Its position on the screen is defined by expression 2 (row, 0-639) and by expression 3 (column 0-335). |

| |
|---|
| **(IB (expression) = INPUT "text", format)** |
| It displays the text indicated in the data input window and stores in the input variable (IBn) the data entered by the user. |

| |
|---|
| **( ODW (expression 1), (expression 2), (expression 3) )** |
| It defines and draws a white window on the screen (1 row and 14 columns). |
| Its position on screen is defined by expression 2 (row) and by expression 3 (column). |

| |
|---|
| **( DW(expression 1) = (expression 2), DW (expression 3) = (expression 4), ... )** |
| It displays in the windows indicated by the value of the expression 1, 3,.. , the numerical data indicated by the expression 2,4,.. |

| |
|---|
| **(SK (expression 1) = "text1" (expression 2) = "text 2", .... )** |
| It defines and displays the new softkey menu indicated. |

| |
|---|
| **( WKEY )** |
| It stops the execution of the program until a key is pressed. |

**( WBUF "text", (expression) )**

It adds the text and value of the expression, once it has been evaluated, to the block that is being edited and within the data entry window.

**( WBUF )**

Enters the block being edited into memory. It can only be used in the screen customizing program to be executed in the Editing mode.

**( SYSTEM )**

It ends the execution of the user screen customizing program and returns to the corresponding standard menu of the CNC.

**B.**

Program control instructions

**FAGOR** ➔

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

·**M**· & ·**EN**· M**ODELS**
S**OFT**: **V02.2x**

·**453**·

**B.**

**FAGOR**

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

# SUMMARY OF INTERNAL CNC VARIABLES.

- The R symbol indicates that the variable can be read.
- The W symbol indicates that the variable can be modified.

### Variables associated with tools.

| Variable | CNC | PLC | DNC | ( section 13.2.2 ) |
|----------|-----|-----|-----|--------------------|
| TOOL | R | R | R | Number of the active tool. |
| TOD | R | R | R | Number of active tool offset. |
| NXTOOL | R | R | R | Number of the next requested tool waiting for M06. |
| NXTOD | R | R | R | Number of the next tool's offset. |
| TMZPn | R | R | - | (n) tool's position in the tool magazine. |
| PTOOL | R | - | - | Magazine position to leave the current tool. |
| PNXTOOL | R | - | - | Magazine position to pick up the next tool. |
| TLFDn | R/W | R/W | - | (n) tool's offset number. |
| TLFFn | R/W | R/W | - | (n) tool's family code. |
| TLFNn | R/W | R/W | - | Nominal life assigned to tool (n). |
| TLFRn | R/W | R/W | - | Real life value of tool (n). |
| TMZTn | R/W | R/W | - | Contents of tool magazine position (n). |
| HTOR | R/W | R | R | Tool radius  being used by the CNC to do the calculations. |
| TORn | R/W | R/W | - | Tool radius value of offset (n). |
| TOLn | R/W | R/W | - | Tool length value of offset (n). |
| TOIn | R/W | R/W | - | Tool radius wear of offset (n). |
| TOKn | R/W | R/W | - | Tool length wear of offset (n). |

### Variables associated with zero offsets.

| Variable | CNC | PLC | DNC | ( section 13.2.3 ) |
|----------|-----|-----|-----|--------------------|
| ORG(X-C) | R | R | - | Active zero offset on the selected axis. The value of the additive offset indicated by the PLC is not included. |
| PORGF | R | - | R | Abscissa coordinate value of polar origin. |
| PORGS | R | - | R | Ordinate coordinate value of polar origin. |
| ORG(X-C)n | R/W | R/W | R | Zero offset (n) value of the selected axis. |
| PLCOF(X-C) | R/W | R/W | R | Value of the additive zero offset activated via PLC. |
| ADIOF(X-C) | R | R | R | Value for the selected axis of the zero offset with additive handwheel. |
| ADDORG (X-C) | R | R | R | Value of the active incremental zero offset corresponding to the selected axis. |
| EXTORG | R | R | R | Value of the active absolute zero offset. |

### Variables associated with function G49

Variables associated with the definition of function G49.

| Variable | CNC | PLC | DNC | ( section 13.2.4 ) |
|----------|-----|-----|-----|--------------------|
| ORGROX | R | R | R | X coordinate of the new part zero with respect to home. |
| ORGROY | R | R | R | Y coordinate of the new part zero with respect to home. |
| ORGROZ | R | R | R | Z coordinate of the new part zero with respect to home. |
| ORGROA | R | R | R | Value assigned to parameter A. |
| ORGROB | R | R | R | Value assigned to parameter B. |
| ORGROC | R | R | R | Value assigned to parameter C. |
| ORGROI | R | R | R | Value assigned to parameter I. |
| ORGROJ | R | R | R | Value assigned to parameter J. |
| ORGROK | R | R | R | Value assigned to parameter K. |
| ORGROQ | R | R | R | Value assigned to parameter Q. |
| ORGROR | R | R | R | Value assigned to parameter R. |
| ORGROS | R | R | R | Value assigned to parameter S. |
| GTRATY | R | R | R | Type of G49 programmed. |

**FAGOR**

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2x

Variables updated by the CNC once function G49 is executed.

| | | | | |
|---|---|---|---|---|
| TOOROF | R/W | R/W | R/W | Position to be occupied by the spindle's main rotary axis. |
| TOOROS | R/W | R/W | R/W | Position to be occupied by the spindle's 2nd rotary axis. |

### *Variables associated with machine parameters.*

| Variable | CNC | PLC | DNC | ( *section 13.2.5* ) |
|---|---|---|---|---|
| MPGn | R | R | - | Value assigned to general machine parameter (n). |
| MP(X-C)n | R | R | - | Value assigned to (X-C) axis machine parameter (n). |
| MPSn | R | R | - | Value assigned to machine parameter (n) of the main spindle. |
| MPSSn | R | R | - | Value assigned to machine parameter (n) of the second spindle. |
| MPASn | R | R | - | Value assigned to machine parameter (n) of the auxiliary spindle. |
| MPLCn | R | R | - | Value assigned to machine parameter (n) of the PLC. |

### *Work zone related variables.*

| Variable | CNC | PLC | DNC | ( *section 13.2.6* ) |
|---|---|---|---|---|
| FZONE | R | R/W | R | Status of work zone 1. |
| FZLO(X-C) | R | R/W | R | Work zone 1. Lower limit along the selected axis (X/C). |
| FZUP(X-C) | R | R/W | R | Work zone 1. Upper limit along the selected axis (X-C). |
| SZONE | R | R/W | R | Status of work zone 2. |
| SZLO(X-C) | R | R/W | R | Work zone 2. Lower limit along the selected axis (X/C). |
| SZUP(X-C) | R | R/W | R | Work zone 2. Upper limit along the selected axis (X-C). |
| TZONE | R | R/W | R | Status of work zone 3. |
| TZLO(X-C) | R | R/W | R | Work zone 3. Lower limit along the selected axis (X/C). |
| TZUP(X-C) | R | R/W | R | Work zone 3. Upper limit along the selected axis (X-C). |
| FOZONE | R | R/W | R | Status of work zone 4. |
| FOZLO(X-C) | R | R/W | R | Work zone 4. Lower limit along the selected axis (X/C). |
| FOZUP(X-C) | R | R/W | R | Work zone 4. Upper limit along the selected axis (X-C). |
| FIZONE | R | R/W | R | Status of work zone 5. |
| FIZLO(X-C) | R | R/W | R | Work zone 5. Lower limit along the selected axis (X/C). |
| FIZUP(X-C) | R | R/W | R | Work zone 5. Upper limit along the selected axis (X-C). |

### *Feedrate related variables.*

| Variable | CNC | PLC | DNC | ( *section 13.2.7* ) |
|---|---|---|---|---|
| FREAL | R | R | R | Real feedrate of the CNC in mm/min or inch/min. |
| FREAL(X-C) | R | R | R | Actual (real) CNC feedrate of the selected axis. |
| FTEO/X-C) | R | R | R | Theoretical CNC feedrate of the selected axis. |

Variables associated with function G94.

| | | | | |
|---|---|---|---|---|
| FEED | R | R | R | Active feedrate at the CNC in mm/min or inch/min. |
| DNCF | R | R | R/W | Feedrate selected via DNC. |
| PLCF | R | R/W | R | Feedrate selected via PLC. |
| PRGF | R | R | R | Feedrate selected by program. |

Variables associated with function G94.

| | | | | |
|---|---|---|---|---|
| FPREV | R | R | R | Active feedrate at CNC, in m/rev or inch/rev. |
| DNCFPR | R | R | R/W | Feedrate selected via DNC. |
| PLCFPR | R | R/W | R | Feedrate selected via PLC. |
| PRGFPR | R | R | R | Feedrate selected by program. |

Variables associated with function G94.

| | | | | |
|---|---|---|---|---|
| PRGFIN | R | R | R | Feedrate selected by program, in 1/min. |

Variables associated with feedrate override (%)

| | | | | |
|---|---|---|---|---|
| FRO | R | R | R | Feedrate Override (%) active at the CNC. |
| PRGFRO | R/W | R | R | Override (%) selected by program. |
| DNCFRO | R | R | R/W | Override (%) selected via DNC. |
| PLCFRO | R | R/W | R | Override (%) selected via PLC. |
| CNCFRO | R | R | R | Override (%) selected from the front panel knob. |
| PLCCFR | R | R/W | R | Override (%) of the PLC execution channel. |

### *Coordinate related variables.*

| Variable | CNC | PLC | DNC | ( *section 13.2.8* ) |
|---|---|---|---|---|
| PPOS(X-C) | R | - | - | Programmed theoretical position value (coordinate). |
| POS(X-C) | R | R | R | Machine coordinates. Real coordinates of the tool base. |
| TPOS(X-C) | R | R | R | Machine coordinates. Theoretical coordinates of the tool base. |
| APOS(X-C) | R | R | R | Part coordinates. Real coordinates of the tool base. |
| ATPOS(X-C) | R | R | R | Part coordinates. Theoretical coordinates of the tool base. |
| DPOS(X-C) | R | R | R | Theoretical position of the probe when the probe touched the part. |
| FLWE(X-C) | R | R | R | Following error of the indicated axis. |
| DEFLEX | R | R | R | Probe deflection along X axis. |
| DEFLEY | R | R | R | Probe deflection along Y axis. |
| DEFLEZ | R | R | R | Probe deflection along Z axis. |
| DIST(X-C) | R/W | R/W | R | Distance traveled by the indicated axis. |
| LIMPL(X-C) | R/W | R/W | R | Second upper travel limit. |
| LIMMI(X-C) | R/W | R/W | R | Second lower travel limit. |
| DPLY(X-C) | R | R | R | Coordinate of the selected axis displayed on the screen. |
| DRPO(X-C) | R | R | R | Position indicated by the Sercos drive of the selected axis. |
| GPOS(X-C)n p | R | - | - | Coordinate of the selected axis, programmed in the (n) block of the program (p). |

### *Variables associated with electronic handwheels.*

| Variable | CNC | PLC | DNC | ( *section 13.2.9* ) |
|---|---|---|---|---|
| HANPF | R | R | - | Pulses received from 1st handwheel since the CNC was turned on. |
| HANPS | R | R | - | Pulses received from 2nd handwheel since the CNC was turned on. |
| HANPT | R | R | - | Pulses received from 3rd handwheel since the CNC was turned on. |
| HANPFO | R | R | - | Pulses received from 4th handwheel since the CNC was turned on. |
| HANDSE | R | R | | For handwheels with a selector button, it indicates whether that button has been pressed or not. |
| HANFCT | R | R/W | R | Multiplying factor different for each handwheel (when having several). |
| HBEVAR | R | R/W | R | HBE handwheel. Reading enabled, axis being jogged and multiplying factor (x1, x10, x100). |
| MASLAN | R/W | R/W | R/W | Linear path angle for "Path handwheel" or "Path Jog" mode. |
| MASCFI | R/W | R/W | R/W | Arc center coordinates for "Path handwheel mode" or "Path jog". |
| MASCSE | R/W | R/W | R/W | Arc center coordinates for "Path handwheel mode" or "Path jog". |

### *Feedback related variables.*

| Variable | CNC | PLC | DNC | ( *section 13.2.10* ) |
|---|---|---|---|---|
| ASIN(X-C) | R | R | R | A signal of the CNC's sinusoidal feedback for the selected axis. |
| BSIN(X-C) | R | R | R | B signal of the CNC's sinusoidal feedback for the selected axis. |
| ASINS | R | R | R | "A" signal of the CNC's sinusoidal feedback for the spindle. |
| BSINS | R | R | R | "B" signal of the CNC's sinusoidal feedback for the spindle. |
| SASINS | R | R | R | "A" signal of the CNC sinusoidal feedback for the second spindle. |
| SBSINS | R | R | R | "B" signal of the CNC sinusoidal feedback for the second spindle. |

### *Variables associated with the main spindle.*

| Variable | CNC | PLC | DNC | ( *section 13.2.11* ) |
|---|---|---|---|---|
| SREAL | R | R | R | Real spindle speed. |
| FTEOS | R | R | R | Theoretical spindle speed. |

**C.**

Summary of internal CNC variables.

**FAGOR**

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

Variables associated with spindle speed.

| | CNC | PLC | DNC | |
|---|---|---|---|---|
| SPEED | R | R | R | Active spindle speed at the CNC. |
| DNCS | R | R | R/W | Spindle speed selected via DNC. |
| PLCS | R | R/W | R | Spindle speed selected via PLC. |
| PRGS | R | R | R | Spindle speed selected by program. |

Variables associated with the spindle override.

| | CNC | PLC | DNC | |
|---|---|---|---|---|
| SSO | R | R | R | Spindle Speed Override (%) active at the CNC. |
| PRGSSO | R/W | R | R | Override (%) selected by program. |
| DNCSSO | R | R | R/W | Override (%) selected via DNC. |
| PLCSSO | R | R/W | R | Override (%) selected via PLC. |
| CNCSSO | R | R | R | Spindle Speed Override (%) selected from front panel. |

Speed limit related variables.

| | CNC | PLC | DNC | |
|---|---|---|---|---|
| SLIMIT | R | R | R | Spindle speed limit active at the CNC. |
| DNCSL | R | R | R/W | Spindle speed limit selected via DNC. |
| PLCSL | R | R/W | R | Spindle speed limit selected via PLC. |
| PRGSL | R | R | R | Spindle speed limit selected by program. |
| MDISL | R | R/W | R | Maximum machining spindle speed. |

Position related variables.

| | CNC | PLC | DNC | |
|---|---|---|---|---|
| POSS | R | R | R | Real spindle position. Reading from the PLC in ten-thousandths of a degree (within ±999999999) and from the CNC in degrees (within ±99999.9999). |
| RPOSS | R | R | R | Real spindle position. Reading from the PLC in ten-thousandths of a degree (between -3600000 and 3600000) and from the CNC in degrees (between -360 and 360). |
| TPOSS | R | R | R | Theoretical spindle position. Reading from the PLC in ten-thousandths of a degree (within ±999999999) and from the CNC in degrees (within ±99999.9999). |
| RTPOSS | R | R | R | Theoretical spindle position. Reading from the PLC in ten-thousandths of a degree (between 0 and 3600000) and from the CNC in degrees (between 0 and 360). |
| DRPOS | R | R | R | Position indicated by the Sercos drive. |
| PRGSP | R | R | R | Position programmed in M19 via program for the main spindle. |

Variables related to the following error.

| | CNC | PLC | DNC | |
|---|---|---|---|---|
| FLWES | R | R | R | Spindle following error. |
| SYNCER | R | R | R | Error of second spindle (synchronized) following the main spindle. |

### *Variables associated with the second spindle.*

| Variable | CNC | PLC | DNC | ( *section 13.2.12* ) |
|---|---|---|---|---|
| SSREAL | R | R | R | Real spindle speed. |
| SFTEOS | R | R | R | Theoretical spindle speed. |

Variables associated with spindle speed.

| | CNC | PLC | DNC | |
|---|---|---|---|---|
| SSPEED | R | R | R | Active spindle speed at the CNC. |
| SDNCS | R | R | R/W | Spindle speed selected via DNC. |
| SPLCS | R | R/W | R | Spindle speed selected via PLC. |
| SPRGS | R | R | R | Spindle speed selected by program. |

Variables associated with the spindle override.

| | CNC | PLC | DNC | |
|---|---|---|---|---|
| SSSO | R | R | R | Spindle Speed Override (%) active at the CNC. |
| SPRGSO | R/W | R | R | Override (%) selected by program. |
| SDNCSO | R | R | R/W | Override (%) selected via DNC. |
| SPLCSO | R | R/W | R | Override (%) selected via PLC. |
| SCNCSO | R | R | R | Spindle Speed Override (%) selected from front panel. |

Speed limit related variables.

| | | | | |
|---|---|---|---|---|
| SSLIMI | R | R | R | Spindle speed limit active at the CNC. |
| SDNCSL | R | R | R/W | Spindle speed limit selected via DNC. |
| SPLCSL | R | R/W | R | Spindle speed limit selected via PLC. |
| SPRGSL | R | R | R | Spindle speed limit selected by program. |

Position related variables.

| | | | | |
|---|---|---|---|---|
| SPOSS | R | R | R | Real spindle position.<br>Reading from the PLC in ten-thousandths of a degree (within ±999999999) and from the CNC in degrees (within ±99999.9999). |
| SRPOSS | R | R | R | Real spindle position.<br>Reading from the PLC in ten-thousandths of a degree (between -3600000 and 3600000) and from the CNC in degrees (between -360 and 360). |
| STPOSS | R | R | R | Theoretical spindle position.<br>Reading from the PLC in ten-thousandths of a degree (within ±999999999) and from the CNC in degrees (within ±99999.9999). |
| SRTPOS | R | R | R | Theoretical spindle position.<br>Reading from the PLC in ten-thousandths of a degree (between 0 and 3600000) and from the CNC in degrees (between 0 and 360). |
| SDRPOS | R | R | R | Position indicated by the Sercos drive. |
| SPRGSP | R | R | R | Position programmed in M19 via program for the second spindle. |

Variables related to the following error.

| | | | | |
|---|---|---|---|---|
| SFLWES | R | R | R | Spindle following error. |

### *Variables associated with the live tool.*

| Variable | CNC | PLC | DNC | ( *section  13.2.13* ) |
|---|---|---|---|---|
| ASPROG | R | R | - | Speed programmed in M45 S (within the associated subroutine). |

### *PLC related variables.*

| Variable | CNC | PLC | DNC | ( *section  13.2.14* ) |
|---|---|---|---|---|
| PLCMSG | R | - | R | Number of the active PLC message with the highest priority. |
| PLCIn | R/W | - | - | 32 PLC inputs starting from (n). |
| PLCOn | R/W | - | - | 32 PLC outputs starting from (n). |
| PLCMn | R/W | - | - | 32 PLC marks starting from (n). |
| PLCRn | R/W | - | - | (n) Register. |
| PLCTn | R/W | - | - | Indicated (n) Timer's count. |
| PLCCn | R/W | - | - | Indicated (n) Counter's count. |
| PLCMMn | R/W | - | - | Modifies the (n) mark of the PLC. |

### *Variables associated with local and global parameters.*

| Variable | CNC | PLC | DNC | ( *section  13.2.15* ) |
|---|---|---|---|---|
| GUP n | - | R/W | - | Global parameter (P100-P299) (n). |
| LUP (a,b) | - | R/W | - | Indicated local (P0-P25) parameter (b) of the nesting level (a). |
| CALLP | R | - | - | Indicates which local parameters have been defined by means of a PCALL or MCALL instruction (calling a subroutine). |

### *Sercos variables.*

| Variable | CNC | PLC | DNC | ( *section  13.2.16* ) |
|---|---|---|---|---|
| SETGE(X-C) | W | W | - | Gear ratio and parameter set of the (X-C) axis drive. |
| SETGES | W | W | - | Gear ratio and parameter set of the main spindle. |
| SSETGS | W | W | - | Gear ratio and parameter set of the second spindle. |
| SVAR(X-C) id | R/W | - | - | Sercos variable sercos for the (X-C) axis "id". |
| SVARS id | R/W | - | - | Sercos variable sercos for the main spindle "id". |
| SSVARS id | R/W | - | - | Sercos variable sercos for the second spindle "id". |
| TSVAR(X-C) id | R | - | - | Third attribute of the sercos variable for the (X-C) axis "id". |
| TSVARS id | R | - | - | Third attribute of the sercos variable for the main spindle "id". |
| TSSVAR id | R | - | - | Third attribute of the sercos variable for the second spindle "id". |

### *Software & hardware configuration variables.*

| Variable | CNC | PLC | DNC | ( *section 13.2.17* ) |
|----------|-----|-----|-----|-----------------------|
| HARCON | R | R | R | It indicates, with bits, the CNC's hardware configuration. |
| HARCOA | R | R | R | It indicates, with bits, the CNC's hardware configuration. |
| IDHARH | R | R | R | Hardware identifier (8 least significant bits). |
| IDHARL | R | R | R | Hardware identifier (4 most significant bits). |
| SOFCON | R | R | R | Software version of the CNC (bits 15-0) and HD (bits 31-16) |
| HDMEGA | R | R | R | Hard disk size (in megabytes). |
| KEYIDE | R | R | R | Keyboard code, according to the auto-identification system. |
| MODEL | R | R | R | It identifies the CNC model, mill or lather. |

### *Variables associated with telediagnosis.*

| Variable | CNC | PLC | DNC | ( *section 13.2.18* ) |
|----------|-----|-----|-----|-----------------------|
| HARSWA | R | R | R | Hardware configuration. |
| HARSWB | R | R | R | Hardware configuration. |
| HARTST | R | R | R | Hardware test. |
| MEMTST | R | R | R | Memory test. |
| NODE | R | R | R | Node number in the Sercos ring. |
| VCHECK | R | R | R | Software version checksum. |
| IONODE | R | R | R | Position of the "ADDRESS" switch of the I/O CAN bus. |
| IOSLOC | R | R | R | Number of local I/O available. |
| IOSREM | R | R | R | Number of remote I/O available. |

### *Operating-mode related variables.*

| Variable | CNC | PLC | DNC | ( *section 13.2.19* ) |
|----------|-----|-----|-----|-----------------------|
| OPMODE | R | R | R | Operation mode. |
| OPMODA | R | R | R | Operating mode when working in the main channel. |
| OPMODB | R | R | R | Type of simulation. |
| OPMODC | R | R | R | Axes selected by handwheel. |

### *Other variables.*

| Variable | CNC | PLC | DNC | ( *section 13.2.20* ) |
|----------|-----|-----|-----|-----------------------|
| NBTOOL | R | - | R | Number of the tool being managed.. |
| PRGN | R | R | R | Number of the program in execution. |
| BLKN | R | R | R | Label number of the last executed block. |
| GSn | R | - | - | Status of the indicated G function (n). |
| GGSA | - | R | R | Status of functions G00 thru G24. |
| GGSB | - | R | R | Status of functions G25 thru G49. |
| GGSC | - | R | R | Status of functions G50 thru G74. |
| GGSD | - | R | R | Status of functions G75 thru G99. |
| GGSE | - | R | R | Status of functions G100 thru G124. |
| GGSF | - | R | R | Status of functions G125 thru G149. |
| GGSG | - | R | R | Status of functions G150 thru G174. |
| GGSH | - | R | R | Status of functions G175 thru G199. |
| GGSI | - | R | R | Status of functions G200 thru G224. |
| GGSJ | - | R | R | Status of functions G225 thru G249. |
| GGSK | - | R | R | Status of functions G250 thru G274. |
| GGSL | - | R | R | Status of functions G275 thru G299. |
| GGSM | - | R | R | Status of functions G300 through G324. |
| GGSN | - | R | R | Status of functions G325 thru G349. |
| GGSO | - | R | R | Status of functions G350 thru G374. |
| GGSP | - | R | R | Status of functions G375 thru G399. |
| GGSQ | - | R | R | Status of functions G400 thru G424. |
| MSn | R | - | - | Status of the indicated M function (n) |
| GMS | - | - | R | Status of M functions: M (0..6, 8, 9, 19, 30, 41..44). |
| PLANE | R | R | R | Abscissa and ordinate axes of the active plane. |

| Variable | CNC | PLC | DNC | ( *section 13.2.20* ) |
|---|---|---|---|---|
| LONGAX | R | R | R | Axis affected by the tool length compensation (G15). |
| MIRROR | R | R | R | Active mirror images. |
| SCALE | R | R | R | General scaling factor applied. Reading from the PLC in ten-thousandths. |
| SCALE(X-C) | R | R | R | Scaling Factor applied only to the indicated axis. Reading from the PLC in ten-thousandths. |
| ORGROT | R | R | R | Rotation angle (G73) of the coordinate system. |
| ROTPF | R | - | - | Abscissa of rotation center. |
| ROTPS | R | - | - | Ordinate of rotation center. |
| PRBST | R | R | R | Returns probe status. |
| CLOCK | R | R | R | System clock in seconds. |
| TIME | R | R | R/W | Time in hours-minutes-seconds format. |
| DATE | R | R | R/W | Date in year-month-day format. |
| TIMER | R/W | R/W | R/W | Clock activated by PLC, in seconds. |
| CYTIME | R | R | R | Time to execute a part in hundredths of a second. |
| PARTC | R/W | R/W | R/W | Parts counter of the CNC. |
| FIRST | R | R | R | First time a program is executed. |
| KEY | R/W | R/W | R/W | keystroke code. |
| KEYSRC | R/W | R/W | R/W | Source of the keys. |
| ANAIn | R | R | R | Voltage (in volts) of the indicated analog input (n). |
| ANAOn | R/W | R/W | R/W | Voltage (in volts) to apply to the indicated output (n). |
| CNCERR | - | R | R | Active CNC error number. |
| PLCERR | - | - | R | Active PLC error number. |
| DNCERR | - | R | - | Number of the error generated during DNC communications. |
| AXICOM | R | R | R | Pairs of axes switched with function G28. |
| TANGAN | R | R | R | Angular position with respect to the path (G45). |
| TPIOUT(X-C) | R | R | R | Output of the PI of the tandem master axis in rpm. |
| DNCSTA | - | R | - | DNC transmission status. |
| TIMEG | R | R | R | Remaining time to finish the dwell block (in hundredths of a second). |
| SELPRO | R/W | R/W | R | When having two probe inputs, it selects the active input. |
| DIAM | R/W | R/W | R | It changes the programming mode for X axis coordinates between radius and diameter. |
| PRBMOD | R/W | R/W | R | Indicates whether a probing error must be displayed or not. |
| RIP | - | - | - | Linear theoretical feedrate resulting from the next loop (in mm/min). |
| TEMPIn | R | R | R | It returns the temperature in tenths of a degree detected by the PT100. |
| TIPPRB | R | R | R | PROBE cycle being executed. |
| TIPDIG | R | R | R | DIGIT cycle being executed. |
| PANEDI | R | R | R | WINDRAW55 application. Number of screen being executed. |
| DATEDI | R | R | R | WINDRAW55 application. Number of element (item) being executed. |
| FBDIF(X-C) | R | R | R | For monitoring the difference between the coordinates (position values) of the first and second feedback on the oscilloscope |
| CYCLEV | R | R | R | It indicates on the conversational model, the tab number being displayed at a particular moment. |
| CYCEDI | R | R | R | It indicates on the conversational model, the cycle or screen number being displayed at a particular moment. |
| DISBLO | R | R | R | It indicates the value of the total programmed distance in blocks with look-ahead. |
| MIXPO(X..C) | R | R | R | Variable that indicates the axis position with the combined feedback. |
| FLWAC(X..C) | R | R | R | Variable that indicates the real error considering the second feedback. |
| DISABMOD | R | R/W | R | Disables some actions or modes. |
| CYCCHORDERR | R/W | - | - | Defines the chordal error of the canned cycles. |

⚠ *The "KEY" variable can be "written" at the CNC only via the user channel.*
*The "NBTOOL" variable can only be used within the tool change subroutine.*

# C.

Summary of internal CNC variables.

**FAGOR**

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

# KEY CODE

## *Alpha-numeric keyboard and monitor*
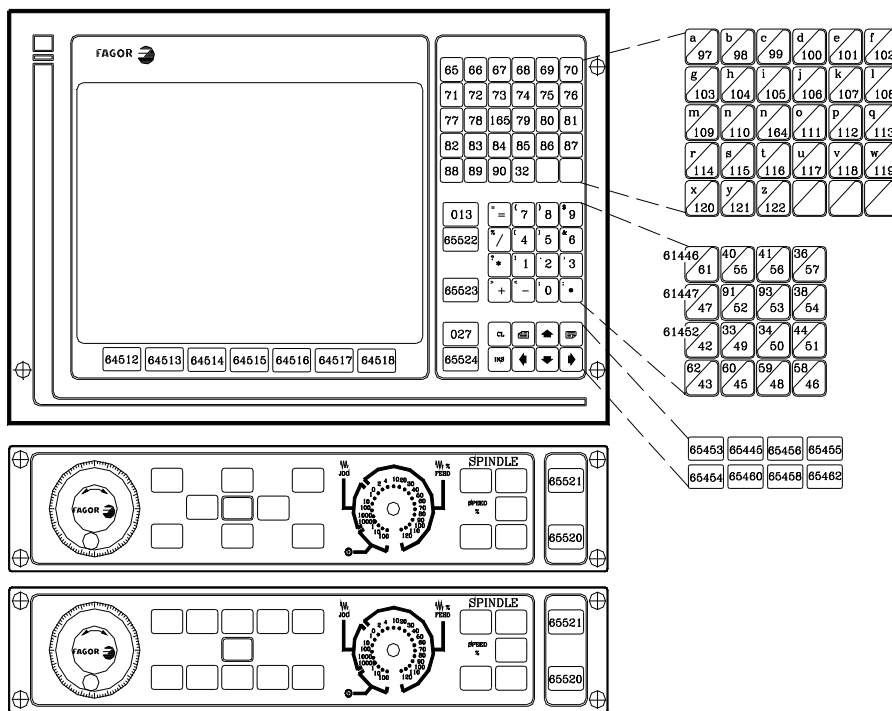


**D.**

Key code
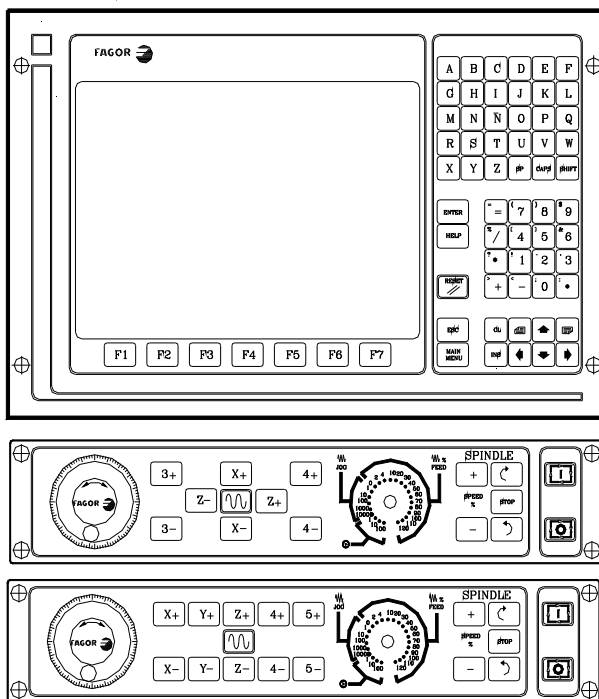
**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

# D.
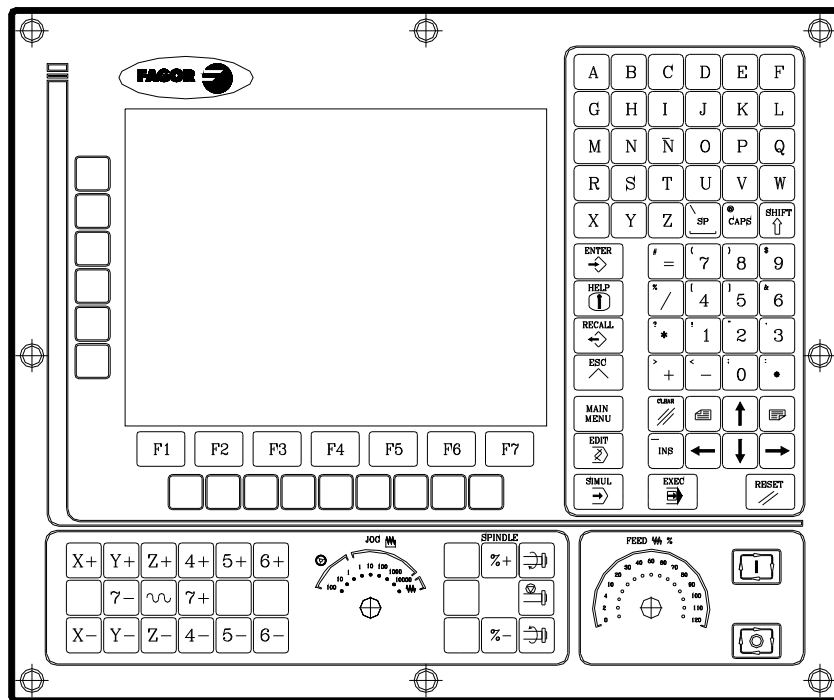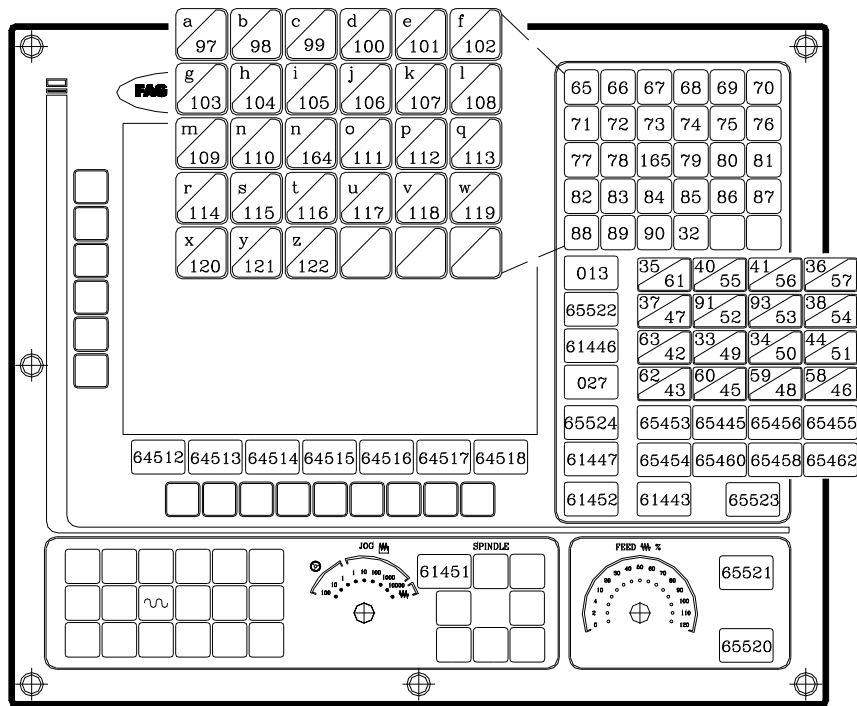
FAGOR

**FAGOR AUTOMATION**
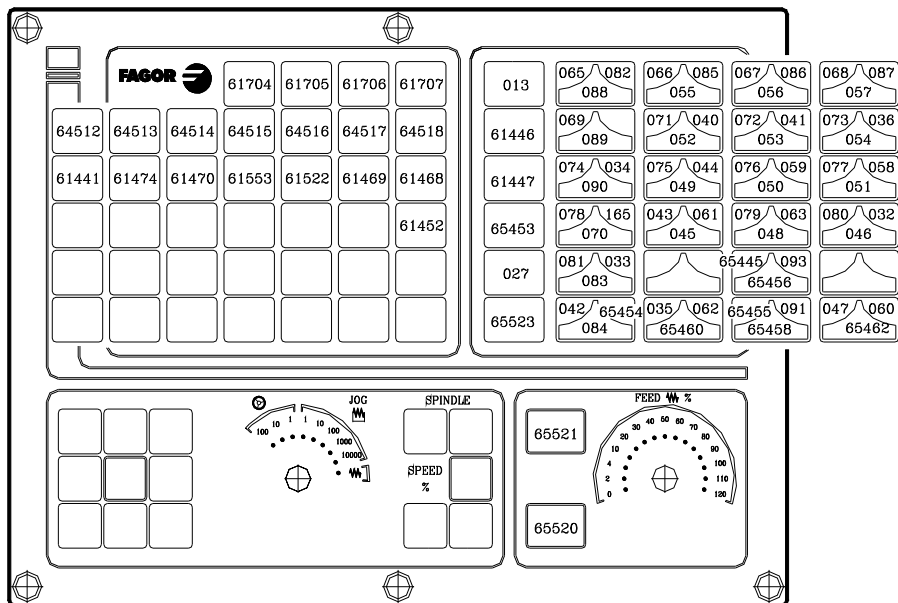
**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

## Alphanumeric operator panel
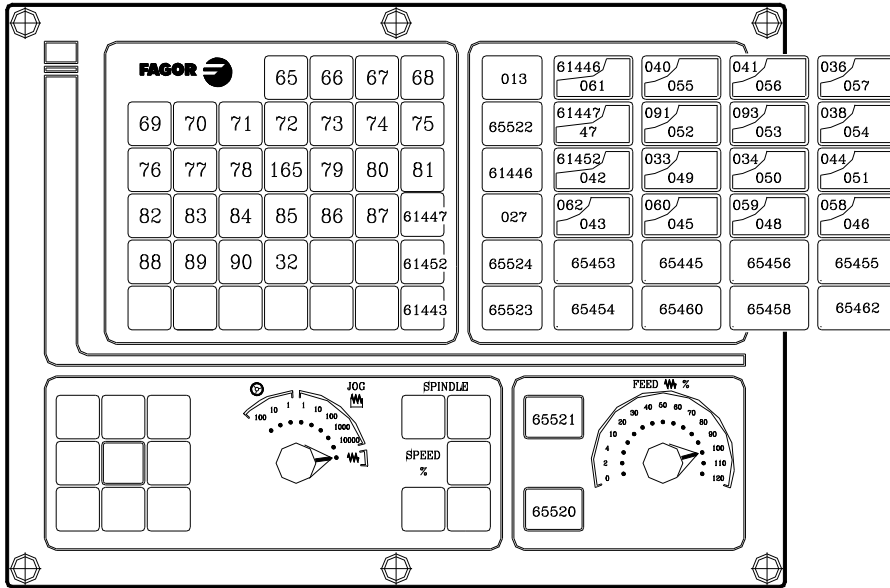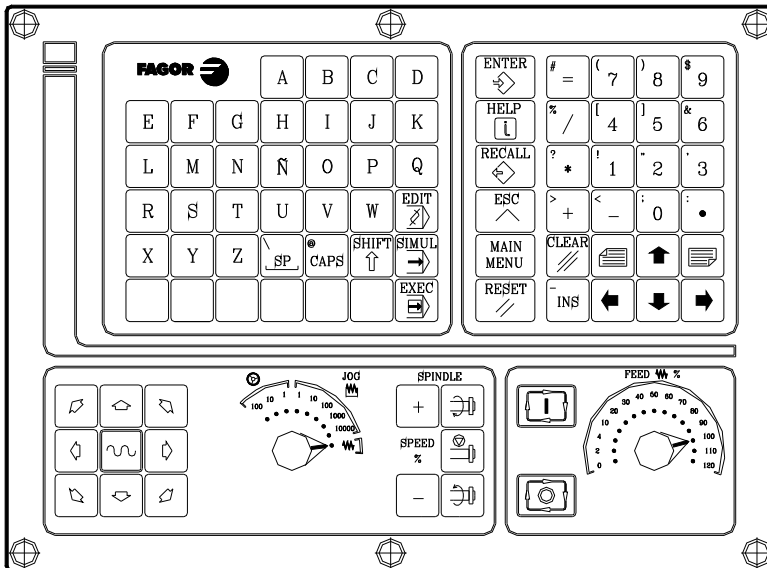
## MC operator panel

**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

·466·
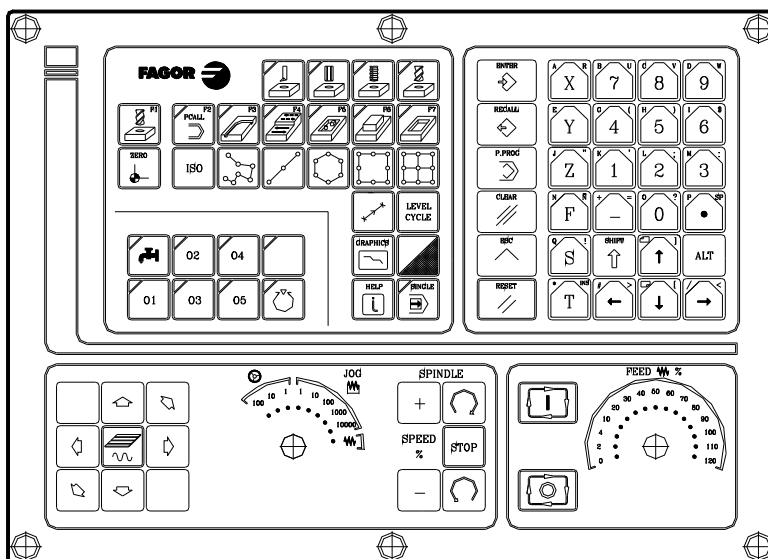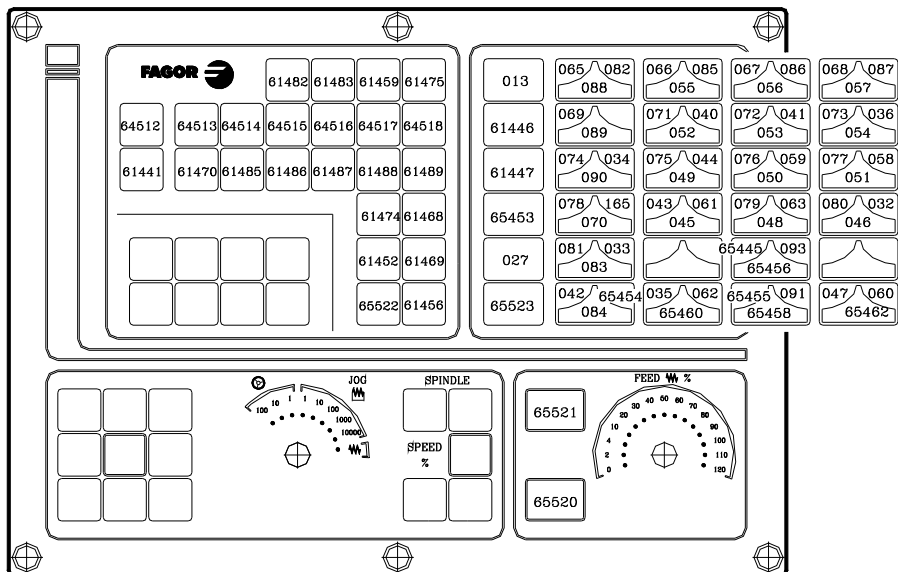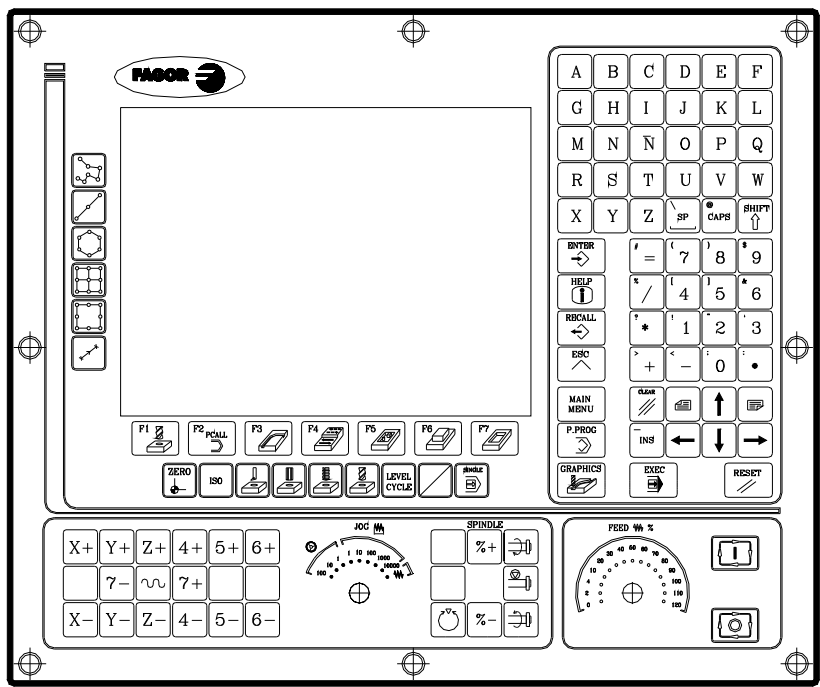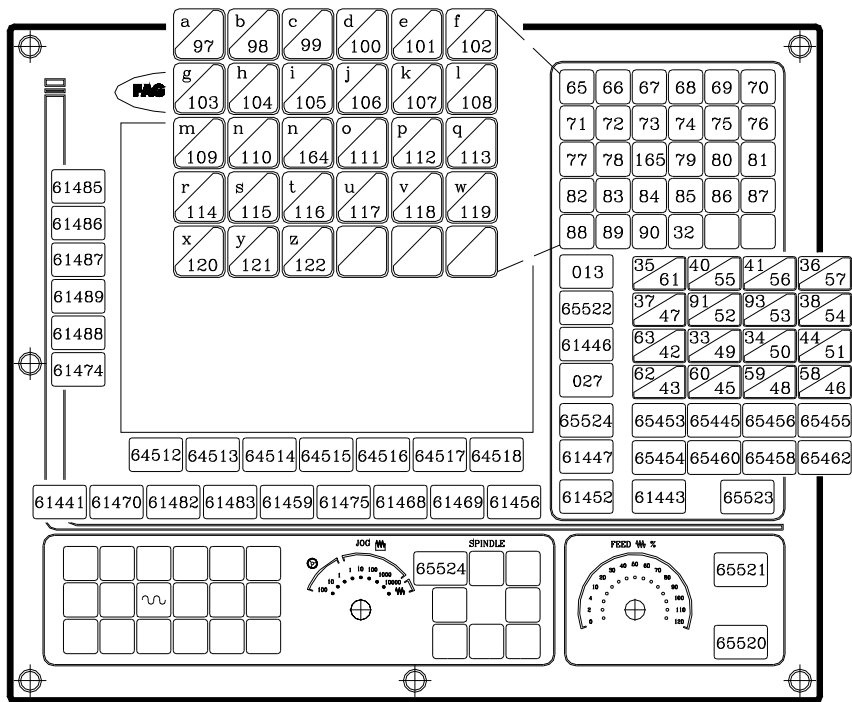
FAGOR

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**
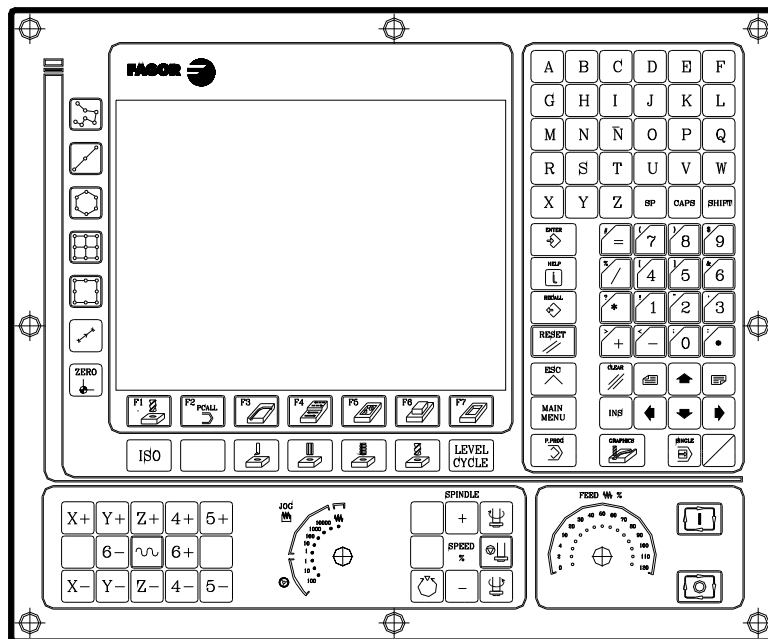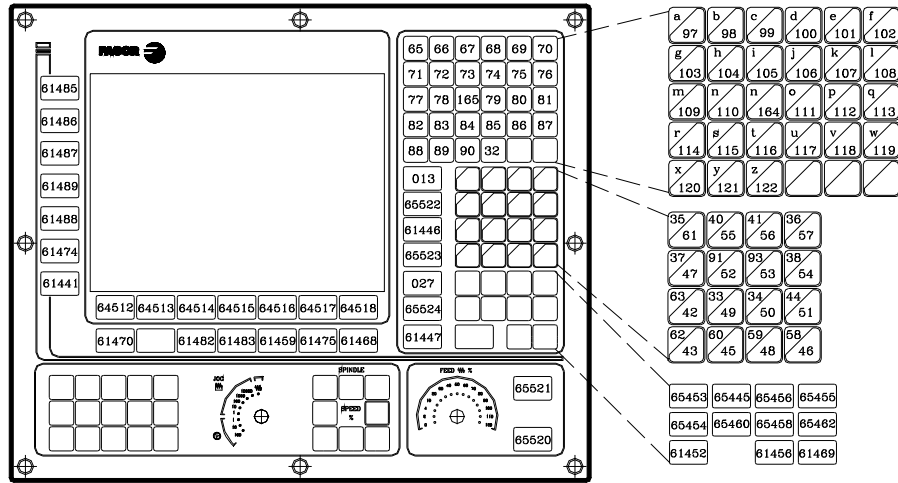
·M· & ·EN· MODELS
SOFT: V02.2x

**FAGOR**

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

### MCO/TCO operator panel

## *Alphanumeric keyboard*

**FAGOR**

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2x

·**470**·

***11" LCD Monitor***

# D.
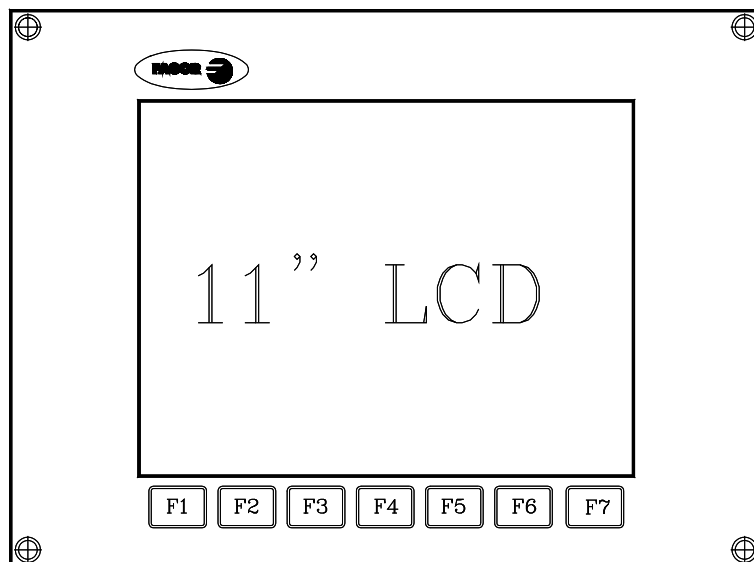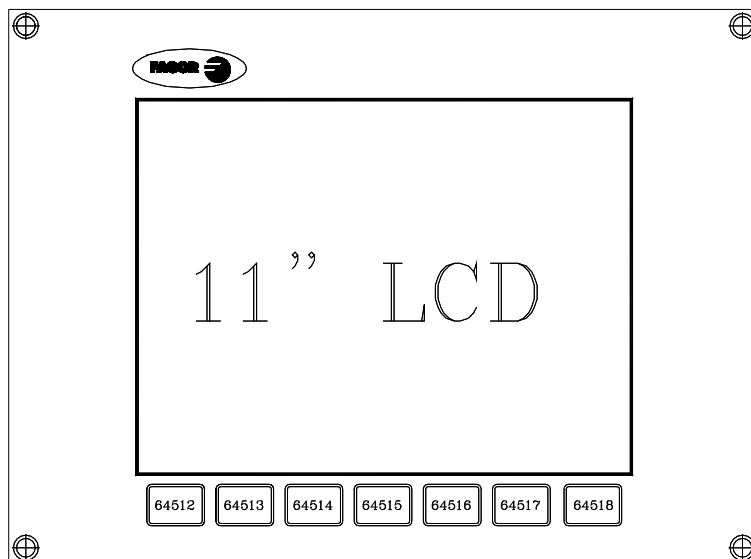
**FAGOR**

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

# PROGRAMMING ASSISTANCE SCREENS OF THE SYSTEM.

These screens (pages) may be displayed using the high level instruction "PAGE". They all belong to the CNC system and are used as help screens of the corresponding functions.

### *Syntax-graphics help screens*

| | |
|---|---|
| Page 1000 | Preparatory functions G00-G09. |
| Page 1001 | Preparatory functions G10-G19. |
| Page 1002 | Preparatory functions G20-G44. |
| Page 1003 | Preparatory functions G53-G59. |
| Page 1004 | Preparatory functions G60-G69. |
| Page 1005 | Preparatory functions G70-G79. |
| Page 1006 | Preparatory functions G80-G89. |
| Page 1007 | Preparatory functions G90-G99. |
| Page 1008 | Auxiliary (miscellaneous) M functions. |
| Page 1009 | Auxiliary (miscellaneous) M functions with the "next page" symbol. |
| Page 1010 | Same as number 250 of the directory if there is one. |
| Page 1011 | Same as number 251 of the directory if there is one. |
| Page 1012 | Same as number 252 of the directory if there is one. |
| Page 1013 | Same as number 253 of the directory if there is one. |
| Page 1014 | Same as number 254 of the directory if there is one. |
| Page 1015 | Same as number 255 of the directory if there is one. |
| Page 1016 | High level language dictionary (from A to G). |
| Page 1017 | High level language dictionary (from H to N). |
| Page 1018 | High level language dictionary (from O to S). |
| Page 1019 | High level language dictionary (from T to Z). |
| Page 1020 | Variables that may be accessed through high level language (part 1). |
| Page 1021 | Variables that may be accessed through high level language (part 2). |
| Page 1022 | Variables that may be accessed through high level language (part 3). |
| Page 1023 | Variables that may be accessed through high level language (part 4). |
| Page 1024 | Variables that may be accessed through high level language (part 5). |
| Page 1025 | Variables that may be accessed through high level language (part 6). |
| Page 1026 | Variables that may be accessed through high level language (part 7). |
| Page 1027 | Variables that may be accessed through high level language (part 8). |
| Page 1028 | Variables that may be accessed through high level language (part 9). |
| Page 1029 | Variables that may be accessed through high level language (part 10). |
| Page 1030 | Variables that may be accessed through high level language (part 11). |
| Page 1031 | Variables that may be accessed through high level language (part 12). |
| Page 1032 | Mathematical operators. |

**E.**

*Syntax help: ISO language*

**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2x

### *Syntax help: CNC tables*

### *Syntax aids: high-level language*

# E.

Programming assistance screens of the system.

**FAGOR** ⟩

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2X

***Syntax help: Canned cycles***

# 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 CNC and, consequently, possible malfunctions.

On the other hand, 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.
- Alcohol.

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

*To check the fuses, first unplug the unit from mains If the CNC does not turn on when flipping the power switch, check that the fuses are the right ones and they are in good condition.*

*Avoid solvents. The action of solvents such as chlorine hydrocarbons, benzole, esters and ether may damage the plastics used to make the front panel of the unit.*

*Do not manipulate 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 connected to main AC power.*
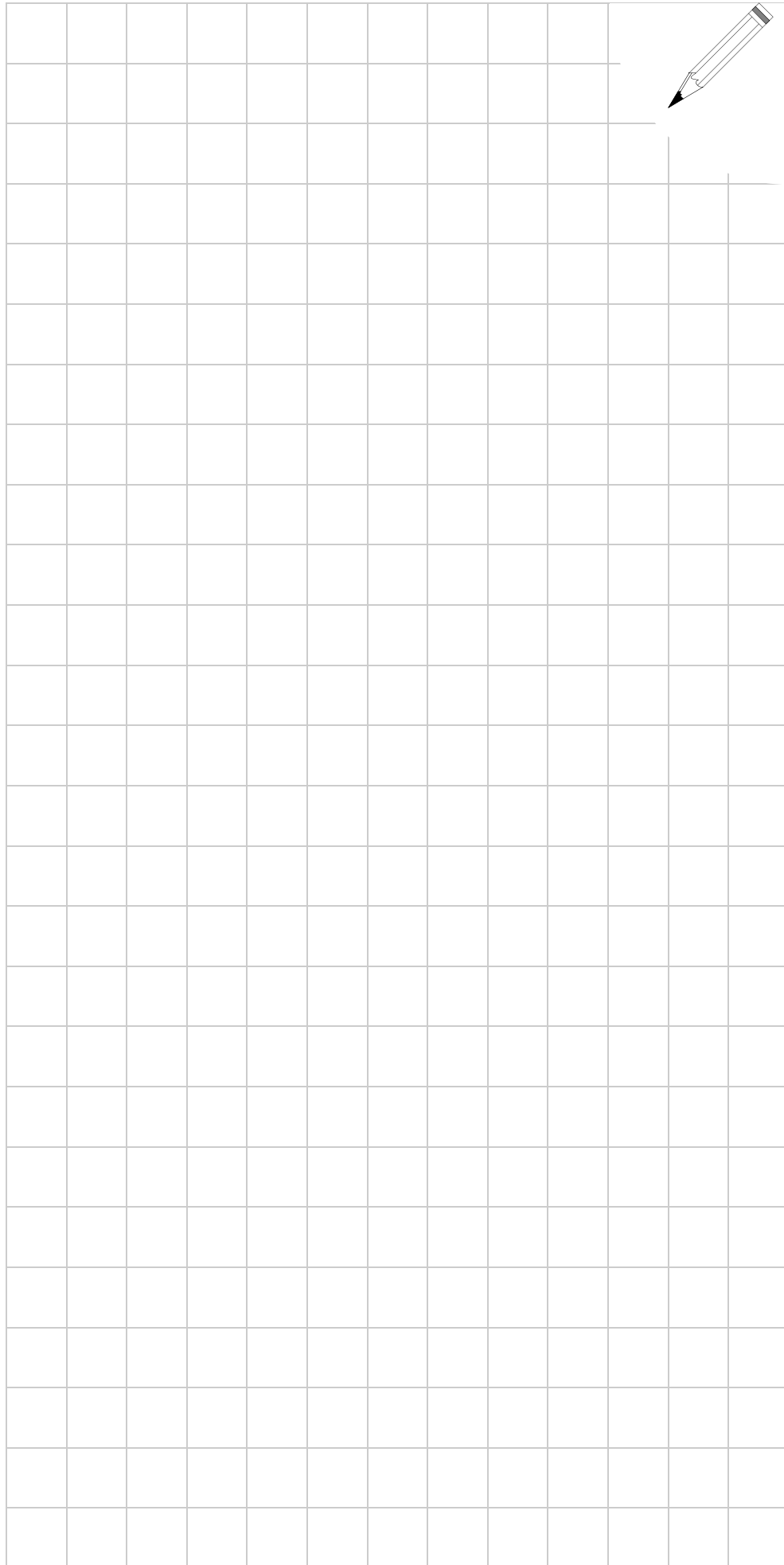
**F.**

Maintenance

**FAGOR**

**FAGOR AUTOMATION**

**CNC 8055**
**CNC 8055i**

·M· & ·EN· MODELS
SOFT: V02.2x

**F.**

FAGOR AUTOMATION

**CNC 8055**
**CNC 8055i**

SOFT: V02.2x

**FAGOR** 

FAGOR AUTOMATION