CNC 8x.00

SIEB & MEYER - Command Set

Description of all SIEB & MEYER commands for software version 12.01.001 and higher (SLM)

D-00000113.2
2021-06-28

Copyright

Trademarks

SIEB & MEYER Worldwide

For questions regarding our products and technical problems please contact us.

SIEB & MEYER AG
Auf dem Schmaarkamp 21
21339 Lueneburg
Germany

Phone: +49 4131 203 0
Fax: +49 4131 203 2000
support@sieb-meyer.de
http://www.sieb-meyer.de

SIEB & MEYER Shenzhen Trading Co. Ltd.
Room A208 2/F,
Internet Innovation and Creation services base Building (2),
No.126, Wanxia road, Shekou, Nanshan district,
Shenzhen City, 518067
P.R. China

Phone: +86 755 2681 1417 / +86 755 2681 2487
Fax: +86 755 2681 2967
sm_china_support1@163.com
http://www.sieb-meyer.cn

SIEB & MEYER Asia Co. Ltd.
4 Fl, No. 532, Sec. 1
Min-Sheng N. Road
Kwei-Shan Hsiang
333 Tao-Yuan Hsien
Taiwan

Phone: +886 3 311 5560
Fax: +886 3 322 1224
smasia@ms42.hinet.net
http://www.sieb-meyer.com

1 General Information

This manual describes all commands that can be included in a SIEB & MEYER part program (Format 5000).

Note

You can download more documentation from the SIEB & MEYER website under http://www.sieb-meyer.de/downloads.html.

Structure of the manual

The manual is divided into subjects. Every subject is introduced by one or several chapters with useful tips and information. Consider that the order in which the commands are described in a chapter is alphabetical, and does not comply to the programming order of the commands in a part program.

The chapter B “SIEB & MEYER Formats”, page 165 includes an overview of all SIEB & MEYER formats.

The chapter A “File Format”, page 155 includes information about the structure of a part program.

Structure of a command description

Any command description includes the following elements:

- The format table shows whether or not the command is available in a format.
  - Minus sign (–): The command is not available in the format.
  - Filled circle (●): The command is available in the format.
  - Resolution/unit (e.g. X1 = 0.001 mm): The command is available in the format. This information also shows the resolution and the unit of measurement to be used for programming the command in the format.
- The following tables inlcude the command syntax and command parameters (also refer to section "Command syntax" below).
- Description: The describing text explains how a command works. Furthermore, possibly required machine equipment and CNC settings are described. In many cases programming examples are added, too.
- Related topics: This section lists related commands or topics of the described command.

Command set

Commands allowed in a SIEB & MEYER part program are referred to as SIEB & MEYER commands. Whenever the term "command" appears in this manual, it is a SIEB & MEYER command. Other commands are explicitly named (e.g. CNC command, sequence command, Excellon command etc.).

Command syntax

You find programming regulations for every individual command.

- Capital letters are part of the SIEB & MEYER command set.
- Small and *italic* letters must be replaced by numbers or text (X*x*Y*y* stands for any XY-coordinates: e.g. X123.456Y234.567, M97,*text* stands for an optional string).

Working method

SIEB & MEYERcommands are effective from the line in which they are programmed.

- Even if a command follows the XY-coordinates in the program, the function already affects the coordinates.
  Example: X..Y..T2 means that the tool is changed to T2 first and then the XY-position is drilled.
- Most of the commands remain valid until they are replaced by another command or until the are deactivated.
  Example: X..Y..G1 means that the G1 function "Rout straight line" is active until another routing command or a T0 block is programmed.
- If contradictory commands are programmed in a program line, the CNC considers the last command of the program line.

Format settings

Length-dependent values depend on the active format.

- The unit and the resolution of the value must be clearly defined in order to interpret a length-dependent value correctly (e.g. XY-coordinate). This is done with the format setting (e.g. CNC command FP).

Note

Unless otherwise defined the programming regulations for the Format 5000 apply for all format-dependent values described in this manual.

For information about interpretation of other formats refer to the manual “CNC 8x.00 − CNC Commands” (s. section “Format Settings”).

Coordinate values

Coordinate values can have two meanings.

- The coordinate includes a working task ("Drill here!" or "Start/stop routing at this position!").
- The coordinate includes an addend. If a program line includes an offset command (M50..M90), the coordinate serves as addend for all coordinates of the previous program section.

`X12.345Y23.456`	Working coordinate (drilling, routing, measuring, pinning etc.)
`X150.Y100.M50`	Offset coordinate During the execution these coordinates are added to each drilling/routing coordinate of the previous program section. The M-command defines a possible rotation or mirroring.

2 Drilling

Generally, coordinate values without additional parameters are interpreted as drilling positions.

- Exception: The coordinate values are programmed within a routing contour (T0-brackets and activated G-function).
- The coordinate values can be programmed as absolute values (G90) or as incremental values.
- A required tool change is defined at the end of the program line.
- The coordinates of the program line are already drilled with the new tool.

`X10.Y10.`	Drill hole at the coordinate X10.Y10.
`X10.Y10.T2`	Tool change to T2; then a hole is drilled at the coordinate X10.Y10.

Depth control

To drill blind holes the working plane can be modified in the part program. Depending from the machine equipment the working plane may be referenced to the table surface or to the board surface.

Peck drilling

The peck drill function divides a drill hole into several partial strokes. The working method of the function depends on the configuration of the CNC (depending on the tool or programmed in the part program).

Cutting parameters

For the cutting parameters required for tools (plunge rate, retract feedrate, speed, max. tool life) please consider the documents of the tool manufacturer. The values are entered into the tool table.

Tool life monitoring

The maximum admissible tool life is defined in the tool table.

- When the maximum admissible tool life is reached, the tool is automatically exchanged.
- The current tool life can also be adapted in the tool table.
- Furthermore, factors for the tool life counting can be defined for different materials and for peck drilling.

Tool measurement

Depending on the machine equipment and configuration the tool is measured after it has been picked up.

- The lenght (absolute and relative) is measured in one measuring cycle.
  - Absolute length = deviation of the actual tool length from an ideally clamped tool
  - Relative length = distance from the lower edge of the pressure foot to the tool tip
- The diameter and runout can be determined in a further measuring cycle:
  - Diameter: The measured diameter serves for monitoring whether the correct tool has been picked up.
  - Runout: Too strong oscillations of a tool may indicate a dirty collet.

2.3 Nibbling

This section describes commands used for programming nibbling processes in the part program.

G84: Nibble Circle

1000	3000	5000
–	•	•

Command	Description
X*x*Y*y* G84 R*r*	Nibbling circle with radius definition

Argument	Description
xy	Center of the circle
r	Radius of the circle (only positive values are allowed) ▸ minimum = 0.500 mm ▸ minimum = 0.0197 inch

With the program command G84 a circle is nibbled at a defined XY-coordinate with the given radius *r*.

In combination with a contact drilling unit, the board surface is only detected during the first nibbling drill stroke. This value applies for the complete nibbling track.

Consider also the following CNC commands in connection to the nibbling function (refer to the manual CNC 8x.00 – CNC Commands for all CNC commands):

- CNC command NONIBO: alternating nibbling
- CNC command NIBO: sequential nibbling
- CNC command BROK2: broken drill detection is off during nibbling

CNC command NONIBO

Alternating nibbling (default).

- The edge roughness is defined with the CNC command NIB.
- The distance between the holes depends on the tool diameter.
- Since the tool diameter is taken into consideration, the hole diameter can be manipulated by changing the diameter value in the tool table.
- If a smaller tool diameter is defined, a bigger hole is produced during the execution and, vice versa.

Nibbling is often used to improve the quality of small drilled holes. The quality of the nibbling edge is set via the edge roughness.

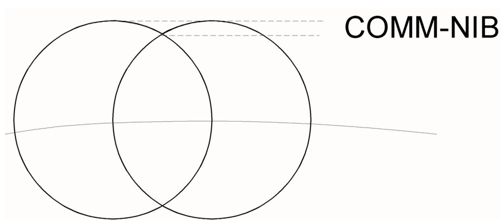


Fig. 1: Edge roughness

CNC command NIBO

Sequential nibbling

- The edge roughness is 13 µm.
- The holes are drilled with the constant distance of 0.381 mm.

- To achieve a satisfying drilling result, the tool must have a diameter of approximately 2.4 mm.

CNC command BROK2

Broken drill detection is off during nibbling and peck drilling.

Example

A hole with a diameter of 10.0 mm shall be nibbled. Ensure that the tool diameter is not larger than the arc radius.

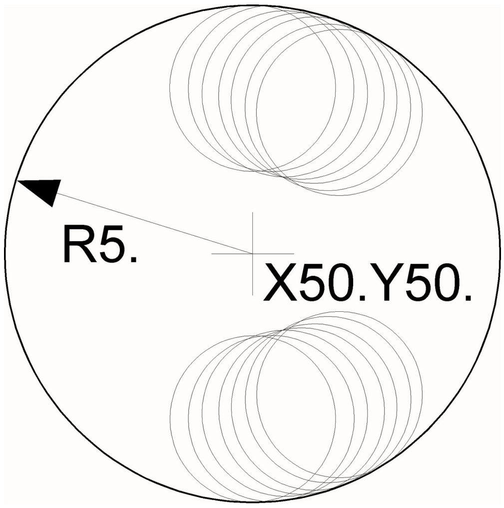


Fig. 2: Nibbled circle

Marginal condition	Setting
Format	5000
Axis version	1

`X50.Y50.G84R5.`	Center and radius of the circle

Related topics

G85: Nibble Slot, page 10

G85: Nibble Slot

1000	3000	5000
–	•	•

Command	Description
*X*x1Y*y1* G85	Nibble slot
*X*x2Y*y2*	

Argument	Description
x1y1	Start point of the slot
x2y2	End point of the slot

Use command G85 to nibble a slot at a defined XY-coordinate. Nibbling means that holes are drilled side-by-side to form a slot.

In combination with a contact drilling unit, the board surface is only detected during the first nibbling drill stroke. This value applies for the complete nibbling track.

Note

The programmed slot length is always one tool diameter shorter than the wanted slot length.

Consider the following CNC commands in connection with the nibbling function (see also the manual CNC 8x.00 - CNC Commands):

- CNC command NONIBO: alternating nibbling
- CNC command NIBO: sequential nibbling
- CNC command BROK2: broken drill detection is off during nibbling

CNC command NONIBO

Alternating nibbling (default).

- The edge roughness is defined with the CNC command NIB.
- The distance between the holes depends on the tool diameter.

The quality of the nibbling edge is set via the edge roughness.

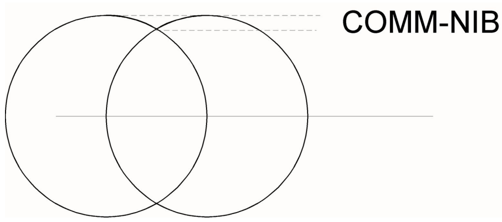


Fig. 3: Edge roughness

CNC command NIBO

Sequential nibbling

- The edge roughness is 13 µm.
- The holes are drilled with the constant distance of 0.381 mm.
- To achieve a satisfying drilling result, the tool must have a diameter of approximately 2.4 mm.

CNC command BROK2

Broken drill detection is off during nibbling and peck drilling.

Calculating the slot length

Length of slot = distance of the programmed points + diameter of drill bit

Example

Requirement

When using a thin tool for nibbling, ensure that alternating nibbling is active.

Depending on the tool diameter the CNC detects the required number of drill strokes. The alternating drilling process (neutral material crowding) allows also the use of a relatively thin tool.

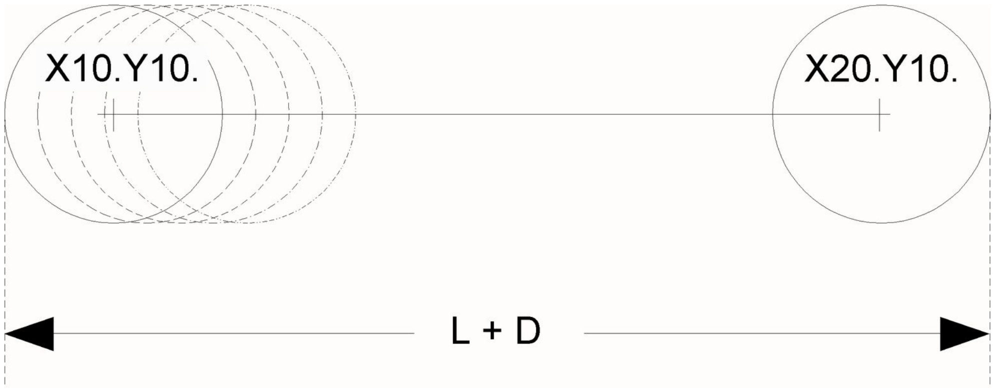


Fig. 4: Nibbled slot

Marginal condition	Setting
Format	5000
Axis version	1

`X10.Y10.G85`	Start point of the slot
`X20.Y10.`	End of the slot (slot length = 10 mm + 2.4 mm = 12.4 mm)

Related topics

G84: Nibble Circle, page 9

2.4 Plain Text Drilling

This section describes commands used for programming a text to be drilled in the part program.

M97: Drill Plain Text in Parallel to the X-axis

1000	3000	5000
–	•	•

Command	Description
X*x*Y*y* M97,*text*	Drill plain text in X-direction

Argument	Description
xy	Reference coordinate for the 1st drill hole
text	Any text

Use the command M97 to program a plain text which is drilled legibly in parallel to the X-axis (applies to axis version 1). Legible in parallel to the X-axis means that the text can be read from the left to the right.

Programming instructions:

- The command, the plain text to be drilled and the corresponding coordinates are programmed in one block.
- A comma must be programmed between the command and the plain text. The comma (,) is not drilled.
- Required tool changes or bracket commands must be programmed before the plain text command.

- The character set is selected in the CNC settings.
- Plain text can contain text variables (see table).
- If the programming station and the production machine have different axis versions, it might be necessary to define settings on the production machine to ensure that the plain text is drilled correctly (it might for example be necessary to define the axis version for the plain text). For detailed information refer to the description of the CNC setting options.

Text variables

You can define text variables in the plain text.

- During execution these text variables will be replaced by current values (e.g. user text, spindle number, run number etc.).
- Requirement for the options with &: Drilling a plain text requires a character set with a fixed character width.

Function	Explanation
*	Alternative text defined with the OPID setting (valid for all character sets)
&D	Date (dd//mm/yyyy)
&T	Time (hh:mm:ss)
&F	File name
&E	File name extension
&M	Module number
&P[1]	(Software version 10.05 and higher) Spindle number as row of holes Spindle 1 = 1 hole, spindle 2 = 2 holes etc.
&S[1]	Spindle number as number
&B	Tool breakage file counter
&N	Run number as decimal number (in the camera mode of the CAMN value)
&R	Drill pattern number as decimal number
&V	Run number as binary row of holes (in the camera mode of the CAMN value)
[1] Since the individual spindles can not be separated in a machine with center drive of the Z-axes drilling of the spindle number is ignored.	

Character repertoire

The character repertoire depends on the selected character set.

Character(s)	3 x 5	4 x 6	5 x 7	Excellon	Hitachi
A..Z	●	●	●	●	●
0..9	●	●	●	●	●
+−/	–	●	●	●	only "−"
.,!?:()"	–	●	●	–	–
#$ %&'*;<=>@[\ ]^ _	-	-	●	-	-

Character set with fixed character width (5 x 7 drill holes)

The figure shows an example for a character repertoire of the character set with a fixed width (5 x 7 drill holes).

Fig. 5: Character set

Structure of a character (5 x 7 drill holes)

Usually, the first drill hole of the first character in the plain text is used for positioning the plain text. See black point 1,7 in the figure.

The character width is 5 drill holes plus the width of one drill hole as free space to the next character.

- The plain text letters are defined within a 6×7 matrix.
- The reference point is always the grid point GP(1.7).
- The size of the character depends on the tool diameter:
  - height of the character = 8.2 x (diameter of the drill bit x)
  - character width = 5.8 x diameter of the drill bit
- The empty space between the characters corresponds to 0.2 diameters of the drill bit.

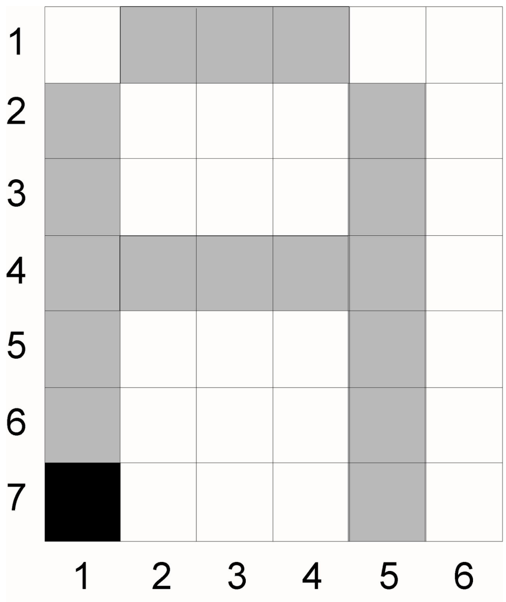


Fig. 6: Character matrix

Example: Plain text

A character set with a fixed character width is selected for plain text drilling (5 x 7 drill holes). The following plain texts are to be drilled:

- The plain text "CNC" is to be drilled in parallel to the X-axis. The character height must be 8.2 mm.
- The plain text "CNC" is to be drilled in parallel to the Y-axis. The character height must be 16.4 mm.

Marginal condition	Setting
Format	5000
Axis version	1

Calculation of the required tool diameters:

Tool	Diameter
T4	8.2 mm / 8.2 = 1.00 mm tool diameter
T5	16.4 mm / 8.2 = 2.00 mm tool diameter

Programming in the part program.

`X30.Y10.T4 M97,CNC`	Dill plain text in parallel to the X-axis
`X30.Y30.T5 M98,CNC`	Drill plain text in parallel to the Y-axis

Plain text can be drilled in parallel to the X-axis or to the Y-axis.

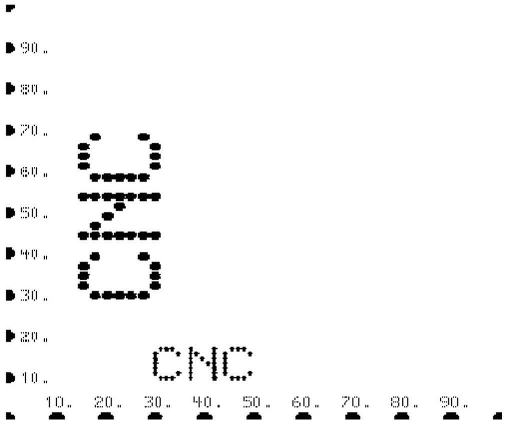


Fig. 7: Drilled plain text

Example: Pattern labeling

To allow clear identification, a specific code shall be drilled in every pattern.

- A drill bit with a diameter of 0.8 mm shall be used for drilling the pattern identification (tool number T7). Since no identical diameter is used in the part program, a separate section is programmed for the pattern code.
- Structure of the number:
  - Component code: identification of the part program ("A2341")
  - Run number: identification of the board code (variable "&N")
  - Pattern number: identification of the patterns (variable "&R")

Marginal condition	Setting
Format	5000
Axis version	1

`X31.Y28.T1M31`	1st drill hole of the tool number T1
`...`	...
`X220.Y40.M50`	1st pattern zero
`X220.Y140.M50`	2nd pattern zero
`X...Y...M50M30`	last pattern zero
`X42.Y13.T2M31`	1st drill hole of the tool number T2
`...`	...
`X220.Y40.M50`	1st pattern zero
`X220.Y140.M50`	2nd pattern zero
`X...Y...M50M30`	Last pattern zero
`( PATTERN LABEL )`	Drill tool change and code

`X30.Y10.T7M31M97,A2341-&N-&R`	"A2341−1−1" to "A2341−1−3"
`X220.Y40.M50`	1st pattern zero
`X220.Y140.M50`	2nd pattern zero
`X...Y...M50M30`	Last pattern zero

Drilled codes

The table lists the drilled plain texts for 2 runs of a part program "A2341" with 3 patterns:

Run	Pattern	Label
1	1	A2341−1−1
1	2	A2341−1−2
1	3	A2341−1−3
2	1	A2341−2−1
2	2	A2341−2−2
2	3	A2341−2−3

Related topics

M98: Drill Plain Text in Parallel to the Y-axis, page 16

M98: Drill Plain Text in Parallel to the Y-axis

1000	3000	5000
–	•	•

Command	Description
XxYy M98,*text*	Drill plain text in Y-direction

Argument	Description
xy	Reference coordinate for the 1st drill hole
text	Any text

Use the command M98 to program a plain text which is drilled legibly in parallel to the Y-axis (applies to axis version 1). Legible to the Y-axis means here that the text can be read from the front to the back when turned to the left.

Programming instructions:

- The command, the plain text to be drilled and the corresponding coordinates are programmed in one block.
- A comma must be programmed between the command and the plain text. The comma (,) is not drilled.
- Required tool changes or bracket commands must be programmed before the plain text command.
- The character set is selected in the CNC settings.
- Plain text can contain text variables (see table).
- If the programming station and the production machine have different axis versions, it might be necessary to define settings on the production machine to ensure that the plain text is drilled correctly (it might for example be necessary to define the axis version for the plain text). For detailed information refer to the description of the CNC setting options.

Text variables

You can define text variables in the plain text.

- During execution these text variables will be replaced by current values (e.g. user text, spindle number, run number etc.).
- Requirement for the options with &: Drilling a plain text requires a character set with a fixed character width.

Function	Explanation
*	Alternative text defined with the OPID setting (valid for all character sets)
&D	Date (dd//mm/yyyy)
&T	Time (hh:mm:ss)
&F	File name
&E	File name extension
&M	Module number
&P[1]	(Software version 10.05 and higher) Spindle number as row of holes Spindle 1 = 1 hole, spindle 2 = 2 holes etc.
&S[1]	Spindle number as number
&B	Tool breakage file counter
&N	Run number as decimal number (in the camera mode of the CAMN value)
&R	Drill pattern number as decimal number
&V	Run number as binary row of holes (in the camera mode of the CAMN value)

[1] Since the individual spindles cannot be separated in a machine with center drive of the Z-axes drilling of the spindle number is ignored.

Character repertoire

The character repertoire depends on the selected character set.

Character(s)	3 x 5	4 x 6	5 x 7	Excellon	Hitachi
A..Z	●	●	●	●	●
0..9	●	●	●	●	●
+−/	–	●	●	●	only "−"
.,!?:()"	–	●	●	–	–
#$ %&'*;<=>@[\ ]^ _	-	-	●	-	-

Character set with fixed character width (5 x 7 drill holes)

The figure shows an example for a character repertoire of the character set with a fixed width (5 x 7 drill holes).

Fig. 8: Character set

Structure of a character (5 x 7 drill holes)

Usually, the first drill hole of the first character in the plain text is used for positioning the plain text. See black point 1,7 in the figure.

The character width is 5 drill holes plus the width of one drill hole as free space to the next character.

- The plain text letters are defined within a 6×7 matrix.
- The reference point is always the grid point GP(1.7).
- The size of the character depends on the tool diameter:
  - height of the character = 7 x (diameter of the drill bit x 1.2)
  - width of the character = 6 x (diameter of the drill bit x 1.2)

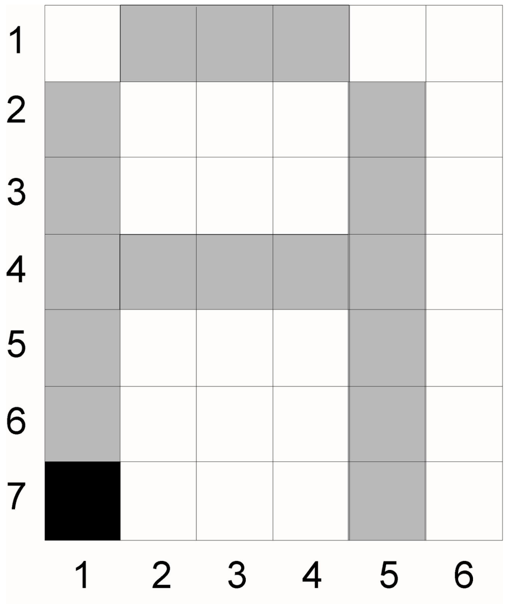


Fig. 9: Character matrix

Example: Plain text

A character set with a fixed character width is selected for plain text drilling (5 x 7 drill holes). The following plain texts are to be drilled:

- The plain text "CNC" is to be drilled in parallel to the X-axis. The character height must be 8.2 mm.
- The plain text "CNC" is to be drilled in parallel to the Y-axis. The character height must be 16.4 mm.

Marginal condition	Setting
Format	5000
Axis version	1

Calculation of the required tool diameters:

Tool	Diameter
T4	8.2 mm / 8.2 = 1.00 mm tool diameter
T5	16.4 mm / 8.2 = 2.00 mm tool diameter

Programming in the part program.

`X30.Y10.T4 M97,CNC`	Dill plain text in parallel to the X-axis
`X30.Y30.T5 M98,CNC`	Drill plain text in parallel to the Y-axis

Plain text can be drilled in parallel to the X-axis or to the Y-axis.

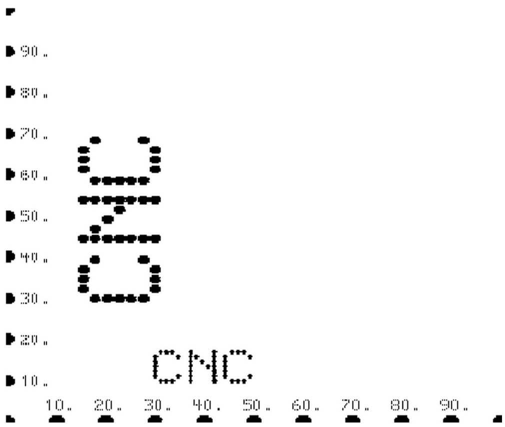


Fig. 10: Drilled plain text

Example: Pattern labeling

To allow clear identification, a specific code shall be drilled in every pattern.

- A drill bit with a diameter of 0.8 mm shall be used for drilling the pattern identification (tool number T7). Since no identical diameter is used in the part program, a separate section is programmed for the pattern code.
- Structure of the number:
  - Component code: identification of the part program ("A2341")
  - Run number: identification of the board code (variable "&N")
  - Pattern number: identification of the patterns (variable "&R")

Marginal condition	Setting
Format	5000
Axis version	1

`X31.Y28.T1M31`	1st drill hole of the tool number T1
`...`	...
`X220.Y40.M50`	1st pattern zero
`X220.Y140.M50`	2nd pattern zero
`X...Y...M50M30`	last pattern zero
`X42.Y13.T2M31`	1st drill hole of the tool number T2
`...`	...
`X220.Y40.M50`	1st pattern zero
`X220.Y140.M50`	2nd pattern zero
`X...Y...M50M30`	Last pattern zero
`( PATTERN LABEL )`	Drill tool change and code
`X30.Y10.T7M31M97,A2341-&N-&R`	"A2341−1−1" to "A2341−1−3"
`X220.Y40.M50`	1st pattern zero
`X220.Y140.M50`	2nd pattern zero
`X...Y...M50M30`	Last pattern zero

Drilled codes

The table lists the drilled plain texts for 2 runs of a part program "A2341" with 3 patterns:

Run	Pattern	Label
1	1	A2341−1−1
1	2	A2341−1−2
1	3	A2341−1−3
2	1	A2341−2−1
2	2	A2341−2−2
2	3	A2341−2−3

Related topics

M97: Drill Plain Text in Parallel to the X-axis, page 12

2.5 Drilled Pattern

This section described commands used for programming complete patterns in the part program (rows of holes, circle based on drill holes etc.).

V1: Drill Dual Row of Holes (Dual-in-line)

1000	3000	5000
•	•	•

Command	Description
X*x1*Y*y1* V1	Dual row of holes: start point
X*x2*Y*y2*	End point (grid = 2.54 mm = 0.1 inch)
X*x2*Y*y2*W*w*	End point (any grid)

Argument	Description
x1y1	1st corner point
x2y2	Diagonal corner point
w	Number of drill holes in a row

The V1 command drills two parallel rows of holes (e.g. for *dual-in-line* components of any size).

- The rows of holes must be axially parallel (otherwise use the V2 command).
- Entering the W value is only required if the standard grid of 2.54 (= 0.1 inch) is to be changed.
- Two diagonal corner points must be programmed.
- If the position of the second corner point does not coincide with the expected grid, the distance of the rows will automatically be corrected.

Position of the rows of holes

The position of the rows of holes depends on two criteria:

- Axis version of the machine
- Diagonals of the rows of holes to be programmed (see figure)
- The order of the programmed corner points is not prescribed.

The position of the drill patterns depends on the set axis version.

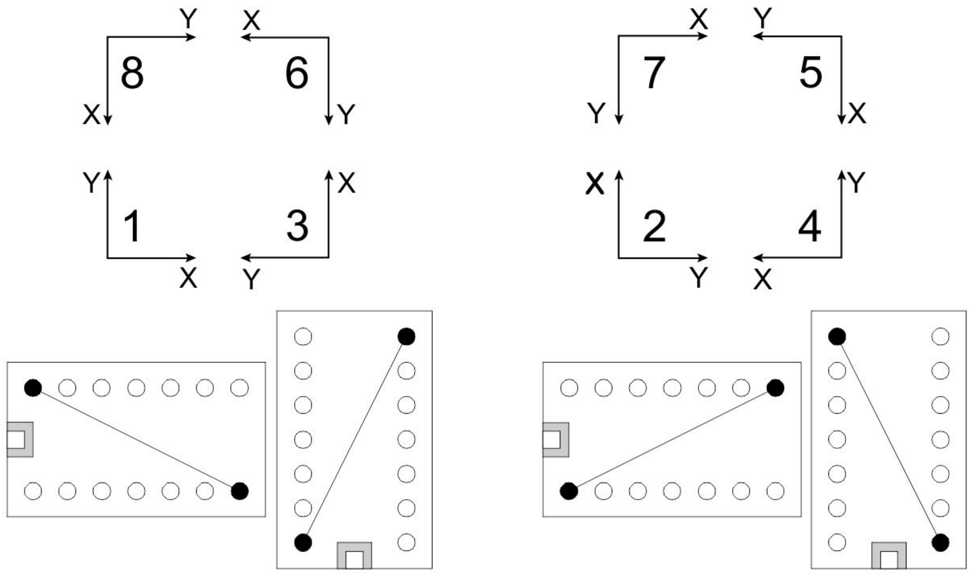


Fig. 11: Position of the rows of holes

Automatic correction of the distance between the rows in the standard grid

In the standard grid, the hole positions are aligned at the inch 0.1"-grid (= 2.54 mm).

The distance between the rows is corrected according to the following criteria (Standard grid = 2.54 mm (=0.1 inch)):

- The diagonal corner point must be programmed with an accuracy of ±1.27 mm.
- Within this deviation the CNC corrects position during the execution.

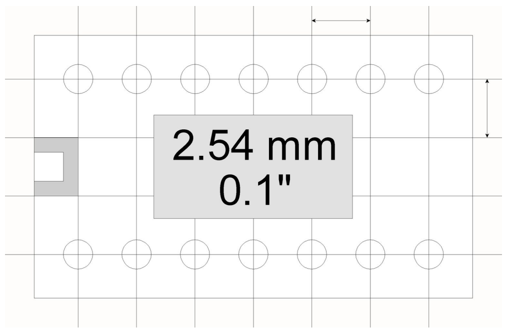


Fig. 12: Standard grid

Correction of the distance between the rows in the freely definable grid

In the free grid the distance between the holes is calculated from the distance of the programmed coordinates.

Programming and calculation of a freely definable grid:

- In the second program line the parameter W defines the number of holes of a row.
- The CNC first calculates the hole center distance (grid) on the basis of the component's position.
- The calculated grid is taken over for the distances of both rows.
- If necessary, the programmed value is executed with corrections.

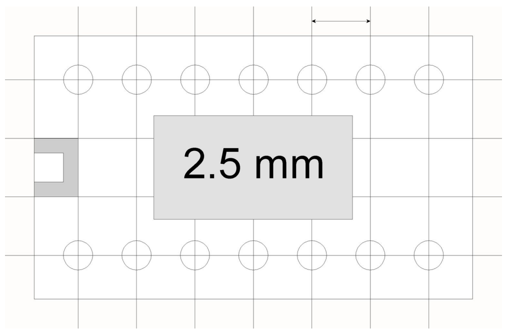


Fig. 13: Free grid

Example (axis version 1)

Marginal condition	Setting
Format	5000
Axis version	1

`( STANDARD GRID )`	Row of holes in the standard grid
`X30.00Y70.00 V1`	Start point and V1 command
`X45.24Y62.38`	Diagonal corner point. If necessary, the programmed position is corrected automatically
`( VARIOUS GRID )`	Row of holes in the freely definable grid
`X70.00Y30.00 V1`	Start point and V1 command
`X77.50Y45.00 W7`	Diagonal corner point. The grid dimension results from the number of holes.

Upper hole pattern: The hole center distance is 2.54 mm (= 0.1 inch). The first programmed drill hole is used as reference point.

Lower hole pattern: The hole center distance is 2.5 mm. Both coordinate values must be programmed exactly.

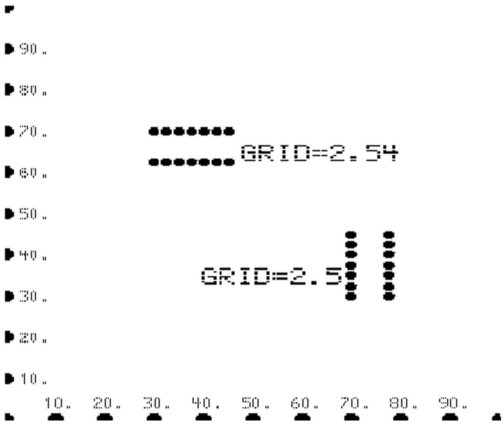


Fig. 14: 2-line drilled rows of holes

Related topics

[V2: Drill Single Row of Holes, page 22]

[V3: Drill Quadruple Row of Holes, page 23]

[V4: Drill Circular Row of Holes, page 25]

V2: Drill Single Row of Holes

1000	3000	5000
•	•	•

Command	Description
X*x1*Y*y1* V2	Single row of holes: start point
X*x2*Y*y2* W*w*	End point and number of holes

Argument	Description
x1y1	Start point of the row of holes
x2y2	End point of the row of holes
w	Number of drill holes in a row

The V2 command allows drilling a row of holes with any angle.

- The parameter W defines the number of holes of a row, including the first and the last hole.

- The holes are drilled with identical distances.

Calculation of the hole center distance

Distance = (P2 − P1) / (W − 1)

- P1: coordinates of the first hole
- P2: coordinates of the last hole
- W: number of holes in the row of holes

Example

A row of holes of 12 holes shall be drilled in X-direction. The distance between the holes shall be 2.54 mm. The first hole is at the position X10.Y10.

The final position can be calculated with the modified formula.

P2 = P1 + (A x (W − 1)) = X10. + (2.54 x 11) = X37.94

Marginal condition	Setting
Format	5000
Axis version	1

`X10.Y10.V2`	Position of the first drill hole
`X37.94.Y10.W12`	Position of the last hole and number of holes

Related topics

V1: Drill Dual Row of Holes (Dual-in-line), page 20

V3: Drill Quadruple Row of Holes, page 23

V4: Drill Circular Row of Holes, page 25

V3: Drill Quadruple Row of Holes

1000	3000	5000
–	–	•

Command	Description
X*x1*Y*y1* V3	Quadruple row of holes: start point
X*x2*Y*y2*	End point (grid = 2.54 mm = 0.1 inch)
X*x2*Y*y2*W*w*	End point (any grid)

Argument	Description
x1y1	Start point of the row of holes
x2y2	End point of the row of holes
w	Number of drill holes in a row

The V3 command allows drilling four rows of holes for components that need rectangular (square) structured rows of holes (e. g. for PLCD, KLCD, etc.).

- The rows of holes must be axially parallel (otherwise use the V2 command).
- Entering a W value is only required if the standard grid of 2.54 (0.1 inch) is to be changed.
- Two diagonal corner points must be programmed.
- The CNC calculates the exact positions of the holes during the execution.

- If the position of the second corner point does not coincide with the expected grid, the distance of the rows will automatically be corrected.

Position of the rows of holes

The position of the rows of holes depends on two criteria:

- Axis version of the machine
- Diagonals of the rows of holes to be programmed (see figure)
- The order of the programmed corner points is not prescribed.

The position of the drill patterns depends on the set axis version.

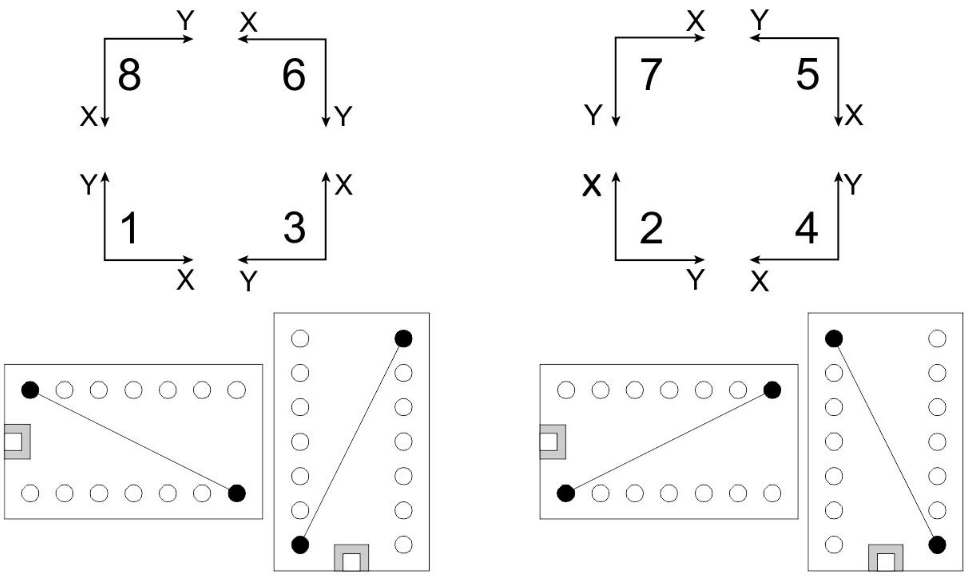


Fig. 15: Position of the rows of holes

Automatic correction of the distance between the rows in the standard grid

In the standard grid, the hole positions are aligned at the inch 0.1"-grid (= 2.54 mm).

The distance between the rows is corrected according to the following criteria (Standard grid = 2.54 mm (=0.1 inch)):

- The diagonal corner point must be programmed with an accuracy of ±1.27 mm.
- Within this deviation the CNC corrects position during the execution.
- Any diagonal can be chosen for the programming of four rows of holes (square or rectangle) in the standard grid.
- If the diagonal point is programmed incorrectly (±1.27 mm = 0.05"), the CNC will position the hole on the next grid point.

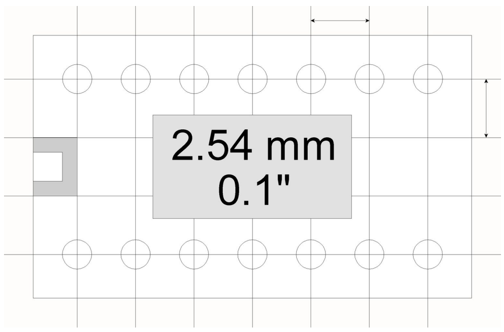


Fig. 16: Standard grid

Correction of the distance between the rows in the freely definable grid

In the free grid the distance between the holes is calculated from the distance of the programmed coordinates.

Programming and calculation of a freely definable grid:

- For the programming of the freely definable grid the CNC first calculates the grid spacing for one direction on the basis of the holes for this direction.
- If necessary, the distance between the two rows is corrected with this grid spacing.

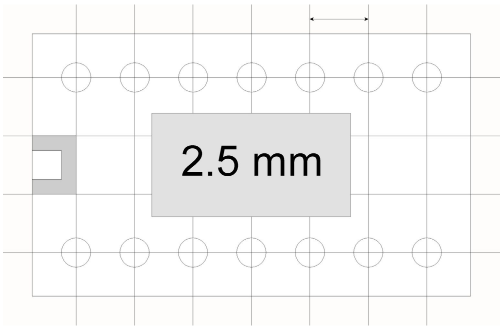


Fig. 17: Free grid

Examples (axis version 1)

`X30.Y10.V3`	Square in standard grid
`X50.Y30.`	Side length = 20.0 mm, grid spacing = 2.54 mm
`X70.Y10.V3`	Rectangle in the standard grid
`X80.Y40.`	X-side length = 10.0 mm, Y-side length = 30.0 mm, grid spacing = 2.54 mm
`X30.Y50.V3`	Square in the freely definable grid
`X50.Y70.W9`	Side length = 20.0 mm, grid spacing = 2.5 mm
`X70.Y50.V3`	Rectangle in the freely definable grid
`X80.Y90.W9`	X-side length = 10.0 mm, Y-side length = 40.0 mm, grid spacing = 2.5 mm

Upper hole pattern: The hole center distance is 2.5 mm. All coordinate values must be programmed exactly.

Lower hole pattern: The hole center distance is 2.54 mm (= 0.1 inch). The first programmed drill hole of a contour is used as reference point.

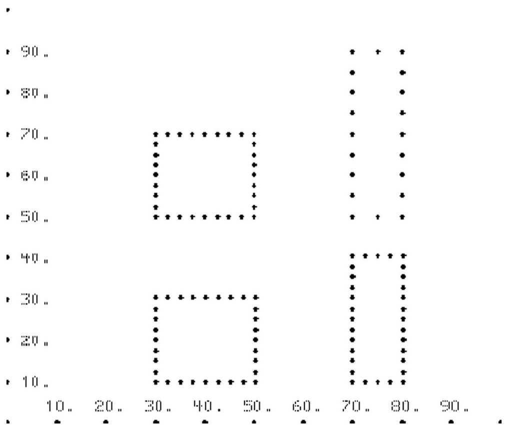


Fig. 18: 4-line drilled rows of holes

Related topics

V1: Drill Dual Row of Holes (Dual-in-line), page 20

V2: Drill Single Row of Holes, page 22

V4: Drill Circular Row of Holes, page 25

V4: Drill Circular Row of Holes

1000	3000	5000
–	–	•

Command	Description
*X*x1*Y*y1 V4	Circular row of holes: center
*X*x2*Y*y2 *W*w	Point on circular arc and number of holes

Argument	Description
x1y1	Center of the circle
x2y2	Any hole on the circular arc
w	Number of holes

The V4 command drills holes on an arc.

- The parameter W defines the number of holes of a row, including the first and the last hole.
- The CNC calculates the exact positions of the holes during the execution.
- The coordinates for the holes on the circular arc can also be determined for a defined radius and the angle α.

Calculation of the position on the circular arc

If the angle for a drilling position is determined on the circular arc the second coordinate can be calculated by means of the angle function.

- *x2* = r x sin(α)
- *y2* = r x cos(α)
- *r* = radius of the circle

Example: program circular row of holes

Marginal condition	Setting
Format	5000
Axis version	1

`X50.Y50.V4T1`	Center of the circle
`X80.Y50.W12`	Any hole on the circular arc and 12 holes
`X50.Y50.V4T2`	Center of the circle
`X82.Y50.W60`	Any hole on the circular arc and 60 holes

Any number of holes can be arranged in a circle with two program lines.

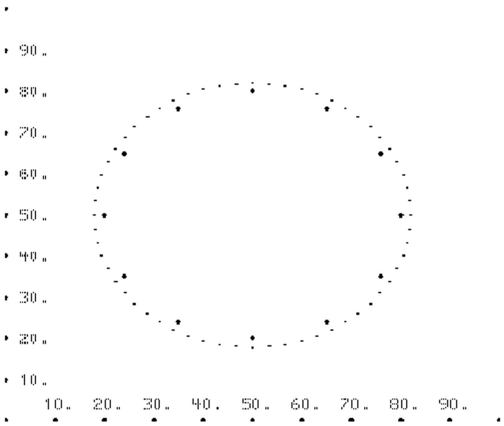


Fig. 19: Circular drilled rows of holes

Related topics

[V1: Drill Dual Row of Holes (Dual-in-line), page 20]

V2: Drill Single Row of Holes, page 22

V3: Drill Quadruple Row of Holes, page 23

2.6 SIEB & MEYER Commands for Peck Drilling

This chapter lists all SIEB & MEYER commands for peck drilling.

Activate and deactivate peck drilling

Peck drilling in the part program

The peck drill function is activated and deactivated via the SIEB & MEYER commands G81 and G80 in the part program.

Peck drilling with tool table

With the following CNC command you can define whether the values from the tool table "Peck drilling" or the values from the part program are used:

CNC command	Meaning
SPEK	Define pecking tools and parameters from the tool table
NOSPEK	Use peck drilling parameters from the part program.

Peck drilling with CNC commands

Use the following CNC commands to activate and define the peck drill function:

CNC command	Meaning
APEK,D*d*I*i*J*j*	Define diameter-dependent pecking and peck drilling parameters
DPEK,D*d*I*i*J*j*	Define diameter-dependent pecking for depth control and peck drilling parameters.
TPEK,I*i*J*j*	Define tool-dependent pecking and peck drilling parameters
PEKM,M	Activate peck drilling mode

Note

The CNC commands APEK, DPEK, TPEK and PEKM,M have priority over the CNC commands SPEK and NOSPEK.

Peck drilling can be activated and deactivated at any time within the part program. Useful applications are, for example:

- Blind holes: The bottom of the hole is not "hollowed out" by the chips because of chip ejection.
- Thin tools: The ejection of chips can prevent tool breakage.
- Thick tools: Improved cooling due to less contact to the material.

Activate and deactivate pulse drilling

Pulse drilling can be activated and defined with the CNC commands APEK, DPEK, TPEK and PEKM,M4. Additionally, pulse drilling can be activated and deactivated in the part program via the SIEB & MEYER commands G89 and G88.

G80: Deactivate Peck Drilling Function

1000	3000	5000
–	–	•

Command	Description
X*x*Y*y* G80	Deactivate peck drilling function

Argument	Description
xy	The coordinate is not peck drilled anymore

The G80 command deactivates the peck drilling function.

- All parameters remain valid.
- The G81 command activates the peck drill function again. The original parameters are used.

Related topics

G81: Activate Peck Drilling, page 28

G81: Activate Peck Drilling

1000	3000	5000
–	–	•

Command	Description
X*x*Y*y* G81	Activate peck drill function with the original parameters
X*x*Y*y* G81I*i*P*p*J*j*W*w*	Define and activate peck drill function

Argument	Description
xy	Peck drilling is active from this coordinate (inclusive)
i	Lowering value and upper turning point of the Z-axis during peck drilling (only positive values are allowed)
p	Percentage value for plunge rate to the lowering value I*i* 1 % to 100 %
j	Height of a partial stroke (only positive values are allowed)
w	Factor for the continuous reduction of the partial strokes 1 ‰ to 1000 ‰

Requirement
The CNC command NOSPEK must be active, so peck drill parameters are considered.

Use the G81 command to activate the peck drill function which means that a hole is drilled by lowering the Z-axis several times (partial strokes). The lowering value I*i* and the height of the partial stroke J*j* must be defined for peck drill holes (see below).

Note
Pulse drilling and peck drilling cannot be used at the same time.

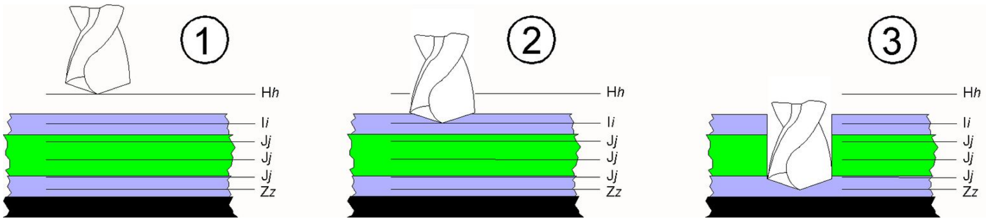


Fig. 20: Planes of a peck drill stroke

1 Traveling plane (H-value or Quick plane). The drill stroke is started in the traveling plane.

2 Lowering value (peck drill parameter I). The lowering value is reached with a reduced infeed rate (peck drill parameter P × tool parameter F).

3 Working plane reached (Z-value or K-value). All partial strokes are finished. The last partial stroke is limited by the working plane.

Lowering value

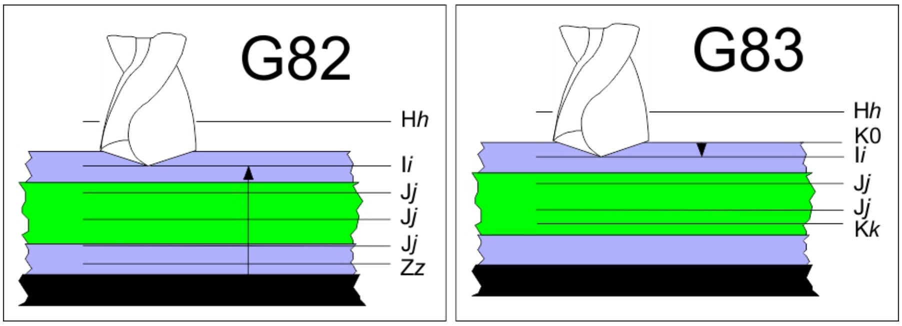


Fig. 21: Reference planes for the lowering value Ii

G82 Reference plane = table surface

G83 Reference plane = board surface

Several parameters are required for the positioning on the lowering value.

- The definition depends on the reference plane (G82 = table surface or G83 = board surface).
- The first drill stroke of a peck drill hole is executed until the lowering value I is reached.
- The infeed rate used to reach the lowering value can be adjusted with the parameter P.
- The lowering value also serves as upper turning point for the following partial strokes.

Parameter	Explanation
I*i*	Lowering value ▸ the value depends on the current reference plane ▸ table surface or board surface
P*p*	Plunge factor ▸ $F_{Lowering}$ = (tool parameter F) x (plunge factor P %)

Partial strokes of a peck drill hole

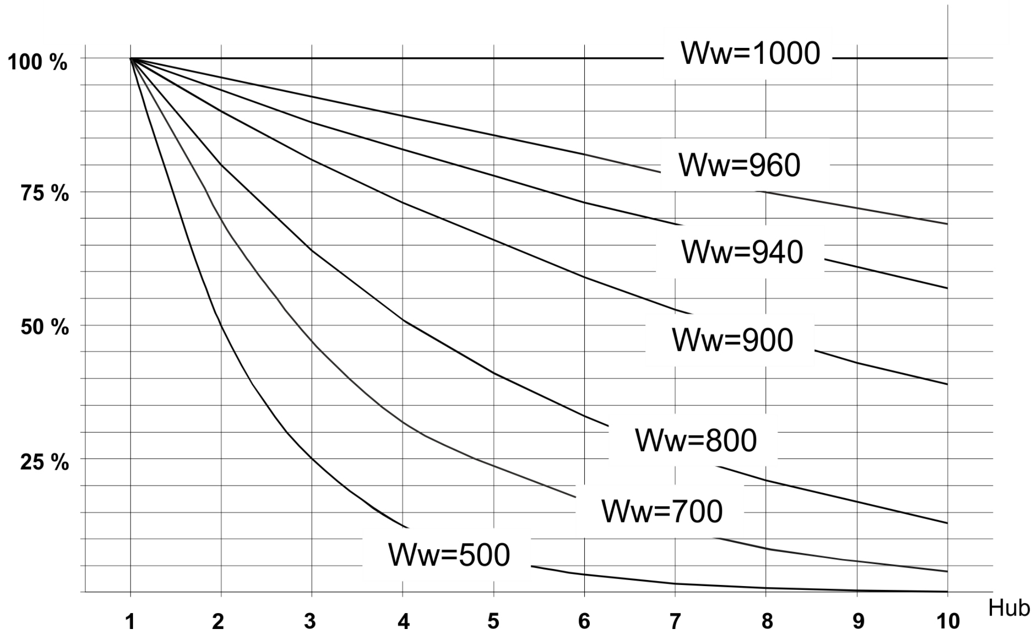


Fig. 22: The heights of the partial strokes of a peck drill hole depends on the W-value.

The height of the partial strokes is reduced depending on the W-value.Several parameters are required for the definition of the partial strokes.

- Partial strokes are strokes required for reboring a hole after predrilling.
- The plunge rate of a partial stroke is defined in the tool parameter F.
- The W-value allows continuous reduction of the partial stroke (e.g. for hard materials). The diagram shows heights of partial strokes in dependance on the number of partial strokes for some peck drilling factors.
- Peck drilling is finished when the target plane is reached. The target plane depends on the reference plane (reference plane = table surface: target plane = Z-plane; reference plane = board surface: target plane = K-plane).
- With CNC command PEKM,D it is possible to force drilling to the programmed target plane Z or K (s. example).

Parameter	Explanation
J*j*	Height of a partial stroke of a peck drill hole
W*w*	Peck drill factor ▸ stroke = $j * (w / 1000)^{n-1}$ ▸ n = partial stroke counter

Example

The heights of the partial strokes for the following program line are listed in the table. The reference plane is the table surface.

X10.Y20. I4.5 P20 J1.2 W500 Z2.

Marginal condition	Setting
Format	5000

Partial stroke	Evaluation	Stroke	reached depth
1	1.2 mm x 0.500^0	1.200 mm	4.500 mm − 1.200 mm = 3.300 mm
2	1.2 mm x 0.500^1	0.600 mm	3.300 mm − 0.600 mm = 2.700 mm

Partial stroke	Evaluation	Stroke	reached depth
3	1.2 mm x 0.500^2	0.300 mm	2.700 mm − 0.300 mm = 2.400 mm
4	1.2 mm x 0.500^3	0.150 mm	2.400 mm − 0.150 mm = 2.250 mm
5	1.2 mm x 0.500^4	0.075 mm	2.250 mm − 0.075 mm = 2.175 mm
6	1.2 mm x 0.500^5	0.037 mm	2.175 mm − 0.037 mm = 2.138 mm
7	1.2 mm x 0.500^6	0.018 mm	2.138 mm − 0.018 mm = 2.120 mm
8	1.2 mm x 0.500^7	0.009 mm	2.120 mm − 0.009 mm = 2.111 mm
9	1.2 mm x 0.500^8	0.004 mm	2.111 mm − 0.004 mm = 2.107 mm
10	1.2 mm x 0.500^9	0.002 mm	2.107 mm − 0.002 mm = 2.105 mm
11	1.2 mm x 0.500^{10}	0.001 mm	2.105 mm − 0.001 mm = 2.104 mm
12	1.2 mm x 0.500^{11}	0.0005 mm	Target plane not reached

Depending on the setting of CNC command PEKM,D the following applies to peck drilling:

- If the CNC command PEKM,-D is active, peck drilling is finished after the 11th partial stroke, the last partial stroke is not performed.
- If the CNC command PEKM,D is active, after the 11th partial stroke one last partial stroke of 0.104 mm is performed to reach the programmed target plane Z.

Process of peck drilling

The process of peck drilling a hole includes the following steps:

- Moving to drilling position.
- Lowering the drill bit with the infeed rate defined for predrilling to the lowering value I.
- The infeed rate for predrilling is the product of the plunge rate (tool parameter F) and the plunge factor (peck drilling parameter P).
- Lowering the drill bit at the standard infeed rate to the first peck drill plane. The default infeed rate is defined in tool parameter F.
- Retracting the drill bit to the upper turning point (s. lowering value I*i*).
- Repeating the last two steps (lowering; retracting) until the target plane is reached (K-plane or Z-plane).
- Retracting the drill bit up to the traveling plane (H-plane or Quick plane).

Example

Marginal condition	Setting
Format	5000

Thickness of the work piece	4.0 mm
Thickness of the backup material	1.5 mm
Reference plane	Table surface (G82)
Working plane	Z1.0
Infeed rate for T2	1.5 m/min (tool parameter F)

The following line has been programmed in the part program:

```
X2.Y7.T2 G81I5.P80 J1.4W1000
```

Parameter	Explanation
I5.	Lowering value, related to the table surface. The first drill stroke is drilled to the lowering value. Since the combined height of the backup material and the work piece is 5.5 mm (4.0 mm + 1.5 mm), the drill bit plunges 0.5 mm into the backup material. For the following partial strokes the lowering value also determines the upper turning point (= 5.0 mm).
P80	Plunge factor. Percentage value of the default infeed rate for lowering. Then, the predrill infeed rate is 1.2 m/min. Infeed rate = 1.5 m/min x 80 % = 1.2 m/min
J1.4	Height of the 1st partial stroke. Each partial stroke is drilled at the plunge rate of 1.5 m/min (tool parameter F).
W1000	Peck drill factor. Since the peck drill factor is defined with 100 % (W = 1000), each partial stroke is 1.4 mm.

The individual partial strokes are calculated as follows:

Partial stroke	Formula	Evaluation	Stroke	Total stroke
1.	$J \times W^0$	$1.4 \text{ mm} \times 1^0$	1.400 mm	1.400 mm
2.	$J \times W^1$	$1.4 \text{ mm} \times 1^1$	1.400 mm	2.800 mm
3.	$J \times W^2$	$1.4 \text{ mm} \times 1^2$	1.400 mm	4.000 mm*

* After every 3rd partial stroke, the drill bit goes below the target plane. During this partial stroke the peck drill hole is finished.

Related topics

G80: Deactivate Peck Drilling Function, page 28

I: Lowering Value of a Peck Drill Hole, page 115

J: Height of Partial Stroke of Peck Drill Hole, page 118

P: Reduce Feed Rate for Peck Drilling, page 118

W: Reduce Height of Partial Strokes for Peck Drill Hole, page 122

2.7 Pulse Drilling

Pulse drilling can be activated and deactivated at any time within the part program. Useful applications are, for example:

- Blind holes. The bottom of the hole is not "hollowed out" by the chips because of chip ejection.
- Thin tools. The ejection of chips can prevent the breakage of drill bits.
- Thick tools. Improved cooling due to less contact to the material.

G88: Deactivate Pulse Drilling

1000	3000	5000
–	–	•

Command	Description
XxYy G88	Deactivate pulse drilling

Argument	Description
xy	The coordinate is drilled as usual

The G88 command deactivates pulse drilling.

- All parameters remain valid.

- The G89 command reactivates the pulse drilling function. The original parameters are used.

Related topics

G89: Activate Pulse Drilling, page 33

G89: Activate Pulse Drilling

1000	3000	5000
–	–	•

Command	Description
X*x*Y*y* G89	Activate pulse drilling with the original parameters
X*x*Y*y* G89I*i*J*j*E*e*R*r*	Define and activate pulse drilling

Argument	Description
xy	Pulse drilling is active from this coordinate (inclusive)
i	Lowering value (only positive values are allowed)
j	Height of a partial stroke (only positive values are allowed)
e	Dwell time in ms
r	Minimum partial stroke of a pulse drilled hole

Requirement

- *The CNC command NOSPEK must be active, so parameters i, j and r are considered.*
- *The parameter e is considered when SPEK or NOSPEK is set.*

The lowering value I*i* and the height of the partial strokes J*j* must be defined for pulse drilling.

Note

Pulse drilling and peck drilling cannot be used at the same time.

Lowering value I*i*)

First, the Z-axis is lowered to the defined lowering value. The pulse process starts from this plane.

- The definition depends on the reference plane (G82 = table surface or G83 = board surface).
- The first drill stroke of pulse drilling is performed to the lowering value I*i*.
- The down-feed rate which applies until the lowering value is reached corresponds to the tool parameter F.
- The dwell time E*e* runs for the first time after the lowering value is reached.

Partial strokes of a pulse drilled hole J*j*

The partial stroke of a pulse drilled hole defines the path by which the Z-axis is lowered during every partial stroke.

- Partial strokes are strokes required for reboring a hole after predrilling.
- The plunge rate of a partial stroke is defined in the tool parameter F.
- The dwell time E*e* runs whenever the next partial plane has been reached.
- Pulse drilling is finished when the target plane is reached. The target plane depends on the reference plane (reference plane = table surface: target plane = Z-plane; reference plane = board surface: target plane = K-plane).

- If the CNC evaluates a partial stroke which is smaller than the remaining path (parameter R), the previous partial stroke will be executed until the target plane is reached.
- The Z-axis is retracted immediately when the working plane is reached.

Minimum partial stroke of a pulse drilled hole R*r*

If the last partial stroke of a pulse drilled hole is below the distance R*r*, the second to last partial stroke will be executed until the traveling plane is reached. This ensures that the partial stroke has a minimum height of R*r*.

Example

Marginal condition	Setting
Format	5000

Thickness of the work piece	4.0 mm
Thickness of the backup material	1.5 mm
Reference plane	Table surface (G82)
Working plane	Z1.8

The following line has been programmed in the part program:

```
X123.456Y789.123T2 G89I5.J1.4 E30 R0.2
```

Parameter	Explanation
I5.	Lowering value, related to the table surface. ▸ The first drill stroke is drilled to the lowering value. ▸ Since the combined height of the backup material and the work piece is 5.5 mm (4.0 mm + 1.5 mm), the drill bit plunges 0.5 mm into the work piece. ▸ The infeed rate is defined with tool parameter F. ▸ After the lowering value has been reached the Z-axis dwells in this position for 30 ms.
J1.4	Height of the partial strokes of a pulse drilled hole: ▸ Every partial stroke of a pulse drilled hole has a height of 1.4 mm. ▸ The infeed rate is defined with tool parameter F. ▸ The Z-axis dwells in this position for 30 ms after every partial stroke.
E30	The dwell time is expired: ▸ after the lowering value has been reached ▸ after a partial stroke has been finished (except for the last partial stroke)
R0.2	If the last partial stroke has a height of less than 0.2 mm, the second to last partial stroke will be executed to the traveling plane.

Related topics

G88: Deactivate Pulse Drilling, page 32

2.8 Check Holes

This section describes commands for programming the check hole area.

M56: Define Check Area in X-direction

1000	3000	5000
–	–	•

Command	Description
X*x*Y*y*M56	Define check area in X-direction

Argument	Description
xy	Start coordinates of the check area

Requirement

The CNC commands CHEK,E and PRGM,C are active.

Dependent on the machine configuration the M56 and M57 commands define the start coordinates and the drilling direction of the area for check holes.

- The check area must have been defined before (for example in the startup-file).
- The coordinates programmed in M56/M57 are the start coordinates of an area. These coordinates replace the start coordinates that have been defined before.

Requirement for the SIEB & MEYER format:

Marginal condition	Setting
Format	5000
Axis version	1

Example

The following example shows the definition of two check areas.

Parameter	Area 1	Area 2
Coordinate of the first drill hole	X10.Y20.	X450.Y20.
Distance in X-direction	2.0	0.0
Distance in Y-direction	0.0	2.0
Number of drill holes in the area	20	10
Drilling direction	positive X-direction	positive Y-direction

The default settings of the check areas are made in the startup file.

`PRGM,C`	Consider M56 and M57 commands in the part program
`CHEK,E`	Several check areas are allowed
`CHEKT1T2T3`	Tool list for check holes
`CHEK,S1XYDX2.DY0.N20`	Define check area 1
`CHEK,S2XYDX0.DY2.N10`	Define check area 2

The program-dependent settings (start positions and drilling directions) are defined in the part program:

`X10.Y20.M56`	Check area 1: start coordinates = X10.Y20.; drilling direction = positive X-direction
`X450.Y20.M57`	Check area 2: start coordinates = X450.Y20.; drilling direction = positive Y-direction

Related topics

M57: Define Check Area in Y-direction, page 36

M57: Define Check Area in Y-direction

1000	3000	5000
–	–	•

Command	Description
X*x*Y*y*M57	Define check area in Y-direction

Argument	Description
xy	Start coordinates of the check area

Requirement

The CNC commands CHEK,E and PRGM,C are active.

Dependent on the machine configuration the M56 and M57 commands define the start coordinates and the drilling direction of the area for check holes.

- The check area must have been defined before (for example in the startup-file).
- The coordinates programmed in M56/M57 are the start coordinates of an area. These coordinates replace the start coordinates that have been defined before.

Requirement for the SIEB & MEYER format:

Marginal condition	Setting
Format	5000
Axis version	1

Example

The following example shows the definition of two check areas.

Parameter	Area 1	Area 2
Coordinate of the first drill hole	X10.Y20.	X450.Y20.
Distance in X-direction	2.0	0.0
Distance in Y-direction	0.0	2.0
Number of drill holes in the area	20	10
Drilling direction	positive X-direction	positive Y-direction

The default settings of the check areas are made in the startup file.

`PRGM,C`	Consider M56 and M57 commands in the part program
`CHEK,E`	Several check areas are allowed
`CHEKT1T2T3`	Tool list for check holes
`CHEK,S1XYDX2.DY0.N20`	Define check area 1
`CHEK,S2XYDX0.DY2.N10`	Define check area 2

The program-dependent settings (start positions and drilling directions) are defined in the part program:

`X10.Y20.M56`	Check area 1: start coordinates = X10.Y20.; drilling direction = positive X-direction

`X450.Y20.M57`	Check area 2: start coordinates = X450.Y20.; drilling direction = positive Y-direction

Related topics

M56: Define Check Area in X-direction, page 35

3 Routing

General information

All standard routing contours can be routed in the SIEB & MEYER format.

- Some routing functions depend on the machine equipment and the configuration. For detailed information refer to the documentation of the machine manufacturer.
- A T0 program line must be programmed at the beginning and at the end of a routing contour.
- Routing functions remain active until a new routing command or a T0 command is programmed.
- Dependent on the used CNC the cutter tool life is monitored during routing.

Routing edges

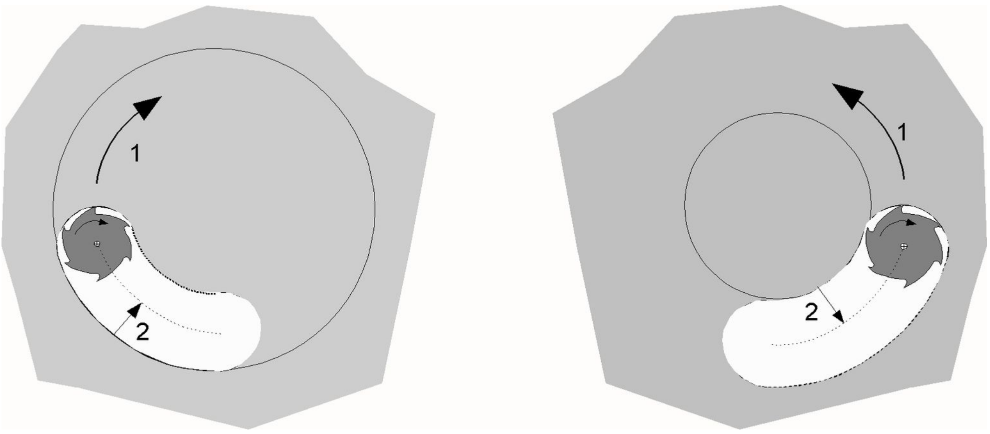
To obtain accurate routing edges, take the following rules into consideration:

- Routing out (hole) is done in clockwise direction. That means: The cutter radius compensation must be programmed to the right.
- Routing along an outer contour is done in counterclockwise direction. That means: The cutter radius compensation must be programmed to the right.
- Therefore, routing programs should not be executed as mirror images!

Note

To obtain accurate routing edges with right-turning cutters, take the following rules into consideration:

- **Left figure**: Routing out (hole) is done in clockwise direction.
- **Right figure**: Routing along an outer contour is done in counterclockwise direction.



Absolute/incremental interpretation

Routing coordinates can be programmed as absolute values (G90) or as incremental values.

- Absolute: coordinate values are referenced to the program zero
- Incremental: coordinate values are referenced to the previous coordinate

Cutter radius compensation

The cutter radius may be compensated with commands G41 or G42. The CNC automatically determines the plunge point.

Double routing coordinates

A directly subsequent program line must not contain identical rout coordinates. Otherwise an error message occurs.

Routing track in a subprogram

Subroutines may contain routing contours. At the end of a subprogram the routing contour is terminated automatically (corresponds to the command T0).

Routing parameters

For the routing parameters required for tools (plunge rate, retract feedrate, speed, maximum tool life) please consider the documents of the tool manufacturer. The values are entered into the tool table.

Tool life monitoring

The maximum admissible tool life is defined in the tool table.

- When the maximum admissible tool life is reached, the tool is automatically exchanged.
- The current tool life can also be adapted in the tool table.
- Furthermore, factors for the tool life counting can be defined for different materials and for peck drilling.

Tool measurement

Depending on the machine equipment and configuration the tool is measured after it has been picked up.

- The lenght (absolute and relative) is measured in one measuring cycle.
  - Absolute length = deviation of the actual tool length from an ideally clamped tool
  - Relative length = distance from the lower edge of the pressure foot to the tool tip
- The diameter and runout can be determined in a further measuring cycle:
  - Diameter: The measured diameter serves for monitoring whether the correct tool has been picked up.
  - Runout: Too strong oscillations of a tool may indicate a dirty collet.

Configuration

Dependent on the used CNC the routing process can be influenced by various settings. Further information can be found in the manuals for your CNC.

3.1 Rout Patterns

This section described commands used for programming routing tracks in the part program (straight lines, circular arc, circle etc.).

G1: Rout Straight Line

1000	3000	5000
•	•	•

Command	Description
X*x*Y*y* G1	Rout straight line

Argument	Description
xy	End position of the routing track

Requirement

A T0 block must be programmed at the beginning of a routing contour.

The G1 command is used for routing a straight line. The G1 function remains active until the next T0 block or until another routing function is programmed.

- If at least one of the two XY-coordinates is not defined the last valid X-value and/or Y-value applies.
- The connecting line between the start and end position is exactly the center of the routing path.
- The cutter radius can either be compensated with the commands G41 or G42.
- The CNC automatically determines the plunge point.

Example

A routing contour is routed consisting of two straight lines.

Marginal condition	Setting
Format	5000
Axis version	1

`X10.Y50.T5 T0`	Start position of the routing track. After a possibly required tool change to T5, the table moves to the position with a retracted Z-axis.
`X50.Y20.G1 F1.`	End position of the 1st routing track. The cutter is lowered to the routing plane and routs to the end position of the straight line.
`X90.Y60.`	End position of the 2nd routing track. The G1 function remains active.
`X90.Y60.T0`	End position of the routing contour. The Z-axis is retracted.

Since cutter radius compensation is off, the routing path is directly on the programmed line.

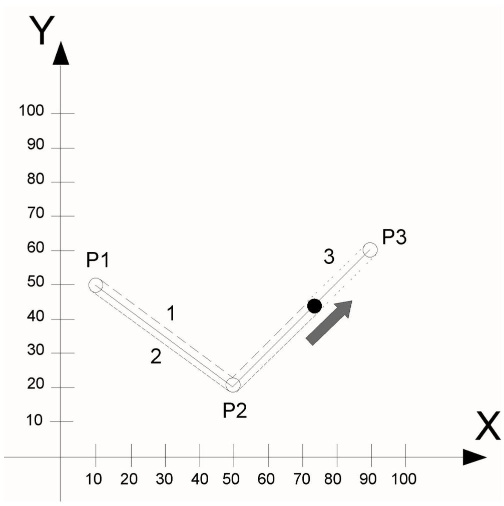


Fig. 23: Routed straight line

Related topics

F: Routing Feed Rate, page 63

G11: Path-dependent Finish-routing Function, page 65

G40: Deactivate Cutter Radius Compensation, page 57

G41: Activate Cutter Radius Compensation to the Left, page 57

G42: Activate Cutter Radius Compensation to the Right, page 59

T: Tool Change, page 120

G2: Rout Circular Arc Clockwise

1000	3000	5000
•	•	•

Command	Description
XxYy G2 Rr	(Format 3000 and higher) Rout circular arc with radius definition in clockwise direction
XxYy G2 IiJj	Rout circular arc with I*i* and J*j* in clockwise direction

Argument	Description
xy	End position of the routing track
r	Radius of the circular arc (e.g. R20.; R−17.5)
i	Y-component of the radius (e.g. I25.J−3.; I−17.5J12.)
j	Y-component of the radius (e.g. I25.J−3.; I−17.5J12.)

Requirement
A T0 block must be programmed at the beginning of a routing contour.

Use the command G2 to rout a circular arc in a clockwise direction (applies to FV1). The G2 function remains active until the next T0 block or until another routing function is programmed.

The following applies for the program command:

- If at least one of the two XY-coordinates is not defined the last valid X-value and/or Y-value applies.
- In addition, the radius of the circular arc *r* or optionally the center of the circle is defined with the parameters I*i* and J*j*, which connects the start and final points of the routing track.
- The connecting line between the start and end position is exactly the center of the routing path.
- The cutter radius can either be compensated with the commands G41 or G42.
- The CNC automatically determines the plunge point.
- The following CNC commands effect the direction of rotation and the compensation direction for routing: CCW, VER, CCWI, ROV1, OCOM and FV. For further information refer to the manual CNC 8x.00 – CNC Command.

The routing angle is specified by the sign of the circular arc radius:

- If the value is positive, the routing angle is ≤ 180°.
- If the value is negative, the routing angle is > 180°.

Routing direction

The routing direction depends on the axis version (CNC command FV).

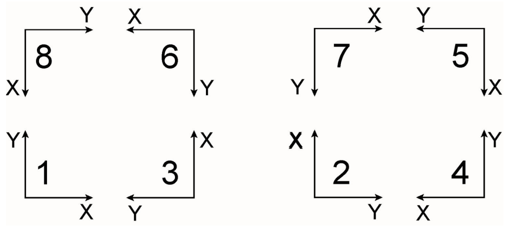


Fig. 24: Axis versions

Axis version	Clockwise	Counterclockwise
1, 3, 6, 8	G2	G3
2, 4, 5, 7	G3	G2

Routing angle

The routing angle depends on the sign of the radius value.

Radius value	Explanation
negative	The routing angle α is larger than 180° (larger than a semicircle)
positive	The routing angle α is smaller than or equal to 180° (maximum a semicircle)

Left figure: The angle α is smaller than 180°: The parameter R must be positive.

Right figure: The angle α is greater than 180°: The parameter R must be negative.

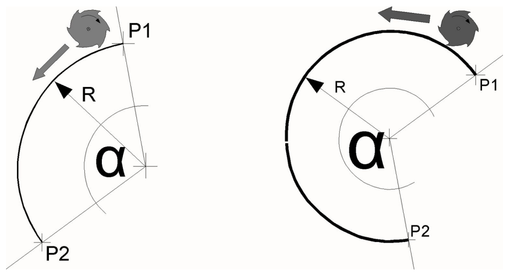


Fig. 25: Routing angle

Example

A contour shall be routed consisting of three arcs.

Marginal condition	Setting
Format	5000
Axis version	1

`X30.Y50.T0 T5`	Start position of the routing track. After a possibly required tool change to T5, the table moves to the position with a retracted Z-axis.
`X40.Y50.G2 R15. F1.2`	End position of the 1st routing track. The cutter is lowered to the routing plane and routs to the end position of the circular arc.
`X50.Y50.R-15.`	End position of the 2nd routing track. The G2 function remains active.
`X60.Y50.G3`	End position of the 3rd routing track. Then, the G3 command is active as routing function.
`X60.Y50.T0`	End position of the routing contour. The Z-axis is retracted.

Even if cutter radius compensation is on, complex routing contours are routed without material damage.

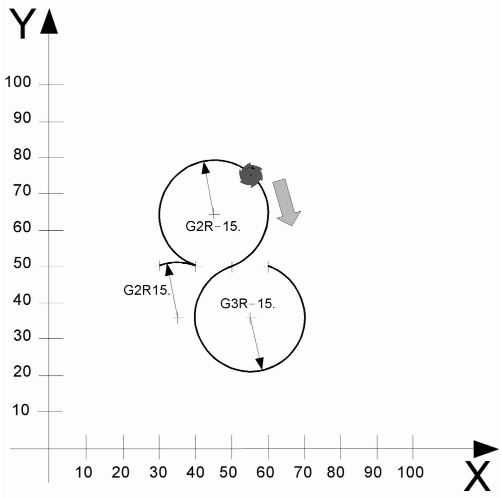


Fig. 26: Routing contour consisting of three combined circular arcs

Related topics

[D: Round Edge, page 61]
[F: Routing Feed Rate, page 63]
[G11: Path-dependent Finish-routing Function, page 65]
[G3: Rout Circular Arc Counterclockwise, page 43]
[G40: Deactivate Cutter Radius Compensation, page 57]
[G41: Activate Cutter Radius Compensation to the Left, page 57]
[G42: Activate Cutter Radius Compensation to the Right, page 59]
[I: Interpolation Parameter I for Routing Commands, page 113]
[J: Interpolation Parameter J for Routing Commands, page 115]
[R: Parameter for Routing Commands, page 119]
[T: Tool Change, page 120]

G3: Rout Circular Arc Counterclockwise

1000	3000	5000
•	•	•

Command	Description
X*x*Y*y* G3 R*r*	(Format 3000 and higher) Rout circular arc with radius definition in counterclockwise direction
X*x*Y*y* G3 I*i*J*j*	Rout circular arc with I*i* and J*j* in counterclockwise direction

Argument	Description
xy	End position of the routing track
r	Radius of the circular arc (e.g. R20.; R−17.5)
i	Y-component of the radius (e.g. I25.J−3.; I−17.5J12.)
j	Y-component of the radius (e.g. I25.J−3.; I−17.5J12.)

Requirement

A T0 block must be programmed at the beginning of a routing contour.

Use the command G3 to rout a circular arc in a counterclockwise direction (applies to FV1). The G3 function remains active until the next T0 block or until another routing function is programmed.

The following applies for the program command:

- If at least one of the two XY-coordinates is not defined the last valid X-value and/or Y-value applies.
- In addition, the radius of the circular arc *r* or optionally the center of the circle is defined with the parameters I*i* and J*j*, which connects the start and final points of the routing track.
- The connecting line between the start and end position is exactly the center of the routing path.
- The cutter radius can either be compensated with the commands G41 or G42.
- The CNC automatically determines the plunge point.
- The following CNC commands effect the direction of rotation and the compensation direction for routing: CCW, VER, CCWI, ROV1, OCOM and FV. For further information refer to the manual CNC 8x.00 – CNC Command.

The routing angle is specified by the sign of the circular arc radius:

- If the value is positive, the routing angle is ≤ 180°.
- If the value is negative, the routing angle is > 180°.

Routing direction

The routing direction depends on the axis version (CNC command FV).

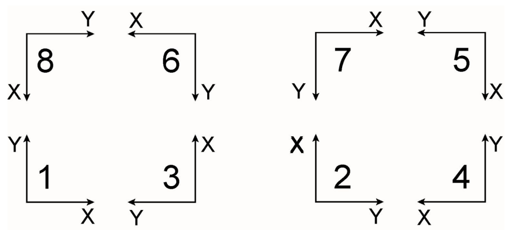


Fig. 27: Axis versions

Axis version	Clockwise	Counterclockwise
1, 3, 6, 8	G2	G3
2, 4, 5, 7	G3	G2

Routing angle

The routing angle depends on the sign of the radius value.

Radius value	Explanation
negative	The routing angle α is larger than 180° (larger than a semicircle)
positive	The routing angle α is smaller than or equal to 180° (maximum a semicircle)

Left figure: The angle α is smaller than 180°: The parameter R must be positive.

Right figure: The angle α is greater than 180°: The parameter R must be negative.

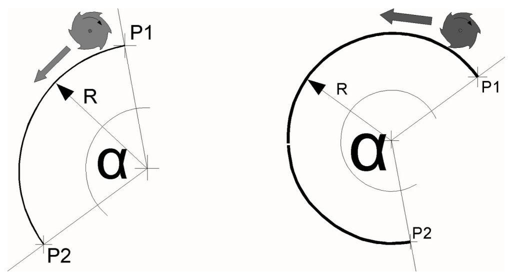


Fig. 28: Routing angle

Example

A contour shall be routed consisting of three arcs.

Marginal condition	Setting
Format	5000
Axis version	1

`X30.Y50.T0 T5`	Start position of the routing track. After a possibly required tool change to T5, the table moves to the position with a retracted Z-axis.
`X40.Y50.G2 R15. F1.2`	End position of the 1st routing track. The cutter is lowered to the routing plane and routs to the end position of the circular arc.
`X50.Y50.R-15.`	End position of the 2nd routing track. The G2 function remains active.
`X60.Y50.G3`	End position of the 3rd routing track. Then, the G3 command is active as routing function.
`X60.Y50.T0`	End position of the routing contour. The Z-axis is retracted.

Even if cutter radius compensation is on, complex routing contours are routed without material damage.

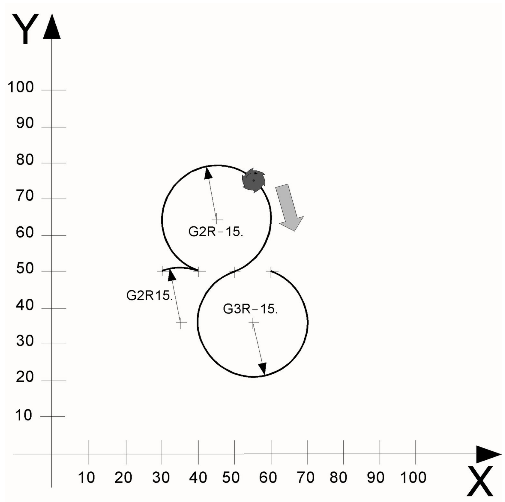


Fig. 29: Routing contour consisting of three combined circular arcs

Related topics

[D: Round Edge, page 61]

[F: Routing Feed Rate, page 63]

[G11: Path-dependent Finish-routing Function, page 65]

[G2: Rout Circular Arc Clockwise, page 41]

[G40: Deactivate Cutter Radius Compensation, page 57]

[G41: Activate Cutter Radius Compensation to the Left, page 57]

[G42: Activate Cutter Radius Compensation to the Right, page 59]

[I: Interpolation Parameter I for Routing Commands, page 113]

[J: Interpolation Parameter J for Routing Commands, page 115]

[R: Parameter for Routing Commands, page 119]

[T: Tool Change, page 120]

G45: Rout Out Full Circle Counterclockwise

1000	3000	5000
•	•	•

Command	Description
X*x*Y*y* G45 R*r*	Rout out full circle with radius definition in counterclockwise direction

Argument	Description
xy	XY-coordinates of the circle center
r	Radius of the circle Only positive values are allowed.

Use the command G45 to rout out a full circle in a counterclockwise direction (applies to FV1).

The following applies for the command:

- The working method of the command depends on the axis version (CNC command FV).
- The cutter radius is automatically compensated to the inside.
- The CNC automatically determines the plunge point.
- Just before the end of the routing track is reached the routing feed rate is reduced.
- The finish-routing function G11 is ignored.
- To obtain a well-routed routing edge the full circle must be routed in clockwise direction.
- If the G43 command or the CNC command ROUT,A is programmed additionally, the surface of the circle is cut completely.

Routing direction

The routing direction depends on the axis version.

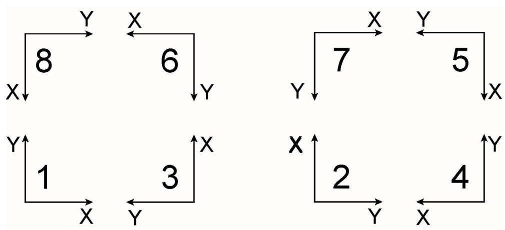


Fig. 30: Axis version

Axis version	Clockwise	Counterclockwise
1, 3, 6, 8	G46	G45
2, 4, 5, 7	G45	G46

Calculate plunge point

The cutter plunges into the inner area of the part to be routed out and moves via an arc to the edge of the circle. For predrillingdrilling the plunge position is calculated as follows:

G45	G46
$X_{plunge} = x + 0.707 \times (r/2 - FD/4)$	$X_{plunge} = x - 0.707 \times (r/2 - FD/4)$
$Y_{plunge} = y + 1.707 \times (r/2 - FD/4)$	$Y_{plunge} = y + 1.707 \times (r/2 - FD/4)$

Parameter	Explanation
xy	Center of the circle
r	Radius

Parameter	Explanation
FD	Cutter diameter

To avoid damage of the hole edge when the cutter plunges into the material, the plunge point is in the inner area.

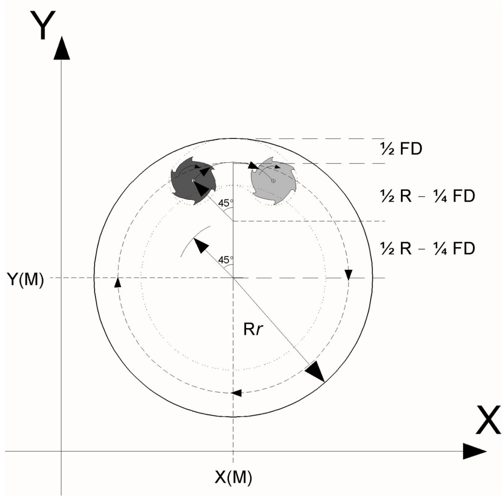


Fig. 31: Routing path of an inner cutout

Example: Routing out full circles

Two full circles shall be routed.

Marginal condition	Setting
Format	5000
Axis version	1

`X35.Y50.T0 T5`	Center of the 1st full circle. After a possibly required tool change to T5, the table moves to the position with a retracted Z-axis.
`X35.Y50.G46R20. F1.2`	Center of the 1st full circle. The cutter is lowered to the routing plane and routs out the complete contour.
`X75.Y30.R15.`	Center of the 2nd full circle. The G46 function remains active.
`X75.Y30.T0`	Center of the 2nd full circle. The Z-axis is retracted.

Example: Calculate plunge point

The plunge point for the example above is calculated for the 1st full circle.

- $X_{plunge1} = 35.0 - 0.707 \times (20.0/2 - 2.00/4) = 28.283$
- $Y_{plunge1} = 50.0 + 1.707 \times (20.0/2 - 2.00/4) = 66.217$

Related topics

G40: Deactivate Cutter Radius Compensation, page 57

G41: Activate Cutter Radius Compensation to the Left, page 57

G42: Activate Cutter Radius Compensation to the Right, page 59

G43: Cut Area of the Stored Pattern Completely, page 66

G46: Rout Out Full Circle Clockwise, page 48

R: Parameter for Routing Commands, page 119

G46: Rout Out Full Circle Clockwise

1000	3000	5000
•	•	•

Command	Description
X*x*Y*y* G46 R*r*	Rout out full circle with radius definition in clockwise direction

Argument	Description
xy	XY-coordinates of the circle center
r	Radius of the circle Only positive values are allowed.

Use the command G46 to rout out a full circle in a clockwise direction (applies to FV1).

The following applies for the command:

- The working method of the command depends on the axis version (CNC command FV).
- The cutter radius is automatically compensated to the inside.
- The CNC automatically determines the plunge point.
- Just before the end of the routing track is reached the routing feed rate is reduced.
- The finish-routing function G11 is ignored.
- To obtain a well-routed routing edge the full circle must be routed in clockwise direction.
- If the G43 command or the CNC command ROUT,A is programmed additionally, the surface of the circle is cut completely.

Routing direction

The routing direction depends on the axis version.

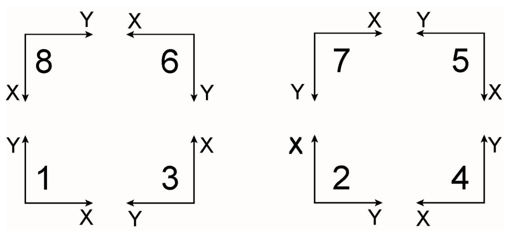


Fig. 32: Axis version

Axis version	Clockwise	Counterclockwise
1, 3, 6, 8	G46	G45
2, 4, 5, 7	G45	G46

Calculate plunge point

The cutter plunges into the inner area of the part to be routed out and moves via an arc to the edge of the circle. For predrillingdrilling the plunge position is calculated as follows:

G45	G46
$X_{plunge} = x + 0.707 \times (r/2 - FD/4)$	$X_{plunge} = x - 0.707 \times (r/2 - FD/4)$
$Y_{plunge} = y + 1.707 \times (r/2 - FD/4)$	$Y_{plunge} = y + 1.707 \times (r/2 - FD/4)$

Parameter	Explanation
xy	Center of the circle
r	Radius
FD	Cutter diameter

To avoid damage of the hole edge when the cutter plunges into the material, the plunge point is in the inner area.

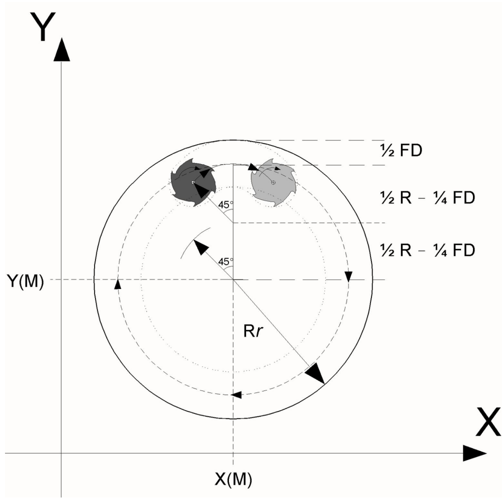


Fig. 33: Routing path of an inner cutout

Example: Routing out full circles

Two full circles shall be routed.

Marginal condition	Setting
Format	5000
Axis version	1

`X35.Y50.T0 T5`	Center of the 1st full circle. After a possibly required tool change to T5, the table moves to the position with a retracted Z-axis.
`X35.Y50.G46R20. F1.2`	Center of the 1st full circle. The cutter is lowered to the routing plane and routs out the complete contour.
`X75.Y30.R15.`	Center of the 2nd full circle. The G46 function remains active.
`X75.Y30.T0`	Center of the 2nd full circle. The Z-axis is retracted.

Example: Calculate plunge point

The plunge point for the example above is calculated for the 1st full circle.

- $X_{plunge1}$ = 35.0 – 0.707 x (20.0/2 – 2.00/4) = 28.283
- $Y_{plunge1}$ = 50.0 + 1.707 x (20.0/2 – 2.00/4) = 66.217

Related topics

G40: Deactivate Cutter Radius Compensation, page 57

G41: Activate Cutter Radius Compensation to the Left, page 57

G42: Activate Cutter Radius Compensation to the Right, page 59

G43: Cut Area of the Stored Pattern Completely, page 66

G45: Rout Out Full Circle Counterclockwise, page 46

R: Parameter for Routing Commands, page 119

G47: Rout Out Disk Counterclockwise

1000	3000	5000
–	•	•

Command	Description
X*x*Y*y* G47 R*r*	Rout out disk with radius definition in counterclockwise direction

Argument	Description
xy	XY-coordinates of the circle center
r	Radius of the circle

Use the command G47 to rout out a disk with a radius R*r* in a counterclockwise direction (applies to FV1). The center is defined with the given XY-coordinates.

The following applies for the command:

- The working method of the command depends on the axis version (CNC command FV).
- The cutter radius is automatically compensated to the outside.
- The CNC automatically determines the plunge point.
- The finish-routing function G11 is considered, if it is programmed.
- To obtain a well-routed routing edge the disk must be routed in counterclockwise direction.

Routing direction

The routing direction depends on the axis version.

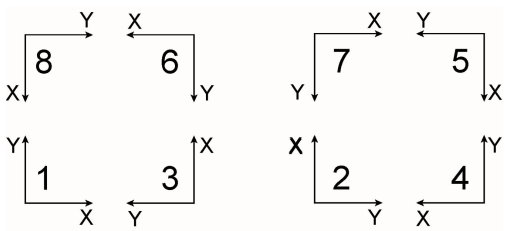


Fig. 34: Axis version

Axis version	Clockwise	Counterclockwise
1, 3, 6, 8	G48	G47
2, 4, 5, 7	G47	G48

Calculate plunge point

The cutter plunges into the outer area of the disk and moves to the edge of the circle. For predrillingdrilling the plunge position is calculated as follows:

G47	G48
$X_{plunge} = x - (r + FD/2)$	$X_{plunge} = x - (r + FD/2)$
$Y_{plunge} = y + FD/2$	$Y_{plunge} = y - FD/2$

Parameter	Explanation
xy	Center of the circle
r	Radius
FD	Cutter diameter

The plunge point is as near as possible to the edge of the hole.

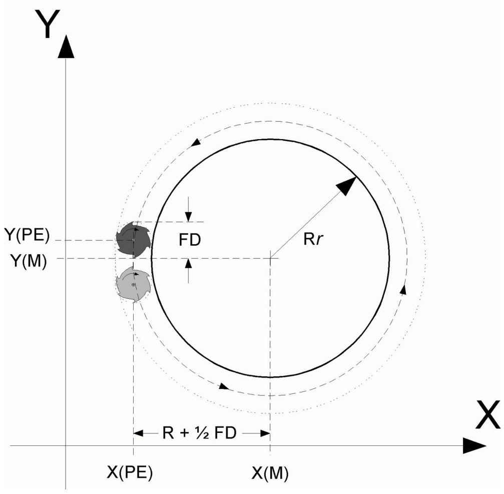


Fig. 35: Routing along an outer contour

Example: Rout out disks

Two disks shall be routed out.

Marginal condition	Setting
Format	5000
Axis version	1

`X35.Y50.T0 T2`	Center of the 1st disk. After a possibly required tool change to T5, the table moves to the position with a retracted Z-axis.
`X35.Y50.G47R20. F1.2`	Center of the 1st disk. The cutter is lowered to the routing plane and routs out the complete contour.
`X75.Y30.R15.`	Center of the 2nd disk. The G47 function remains active.
`X75.Y30.T0`	Center of the 2nd disk. The Z-axis is retracted.

Example: Calculate plunge point

The plunge point for the example above is calculated for the 1st disk.

- $X_{plunge1} = 35.0 - (20.0 + 2.00/2) = 14.0$
- $Y_{plunge1} = 50.0 + (2.00/2) = 51.0$

Related topics

F: Routing Feed Rate, page 63

G40: Deactivate Cutter Radius Compensation, page 57

G41: Activate Cutter Radius Compensation to the Left, page 57

G42: Activate Cutter Radius Compensation to the Right, page 59

G48: Rout Out Disk Clockwise, page 51

R: Parameter for Routing Commands, page 119

G48: Rout Out Disk Clockwise

1000	3000	5000
–	•	•

Command	Description
X*x*Y*y* G48 R*r*	Rout out disk with radius definition in clockwise direction

Argument	Description
xy	XY-coordinates of the circle center
r	Radius of the circle

Use the command G48 to rout out a disk with a radius R*r* in a clockwise direction (applies to FV1). The center is defined with the given XY-coordinates.

The following applies for the command:

- The working method of the command depends on the axis version (CNC command FV).
- The cutter radius is automatically compensated to the outside.
- The CNC automatically determines the plunge point.
- The finish-routing function G11 is considered, if it is programmed.
- To obtain a well-routed routing edge the disk must be routed in counterclockwise direction.

Routing direction

The routing direction depends on the axis version.

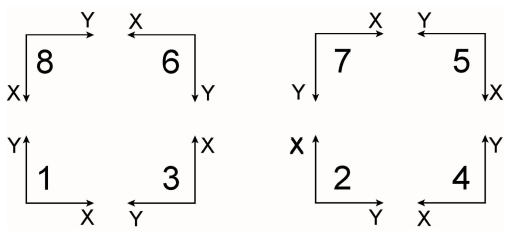


Fig. 36: Axis version

Axis version	Clockwise	Counterclockwise
1, 3, 6, 8	G48	G47
2, 4, 5, 7	G47	G48

Calculate plunge point

The cutter plunges into the outer area of the disk and moves to the edge of the circle. For predrillingdrilling the plunge position is calculated as follows:

G47	G48
$X_{plunge} = x - (r + FD/2)$	$X_{plunge} = x - (r + FD/2)$
$Y_{plunge} = y + FD/2$	$Y_{plunge} = y - FD/2$

Parameter	Explanation
xy	Center of the circle
r	Radius
FD	Cutter diameter

The plunge point is as near as possible to the edge of the hole.

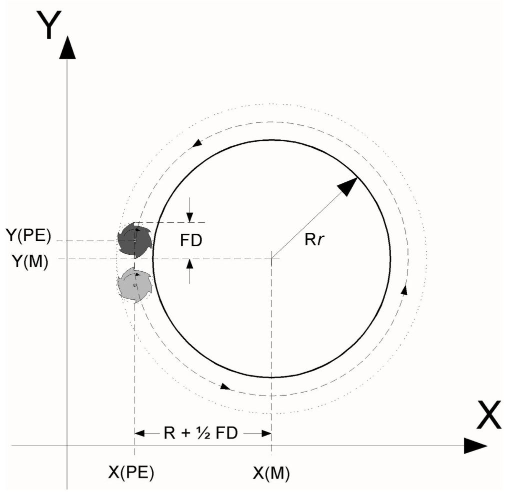


Fig. 37: Routing along an outer contour

Related topics

[F: Routing Feed Rate, page 63]

[G40: Deactivate Cutter Radius Compensation, page 57]

[G41: Activate Cutter Radius Compensation to the Left, page 57]

[G42: Activate Cutter Radius Compensation to the Right, page 59]

[G47: Rout Out Disk Counterclockwise, page 49]

[R: Parameter for Routing Commands, page 119]

G49: Rout Out Rectangle Counterclockwise

1000	3000	5000
•	•	•

Command	Description
X*x*Y*y* G49 R*r*	Rout out square with definition of the side length R*r* in counterclockwise direction
X*x*Y*y* G49 I*i* J*j*	Rout out rectangle with definition of the side lengths I*i* and J*j* in counterclockwise direction

Argument	Description
xy	Center of the rectangle/square
r	Side length of a square
i	Side length of a rectangle in X-direction
j	Side length of a rectangle in Y-direction

Use the command G49 to rout out a rectangular hole in a counterclockwise direction. The side lengths are defined with corresponding parameters.

The following applies for this command:

- The working method of the commands depends on the axis version (CNC command FV).
- The cutter radius is automatically compensated.
- The CNC automatically determines the plunge point.
- The finish-routing function G11 is ignored.
- Just before the end of the routing track is reached the routing feed rate is reduced.

- If the G43 command or the CNC command ROUT,A is programmed additionally, the surface of the rectangle is cut completely.
- To obtain a well-routed routing edge the rectangle must be routed in clockwise direction.

Routing direction

The routing direction depends on the axis version.

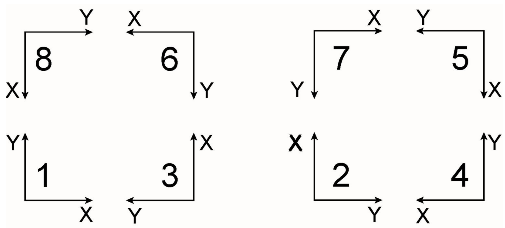


Fig. 38: Axis version

Calculate plunge point

Axis version	Clockwise	Counterclockwise
1, 3, 6, 8	G50	G49
2, 4, 5, 7	G49	G50

The cutter plunges into the inner area of the rectangle and moves to the edge of the rectangle. For predrillingdrilling the plunge position is calculated as follows:

G49	G50
$X_{plunge} = x$	$X_{plunge} = x$
$Y_{plunge} = y + j/4 - FD/4$	$Y_{plunge} = y + j/4 - FD/4$

Parameter	Explanation
xy	Center of the rectangle
j	Side length in Y-direction
FD	Cutter diameter

To avoid damage of the hole edge when the cutter plunges into the material, the plunge point is in the inner area.

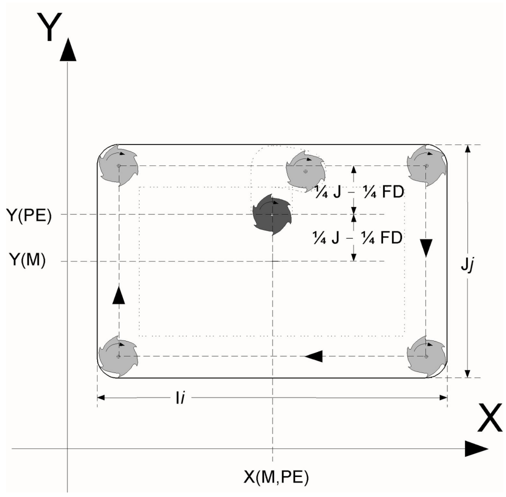


Fig. 39: Rout out a rectangle

Related topics

F: Routing Feed Rate, page 63

G40: Deactivate Cutter Radius Compensation, page 57

G41: Activate Cutter Radius Compensation to the Left, page 57

G42: Activate Cutter Radius Compensation to the Right, page 59

G43: Cut Area of the Stored Pattern Completely, page 66

G50: Rout Out Rectangle Clockwise, page 55

I: Interpolation Parameter I for Routing Commands, page 113

J: Interpolation Parameter J for Routing Commands, page 115

R: Parameter for Routing Commands, page 119

G50: Rout Out Rectangle Clockwise

1000	3000	5000
•	•	•

Command	Description
XxYy G50 R*r*	Rout out square with definition of the side length R*r* in clockwise direction
XxYy G50 I*i* J*j*	Rout out rectangle with definition of the side lengths I*i* and J*j* in clockwise direction

Argument	Description
xy	Center of the rectangle/square
r	Side length of a square
i	Side length of a rectangle in X-direction
j	Side length of a rectangle in Y-direction

Use the command G50 to rout out a rectangular hole in a clockwise direction. The side lengths are defined with corresponding parameters.

The following applies for this command:

- The working method of the commands depends on the axis version (CNC command FV).
- The cutter radius is automatically compensated.
- The CNC automatically determines the plunge point.
- The finish-routing function G11 is ignored.
- Just before the end of the routing track is reached the routing feed rate is reduced.
- If the G43 command or the CNC command ROUT,A is programmed additionally, the surface of the rectangle is cut completely.
- To obtain a well-routed routing edge the rectangle must be routed in clockwise direction.

Routing direction

The routing direction depends on the axis version.

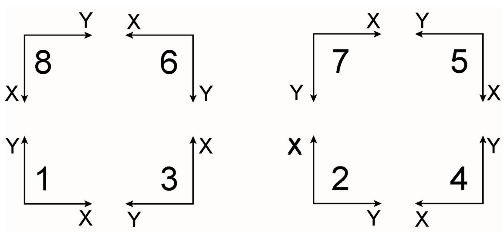


Fig. 40: Axis version

Calculate plunge point

Axis version	Clockwise	Counterclockwise
1, 3, 6, 8	G50	G49

Axis version	Clockwise	Counterclockwise
2, 4, 5, 7	G49	G50

The cutter plunges into the inner area of the rectangle and moves to the edge of the rectangle. For predrillingdrilling the plunge position is calculated as follows:

G49	G50
$X_{plunge} = x$	$X_{plunge} = x$
$Y_{plunge} = y + j/4 - FD/4$	$Y_{plunge} = y + j/4 - FD/4$

Parameter	Explanation
xy	Center of the rectangle
j	Side length in Y-direction
FD	Cutter diameter

To avoid damage of the hole edge when the cutter plunges into the material, the plunge point is in the inner area.

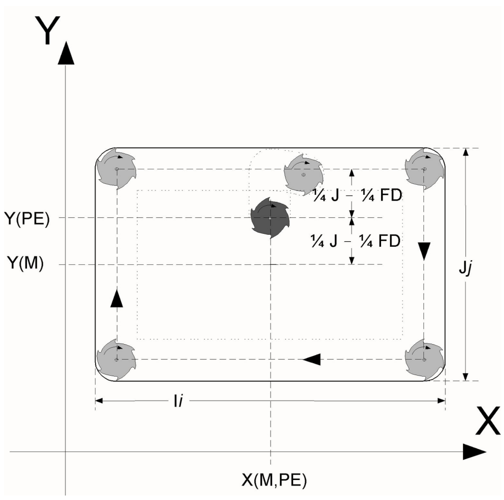


Fig. 41: Rout out a rectangle

Example: Rout out rectangles

Rout out two rectangles.

Marginal condition	Setting
Format	5000
Axis version	1

Code	Description
`X30.Y50.T0 T5`	Center of the 1st rectangle. After a possibly required tool change to T5, the table moves to the position with a retracted Z-axis.
`X30.Y50.G50I40.J30. F1.2`	Center of the 1st rectangle. The cutter is lowered to the routing plane and routs out the complete contour.
`X75.Y50.R40.`	Center of the 2nd rectangle. The G50 function remains active.
`X75.Y50.T0`	Center of the 2nd rectangle. The Z-axis is retracted.

Example: Calculate plunge point

The plunge point for the example above is calculated for the 1st disk.

- $X_{plunge1} = 30.0$
- $Y_{plunge1} = 50.0 + 30.0/4 - 2.00/4 = 57.0$

Related topics

[F: Routing Feed Rate, page 63]
[G40: Deactivate Cutter Radius Compensation, page 57]
[G41: Activate Cutter Radius Compensation to the Left, page 57]
[G42: Activate Cutter Radius Compensation to the Right, page 59]
[G43: Cut Area of the Stored Pattern Completely, page 66]
[G49: Rout Out Rectangle Counterclockwise, page 53]
[I: Interpolation Parameter I for Routing Commands, page 113]
[J: Interpolation Parameter J for Routing Commands, page 115]
[R: Parameter for Routing Commands, page 119]

3.2 Cutter radius compensation

This section describes commands used for programming the cutter radius compensation in the part program.

G40: Deactivate Cutter Radius Compensation

1000	3000	5000
–	•	•

Command	Description
X*x*Y*y* G40	Deactivate cutter radius compensation

Argument	Description
xy	End position of the routing track

The command G40 deactivates the compensations activated by G41 or G42. The center of the cutter is guided exactly on the programmed contour. The cutter radius compensation remains deactivated, until a G41 or G42 command appears in the part program.

If the CNC command ACCM is active, the contour programmed with ACCM still applies.

Note

Functions programmed with the commands G45, G46, G47, G48, G49 and G50 cannot be deactivated with G40. They will be compensated during execution.

Related topics

[G41: Activate Cutter Radius Compensation to the Left, page 57]
[G42: Activate Cutter Radius Compensation to the Right, page 59]

G41: Activate Cutter Radius Compensation to the Left

1000	3000	5000
•	•	•

Command	Description
X*x*Y*y* G41	Activate cutter radius compensation to the left relative to routing direction

Argument	Description
xy	End position of the routing track

With command G41 you activate the cutter radius compensation to the left relative to the routing direction. The routing tool is moved along the programmed contour compensated by half of the tool diameter to the left.

The following applies for the command:

- The command is only effective at the beginning of a routing contour.
- The cutter radius compensation considers the tool parameter D (diameter) during routing.
- The actual routing path runs parallel as displaced curve next to the programmed routing path.
- Changing the compensation direction within a routing track is not recommended, since it may result in faulty routing results.
- The cutter radius compensation remains on
  - until the next compensation command or
  - until the next G40 command or
  - until the T0 command.

Compensation direction

The direction of the cutter radius compensation depends on the axis version.

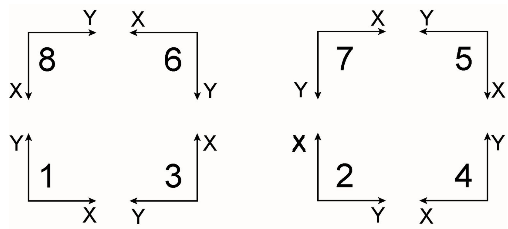


Fig. 42: Axis versions

Axis version	In routing direction to the right	In routing direction to the left
1, 3, 6, 8	G42	G41
2, 4, 5, 7	G41	G42

Note

To obtain accurate routing edges, take the following rules into consideration:

- Routing out (hole) is done **in clockwise direction**.
- Routing along an outer contour is done **in counterclockwise direction**.

When the above mentioned rules are observed, only the cutter radius compensation into the right routing direction is reasonable.

CNC commands

- If the CNC command INDX,D is set, the effect of the compensation commands G41 and G42 is inverted.
- (Software version 21.01.001 and higher) If the CNC commands INDX,D and INDX,I are set and if the index table contains negative diameter values, the effect of the compensation commands G41 and G42 is not inverted.

Related topics

G40: Deactivate Cutter Radius Compensation, page 57

G42: Activate Cutter Radius Compensation to the Right, page 59

G42: Activate Cutter Radius Compensation to the Right

1000	3000	5000
•	•	•

Command	Description
XxYy G42	Activate cutter radius compensation to the right relative to routing direction

Argument	Description
xy	End position of the routing track

With command G42 you activate the cutter radius compensation to the right relative to the routing direction. The routing tool is moved along the programmed contour compensated by half of the tool diameter to the right.

The following applies for the command:

- The command is only effective at the beginning of a routing contour.
- The cutter radius compensation considers the tool parameter D (diameter) during routing.
- The actual routing path runs parallel as displaced curve next to the programmed routing path.
- Changing the compensation direction within a routing track is not recommended, since it may result in faulty routing results.
- The cutter radius compensation remains on
  - until the next compensation command or
  - until the next G40 command or
  - until the T0 command.

Compensation direction

The direction of the cutter radius compensation depends on the axis version.

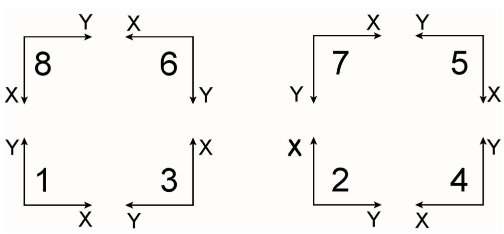


Fig. 43: Axis versions

Axis version	In routing direction to the right	In routing direction to the left
1, 3, 6, 8	G42	G41
2, 4, 5, 7	G41	G42

Note

To obtain accurate routing edges, take the following rules into consideration:

- Routing out (hole) is done **in clockwise direction**.
- Routing along an outer contour is done **in counterclockwise direction**.

When the above mentioned rules are observed, only the cutter radius compensation into the right routing direction is reasonable.

CNC commands

- If the CNC command INDX,D is set, the effect of the compensation commands G41 and G42 is inverted.

- (Software version 21.01.001 and higher) If the CNC commands INDX,D and INDX,I are set and if the index table contains negative diameter values, the effect of the compensation commands G41 and G42 is not inverted.

Left figure: Compensation at a straight line.

Right figure: Compensation at a circular arc. The plunge position is calculated by the CNC. FD = cutter diameter, FR = center path of the cutter

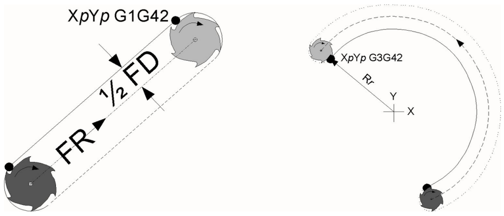


Fig. 44: Cutter radius compensation

Example

Several slots shall be routed.

Marginal condition	Setting
Format	5000
Axis version	1

`X10.Y50.T5 T0`	Start position of the routing track. Tool change to T5 and positioning and to the plunge point. The positioning already considers the cutter radius compensation of the following program line.
`X50.Y30.G1 G42 F1.2`	Programmed end position of the 1st straight line. The G42 command displaces the actual routing path to the right by the cutter radius.
`X60.Y55.G40`	End position of the 2nd straight line. The G40 command deactivates the cutter radius compensation and routs exactly on the programmed path. G1 is still active.
`X50.Y30.`	End position of the 3rd straight line. To obtain accurate two routing edges routing is repeated in the same slot in the opposite direction. G1 and G40 are still active.
`X85.Y45.G42`	Programmed end position of the 4th straight line. The G42 command displaces the actual routing path to the right by the cutter radius. G1 is still active.
`X85.Y45.T0`	End position of the routing track. The T0 command deactivates the cutter radius compensation.

Meaning of the lines in the figure:

A **continuous line** means an accurate routing edge; a **dotted line** means an inaccurate routing edge.

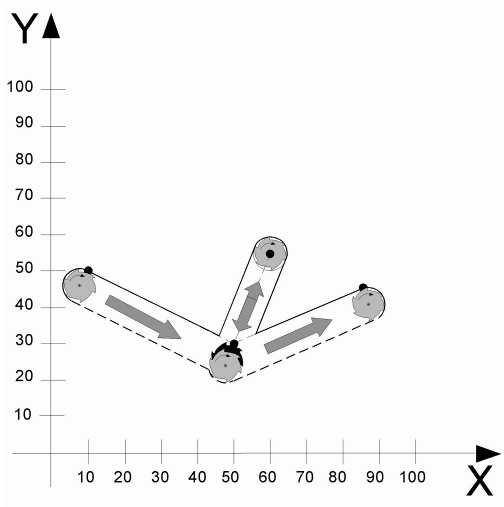


Fig. 45: Review of the routing edges

Related topics

G40: Deactivate Cutter Radius Compensation, page 57

G41: Activate Cutter Radius Compensation to the Left, page 57

3.3 Routing Conditions

This section describes commands which influence the execution of the routing functions.

D: Round Edge

1000	3000	5000
–	D1 = 0.001 mm	D1 = 0.001 mm

Command	Description
X*x*Y*y* D*r*	Round edge with radius definition

Argument	Description
xy	Coordinates of the intersection point
r	Radius of the rounded edge Only positive values are allowed.

The D command is used for rounding edges.

- The D value defines the radius of the circular arc with which the edge is rounded.
- The D command is programmed in the program line of which the routing track is to be rounded at the end.
- Only positive values are admitted.
- The routing function "Rounding edge" is only effective for the program line in which it was programmed.
- The D value must be smaller than the shorter of the adjacent straight line. Otherwise the routing contour is faulty.

Sequence of routing track

Table with the admissible passages:

Routing path	Straight line	Circular arc
Straight line	●	●
Circular arc	●	○

In combination with the function "Round edge" the following transitions are allowed:

- Left figure: straight line – straight line
- Center figure: straight line – circular arc
- Right figure: circular arc – straight line

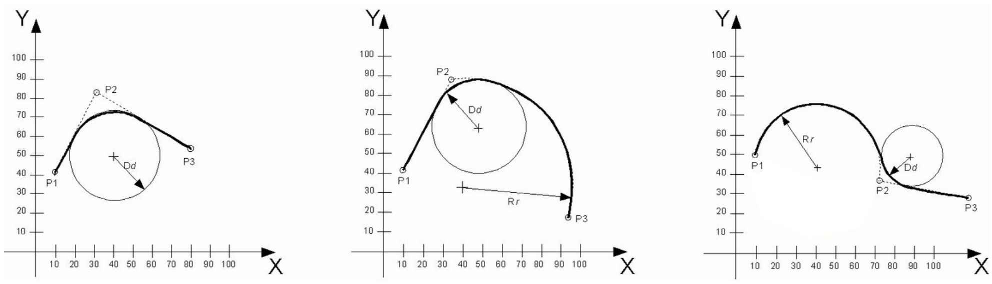


Fig. 46: Allowed routing transitions

Example

The passage from a circular arc to a straight line within a routing track shall be rounded.

Marginal condition	Setting
Format	5000
Axis version	1

`X10.Y50.T0T5`	Start position of the routing track. After a possibly required tool change to T5, the table moves to the position with a retracted Z-axis.
`X70.Y40.G2 D20. F1.2`	End position of the 1st routing track. The cutter is lowered to the routing plane and the routing process is started. Before the programmed end point is reached the system continues routing on the previously calculated edge.
`X30.Y130.G1`	End position of the 2nd routing track. The G1 function defines a straight line.
`X30.Y130.T0`	End position of the routing contour. The Z-axis is retracted.

The transition between a circular arc and a straight line is rounded.

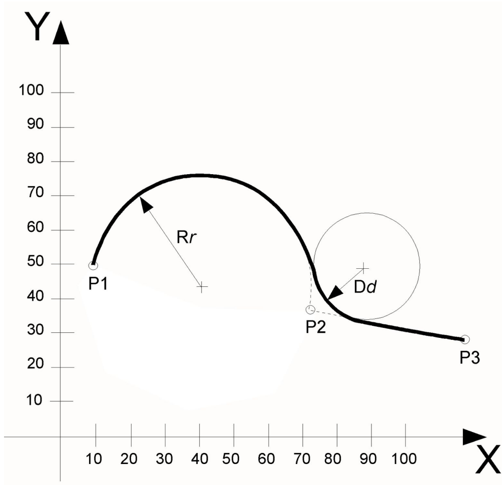


Fig. 47: Rounded edge

Related topics

[G1: Rout Straight Line, page 39]

[G2: Rout Circular Arc Clockwise, page 41]

[G3: Rout Circular Arc Counterclockwise, page 43]

F: Routing Feed Rate

1000	3000	5000
F1 = 0.1 m/min	F1 = 0.1 m/min	F1 = 0.001 m/min

Command	Description
X*x*Y*y* F*f*	Define feed rate

Argument	Description
xy	Coordinate reached with the new feed rate
f	Feed rate for the XY-axes ▸ unit in Format 1000: 0.1 m/min (Points in the numerical value are not allowed.) ▸ unit in Format 3000: 0.1 m/min (Points in the numerical value are not allowed.) ▸ unit in Format 5000: 0.001 m/min (Points in the numerical value are allowed.)

The F command defines the routing feed rate of the XY-axes.

- The routing feed rate must be defined for the first routing operation within a part program.
- The setting is active until it is replaced by another definition.
- The routing feed rate can be modified in the part program at any time.
- If no routing feed rate is programmed the machine routs at the default routing feed rate defined by the machine manufacturer.
- The routing feed rate is only effective during routing.
- The system always moves to drilling coordinates at the highest positioning speed of the machine.

Example

The routing speed is defined for the 1st routing contour and remains active for all routing contours within the part program.

Marginal condition	Setting
Format	5000
Axis version	1

`X30.Y60.T0 T5`	Start position of the 1st routing contour
`X60.Y30.G1 F2.`	Positioning speed = 2.0 m/min
`X60.Y30.T0`	
`(`	
`X60.Y40.T1`	Positioning speed = maximum speed
`X30.Y40.`	
`(`	
`X30.Y30.T0 T6`	Start position of the 2nd routing contour
`X60.Y60.G1`	Positioning speed = 2.0 m/min
`X60.Y60.T0`	

Related topics

G1: Rout Straight Line, page 39

G2: Rout Circular Arc Clockwise, page 41

G3: Rout Circular Arc Counterclockwise, page 43

G45: Rout Out Full Circle Counterclockwise, page 46
G46: Rout Out Full Circle Clockwise, page 48
G47: Rout Out Disk Counterclockwise, page 49
G48: Rout Out Disk Clockwise, page 51
G49: Rout Out Rectangle Counterclockwise, page 53
G50: Rout Out Rectangle Clockwise, page 55

G6: Lower Z-axis Linearly During Routing

1000	3000	5000
–	–	•

Command	Description
XxYy Zz G6	Lower Z-axis linearly during routing

Argument	Description
xy	End position of the routing track
z	Z-target plane

Note

The parameter Zz of the G6 command is used for modifying the working plane. Therefore, it must be ensured that a correct working plane is defined again for the following routing tracks and drill strokes.

The Z-plane of the routing path changed continuously.

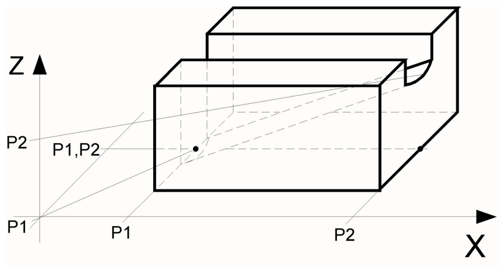


Fig. 48: Rout ramp

Use the command G6 to lower the Z-axis linearly during routing in order to rout a ramp.

- The programmed Z-target plane remains active until the next Z-command, which means that the following holes are also drilled up to the new Z-plane.
- If the start point of the routing track is a rounded edge (the previous routing track was programmed with the D command), the ramp only begins at the end of a rounding.
- If the end position of a routing track is a rounded edge (programmed in the same routing block with the D command), the ramp already ends at the beginning of a rounding.

Example

A ramp is to be routed from 2.0 mm to 3.5 mm. For the remaining part program a working plane of 1.0 mm applies.

Marginal condition	Setting
Format	5000
Axis version	1

`X..Y..T1 Z1.0`	Working plane = 1.0 mm
`...`	
`X20.Y40.T0T2Z2.0`	Working plane = 2.0 mm (start plane of the ramp)
`X70.Y40.G1F.8 G6Z3.5.`	Working plane = 3.5 mm (target plane of the ramp)
`X70.Y40. T0`	
`X..Y..T3Z1.0`	Working plane = 1.0 mm
`...`	

A routed ramp can for example be useful for the start/end position of a slot, if the slot is required for insertion of an attachment.

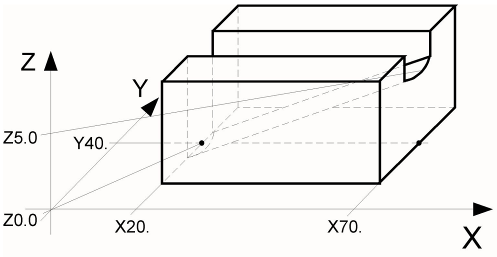


Fig. 49: Routed straight line with ramp

Related topics

[G1: Rout Straight Line, page 39]
[G2: Rout Circular Arc Clockwise, page 41]
[G3: Rout Circular Arc Counterclockwise, page 43]

G11: Path-dependent Finish-routing Function

1000	3000	5000
•	•	•

Command	Description
Xx Yy G11	Activate path-dependent finish-routing function

Note

The mode of functioning of the finish-routing function depends on the equipment and the configuration of a machine. For detailed information, refer to the documentation of the machine manufacturer.

Path-dependent finish-routing function

The G1 command activates the finish-routing function.

- The finish-routing function is activated shortly before a routing track is finished.
- For this purpose the machine manufacturer must have defined the distance which is active during the finish-routing function.
- During this distance the routing feed rate is reduced and an output is activated.
- The output allows for example activating/deactivating the pressure foot and the vacuum unit.

Time-dependent vacuum operations

The vacuum unit can also be time-controlled and can be activated and deactivated.

- The switch-on/off moment can be influenced via the CNC commands.

- For this purpose the G11 command is not required.

Method of functioning for routing commands

Routing command	G11 function	Time-dependent vacuum operations
G1	●	●
G2	●	●
G3	●	●
G45	○	●
G46	○	●
G47	●	●
G48	●	●
G49	○	●
G50	○	●

Example

In the machine parameters a distance of 5.0 mm is defines as G11. As soon as the remaining routing track is only 5.0 mm the following steps are carried out.

- the routing feed rate of the XY-axes is reduced,
- the air pressure for the pressure foot is increased and
- the vacuum unit is switched off.

Marginal condition	Setting
Format	5000

```
X50.Y50.T0T2
X70.Y50.G2R10. F1.2
X110.Y90.G11
```

Activate finish-routing function. When the remaining routing track is reached the G11 functions are automatically executed.

```
X110.Y90.T0
```

Related topics

G1: Rout Straight Line, page 39

G2: Rout Circular Arc Clockwise, page 41

G3: Rout Circular Arc Counterclockwise, page 43

G47: Rout Out Disk Counterclockwise, page 49

G48: Rout Out Disk Clockwise, page 51

G49: Rout Out Rectangle Counterclockwise, page 53

G50: Rout Out Rectangle Clockwise, page 55

G43: Cut Area of the Stored Pattern Completely

1000	3000	5000
–	–	●

Command	Description
XxYy Gnn G43	Cut area of the stored pattern completely

Argument	Description
xy	XY-coordinates of the circle center
nn	Routing function for stored patterns: G45, G46, G49 or G50

The command G43 defines that a stored pattern programmed with a finish-routing function is cut out completely. This applies to the following finish-routing functions:

- G45/G46: rout out full circle
- G49/G50: rout out rectangle or square
- The G43 function is not saved. It only applies in the programmed block.

The material in the inner area of a routing contour is cut out completely. This prevents for example blocking the vacuum unit by too many chips.

Fig. 50: Cut the inner area completely

Related topics

G45: Rout Out Full Circle Counterclockwise, page 46

G46: Rout Out Full Circle Clockwise, page 48

G49: Rout Out Rectangle Counterclockwise, page 53

G50: Rout Out Rectangle Clockwise, page 55

3.4 Program Two Routing Contours

Every routing contour is considered as individual routing operation.

- Every routing contour is started and finished by the T0 commands.
- This ensures that for example complete routing profiles can be copied to another position in the part program.

Example

The exemplary programs show the programming of two straight lines each.

- The first routing contour starts at X100.Y100. and ends at X110.Y100.
- The second routing contour starts at X200.Y200. and ends at X220.Y200.
- The tool with the number T5 is used.
- The routing feed rate is 1.5 m/min.

Marginal condition	Setting
Format	5000

`%%5000`	Start position of the 1st routing contour. Tool change to the cutter. Positioning with retracted Z-axis
`( 1. CONTOUR )`	
`X100.Y100.T0T5`	End position of the straight line. Routing function = straight line. Routing feed rate = 1.5 m/min
`X110.Y100.G1F1.5`	End position of the 1st routing track. extraction of the cutter
`X110.Y100.T0`	End position of the 1st routing contour. The routing function is deactivated. The Z axis is retracted
`( 2. CONTOUR )`	
`X200.Y200.T0`	Start position of the 2nd routing contour. Positioning with retracted Z axis. Tool number = T5
`X220.Y200.G1`	End position of the straight line. Routing function = straight line. Routing feed rate = 1.5 m/min
`X220.Y200.T0`	End position of the 2nd routing contour. The routing function is deactivated. The Z axis is retracted

3.5 Connected Routing Contours

Connected routing contours are programmed as follows.

- Within a routing contour either track-dependent routing functions (G3 .. G1 .. G2 ..) or finished contours (G45 .. G46 .. G50 ..) can be connected in any order .
- A routing command remains active until another routing command or the T0 command occurs.

Example

The following routing functions are used in the exemplary program:

- a circular arc with a radius of 25.0 mm
- two straight lines

Marginal condition	Setting
Format	5000

`X100.Y100.T5T0`	Start position of a routing contour. The cutter is T5. Positioning with retracted Z-axis
`X150.Y100.G3R25.F1.5`	End position of the circular arc. Routing function = circular arc. Radius = 25.0 mm. Routing feed rate = 1.5 m/min
`X200.Y100.G1`	End position of the 1st straight line. Routing function = even
`X200.Y150.`	End position of the 2nd straight line
`X200.Y150.T0`	End position of the routing contour. The Z axis is retracted

4 Scaling

This section describes commands for up-sized/down-sized execution of a program section.

M92: Deactivate Scaling of Section

1000	3000	5000
–	–	•

Command	Description
X*x*Y*y* M92	Deactivate scaling function
X*x*Y*y* M93	Activate scaling function

Argument	Description
xy	Scaled execution already applies for these coordinates

Requirement
The CNC command NOSMDS is set.

The scaling commands can be used for limiting the scaling function to certain program sections.

- The scaling parameters are defined with the CNC command SA.
- The scaling center is defined with the CNC command SAZ.

Example

The holes for board finishing shall not be scaled in a part program.

Marginal condition	Setting
Format	5000

`%%5000`	
`M99,RESET`	Call subprogram "RESET". Therein you program the default settings of all CNC commands.
`M49,SAX3.Y100.`	Define scaling factors
`M49,SAZX200.Y400.`	Define scaling center
`X12.345Y34.567T1M31` **`M93`**	Beginning of the program section to be scaled
`...`	
`X45.6Y43.21M50M30` **`M92`**	End of the program section to be scaled
`M99,FINAL`	Execute drill holes for production
`M99,RESET`	Call subprogram "RESET". All used CNC commands are reset.

Related topics

M93: Activate Scaling of Section, page 70

M93: Activate Scaling of Section

1000	3000	5000
–	–	•

Command	Description
X*x*Y*y* M92	Deactivate scaling function
X*x*Y*y* M93	Activate scaling function

Argument	Description
xy	Scaled execution already applies for these coordinates

Requirement

The CNC command NOSMDS is set.

The scaling commands can be used for limiting the scaling function to certain program sections.

- The scaling parameters are defined with the CNC command SA.
- The scaling center is defined with the CNC command SAZ.

Example

The holes for board finishing shall not be scaled in a part program.

Marginal condition	Setting
Format	5000

`%%5000`	
`M99,RESET`	Call subprogram "RESET". Therein you program the default settings of all CNC commands.
`M49,SAX3.Y100.`	Define scaling factors
`M49,SAZX200.Y400.`	Define scaling center
`X12.345Y34.567T1M31` **`M93`**	Beginning of the program section to be scaled
`...`	
`X45.6Y43.21M50M30` **`M92`**	End of the program section to be scaled
`M99,FINAL`	Execute drill holes for production
`M99,RESET`	Call subprogram "RESET". All used CNC commands are reset.

Related topics

M92: Deactivate Scaling of Section, page 69

5 Repeat, Offset and Mirror Patterns

Every part program and every section in a part program can be offset as desired.

- Define a new zero point for the section concerned.
- Use the M31 to M30 commands to bracket a program section.
- Up to 6 (CNC 8x.00) bracketed program sections can be nested (including called subprograms).
- Bracketing program sections facilitates manual creation of a part program.
- Use the following commands to bracket a program section:
  - M31: LEFT Bracket
  - M30: RIGHT Bracket
- Each programmed M31 command expects an M30 command.

Note

- Bracketing commands must be programmed consistently. Avoid program un-bracketed program section.
- Using camera commands in a nested program section (nested Step-and-Repeat) requires verification of the complete part program.

Additional offset possibilities

In addition to zero point offset further rotating and mirroring commands can be programmed.

M command	Effect
M50	No rotation/mirroring
M60	Versions 1,3,6,8: counterclockwise rotation by 90° Versions 2,4,5,7: clockwise rotation by 90°
M70	Mirroring around the Y-axis
M80	Mirroring around the X-axis
M90	Rotation by 180°
M60M70	Versions 1,3,6,8: mirroring around the Y-axis and counterclockwise rotation by 90° Versions 2,4,5,7: Mirroring around the Y-axis and clockwise rotation by 90°
M60M80	Versions 1,3,6,8: mirroring around the X-axis and counterclockwise rotation by 90° Versions 2,4,5,7: mirroring around the X-axis and clockwise rotation by 90°
M60M90	Versions 2,4,5,7: Mirroring around the Y-axis and clockwise rotation by 90° Versions 1,3,6,8: mirroring around the Y-axis and counterclockwise rotation by 90°

Calculating the zero point

To calculate the offset coordinates, follow these steps:

- Actual position: Determine the current position of a distinctive coordinate (measurement, graphic display, coordinate value).
- Reference position: Determine the target position (measurement, graphic display).
- Enter the values in the corresponding offset command.

5.1 Tool Change

If several tools are required in a part program with multiple patterns it might happen that possibly unnecessary tool changes are carried out.

- If the offset commands are only entered at the end of the part program (first example), all tool changes are unnecessarily carried out by the CNC during the execution of every single pattern.
- The correct programming is shown in the second example: Before every tool change it must be ensured that all patterns are drilled with this tool.

Example: Unnecessary tool changes

Since the offset commands are only programmed at the end of the program, the execution of the program is repeated (including all tool changes). This is not recommended for programming programs.

Marginal condition	Setting
Format	5000

`( START: T01 )`	
`X0.000 Y0.000 T00 T01 M25`	Section: T01 (start)
`X0.000 Y0.000G45R20.F10`	
`X0.000 Y0.000 T00`	Section: T01 (end)
`( START: T02 )`	Section: T02 (start)
`X10.000 Y-20.000 T02`	
`X20.000 Y-10.000`	Section: T02 (end)
`( OFFSET COMMANDS )`	
`X25.000 Y25.000 M50`	Pattern programming
`X75.000 Y75.000 M50 M08`	

Example: Correct programming

This example shows that all necessary offset commands must be programmed at the end of a program section to avoid unnecessary tool changes.

Marginal condition	Setting
Format	5000

`( START: T01 )`	
`X0.000 Y0.000 T00 T01 M08`	Section: T01 (start)
`X0.000 Y0.000G45R20.F10`	
`X0.000 Y0.000 T0`	
`( OFFSET COMMANDS )`	
`X25.000 Y25.000 M50`	Pattern programming
`X75.000 Y75.000 M50 M25`	Section: T01 (end)
`( START: T02 )`	
`X10.000 Y-20.000 T02 M25`	Section: T02 (start)
`X20.000 Y-10.000`	

`( OFFSET COMMANDS )`	
`X25.000 Y25.000 M50`	Pattern programming
`X75.000 Y75.000 M50 M08`	Section: T02 (end)

5.2 Program Section

This chapter describes commands for dividing a part program into several sections.

M31: LEFT Bracket

1000	3000	5000
–	•	•

Command	Description
X*x*Y*y* M31	LEFT bracket (beginning of a program section)
X*x*Y*y* M30	RIGHT bracket (end of a program section)

Argument	Description
xy	The coordinates already belong to the bracketed section.

The commands M31 and M30 are used for bracketing program sections.

- The M31 introduces a new program section.
- The M30 command terminates the last not yet finished program section.
- If several program sections have been introduced by a M31 command, the M30 command terminates the section that has been introduces by the last M31 command.
- All open program sections (M31) must be closed again (M30).
- The maximum admissible number of program sections is 6 (CNC 8x.00) (nesting).

Note

Camera commands are not allowed nested program sections (nested Step-and-Repeat, determine compensation values).

Example

A complete offset is entered at the end of a simple program.

Marginal condition	Setting
Format	5000

`X10.Y30.T1 M31 M31`	Open section T1 and left bracket of complete bracketing.
`...`	
`X30.Y40.M50`	
`X70.Y20.M50 M30`	Close section T1
`X10.Y30.T2 M31`	Open section T2
`...`	
`X30.Y40.M50`	

`X70.Y20.M50 M30`	Close section T2
`( FINAL OFFSET )`	
`X123.45Y234.67M50 M30`	Right bracket of complete bracketing

Related topics

M30: RIGHT Bracket, page 74

M30: RIGHT Bracket

1000	3000	5000
•	•	•

Command	Description
X*x*Y*y* M31	LEFT bracket (beginning of a program section)
X*x*Y*y* M30	RIGHT bracket (end of a program section)

Argument	Description
xy	The coordinates already belong to the bracketed section.

The commands M31 and M30 are used for bracketing program sections.

- The M31 introduces a new program section.
- The M30 command terminates the last not yet finished program section.
- If several program sections have been introduced by the M31 command, the M30 command terminates the section that has been introduced by the last M31 command.
- All open program sections opened with M31 must be closed again.
- The maximum admissible number of program sections is 6 (CNC 8x.00) (nesting).

Note

Make sure for nested Step-and-Repeat to program the camera commands in the same program section as the corresponding drilling/routing. Otherwise the nested Step-and-Repeat programming of the camera commands must be of the same structure as the drilling/routing in order to ensure correct assignment of "G39,SAVE" and "G39,LOAD" (if used).

Example

A complete offset is entered at the end of a simple program.

Marginal condition	Setting
Format	5000

`X10.Y30.T1 M31 M31`	Open section T1 and left bracket of complete bracketing.
`...`	
`X30.Y40.M50`	
`X70.Y20.M50 M30`	Close section T1.
`X10.Y30.T2 M31`	Open section T2.
`...`	
`X30.Y40.M50`	

`X70.Y20.M50 M30`	Close section T2.
`( FINAL OFFSET )`	
`X123.45Y234.67M50 M30`	Right bracket of complete bracketing

Related topics

M31: LEFT Bracket, page 73

5.3 Step-and-Repeat

Offsetting and repeating program sections

Information on offset commands:

- Generally, coordinate values are interpreted as absolute values.
- If at least one offset command is programmed at the end of a program section, only this (or these) offset command(s) is (are) considered for the execution of the program section.
- That means: For execution of a program section exactly at the programmed position and additional shifted execution at least two offset lines must be programmed.
- The offset structure must be programmed after every program section.
- A typical program section is the program section for a tool number.
- Every program section must be bracketed with the commands M31 to M30.
- Only this program structure allows shifting a complete part program by means of an additional M31-M30 bracket (*nested Step-and-Repeat*).

Marginal condition	Setting
Format	5000

`( SECTION: T5 )`	
`X0.000Y0.000T0T5 M31 M31`	
`X0.000Y0.000G45R20.F1.2`	
`X0.000Y0.000T0`	
`( STEP-AND-REPEAT )`	
`X0Y0 M50`	Execution at the programmed position
`X75.000Y75.000 M50 M30`	Repeated and shifted execution
`( SECTION: T6 )`	
`X30.000Y40.000T0T6 M31`	Beginning of the section T6
`X30.000Y40.000G45R20.F1.2`	
`X30.000Y40.000T0`	
`( STEP-AND-REPEAT )`	
`X0Y0 M50`	Execution at the programmed position
`X75.000Y75.000 M50 M30`	Repeated and shifted execution
`( FINAL OFFSET )`	
`X123.43Y536.224 M50 M30`	Right bracket of complete bracketing

5.4 Zero Point Calculation

Example for zero point calculation

Determining offset condition

- Based on the graphic view you determined that the program must be rotated by 90° with the commands M60M90 for the production.
- This rotation is programmed by additional M31-M30 bracketing the complete program.

`%%5000`	
`X..Y..T1M31` **`M31`**	Beginning of section T1: Left bracket for the complete rotation
`...`	
`XYM50`	Offset commands for pattern repetition
`X..Y..M50M30`	End of section T1
`X..Y..T2M31`	Beginning of section T2
`...`	
`XYM50`	Offset commands for pattern repetition
`X..Y..M50M30`	End of section T2
`( ROTATION )`	
`XY M60M90 M30`	Complete rotation and right bracket for the complete rotation

Determining offset value

- In the graph you will see that the display is shifted out of the working area.
- Determine the ACTUAL position and the REFERENCE position for a distinctive coordinate.
- The difference between these two coordinate value is the required offset.
- Then, program this simple offset by another M31-M30 bracketing (M50 command).

`%%5000`	
`X..Y..T1M31 M31` **`M31`**	Beginning of section T1: Left bracket for the complete rotation. Left bracket for the complete offset
`...`	
`XYM50`	Offset commands for pattern repetition
`X..Y..M50M30`	End of section T1
`X..Y..T2M31`	Beginning of section T2
`...`	
`XYM50`	Offset commands for pattern repetition
`X..Y..M50M30`	End of section T2
`( COMPLETE ROTATION )`	
`XY M60M90 M30`	Complete rotation and right bracket for the complete rotation
`( FINAL OFFSET )`	
`X123.34Y-223.11 M50 M30`	Complete offset and right bracket for the complete offset

5.5 Additional Offset Possibilities

This section describes commands allowing to offset, rotate or mirror program sections.

M50: Simple Offset

1000	3000	5000
•	•	•

Command	Description
X*x*Y*y* M50	Shift program section (no rotation/mirroring)

Argument	Description
xy	Offset coordinates

The M50 command is used for programming a simple offset.

- The offset coordinates define the new zero point of the bracketed program section.
- If at least one offset command is programmed at the end of a program section, only this (or these) offset command(s) is (are) considered for the execution of the program section.

Calculating offset coordinates

The offset coordinates are calculated based on a drill hole or on a routing track. Determine the actual coordinate position (ACTUAL) and the new coordinate position (REFERENCE) and enter the values into the formula.

- $X_{M50} = X_{Ref} - X_{Act}$
- $Y_{M50} = Y_{Ref} - Y_{Act}$

Example

The programmed routing contour shall be executed twice after shifting.

- Zero point of the 1st execution = X30.Y40.
- Zero point of the 2nd execution = X70.Y20.

Marginal condition	Setting
Format	5000
Axis version	1

`X10.Y30.T0T2 M31`	Beginning of the routing contour. Since offset commands are programmed at the beginning of the section, the section is only executed under consideration of the offset coordinates.
`X10.Y40.G1F1.2`	Position of the 1st execution: X40.Y80.; Position of the 2nd execution: X80.Y60.
`X20.Y40.`	Position of the 1st execution: X50.Y80.; Position of the 2nd execution: X90.Y60.
`X20.Y10.`	Position of the 1st execution: X50.Y50.; Position of the 2nd execution: X90.Y30.
`X10.Y10.`	Position of the 1st execution: X40.Y50.; Position of the 2nd execution: X80.Y30.
`X10.Y10.T0`	End of the routing contour

`X30.Y40.M50`	First offset block. The XY-coordinates are added to every coordinate of the above section and then executed. X_{M50} = X40. − X10. = X30.; Y_{M50} = Y50. − Y10. = Y40.
`X70.Y20.M50 M30`	Second offset block. The XY-coordinates are added to every coordinate of the above section and then executed. X_{M50} = X80. − X10. = X70.; Y_{M50} = Y30. − Y10. = Y20.

Two repetitions are programmed with two offset commands. Since every offset command includes a zero offset (offset coordinates), no execution will be done at the programmed position (thin contour).

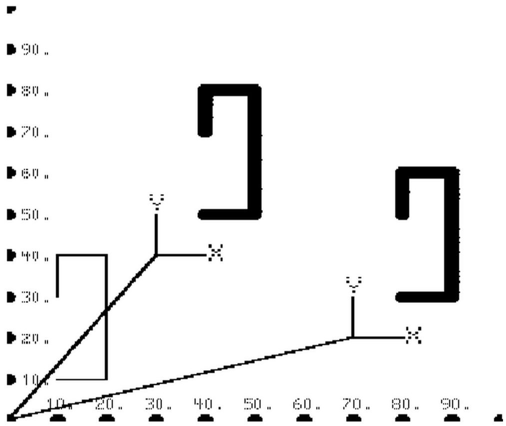


Fig. 51: Two routed patterns

Note

The **thick lines** show the programmed contours under consideration of the offset commands. The **thin lines** only serve for a clear view. They are not part of the part program.

M50V2: Pattern Repetition

1000	3000	5000
–	•	•

Command	Description
XY M50V2	Activate V2-repetition
X*x*Y*y* M50W*w* M30	Define the number of patterns.

Argument	Description
xy	Distance between the 1st and the last zero point of the pattern row
w	Number of patterns in the pattern row

The command combination M50V2 is used for programming the repetition of a pattern.

- This allows easy programming of any pattern fields.
- Only the M50 command is recommended as offset condition.
- The M50 command must be programmed in every offset line.
- The complete section must be bracketed with the necessary M31-M30 commands.
- To obtain a clear structure of the program, you are recommended to follow the examples.
- Consider the possibility that the exact machine position can be realized by completely M31-M30 bracketing the program (see examples).
- Take care that the M31-M30 are correctly used.
- The CNC calculates the exact definition between the patterns during the execution.

Example: pattern row

Five patterns are to be programmed into the X-direction.

- pattern width = 100 mm
- pattern distance = 10 mm

Calculation for the zero point of the last pattern:

- last zero point = (4 x 100 mm) + (4 x 10 mm) = 440 mm

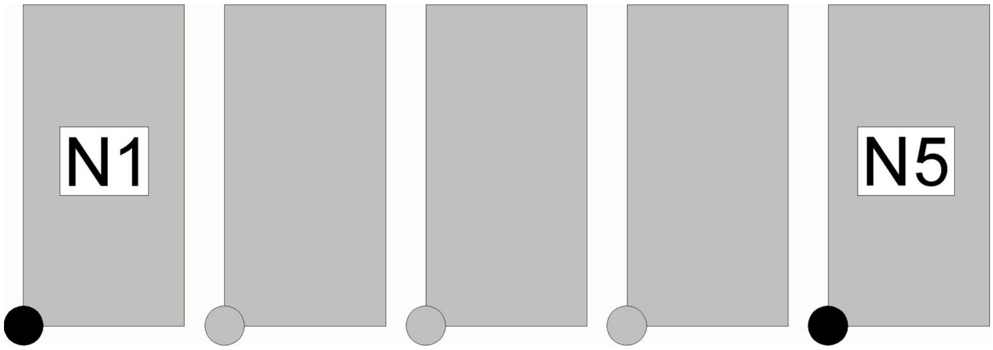


Fig. 52: Pattern row

Marginal condition	Setting
Format	5000
Axis version	1

`%%5000`	
`( SECTION T1 )`	
`X..Y..T1 M31 M31`	M31: LEFT bracket for complete program; M31: LEFT bracket for section T1
`...`	
`X0Y0 M50V2`	Activate V2-repetition
`X440.Y M50W5 M30`	Define last zero point and number: M30: RIGHT bracket for section T1
`( SECTION T2 )`	
`X..Y..T2 M31`	M31: LEFT bracket for section T2
`...`	
`X0Y0 M50V2`	Activate V2-repetition
`X440.Y M50W5 M30`	M30: RIGHT bracket for section T2
`( SECTION T.. )`	
`...`	Identical programming for every section
`( FINAL OFFSET )`	
`X123.45Y23.65 M50 M30`	Complete offset = M30: RIGHT bracket for complete program

Example: pattern column

Three patterns are to be programmed into the Y-direction.

- pattern height = 150 mm
- pattern distance = 10 mm

Calculation for the zero point of the last pattern:

- last zero point = (2 x 150 mm) + (2 x 10 mm) = 320 mm

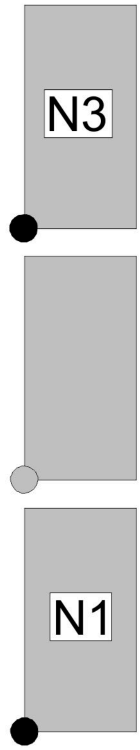


Fig. 53: Pattern column

Marginal condition	Setting
Format	5000
Axis version	1

`%%5000`	
`( SECTION T1 )`	
`X..Y..T1 M31 M31`	M31: LEFT bracket for complete program; M31: LEFT bracket for section T1
`...`	
`X0Y0 M50V2`	Activate V2-repetition
`XY320. M50W3 M30`	Define last zero point and number: M30: RIGHT bracket for section T1
`( SECTION T2 )`	
`X..Y..T2 M31`	M31: LEFT bracket for section T2
`...`	
`X0Y0 M50V2`	Activate V2-repetition
`XY320. M50W3 M30`	M30: RIGHT bracket for section T2
`( SECTION T.. )`	
`...`	Identical programming for every section
`( FINAL OFFSET )`	
`X123.45Y23.65 M50 M30`	Complete offset = M30: RIGHT bracket for complete program

Example: pattern field

Fifteen patterns are to be programmed (5 patterns into X-direction and 3 rows of five into Y-direction).

- pattern width = 100 mm
- pattern height = 150 mm
- pattern distance = 10 mm

Calculation for the zero points of the last patterns:

- last X-axis zero point = (4 x 100 mm) + (4 x 10 mm) = 440 mm
- last Y-axis zero point = (2 x 150 mm) + (2 x 10 mm) = 320 mm

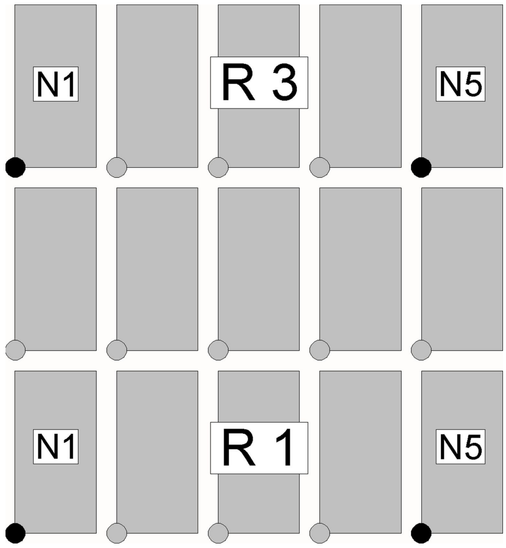


Fig. 54: Pattern field

Marginal condition	Setting
Format	5000
Axis version	1

Note: Two M31 commands are at the beginning of every section. There are even three M31 commands at the beginning of the 1st section (one command for complete bracketing).

`%%5000`	
`( SECTION T1 )`	
`X..Y..T1 M31M31 M31`	M31: LEFT bracket for complete program; M31: LEFT bracket for section T1 (column); M31: LEFT bracket for section T1 (row)
`...`	
`X0Y0 M50V2`	Activate V2-repetition (row).
`X440.Y M50W5 M30`	Define last zero point and number: M30: RIGHT bracket for section T1 (row)
`X0Y0 M50V2`	Activate V2-repetition (column).
`XY320. M50W3 M30`	Define last zero point and number: M30: RIGHT bracket for section T1 (column)
`( SECTION T2 )`	
`X..Y..T2 M31M31`	M31: LEFT bracket for section T1 (column); M31: LEFT bracket for section T1 (row)
`...`	
`X0Y0 M50V2`	Activate V2-repetition (row).
`X440.Y M50W5 M30`	M30: RIGHT bracket for section T2
`X0Y0 M50V2`	Activate V2-repetition (column).
`XY320. M50W3 M30`	Define last zero point and number: M30: RIGHT bracket for section T1
`( SECTION T.. )`	
`...`	Identical programming for every section

(FINAL OFFSET)	
X123.45Y23.65 M50 M30	Complete offset = M30: RIGHT bracket for complete program

Related topics

V2: Drill Single Row of Holes, page 22

M60: Offset with Rotation

1000	3000	5000
•	•	•

Command	Description
X*x*Y*y* M60	Shift program section and rotate it by 90°

Argument	Description
xy	Offset coordinates

The M60 command is used for programming an offset and rotation.

- The offset coordinates define the new zero point of the bracketed program section.
- If at least one offset command is programmed at the end of a program section, only this (or these) offset command(s) is (are) considered for the execution of the program section.

Direction of rotation

The direction of rotation depends on the axis version.

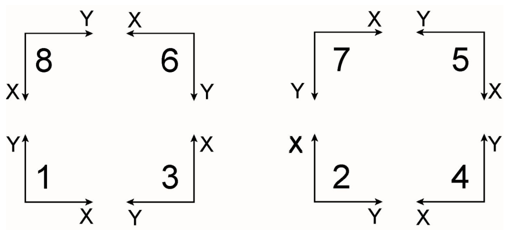


Fig. 55: Axis versions

Axis version	Direction of rotation
1, 3, 6, 8	Counterclockwise
2, 4, 5, 7	Clockwise

Calculating offset coordinates

The offset coordinates are calculated based on a drill hole or on a routing track. Determine the actual coordinate position (ACTUAL) and the new coordinate position (REFERENCE) and enter the values into the formula.

- $X_{M60} = X_{Ref} + Y_{Act}$
- $Y_{M60} = Y_{Ref} - X_{Act}$

Example

The programmed routing contour shall be executed twice after shifting. In addition, every pattern shall be rotated by 90° into counterclockwise direction. The axis version is version 1.

- Zero point of the 1st pattern = X60.Y60.

- Zero point of the 2nd pattern = X90.Y10.
- Required offset condition = M60

Marginal condition	Setting
Format	5000
Axis version	1

`X10.Y30.T0T2 M31`	Beginning of the routing contour. Since offset commands are programmed at the beginning of the section, the section is only executed under consideration of the offset coordinates.
`X10.Y40.G1 F1.2`	Position of the 1st execution: X20.Y70.; Position of the 2nd execution: X50.Y20.
`X20.Y40.`	Position of the 1st execution: X20.Y80.; Position of the 2nd execution: X50.Y30.
`X20.Y10.`	Position of the 1st execution: X50.Y80.; Position of the 2nd execution: X80.Y30.
`X10.Y10.`	Position of the 1st execution: X50.Y70.; Position of the 2nd execution: X80.Y20.
`X10.Y10.T0`	End of the routing contour
`X60.Y60.M60`	First offset block. The XY-coordinates are added to every coordinate of the above section and then executed. X_{M60} = X30. + Y30. = X60.; Y_{M60} = Y70. − X10. = Y60.
`X90.Y10.M60 M30`	Second offset block. The XY-coordinates are added to every coordinate of the above section and then executed. X_{M60} = X60. + Y30. = X90.; Y_{M60} = Y20. − X10. = Y10.

Two repetitions are programmed with two offset commands. Since every offset command includes a zero offset (offset coordinates), no execution will be done at the programmed position (thin contour).

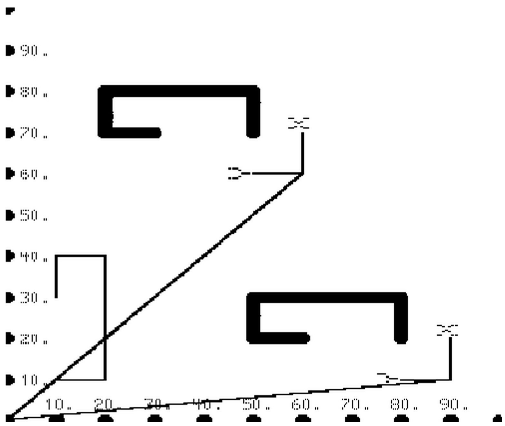


Fig. 56: Two routed patterns

Note

The **thick lines** show the programmed contours under consideration of the offset commands. The **thin lines** only serve for a clear view. They are not part of the part program.

M60M70: Offset with Mirroring and Rotation

1000	3000	5000
•	•	•

Command	Description
X*x*Y*y* M60M70	Shift program section, mirror it around the Y-axis and rotate it by 90°.

Argument	Description
xy	Offset coordinates

The M60M70 command is used for defining an offset, mirroring and rotation.

- The offset coordinates define the new zero point of the bracketed program section.
- If at least one offset command is programmed at the end of a program section, only this (or these) offset command(s) is (are) considered for the execution of the program section.
- First, mirroring is carried out on the basis of the machine zero. Then, the program section is rotated under consideration of the new axis version.

Direction of rotation

The direction of rotation depends on the axis version resulting from mirroring.

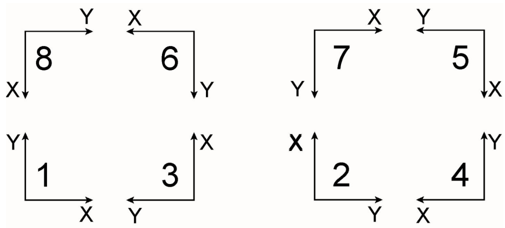


Fig. 57: Axis versions

Axis version	Direction of rotation of the M60 command
1, 3, 6, 8	Counterclockwise
2, 4, 5, 7	Clockwise

Example: First, the M70 command triggers mirroring the program section around the Y-axis on the basis of the axis version 1. Now, the resulting axis version is 4! In this axis version, the M60 command causes a rotation by 90° in clockwise direction.

Calculating offset coordinates

The offset coordinates are calculated based on a drill hole or on a routing track. Determine the actual coordinate position (ACTUAL) and the new coordinate position (REFERENCE) and enter the values into the formula.

- $X_{M60M70} = X_{Ref} - Y_{Act}$
- $Y_{M60M70} = Y_{Ref} - X_{Act}$

Example

The programmed routing contour shall be executed twice after shifting. In addition, every pattern shall be mirrored around the Y-axis first and then be rotated by 90° into counterclockwise direction. The axis version is version 1.

- Zero point of the 1st pattern = X10.Y60.
- Zero point of the 2nd pattern = X30.Y10.
- Required offset condition = M6070

Marginal condition	Setting
Format	5000
Axis version	1

```
X10.Y30.T0T2 M31
```

Beginning of the routing contour. Since offset commands are programmed at the beginning of the section, the section is only executed under consideration of the offset coordinates.

`X10.Y40.G1 F1.`	Position of the 1st execution: X50.Y70.; Position of the 2nd execution: X70.Y20.
`X20.Y40.`	Position of the 1st execution: X50.Y80.; Position of the 2nd execution: X70.Y30.
`X20.Y10.`	Position of the 1st execution: X20.Y80.; Position of the 2nd execution: X40.Y30.
`X10.Y10.`	Position of the 1st execution: X20.Y70.; Position of the 2nd execution: X40.Y20.
`X10.Y10.T0`	End of the routing contour
`X10.Y60.M60M70`	First offset block. The XY-coordinates are added to every coordinate of the above section and then executed. X_{M60M70} = X40. − Y30. = X10.; Y_{M60M70} = Y70. − X10. = Y60.
`X30.Y10.M60M70 M30`	Second offset block. The XY-coordinates are added to every coordinate of the above section and then executed. X_{M60M70} = X60. − Y30. = X30.; Y_{M60M70} = Y20. − X10. = Y10.

Two repetitions are programmed with two offset commands. Since every offset command includes a zero offset (offset coordinates), no execution will be done at the programmed position (thin contour).

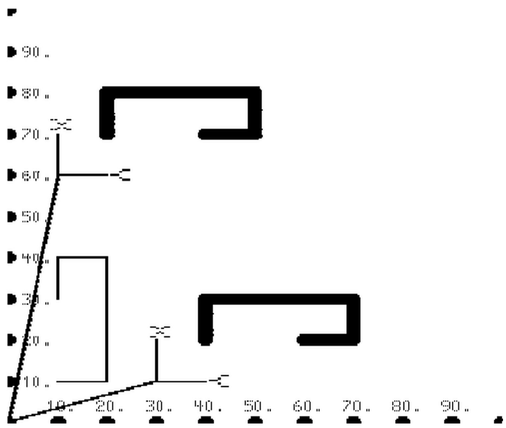


Fig. 58: Two routed patterns

Note

The **thick lines** show the programmed contours under consideration of the offset commands. The **thin lines** only serve for a clear view. They are not part of the part program.

M60M80: Offset with Mirroring and Rotation

1000	3000	5000
•	•	•

Command	Description
X*x*Y*y* M60M80	Shift program section, mirror it around the X-axis and rotate it by 90°

Argument	Description
xy	Offset coordinates

The M60M80 command is used for defining an offset, mirroring and rotation.

- The offset coordinates define the new zero point of the bracketed program section.
- If at least one offset command is programmed at the end of a program section, only this (or these) offset command(s) is (are) considered for the execution of the program section.
- First, mirroring is carried out on the basis of the machine zero. Then, the program section is rotated under consideration of the new axis version.

Direction of rotation

The direction of rotation depends on the axis version resulting from mirroring.

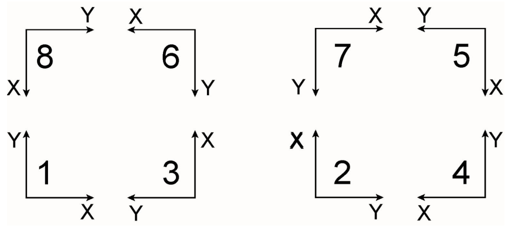


Fig. 59: Axis version

Axis version	Direction of rotation of the M60 command
1, 3, 6, 8	Counterclockwise
2, 4, 5, 7	Clockwise

Example: First, the M80 command triggers mirroring the program section around the X-axis on the basis of the axis version 1. Now, the resulting axis version is 7! In this axis version, the M60 command causes a rotation by 90° in clockwise direction.

Calculating offset coordinates

The offset coordinates are calculated based on a drill hole or on a routing track. Determine the actual coordinate position (ACTUAL) and the new coordinate position (REFERENCE) and enter the values into the formula.

- $X_{M60M80} = X_{Ref} + Y_{Act}$
- $Y_{M60M80} = Y_{Ref} + X_{Act}$

Example

The programmed routing contour shall be executed twice after shifting. In addition, every pattern shall be mirrored around the X-axis first and then be rotated by 90° into counterclockwise direction. The axis version is version 1.

- Zero point of the 1st pattern = X50.Y80.
- Zero point of the 2nd pattern = X90.Y40.
- Required offset condition = M6080

Marginal condition	Setting
Format	5000
Axis version	1

`X10. Y30.T0T2 M31`	Beginning of the routing contour. Since offset commands are programmed at the beginning of the section, the section is only executed under consideration of the offset coordinates.
`X10. Y40. G1 F1.`	Position of the 1st execution: X10.Y70.; Position of the 2nd execution: X50.Y30.
`X20. Y40.`	Position of the 1st execution: X10.Y60.; Position of the 2nd execution: X50.Y20.
`X20. Y10.`	Position of the 1st execution: X40.Y60.; Position of the 2nd execution: X80.Y20.
`X10. Y10.`	Position of the 1st execution: X40.Y70.; Position of the 2nd execution: X80.Y30.
`X10. Y10. T0`	End of the routing contour

`X50. Y80. M60M80`	First offset block. The XY-coordinates are added to every coordinate of the above section and then executed. X_{M60M80} = X20. − Y30. = X50.; Y_{M60M80} = Y70. − X10. = Y80.
`X90. Y40. M60M80 M30`	Second offset block. The XY-coordinates are added to every coordinate of the above section and then executed. X_{M60M80} = X60. − Y30. = X90.; Y_{M60M80} = Y30. − X10. = Y40.

Two repetitions are programmed with two offset commands. Since every offset command includes a zero offset (offset coordinates), no execution will be done at the programmed position (thin contour).

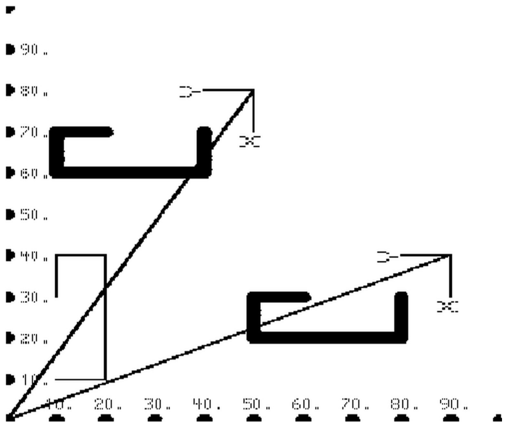


Fig. 60: Two routed patterns

Note

The **thick lines** show the programmed contours under consideration of the offset commands. The **thin lines** only serve for a clear view. They are not part of the part program.

M60M90: Offset with Rotation

1000	3000	5000
•	•	•

Command	Description
X*x*Y*y* M60M90	Shift program section and rotate it by 90°

Argument	Description
xy	Offset coordinates

The M60M90 command is used for programming an offset and rotation.

- The offset coordinates define the new zero point of the bracketed program section.
- If at least one offset command is programmed at the end of a program section, only this (or these) offset command(s) is (are) considered for the execution of the program section.

Direction of rotation

The direction of rotation depends on the axis version.

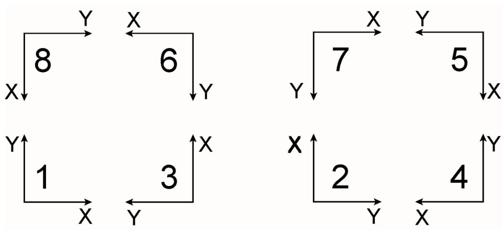


Fig. 61: Axis version

Axis version	Direction of rotation
1, 3, 6, 8	Clockwise
2, 4, 5, 7	Counterclockwise

Calculating offset coordinates

The offset coordinates are calculated based on a drill hole or on a routing track. Determine the actual coordinate position (ACTUAL) and the new coordinate position (REFERENCE) and enter the values into the formula.

- $X_{M60M90} = X_{Ref} - Y_{Act}$
- $Y_{M60M90} = Y_{Ref} + X_{Act}$

Example

The programmed routing contour shall be executed twice after shifting. In addition, every pattern shall be rotated by 90° into a clockwise direction. The axis version is version 1.

- Zero point of the 1st pattern = X10.Y90.
- Zero point of the 2nd pattern = X50.Y50.
- Required offset condition = M6090

Marginal condition	Setting
Format	5000
Axis version	1

`X10. Y 30.T0T2 M31`	Beginning of the routing contour. Since offset commands are programmed at the beginning of the section, the section is only executed under consideration of the offset coordinates.
`X10. Y40. G1 F1.`	Position of the 1st execution: X50.Y80.; Position of the 2nd execution: X90.Y40.
`X20. Y40.`	Position of the 1st execution: X50.Y70.; Position of the 2nd execution: X90.Y30.
`X20. Y10.`	Position of the 1st execution: X20.Y70.; Position of the 2nd execution: X60.Y30.
`X10. Y10.`	Position of the 1st execution: X20.Y80.; Position of the 2nd execution: X60.Y40.
`X10. Y10. T0`	End of the routing contour
`X10. Y90. M60M90`	First offset block. The XY-coordinates are added to every coordinate of the above section and then executed. X_{M60M90} = X40. − Y30. = X10.; Y_{M60M90} = Y80. + X10. = Y90.
`X50. Y50. M60M90 M30`	Second offset block. The XY-coordinates are added to every coordinate of the above section and then executed. X_{M60M90} = X80. − Y30. = X50.; Y_{M60M90} = Y40. + X10. = Y50.

Two repetitions are programmed with two offset commands. Since every offset command includes a zero offset (offset coordinates), no execution will be done at the programmed position (thin contour).

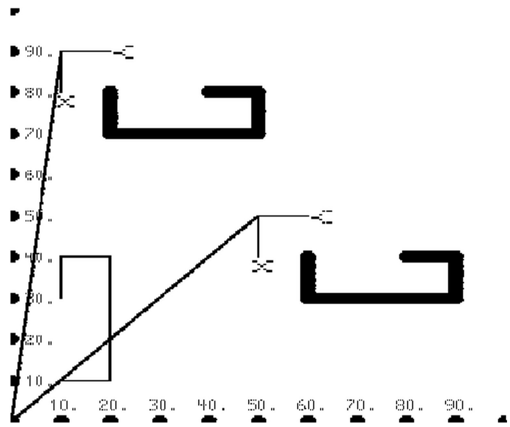


Fig. 62: Two routed patterns

Note

The **thick lines** show the programmed contours under consideration of the offset commands. The **thin lines** only serve for a clear view. They are not part of the part program.

M70: Offset with Mirroring around the Y-axis

1000	3000	5000
•	•	•

Command	Description
X*x*Y*y* M70	Shift program section and mirror it around the Y-axis

Argument	Description
xy	Offset coordinates

The M70 command is used for programming an offset and mirroring.

- The offset coordinates define the new zero point of the bracketed program section.
- If at least one offset command is programmed at the end of a program section, only this (or these) offset command(s) is (are) considered for the execution of the program section.

Calculating offset coordinates

The offset coordinates are calculated based on a drill hole or on a routing track. Determine the actual coordinate position (ACTUAL) and the new coordinate position (REFERENCE) and enter the values into the formula.

- $X_{M70} = X_{Ref} + X_{Act}$
- $Y_{M70} = Y_{Ref} - Y_{Act}$

Example

The programmed routing contour shall be executed twice after shifting. In addition, every pattern shall be mirrored/rotated around the Y-axis. The axis version is version 1.

- Zero point of the 1st pattern = X40.Y50.
- Zero point of the 2nd pattern = X80.Y10.
- Required offset condition = M70

Marginal condition	Setting
Format	5000

Marginal condition	Setting
Axis version	1

`X10. Y 30.T0T2 M31`	Beginning of the routing contour. Since offset commands are programmed at the beginning of the section, the section is only executed under consideration of the offset coordinates.
`X10. Y40. G1 F1.`	Position of the 1st execution: X30.Y80.; Position of the 2nd execution: X70.Y40.
`X20. Y40.`	Position of the 1st execution: X20.Y80.; Position of the 2nd execution: X60.Y40.
`X20. Y10.`	Position of the 1st execution: X20.Y60.; Position of the 2nd execution: X60.Y20.
`X10. Y10.`	Position of the 1st execution: X30.Y60.; Position of the 2nd execution: X70.Y20.
`X10. Y10. T0`	End of the routing contour
`X40. Y50. M70`	First offset block. The XY-coordinates are added to every coordinate of the above section and then executed. X_{M70} = X30. + X10. = X40.; Y_{M70} = Y80. − Y30. = Y50.
`X80. Y10. M70 M30`	Second offset block. The XY-coordinates are added to every coordinate of the above section and then executed. X_{M70} = X70. + X10. = X80.; Y_{M70} = Y40. − Y30. = Y10.

Two repetitions are programmed with two offset commands. Since every offset command includes a zero offset (offset coordinates), no execution will be done at the programmed position (thin contour).

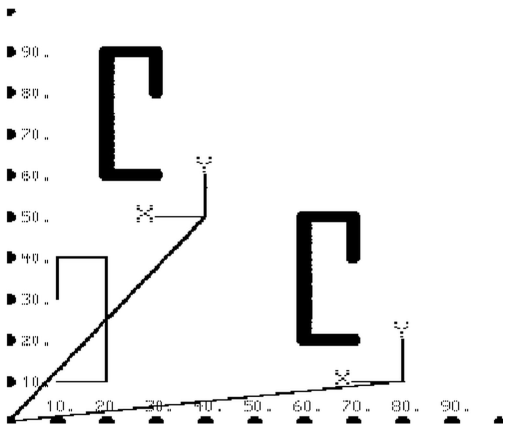


Fig. 63: Two routed patterns

Note

The **thick lines** show the programmed contours under consideration of the offset commands. The **thin lines** only serve for a clear view. They are not part of the part program.

M80: Offset with Mirroring Around the X-axis

1000	3000	5000
•	•	•

Command	Description
X*x*Y*y* M80	Shift program section and mirror it around the X-axis

Argument	Description
xy	Offset coordinates

The M80 command is used for programming an offset and a mirroring.

- The offset coordinates define the new zero point of the bracketed program section.
- If at least one offset command is programmed at the end of a program section, only this (or these) offset command(s) is (are) considered for the execution of the program section.

Calculating offset coordinates

The offset coordinates are calculated based on a drill hole or on a routing track. Determine the actual coordinate position (ACTUAL) and the new coordinate position (REFERENCE) and enter the values into the formula.

- $X_{M80} = X_{Ref} - X_{Act}$
- $Y_{M80} = Y_{Ref} + Y_{Act}$

Example

The programmed routing contour shall be executed twice after shifting. In addition, every pattern shall be mirrored/rotated around the X-axis. The axis version is version 1.

- Zero point of the 1st pattern = X10.Y90.
- Zero point of the 2nd pattern = X60.Y60.
- Required offset condition = M80

Marginal condition	Setting
Format	5000
Axis version	1

`X10. Y 30.T0T2 M31`	Beginning of the routing contour. Since offset commands are programmed at the beginning of the section, the section is only executed under consideration of the offset coordinates.
`X10. Y40. G1 F1.`	Position of the 1st execution: X20.Y50.; Position of the 2nd execution: X70.Y20.
`X20. Y40.`	Position of the 1st execution: X30.Y50.; Position of the 2nd execution: X80.Y20.
`X20. Y10.`	Position of the 1st execution: X30.Y80.; Position of the 2nd execution: X80.Y50.
`X10. Y10.`	Position of the 1st execution: X20.Y80.; Position of the 2nd execution: X70.Y50.
`X10. Y10. T0`	End of the routing contour
`X10. Y90. M80`	First offset block. The XY-coordinates are added to every coordinate of the above section and then executed. X_{M80} = X20. − X10. = X10.; Y_{M80} = Y60. + Y30. = Y90.
`X60. Y60. M80 M30`	Second offset block. The XY-coordinates are added to every coordinate of the above section and then executed. X_{M80} = X70. − X10. = X60.; Y_{M80} = Y30. + Y30. = Y60.

Two repetitions are programmed with two offset commands. Since every offset command includes a zero offset (offset coordinates), no execution will be done at the programmed position (thin contour).

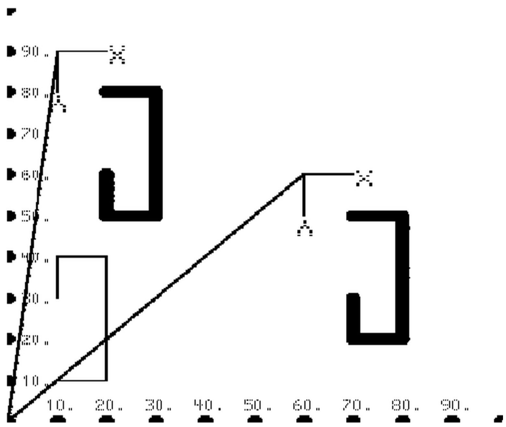


Fig. 64: Two routed patterns

Note

The **thick lines** show the programmed contours under consideration of the offset commands. The **thin lines** only serve for a clear view. They are not part of the part program.

M90: Offset with Rotation by 180°

1000	3000	5000
•	•	•

Command	Description
X*x*Y*y* M90	Shift program section and rotate it by 180°

Argument	Description
xy	Offset coordinates

The M90 command is used for programming an offset and rotation.

- The offset coordinates define the new zero point of the bracketed program section.
- If at least one offset command is programmed at the end of a program section, only this (or these) offset command(s) is (are) considered for the execution of the program section.

Calculating offset coordinates

The offset coordinates are calculated based on a drill hole or on a routing track. Determine the actual coordinate position (ACTUAL) and the new coordinate position (REFERENCE) and enter the values into the formula.

- $X_{M90} = X_{Ref} + X_{Act}$
- $Y_{M90} = Y_{Ref} + Y_{Act}$

Example

The programmed routing contour shall be executed twice after shifting. In addition, every pattern shall be rotated by 180° into clockwise direction. The axis version is version 1.

- Zero point of the 1st pattern = X40.Y90.
- Zero point of the 2nd pattern = X80.Y60.
- Required offset condition = M90

Marginal condition	Setting
Format	5000
Axis version	1

`X10. Y 30.T0T2 M31`	Beginning of the routing contour. Since offset commands are programmed at the beginning of the section, the section is only executed under consideration of the offset coordinates.
`X10. Y40. G1 F1.`	Position of the 1st execution: X30.Y50.; Position of the 2nd execution: X70.Y20.
`X20. Y40.`	Position of the 1st execution: X20.Y50.; Position of the 2nd execution: X60.Y20.
`X20. Y10.`	Position of the 1st execution: X20.Y80.; Position of the 2nd execution: X60.Y50.
`X10. Y10.`	Position of the 1st execution: X30.Y80.; Position of the 2nd execution: X70.Y50.
`X10. Y10. T0`	End of the routing contour
`X40. Y90. M90`	First offset block. The XY-coordinates are added to every coordinate of the above section and then executed. X_{M90} = X30. + X10. = X40.; Y_{M90} = Y60. + Y30. = Y90.
`X80. Y60. M90 M30`	Second offset block. The XY-coordinates are added to every coordinate of the above section and then executed. X_{M90} = X70. + X10. = X80.; Y_{M90} = Y30. + Y30. = Y60.

Two repetitions are programmed with two offset commands. Since every offset command includes a zero offset (offset coordinates), no execution will be done at the programmed position (thin contour).

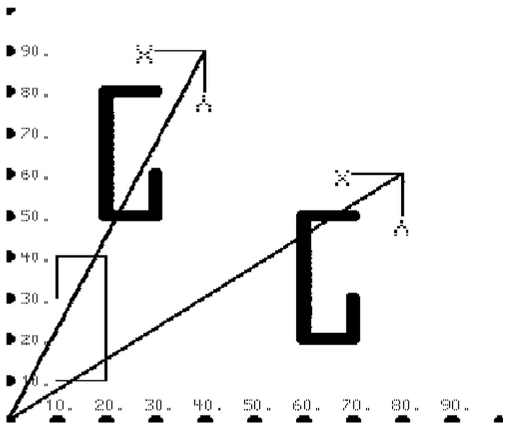


Fig. 65: Two routed patterns

Note

The **thick lines** show the programmed contours under consideration of the offset commands. The **thin lines** only serve for a clear view. They are not part of the part program.

6 Subprograms

Subprograms are useful, if a program section

- is used in different part programs (e.g. tooling holes for the production, routing out etc.)
- is required at different positions within a part program

Requirement

Format 5000 and higher allows definition and execution of subprograms.

The CNC can execute subroutines. The SIEB & MEYER CNCs CNC 4x.00, CNC 8x.00 and all following generations are able to cope with the subprogram technique.

Creation of subprogram (general information)

- Subprograms (file extension .SS5) can be created and edited in a standard ASCII editor.
- Subprograms are labeled by subprogram names. In the subprogram the file the name of the subprogram is programmed with an additional prefixed @ sign (e.g. @EXAMPLE-SUBPROGRAM).The name of a subprogram defined in the part program must be identical to the name of the subprogram file.
- A subprogram is called in the part program or in another subprogram. However, a subprogram cannot call itself.
- Depending on the CNC configuration a subprogram file can include one or several subprograms.
- A SIEB & MEYER subprogram file with Format 5000 can include one or several subprograms. Every subprogram begins with the line "@upname". This file is loaded manually before starting the production.
  In the CNC 44.00 the subroutines remain in the main memory.
- Subprograms are called with the M99 command. A separating comma must be written between the M99 command and the subprogram name (e.g. X100.Y100.M99,EXAMPLE-SUBPROGRAM).
- Additional commands (e.g. tool change T, bracket command M31 etc. must be programmed ahead of the M99 command; otherwise they are interpreted as subprogram names.
- The maximum number of nested bracket levels is 6 (CNC 8x.00). Calling the subprogram is counted as one nesting depth.
- Mixing working blocks (drilling/routing) and offset blocks in subroutines is not allowed.
- At the end of a subprogram the routing track is terminated automatically (corresponds to the command T0).
- The storage location for subprograms can be defined in the CNC.

Notes for the CNCs

Handling subroutines depends in the used CNC and its configuration.

CNC	CNC command	Working procedure
CNC 25.05	–	not available
CNC 35.00	–	not available
CNC 44.00	Standard	Before the execution is started, all required subroutines are saved in the memory of the CNC.
CNC 45.00	Standard	Before the execution is started, all required subroutines are saved in the memory of the CNC.

CNC	CNC command	Working procedure
CNC 46.00 CNC 48.00 SYSTEM 56.00 CNC 8x.00	CSUB	Every subprogram is saved in a separate file. The file name and the subprogram name are the same (except for "@").
	NOCSUB	Before the execution is started, all required subroutines are saved in the memory of the CNC.

Note

In many cases subroutines are no longer needed when a CAD/CAM software is used.

M99: Call Subprogram

1000	3000	5000
–	–	•

Command	Description
XxYy M99,*name*	Call subprogram in a part program
@*name*	Define subprogram (within a subprogram file)

Argument	Description
xy	Zero point of the subprogram
name	Name of the subprogram (capital letters and numbers)

Use the command M99 to call the subprogram *name*. The executed part program is quit in order to execute the subprogram.

The subprogram is executed relatively to the coordinates X*x*Y*y* given in the program block. This means the defined XY-coordinates set the zero point of the subprogram. These values are added to all coordinates of the subprogram during the execution.

Programming instructions:

- Program the coordinates within the subprogram with reference to a neutral zero point.
- The actual coordinate position is determined by the M99 call.

General information for subprograms see chapter 6 “Subprograms”, page 94.

Example

Two subprograms are saved in one subprogram file:

- RESET: list of CNC commands set to a default value
- FINAL: holes for finishing patterns (electroplating, mounting etc.)

Marginal condition	Setting
Format	5000

`@RESET`	Beginning of the subroutine RESET
`M49,NOFA`	Reset program zero.
`M49,NORA`	Reset rotation values.
`M49,NOSA`	Reset scaling values.
`...`	
`@FINAL`	Beginning of the subprogram FINAL
`XYT99`	

```
XY500.
X300.Y500.
X300.Y
```

To call the subprogram in the part program enter the exact name of the suprogram file. The coordinate values are programmed are offset values.

`%%5000`	Part program in the Format 5000
`XYM99,RESET`	Execute the subprogram RESET.
`M49,FAX123.432Y256.753`	Example for program zero setting via CNC command
`...`	
`XYM99,FINAL`	Execute the subprogram FINAL.
`XYM99,RESET`	Execute the subprogram RESET again.

Information on the subprogram **RESET**

- This subprogram is useful if CNC commands shall be used for settings in a part program.
- To ensure that identical conditions are given for every execution, all CNC commands are reset to default values at the beginning of the program.
- Whenever you have to use a new CNC command in a program, you just add the CNC command to the command list in the subprogram RESET.
- Thus, you ensure that this new CNC command will not cause problems when another program is executed.
- As a precaution all CNC settings are reset when an execution is finished normally.

Information on the subprogram **FINAL**

- If additional holes are required for PCB finishing it is useful to program them in a subprogram.
- If the holes have to be drilled in other positions, for example after means of production have been changed, only the subprogram needs to be adapted.

Related topics

@: Define Subprogram, page 96

@: Define Subprogram

1000	3000	5000
–	–	•

Command	Description
X*x*Y*y* M99,*name*	Call subprogram in a part program
@*name*	Define subprogram (within a subprogram file)

Argument	Description
xy	Zero point of the subprogram
name	Name of the subprogram (capital letters and numbers)

Call subprogram

Subprograms are called with the M99 command.

- The XY-coordinates define the zero point of the subprogram. These values are added to all coordinates of the subprogram during the execution.

- The name of a subprogram defined in the part program must be identical to the name of the subprogram file.
- In the subprogram file the name must be prefixed by an @ sign.
- The number of subprogram calls within a part program is not limited.
- It depends on the CNC and the configuration how subprograms are physically accessed .
- The maximum nesting depth of 6 (CNC 8x.00) must not be exceeded. This applies inclusive brackets within a subprogram. The subprogram call is counted as one nesting depth.
- A subprogram may call another subprogram (consider the nesting depth).
- Recursive calls are forbidden. That means: A subprogram cannot call itself (also not indirectly).

Define a subprogram

Subprograms are saved in one or several subprogram files.

- According to the CNC configuration a subprogram file can include one or several subprograms.
- Every subprogram begins with the name of the subprogram.
- The name must be prefixed by the @ sign.
- The number of characters for a subprogram name is not limited.
- Then, the actual subprogram follows.
- In subprogram all commands of the SIEB & MEYER command set can be used.
- A subprogram ends when a new subprogram begins.
- Subprograms can be called by other subprograms (recursive calls are not allowed.).

Notes for programming

Programming subprograms is useful, if a certain program section would have to be programmed several times within one program or in several programs.

- Program the coordinates within the subprogram with reference to a neutral zero point.
- The actual coordinate position is determined by the M99 call.

Notes for the CNCs

Handling subprograms depends in the used CNC and its configuration.

CNC	CNC command	Working procedure
CNC 25.05	–	not available
CNC 35.00	–	not available
CNC 44.00	Standard	Before the execution is started, all required subprograms are saved in the memory of the CNC.
CNC 45.00	Standard	Before the execution is started, all required subprograms are saved in the memory of the CNC.
CNC 46.00 CNC 48.00 SYSTEM 56.00 CNC 8x.00	CSUB	Every subprogram is saved in a separate file. The file name and the subprogram name are the same (except for "@").
	NOCSUB	Before the execution is started, all required subprograms are saved in the memory of the CNC.

Note

In many cases subprograms are no longer needed when a CAD/CAM software is used.

Example

Two subprograms are saved in one subprogram file:

- RESET: list of CNC commands set to a default value
- FINAL: holes for finishing patterns (electroplating, mounting etc.)

Marginal condition	Setting
Format	5000

`@RESET`	Beginning of the subprogram "RESET"
`M49,NOFA`	Reset program zero.
`M49,NORA`	Reset rotation values.
`M49,NOSA`	Reset scaling values.
`...`	
`@FINAL`	Beginning of the subprogram "FINAL"
`XYT99`	
`XY500.`	
`X300.Y500.`	
`X300.Y`	

In the part program subprograms are called via the exact name of the subprogram. The coordinate values are programmed are offset values.

`%%5000`	Part program in the Format 5000
`XYM99,RESET`	Execute the subprogram "RESET".
`M49,FAX123.432Y256.753`	Example for program zero setting via CNC command
`...`	
`XYM99,FINAL`	Execute the subprogram "FINAL".
`XYM99,RESET`	Execute the subprogram "RESET" again.

Information on the subprogram **RESET**

- This subprogram is useful if CNC commands shall be used for settings in a part program.
- To ensure that identical conditions are given for every execution, all CNC commands are reset to default values at the beginning of the program. .
- Whenever you have to use a new CNC command in a program, you just add the CNC command to the command list in the subprogram RESET.
- Thus, you ensure that this new CNC command will not cause problems when another program is executed.
- As a precaution all CNC commands are reset when an execution is finished normally.

Information on the subprogram **FINAL**

- If additional holes are required for PCB finishing it is useful to program them in a subprogram.

- If the holes have to be drilled in other positions, for example after means of production have been changed, only the subprogram needs to be adapted.

Related topics

M99: Call Subprogram, page 95

7 Tool Breakage

M94: Deactivate Broken Tool Monitoring

1000	3000	5000
–	–	•

Command	Description
X*x*Y*y* M94	Deactivate broken tool monitoring
X*x*Y*y* M95	Activate broken tool monitoring

Argument	Description
xy	The function already applies for the coordinates in the block

Requirement
The CNC command PRGM,−I is set.

Use this command to deactivate broken tool monitoring.

Related topics

M95: Activate Broken Tool Monitoring, page 100

M95: Activate Broken Tool Monitoring

1000	3000	5000
–	–	•

Command	Description
X*x*Y*y* M94	Deactivate broken tool monitoring
X*x*Y*y* M95	Activate broken tool monitoring

Argument	Description
xy	The function already applies for the coordinates in the block

Requirement
The CNC command PRGM,−I is set.

Use this command to activate broken tool monitoring.

Related topics

M94: Deactivate Broken Tool Monitoring, page 100

8 Various Commands

8.1 Reference Points for Coordinates

This section describes commands used for modifying the reference point of the programmed coordinates (absolute/incremental).

G90: Absolute Working Coordinates

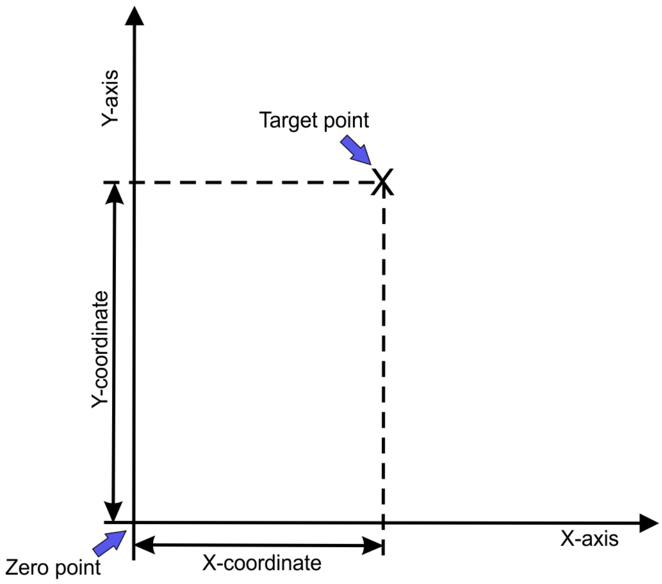
1000	3000	5000
•	•	•

Command	Description
X*x*Y*y* G90	Absolute interpretation of coordinates

Argument	Description
xy	Working coordinate The setting "absolute" is already effective for the current program line.

Use the G90 command to define that all following coordinates will be related to the current program zero.

- The G90 setting is the default setting.
- In SIEB & MEYER part programs a G90/G91 programming is effective from the block in which the commands are programmed.
- To describe a target point you have to define the complete path to the program zero on the X-axis and Y-axis.



Note

- Offset coordinates must always be programmed as absolute values.
- Before an offset command is executed the CNC automatically switches to the absolute interpretation.

Marginal condition	Setting
Format	5000
Axis version	1

`X20.Y-30. T0T5 M31`	Absolute working coordinate
`X20.Y-20. G1F1.2G42`	
`X30.Y-20.`	
`X30.Y`	
`X20.Y`	
`X20.Y30.`	
`X-20.Y10.`	
`Y-20.X G91`	Incremental working coordinate
`X-20.Y`	
`XY-10.`	
`X10.Y-10.`	
`X20.Y-30. G90`	Absolute working coordinate
`X20.Y-30. T0`	
`(FINAL OFFSET)`	
`X80.Y70.M50M30`	Absolute offset coordinate.

Related topics

G91: Incremental Working Coordinate, page 102

G91: Incremental Working Coordinate

1000	3000	5000
•	•	•

Command	Description
X*x*Y*y* G91	Incremental working coordinate

Argument	Description
xy	Working coordinate The setting "incremental" is already effective for the current program line.

Use the G91 command to define that all following coordinates will be interpreted relatively to each other. That means: The current coordinate relates to the previous position. This reference will be set to the current reference point for the following actions:

- when the system is started
- at the end of the part program
- at the end of a Step-and-Repeat pattern.
- In SIEB & MEYER part programs a G90/G91 programming is effective from the block in which the commands are programmed.

The first coordinate of a relative coordinate definition relates to the last used absolute coordinate. At the beginning of the program the last used absolute coordinate will be set to X0Y0 (program zero).

Incremental coordinates are always defined in relation to the last coordinate. This coordinate serves as reference for the target point. The coordinates are defined as modification of the X-axis and Y-axis to the reference point.

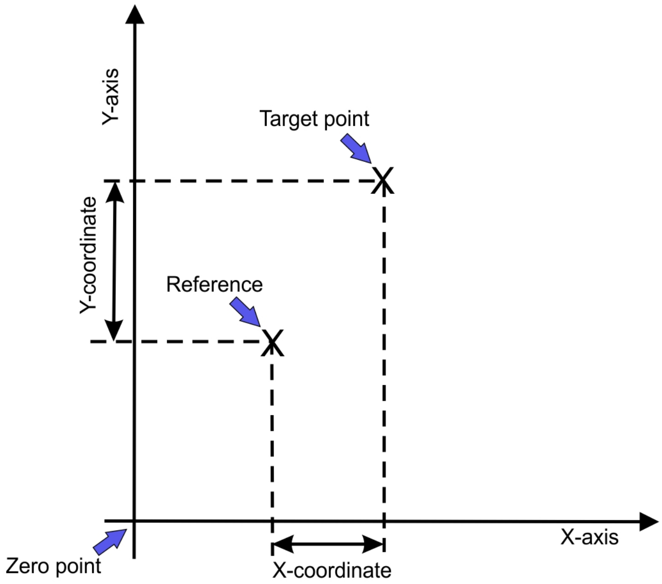


Fig. 66: Working coordinates

Note

Before an offset command is executed the CNC automatically switches to the absolute interpretation.

Marginal condition	Setting
Format	5000
Axis version	1

`X20.Y-30. T0T5 M31`	Absolute working coordinate
`X20.Y-20. G1F1.2G42`	
`X30.Y-20.`	
`X30.Y`	
`X20.Y`	
`X20.Y30.`	
`X-20.Y10.`	
`Y-20.X G91`	Incremental working coordinate
`X-20.Y`	
`XY-10.`	
`X10.Y-10.`	
`X20.Y-30. G90`	Absolute working coordinate
`X20.Y-30. T0`	
`(FINAL OFFSET)`	
`X80.Y70.M50M30`	Absolute offset coordinate.

Related topics

G90: Absolute Working Coordinates, page 101

8.2 Set Drill Stroke

This section describes command for programming the settings for traveling and working planes in the part program.

H: Absolute Traveling Plane

1000	3000	5000
–	H1 = 0.01 mm	H1 = 0.001 mm

Command	Description
X*x*Y*y* H*h*	Absolute traveling plane

Argument	Description
xy	Coordinate, from which the new plane is effective
h	Distance between the table surface and the absolute traveling plane (above the PCB) ▸ Only positive values are allowed. ▸ The unit depends on the set format: – unit in Format 3000: 0.01 mm (millimeter point allowed) – unit in Format 5000: 0.001 mm (millimeter point allowed)

Note

During the execution the programmed H-values, K-values and Z-values overwrite the manual settings. Ensure that the correct settings are reset after an execution is finished to guarantee trouble-free positioning of the XY-axes.

Absolute traveling plane

The Z0-plane is used as reference plane for the absolute traveling plane (e.g. table surface).

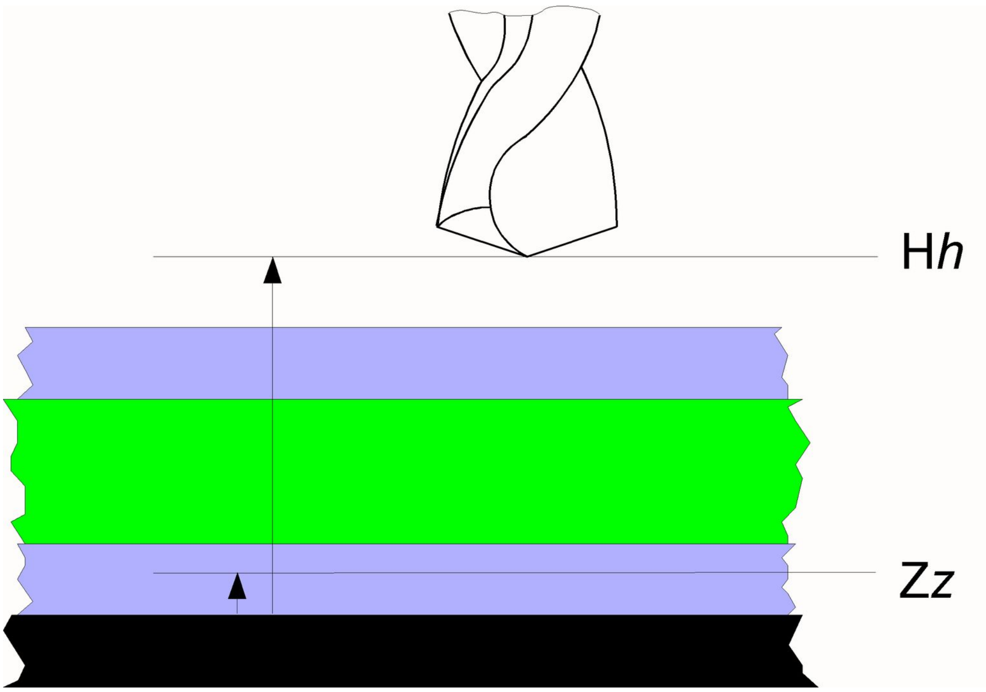


Fig. 67: Absolute traveling plane

The absolute traveling plane is defined with the H-value.

- Reference plane = Z0 (normally the table surface)

- The Z-axis is retracted to the traveling plane after every drill stroke.
- The distance from the tool and the pressure foot to the work piece is sufficient so that the axes can move to the next working position without any risk.
- For the Quick function (relative traveling plane) the H-value is the maximum limit for the Z-axis stroke.

Related topics

Z: Absolute Working Plane, page 106

K: Relative Working Plane

1000	3000	5000
–	K1 = 0.01 mm	K1 = 0.001 mm

Command	Description
XxYy K*k*	Relative working plane (plunge depth)

Argument	Description
xy	Coordinate, from which the new plane is effective
k	Distance between the board surface and the relative working plane ▸ Only positive values are allowed. ▸ The unit depends on the set format: – unit in Format 3000: 0.01 mm (millimeter point allowed) – unit in Format 5000: 0.001 mm (millimeter point allowed)

Relative working plane

Important

During the execution the programmed H-values, K-values and Z-values overwrite the manual settings. Ensure that the correct settings are reset after an execution is finished to guarantee trouble-free positioning of the XY-axes.

Requirement

The machine must be equipped with a device for depth control (e.g. contact drilling device, second scale etc.).

The K0-plane used as reference plane for the relative working plane (e.g. board surface).

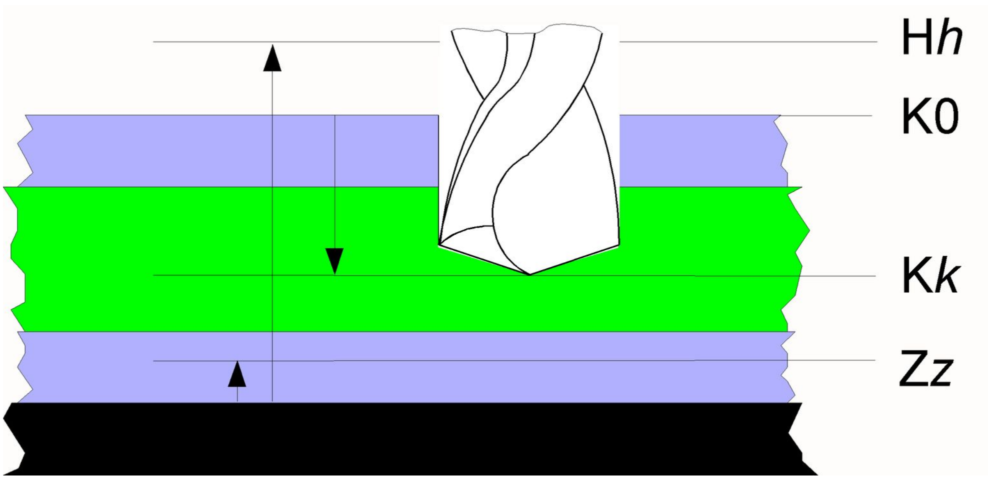


Fig. 68: Relative working plane

The relative working plane for depth control is defined with the K-value.

- Reference plane = K0 (board surface)
- The relative working plane is also referred to as plunge depth and is limited by the Z-value.
- The depth control is activated/deactivated with the G83 command.
- During one working step (drilling/routing) the Z-axis is lowered to the working plane.

Related topics

Z: Absolute Working Plane, page 106

Z: Absolute Working Plane

1000	3000	5000
–	Z1 = 0.01 mm	Z1 = 0.001 mm

Command	Description
X*x*Y*y* Z*z*	Absolute working plane

Argument	Description
xy	Coordinate, from which the new plane is effective
z	Distance between the table surface and the absolute working plane ▸ Only positive values are allowed. ▸ The unit depends on the set format: – unit in Format 3000: 0.01 mm (millimeter point allowed) – unit in Format 5000: 0.001 mm (millimeter point allowed)

Note

During the execution the programmed H-values, K-values and Z-values overwrite the manual settings. Ensure that the correct settings are reset after an execution is finished to guarantee trouble-free positioning of the XY-axes.

Absolute working plane

The Z0-plane is used as reference plane for the absolute working plane (e.g. table surface).

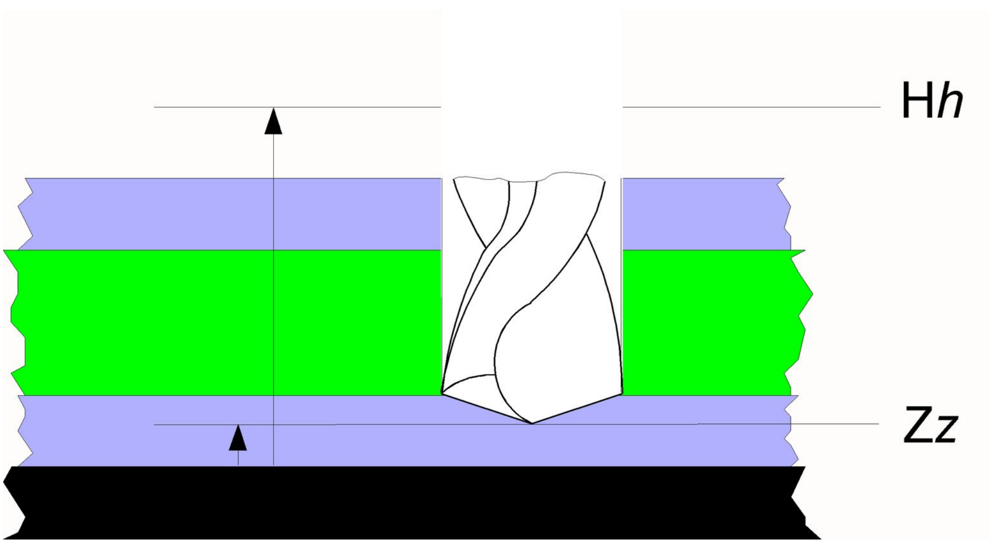


Fig. 69: Absolute working plane

The absolute working plane is defined with the Z-value.

- Reference plane = Z0 (normally the table surface)
- During one working step (drilling/routing) the Z-axis is lowered to the working plane.
- For the depth control (relative working plane) the Z-value is the minimum limit for lowering the Z-axis.

Related topics

H: Absolute Traveling Plane, page 104

K: Relative Working Plane, page 105

8.3 Machine Functions

This section describes commands which activate functional sequences in the machine.

Mxx: Execute Machine Function

1000	3000	5000
–	–	•

Command	Description
M22	Execute machine function M22
...	...
M27	Execute machine function M27
M34	Execute machine function M34
...	...
M41	Execute machine function M41
M52	Execute machine function M52
...	...
M55	Execute machine function M55
M49,M101	Execute machine function M101
...	...
M49,M116	Execute machine function M116

NOTICE

Damage to the machine

Improper use may cause damage to the machine.

→ Never use a machine function without having contacted the machine manufacturer.

This commands are used for executing functions programmed by the machine manufacturer.

- The number of machine functions depends on the machine type.
- The effect of the machine functions can be found in the machine manufacturer's documentation.
- Machine functions are executed according to the program line in which it was programmed.
- Machine functions must not be executed within a routing contour. This may result in faulty routing results.

Example

The following machine functions are to be executed:.

Machine function	Position in the program
M38	Beginning of the program
M34	At every beginning of a routing track
M35	At every end of a routing track
M39	End of program

Marginal condition	Setting
Format	5000

%%5000	
M38	Beginning of the program
X0Y0	
M34	
X..Y..T0T1	Beginning of routing
...	
X..Y..T0	End of routing
M35	
M34	
X..Y..T0T1	Beginning of routing
...	
X..Y..T0	End of routing
M35	
M34	
X..Y..T0T1	Beginning of routing
...	
X..Y..T0	End of routing
M35	
X0Y0	End of program
M39	

M20: Contidional Stop

1000	3000	5000
•	•	•

Command	Description
X*x*Y*y* M20	The machine stops when the function "Optional stop" is active

Argument	Description
xy	The machine is stopped before this coordinate is executed

Use these commands for programming a stop condition in the part program.

- The execution is interrupted, if the function "Optional Stop" is activated in a SIEB & MEYER CNC.
- The system waits, until previous drilling, routing and positioning commands are finished.
- The machine moves to the machine zero or XY (if coordinates are defined).

- The message “Optional Stop” appears on the status bar of the user interface.
- Press the START key to continue the execution.
- Programming the command is only allowed
  - in drilling blocks and
  - in a T0 block of a routing track

M21: Activate Output M21

1000	3000	5000
•	•	•

Command	Description
X*x*Y*y* M21	Activate the output M21

Argument	Description
xy	The output M21 is activated when this coordinate is executed

NOTICE

Damage to the machine

Improper use may cause damage to the machine.

→ Never use a machine function without having contacted the machine manufacturer.

The M21 command activated the output M21.

- The output remains active until the execution of the next program line starts.
- During routing the output M21 may be active for the execution time of two program lines (round edge, broke tip and connection arc)
- Further information can be found in the documentation of the machine manufacturer.

M28: Move to Machine Zero

1000	3000	5000
–	•	•

Command	Description
X*x*Y*y* M28	Move the XY-axes to machine zero

Argument	Description
xy	The axes are positioned after the coordinate has been executed

This command moves the XY-axes are moved to the machine zero.

- The machine zero is identical to the calibrating point.
- Dependent on the machine type the tool is released before.
- The spindles are decelerated.
- The vacuum unit is switched off.
- Press the START key to continue the execution.
- The following applies for programming:
  - The command can be programmed alone in a line with and without XY-coordinates: If the command is programmed without XY-coordinates, the XY-axes move directly to the machine zero.

If the command is programmed with XY-coordinates, the XY-axes move to the defined XY-position first and then to the machine zero.
- Programming the ciommand with XY-coordinates is only allowed in drilling and routing blocks.
- The definition of XY-coordinates in a T0 block of a routing path is optional.

Related topics

M29: Move to Park Position, page 110

M29: Move to Park Position

1000	3000	5000
–	•	•

Command	Description
X*x*Y*y* M29	Move XY-axes to the park position

Argument	Description
xy	The axes are positioned after the coordinate has been executed.

The M29 command moves the XY-axes to the park position.

- Dependent on the machine type the tool is released before.
- The spindles are decelerated.
- The vacuum unit is switched off.
- Press the START key to continue the execution.
- The following applies for programming:
  - The command can be programmed alone in a line with and without XY-coordinates: If the command is programmed without XY-coordinates, the XY-axes move directly to the park position.
    If the command is programmed with XY-coordinates, the XY-axes move first to the defined XY-position and then to the park position.
  - Programming the ciommand with XY-coordinates is only allowed in drilling and routing blocks.
  - The definition of XY-coordinates in a T0 block of a routing path is optional.

Related topics

M28: Move to Machine Zero, page 109

8.4 Influence Program Sequence

This section describes commands used for interrupting the execution or for programming optimal settings for the production.

M47: Show User Message

1000	3000	5000
–	•	•

Command	Description
M47,*text*	Stop the execution and show user message
M47*text*	Stop the execution and show user message

Command	Description
M47,&P*text*	Move to park position and show user message
M47,!*text*	Stop the execution and show user message and additional prompt

Argument	Description
text	Any text without ^|<|#

The text of the M47 command is shown as message on the screen. The maximum text length depends on the used CNC.

- The machine is stopped during the execution.
- The programmed text appears on the screen.
- The next line of the part program appears.
- Press the START key to continue the execution. If the text begins with an exclamation mark "!", another message appears which must be confirmed (e.g. M47,!TURN BOARD).
- If the M47 text begins with the string "&P", the XY-axes are moved to the park position before a message is displayed (e.g. M47,&PTURN BOARD).
- The options exclamation mark and "&P" can be combined as desired (e.g. M47,&PTURN BOARD or M47,&P!TURN BOARD).

Related topics

M49: Execute CNC Command, page 111

M49: Execute CNC Command

1000	3000	5000
–	●	●

Command	Description
M49, *command*	Execute CNC Command

Argument	Description
command	CNC command

The M49 command is used for programming a CNC command.

- The programmed CNC commands are automatically processed during the execution.
- The M49 command is programmed in a separate program line.
- A separating comma must be entered between the M49 command and the CNC command.
- The programmed CNC commands remain active after the execution is finished. Tip: Subroutines allow resetting all used CNC commands to default values (refer to the M99 command).
- If the CNC command COMM,E is active, error messages are displayed on the screen during the execution and the execution is interrupted.

Notes for the CNC 35.00:

- The CNC 35.00 interprets the M49 commands as M47 commands.
- The machine interrupts the execution and shows the text "M47" in the status bar.
- The current and the following program line are shown on the bottom of the screen.

Feasibility of CNC commands

If the execution is not started at the beginning of a part program, the execution of CNC commands that are programmed ahead of the start block may be influenced

CNC command	Explanation
MCOM	After start with block, hole or tool also the CNC commands are executed that are programmed ahead of the start position.
NOMCOM	After start with block, hole or tool the CNC commands that are programmed ahead of the start position are not executed.

Example

CNC commands are programmed in a part program.

Marginal condition	Setting
Format	5000

`M99,RESET`	Call subprogram "RESET". Therein you program the default settings of all CNC commands.
`M49,FV1`	Switch axis version (in this case axis version 1).
`M49,CHEKX100.Y10.DXDY10.`	Define check hole
`( PROGRAM START )`	
`X3.56Y-1.02T1M31`	Work program
`...`	
`M99,RESET`	Call subprogram "RESET". All used CNC commands are reset.

Related topics

M47: Show User Message, page 110

M58: Forbidden Areas

1000	3000	5000
–	–	•

Command	Description
X*x*Y*y*M58D*d*	Define Forbidden Area

Argument	Description
d	Diameter or edge length of the Forbidden Area ▸ minimum = 0.001 mm ▸ minimum = 0.0001 inch ▸ maximum = as desired
xy	Center of the Forbidden Area

M58

Requirement
The CNC command PRGM,C must be active.

Use this programming command to activate the Forbidden Area for the *production mode*. During the execution of a part program, drilling is not allowed within the Forbidden Areas. If a drill hole is within a Forbidden Area, the drill hole is omitted without an error message.

- The Forbidden Areas are defined by means of a center *xy* and a diameter. The diameter serves as edge length, since the Forbidden Areas are considered as squares by the CNC.
- (Up to software version 12.02.001) A maximum of 10 Forbidden Areas can be defined.
  (Software version 12.02.003 and higher) A maximum of 99 Forbidden Areas can be defined.
- (Software version 12.02.009 and higher) The program command M58 can only be used in a Step-and-Repeat programming, if the CNC command FORB,A is active. If the CNC command FORB,M or FORB,P is set, the program command M58 is not considered in a Step-and-Repeat programming and must be programmed outside of Step-and-Repeat therefore.

Use the CNC command FORB to set Forbidden Areas and other settings. For further information refer to the manual CNC 8x.00 - CNC Commands.

8.5 Other Commands

This section describes global CNC commands which do not specially apply to one of the other subjects.

I: Interpolation Parameter I for Routing Commands

1000	3000	5000
I1 = 0.01 mm	I1 = 0.001 mm	I1 = 0.001 mm

Command	Description
X*x*Y*y* G2 I*i*J*j*	Rout circular arc I = X-share of the radius J = Y-share of the radius
X*x*Y*y* G3 I*i*J*j*	Rout circular arc I = X-share of the radius J = Y-share of the radius
X*x*Y*y* G49 I*i*J*j*	Rout out a rectangle I = side length in X-direction J = side length in Y-direction
X*x*Y*y* G50 I*i*J*j*	Rout out a rectangle I = side length in X-direction J = side length in Y-direction

Argument	Description
i, j	The unit depends on the set format: ▸ unit in Format 1000: 0.01 mm (no millimeter point and no negative value allowed) ▸ unit in Format 3000: 0.001 mm (millimeter point allowed) ▸ unit in Format 5000: 0.001 mm (millimeter point allowed)

The interpolation parameters I and J are used for different functions:

- programming a circular arc
- programming a rectangle

- programming peck drilling parameters
- all
  Internally in the CNC the routing and peck drilling parameters are organized in different variables. Thus, the last settings remain active, if it is switched between routing and peck drilling.

Programming circular arc with I and J

The interpolation parameters I and J are used for programming a circular arc (also refer to the G2 and G3 command).

Interpolation parameters	Explanation
I	X distance between the start position of the circular arc and the center of the circle
J	Y distance between the start position of the circular arc and the center of the circle

Note

The calculation of the interpolation parameters I and J automatically lead to the correct signs.

No negative signs must be programmed in the format 1000. Furthermore, a separate program line must programmed for each quadrant of the circular arc.

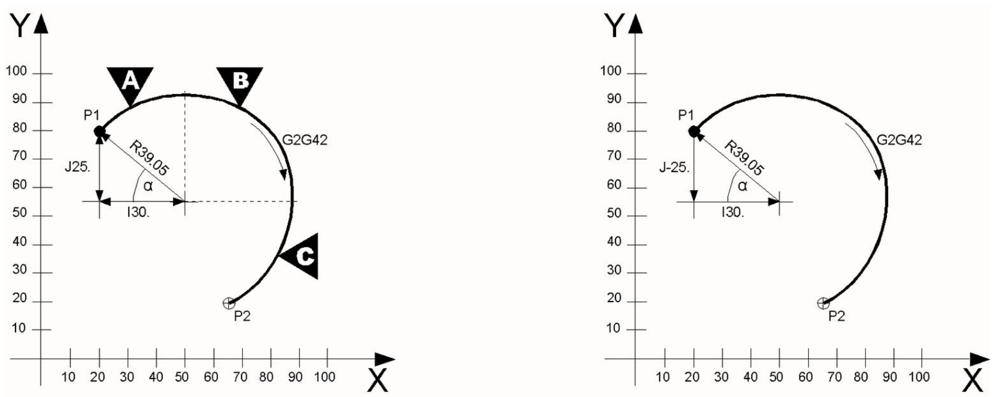


Fig. 70: Circular arc with I and J

- **Left:** Definition in format 1000. Only positive IJ values are allowed. One block must be programmed for every quadrant [A, B and C].
- **Right:** When calculating circular arc parameters in Format 3000 or higher, the correct signs are evaluated automatically.

Calculate I and J on the basis of the coordinate values.

The start position and the end position of the circular arc given.

- $I = X_{\text{Start position for routing}} - X_{\text{Center}}$
- $I = 20.0 - 50.0 = -30.0$
- $J = Y_{\text{Start position for routing}} - Y_{\text{Center}}$
- $J = 80.0 - 55.0 = 25.0$

The Format 3000 results in the following program section:

```
X20.Y80.T5T0
X65.Y20.G2I-30.J25.F12
X65.Y20.T0
```

Calculate I and J on the basis of an angle value. The radius and the angle α between the X-axis and the start position of the circular arc are given.

- `I = r x sin(α)`
- `J = r x cos(α)`

Programming a rectangle with I and J

The interpolation parameters I and J are used for programming a circular arc (also refer to the G49 and G50 command).

Interpolation parameters	Explanation
I	X side length
J	Y side length

Related topics

G2: Rout Circular Arc Clockwise, page 41

G3: Rout Circular Arc Counterclockwise, page 43

G49: Rout Out Rectangle Counterclockwise, page 53

G50: Rout Out Rectangle Clockwise, page 55

J: Interpolation Parameter J for Routing Commands, page 115

I: Lowering Value of a Peck Drill Hole

1000	3000	5000
–	–	•

Command	Description
XxYy G81 *IiPpJjWw*	Activate peck drilling I = lowering value J = height of a partial stroke of a peck drill hole

Programming peck drill parameters I and J

The parameters I and J are used for programming different planes for peck drilling (also refer to the G81 command).

Parameter	Explanation
I	Lowering value The value depends on the current reference plane (table surface or board surface)
J	Height of a partial stroke of a peck drill hole

Related topics

G81: Activate Peck Drilling, page 28

J: Height of Partial Stroke of Peck Drill Hole, page 118

P: Reduce Feed Rate for Peck Drilling, page 118

J: Interpolation Parameter J for Routing Commands

1000	3000	5000
J1 = 0.01 mm	J1 = 0.001 mm	J1 = 0.001 mm

Command	Description
X*x*Y*y* G2 I*i*J*j*	Rout circular arc I = X-share of the radius J = Y-share of the radius
X*x*Y*y* G3 I*i*J*j*	Rout circular arc I = X-share of the radius J = Y-share of the radius
X*x*Y*y* G49 I*i*J*j*	Rout out a rectangle I = side length in X-direction J = side length in Y-direction
X*x*Y*y* G50 I*i*J*j*	Rout out a rectangle I = side length in X-direction J = side length in Y-direction

Argument	Description
i, j	The unit depends on the set format: ▸ unit in Format 1000: 0.01 mm (no millimeter point and no negative value allowed) ▸ unit in Format 3000: 0.001 mm (millimeter point allowed) ▸ unit in Format 5000: 0.001 mm (millimeter point allowed)

The interpolation parameters I and J are used for different functions:

- programming a circular arc
- programming a rectangle
- programming peck drilling parameters
- all
  Internally in the CNC the routing and peck drilling parameters are organized in different variables. Thus, the last settings remain active, if it is switched between routing and peck drilling.

Programming circular arc with I and J

The interpolation parameters I and J are used for programming a circular arc (also refer to the G2 and G3 command).

Interpolation parameters	Explanation
I	X distance between the start position of the circular arc and the center of the circle
J	Y distance between the start position of the circular arc and the center of the circle

Note

The calculation of the interpolation parameters I and J automatically lead to the correct signs.

No negative signs must be programmed in the format 1000. Furthermore, a separate program line must programmed for each quadrant of the circular arc.

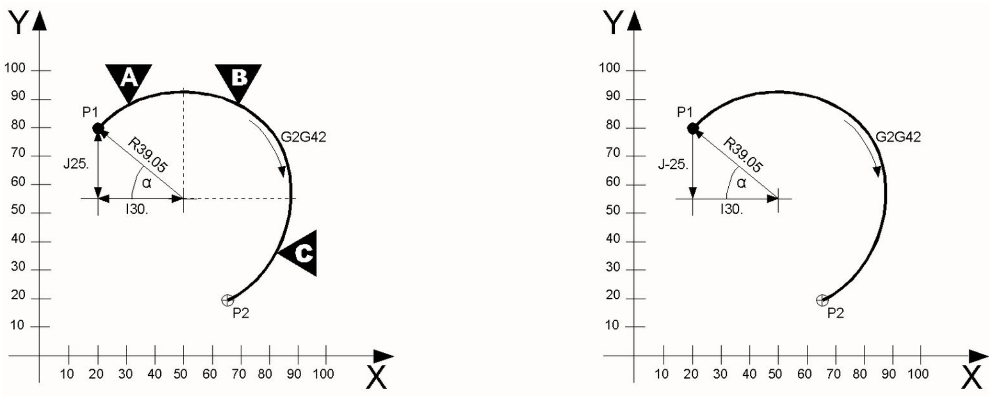


Fig. 71: Circular arc with I and J

- **Left:** Definition in format 1000. Only positive IJ values are allowed. One block must be programmed for every quadrant [A, B and C].
- **Right:** When calculating circular arc parameters in Format 3000 or higher, the correct signs are evaluated automatically.

Calculate I and J on the basis of the coordinate values.

The start position and the end position of the circular arc given.

- $I = X_{Start\ position\ for\ routing} - X_{Center}$
- `I = 20.0 - 50.0 = -30.0`
- $J = Y_{Start\ position\ for\ routing} - Y_{Center}$
- `J = 80.0 - 55.0 = 25.0`

The Format 3000 results in the following program section:

```
X20.Y80.T5T0
X65.Y20.G2I-30.J25.F12
X65.Y20.T0
```

Calculate I and J on the basis of an angle value. The radius and the angle α between the X-axis and the start position of the circular arc are given.

- $I = r \times \sin(\alpha)$
- $J = r \times \cos(\alpha)$

Programming a rectangle with I and J

The interpolation parameters I and J are used for programming a circular arc (also refer to the G49 and G50 command).

Interpolation parameters	Explanation
I	X side length
J	Y side length

Related topics

G2: Rout Circular Arc Clockwise, page 41

G3: Rout Circular Arc Counterclockwise, page 43

G49: Rout Out Rectangle Counterclockwise, page 53

G50: Rout Out Rectangle Clockwise, page 55

I: Interpolation Parameter I for Routing Commands, page 113

J: Height of Partial Stroke of Peck Drill Hole

1000	3000	5000
–	–	•

Command	Description
X*x*Y*y* G81 I*i*P*p*J*j*W*w*	Activate peck drilling I = lowering value J = height of a partial stroke of a peck drill hole

Programming peck drill parameters I and J

The parameters I and J are used for programming different planes for peck drilling (also refer to the G81 command).

Parameter	Explanation
I	Lowering value The value depends on the current reference plane (table surface or board surface)
J	Height of a partial stroke of a peck drill hole

Related topics

G81: Activate Peck Drilling, page 28

I: Lowering Value of a Peck Drill Hole, page 115

W: Reduce Height of Partial Strokes for Peck Drill Hole, page 122

M76: Transfer String

1000	3000	5000
–	–	•

Command	Description
M76,*text*	Transfer string

The M76 command transmits a character string.

- The text to be transmitted depends on the recipient.
- For further information refer to the documentation of the machine manufacturer.

Specific notes for the machine manufacturer:

- The TCP/IP address and all other settings must be defined in the machine parameters (Parameter Editor).
- The transmission must be programmed in the sequence INT-FUNC (16th function), for example the command SEND M76 for the actual transmission, troubleshooting, text display, interruption of the execution etc.

P: Reduce Feed Rate for Peck Drilling

1000	3000	5000
–	–	•

Command	Description
X*x*Y*y* G81I*i*P*p*J*j*W*w*	Define and activate peck drill function

Argument	Description
p	Percentage value for plunge rate to the lowering value I*i* 0 to 100 %

To obtain optimum drilling results you can reduce the down-feed rate to the lowering value (s. peck drill parameter I*i*) of a peck drill hole. The value is determined as percentage value. The percentage value is related to the feed rate defined for the current tool in the tool table (tool parameter S). When the lowering value is reached the value for the feed rate changes to the value defined with tool parameter S.

Related Topics

[G81: Activate Peck Drilling, page 28]

[I: Lowering Value of a Peck Drill Hole, page 115]

R: Parameter for Routing Commands

1000	3000	5000
–	R1 = 0.001 mm	R1 = 0.001 mm

Command	Description
X*x*Y*y* G2 R*r*	Rout circular arc R = radius of the circular arc
X*x*Y*y* G3 R*r*	Rout circular arc R = radius of the circular arc
X*x*Y*y* G49 R*r*	Rout a square R = side length of a square
X*x*Y*y* G50 R*r*	Rout a square R = side length of a square

Argument	Description
r	Radius of the circular arc (e.g. R20.; R−17.5)
r	Side length of the square (e.g. R20.)

The R parameter is used for different functions:

- program a circular arc
- programming a square

Programming a circular arc with R

The R parameter is used for programming a circular arc (also refer to the G2 and G3 command).

Parameter	Explanation
R	Radius (distance from the center of the circle to the circular arc)

The routing angle depends on the sign of the radius value.

Radius value	Explanation
negative	The routing angle α is larger than 180° (larger than a semicircle)
positive	The routing angle α is smaller than or equal to 180° (maximum a semicircle)

Left figure: The angle α is smaller than 180°: The parameter R must be positive.

Right figure: The angle α is greater than 180°: The parameter R must be negative.

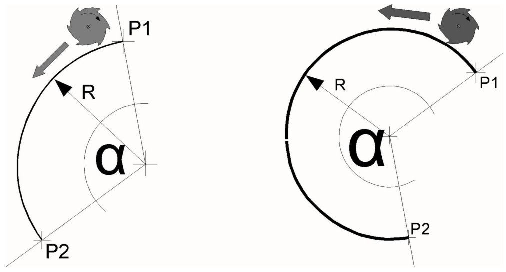


Fig. 72: Routing angle

Programming a square with R

The R parameter is used for programming the side length of a square (also refer to the G49 and G50 command).

Parameter	Explanation
R	Side length of a square

Related topics

G2: Rout Circular Arc Clockwise, page 41

G3: Rout Circular Arc Counterclockwise, page 43

G45: Rout Out Full Circle Counterclockwise, page 46

G46: Rout Out Full Circle Clockwise, page 48

G47: Rout Out Disk Counterclockwise, page 49

G48: Rout Out Disk Clockwise, page 51

G49: Rout Out Rectangle Counterclockwise, page 53

G50: Rout Out Rectangle Clockwise, page 55

T: Tool Change

1000	3000	5000
•	•	•

Command	Description
X*x*Y*y* T*t*	Tool change

Argument	Description
xy	These coordinates are already executed with the new tool

The T-command is used for programming a tool change.

- A tool change is initiated with the T-command.
- The coordinates of the program line are already executed with the new tool.
- The tool number is just a name and not necessarily the same as the tool number in the machine (see CNC command OD: tool number conversion).
- When the T-command is executed a tool change is carried out. If there is already a tool in of the selected tool number in the spindle, the tool change is not carried out.
- Dependent on the machine type the tool is measured (length, diameter, runout).
- The required cutting parameters are in the tool table of the CNC.
- The number of maximum possible tool numbers depends on the machine equipment and on the set magazine management.

The T0 command

The T0 command allows programming special working situations.

- A T0-command must be programmed at the beginning of a routing contour so that the axes move to the start position while the Z-axis is retracted.
- A T0 command must be programmed at the end of a routing track so that the Z-axis is retracted before the next positioning process is started.
- A T0 command programmed within a drilling program prevents lowering the Z-axis.

Example

Part program with tool change commands.

Marginal condition	Setting
Format	5000
Axis version	1

Program	Comment
`X0.Y0.V2 T1 M31 M31`	T1 initiates a tool change
`...`	
`XYM50`	
`XY100.M50 M30`	
`X25.Y50. T0 T2 M31`	T2 initiates a tool change. T0 indicates the beginning of a routing track.
`X25.Y50. G45 R20.`	
`X25.Y50. T0`	T0 indicates the end of a routing track
`XYM50`	
`XY100.M50 M30`	
`( FINAL OFFSET )`	
`X123.45Y345.67M50 M30`	

W: Number of Executions

1000	3000	5000
•	•	•

Command	Description
X*x2*Y*y2*W*w*	End point (any grid)

Argument	Description
w	Number of drill holes/patterns in a row

Number of repetitions

The parameter W defines the total number of executions for the SIEB & MEYER commands V1, V2, V3 and V4.

- Depending on the used command the parameter defines the number of drill holes or the number of patterns.
- The parameter W is always defined in the 2nd block of a repetition structure. Such a repetition structure always has at least 2 program lines:
  - 1st line: start coordinates and the function command (V1, V2, V3 or V4)
  - 2nd line: end coordinate and total number (parameter W)

Related topics

V1: Drill Dual Row of Holes (Dual-in-line), page 20

V2: Drill Single Row of Holes, page 22

V3: Drill Quadruple Row of Holes, page 23

V4: Drill Circular Row of Holes, page 25

W: Reduce Height of Partial Strokes for Peck Drill Hole

1000	3000	5000
–	–	•

Command	Description
X*x*Y*y* G81	Activate peck drill function with the original parameters
X*x*Y*y* G81I*i*P*p*J*j*W*w*	Define and activate peck drill function

Argument	Description
w	Factor for the continuous reduction of the partial strokes 0 to 1000 ‰

Partial strokes of a peck drill hole

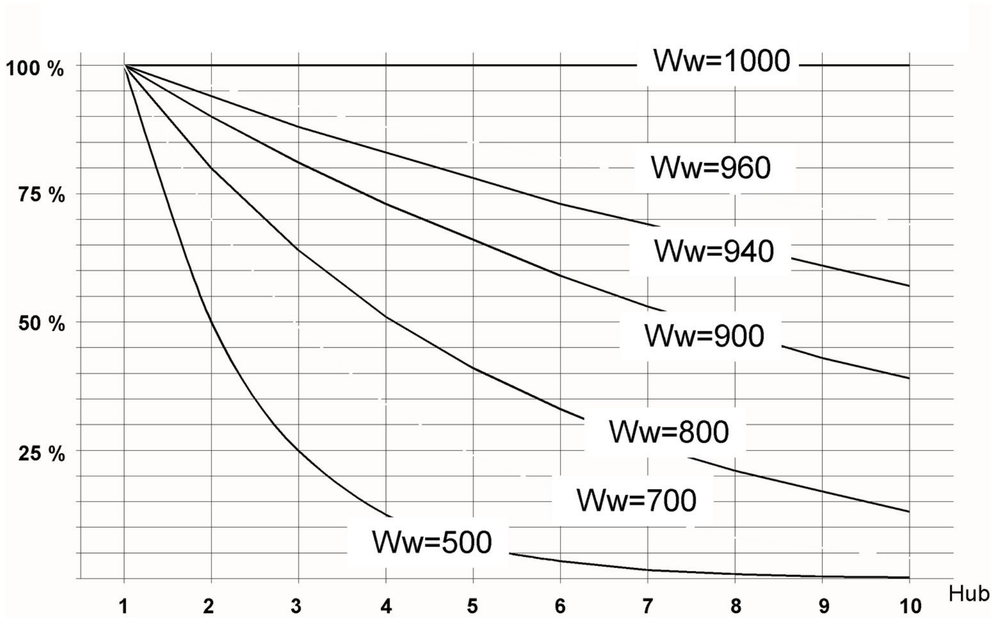


Fig. 73: The heights of the partial strokes of a peck drill hole depends on the W-value.

The height of the partial strokes is reduced depending on the W-value.Several parameters are required for the definition of the partial strokes.

- Partial strokes are strokes required for reboring a hole after predrilling.
- The plunge rate of a partial stroke is defined in the tool parameter F.
- The W-value allows continuous reduction of the partial stroke (e.g. for hard materials). The diagram shows heights of partial strokes in dependance on the number of partial strokes for some peck drilling factors.
- Peck drilling is finished when the target plane is reached. The target plane depends on the reference plane (reference plane = table surface: target plane = Z-plane; reference plane = board surface: target plane = K-plane).
- If the CNC calculates a partial stroke that is smaller than 1 µm, drilling the hole is canceled and a message appears that the target plane was not reached.

Parameters	Explanation
J*j*	Height of a partial stroke of a peck drill hole
W*w*	Peck drill factor ‣ stroke = *j* * (*w* / 1000)$^{n-1}$ ‣ n = partial stroke counter

Example

The heights of the partial strokes for the following program line are listed in the table.

X10.Y20. I4.5 P20 J1.2 W500 Z2.

Marginal condition	Setting
Format	5000

Partial stroke	Evaluation	Stroke	reached depth
1	1.2 mm x 0.500^0	1.200 mm	4.500 mm − 1.200 mm = 3.300 mm
2	1.2 mm x 0.500^1	0.600 mm	3.300 mm − 0.600 mm = 2.700 mm
3	1.2 mm x 0.500^2	0.300 mm	2.700 mm − 0.300 mm = 2.400 mm
4	1.2 mm x 0.500^3	0.150 mm	2.400 mm − 0.150 mm = 2.250 mm
5	1.2 mm x 0.500^4	0.075 mm	2.250 mm − 0.075 mm = 2.175 mm
6	1.2 mm x 0.500^5	0.037 mm	2.325 mm − 0.037 mm = 2.138 mm
7	1.2 mm x 0.500^6	0.018 mm	2.288 mm − 0.018 mm = 2.119 mm
8	1.2 mm x 0.500^7	0.009 mm	2.119 mm − 0.009 mm = 2.110 mm
9	1.2 mm x 0.500^8	0.004 mm	2.110 mm − 0.004 mm = 2.106 mm
10	1.2 mm x 0.500^9	0.002 mm	2.106 mm − 0.002 mm = 2.104 mm
11	1.2 mm x 0.500^{10}	0.001 mm	2.104 mm − 0.001 mm = 2.103 mm
12	1.2 mm x 0.500^{11}	0.0005 mm	Target plane not reached

In this example the drilling operation is canceled after the 11th partial stroke and a message appears.

Related Topics

G81: Activate Peck Drilling, page 28

J: Height of Partial Stroke of Peck Drill Hole, page 118

X: X-coordinate

1000	3000	5000
X1 = 0.01 mm	X1 = 0.001 mm	X1 = 0.001 mm

Command	Description
X*x*Y*y*	XY-coordinates (working coordinates, offset coordinates)

Argument	Description
xy	XY-coordinates can be programmed as positive or negative values The unit depends on the set format: ‣ unit in Format 1000: 0.01 mm (millimeter point not allowed) ‣ unit in Format 3000: 0.001 mm (millimeter point allowed) ‣ unit in Format 5000: 0.001 mm (millimeter point allowed)

XY-working coordinates

The working coordinates define all positions at which holes are drilled or tracks are routed.

- Drilling coordinates are all coordinates to which no routing function can be assigned.
- Routing coordinates are programmed in conjunction with a routing function (G1..G50).
- Working coordinates can be interpreted as absolute values (G90) or as incremental values (G91).
- The value range depends on the dimensions of the production machine.

XY-offset coordinates

The offset coordinates can be used for defining a new zero point.

- Offset coordinates are programmed in conjunction with an offset function (M50..M90).
- Offset coordinates are always interpreted as absolute values.
- The value range depends on the dimensions of the production machine.

Related topics

Y: Y-coordinate, page 124

Y: Y-coordinate

1000	3000	5000
Y1 = 0.01 mm	Y1 = 0.001 mm	Y1 = 0.001 mm

Command	Description
X*x*Y*y*	XY-coordinates (working coordinates, offset coordinates)

Argument	Description
xy	XY-coordinates can be programmed as positive or negative values The unit depends on the set format: ▸ unit in Format 1000: 0.01 mm (millimeter point not allowed) ▸ unit in Format 3000: 0.001 mm (millimeter point allowed) ▸ unit in Format 5000: 0.001 mm (millimeter point allowed)

XY-working coordinates

The working coordinates define all positions at which holes are drilled or tracks are routed.

- Drilling coordinates are all coordinates to which no routing function can be assigned.
- Routing coordinates are programmed in conjunction with a routing function (G1..G50).
- Working coordinates can be interpreted as absolute values (G90) or as incremental values (G91).
- The value range depends on the dimensions of the production machine.

XY-offset coordinates

The offset coordinates can be used for defining a new zero point.

- Offset coordinates are programmed in conjunction with an offset function (M50..M90).

- Offset coordinates are always interpreted as absolute values.
- The value range depends on the dimensions of the production machine.

Related topics

X: X-coordinate, page 123

(: Comment

1000	3000	5000
–	•	•

Command	Description
(*text*)	Enter a comment
XxYy (*text*)	Enter a comment

Argument	Description
text	Any text without ^|<|#

Comments can be labeled with round brackets (.

- Comments are entered in round brackets and can be programmed alone in a line or at the end of a program line.
- Comments have no effect on the execution of the part program.
- Comments serve for a better overview in a program.

/: Label Program Line

1000	3000	5000
–	•	•

Command	Description
/XxYy	Label program line

The slash "/" can be used for labeling blocks. The effect depends on the setting of the command CNC command BLKD.

CNC command	Explanation
BLKD	The / program line is ignored.
NOBLKD	The / program line is executed.

9 Optical Measurement

The mode of operation of the optical measurement depends on the machine type. For detailed information refer to the documentation of the machine manufacturer.

The following correction values can be determined during a measuring cycle:

- Offset value
- Scaling value
- Rotation value

If the corrective function has been activated the correction values will be considered during the execution.

Note

Programming other commands after a camera command in the same program line is not allowed.

G30: Deactivate Corrective Function

1000	3000	5000
–	–	•

Command	Description
G30	Deactivate corrective function.
G31	Calculate correction values and activate corrective function.
G34	Deactivate corrective function and clear correction values.

The G31 to G30 commands are used for bracketing the program section in which the correction values are considered.

- The correction values are cleared with the G34 command or via two consecutive G30 blocks.

Example

Note for the example:

- Two positions are measured in the G32 blocks at the beginning of the program.
- The correction values are calculated from the measured deviations.
- These values are considered during the execution of the bracketed program section (G31-G30).

Marginal condition	Setting
Format	5000

`%%5000`	
`X..Y..G32`	Measuring point P1
`X..Y..G32`	Measuring point P2
`G31`	Calculate correction values and activate corrective function.
`( SECTION: T1 )`	
`X..Y..T1 M31 M31`	
`...`	

`XYM50`	
`X..Y..M50M30`	
`( SECTION: T2 )`	
`X..Y..T2 M31`	
`...`	
`XYM50`	
`X..Y..M50M30`	
`G30`	Deactivate corrective function.
`( FINAL OFFSET )`	
`X123.45Y452.34 M50M30`	

Related topics

G31: Calculate Correction Values and Activate Corrective Function, page 127

G31: Calculate Correction Values and Activate Corrective Function

1000	3000	5000
–	–	•

Command	Description
G30	Deactivate corrective function.
G31	Calculate correction values and activate corrective function.
G34	Deactivate corrective function and clear correction values.

The G31-G30 commands are used for bracketing the program section in which the correction values are considered.

- The correction values are cleared with the G34 command or via two consecutive G30 blocks.

Example

Note for the example:

- Two positions are measured in the G32 blocks at the beginning of the program.
- The correction values are calculated from the measured deviations.
- These values are considered during the execution of the bracketed program section (G31-G30).

Marginal condition	Setting
Format	5000

`%%5000`	
`X..Y..G32`	Measuring point P1
`X..Y..G32`	Measuring point P2
`G31`	Calculate correction values and activate corrective function.
`( SECTION: T1 )`	
`X..Y..T1 M31 M31`	
`...`	
`XYM50`	

`X..Y..M50M30`	
`( SECTION: T2 )`	
`X..Y..T2 M31`	
`...`	
`XYM50`	
`X..Y..M50M30`	
`G30`	Deactivate corrective function.
`( FINAL OFFSET )`	
`X123.45Y452.34 M50M30`	

Related topics

G30: Deactivate Corrective Function, page 126

G32: Run Measurement

1000	3000	5000
–	–	•

Command	Description
X*x*Y*y*G32	Run measurement

A measurement is carried out at the XY-position with the G32 command.

- The measurement method depends on the set measurement mode.
- If several layers are measured one after the other, the measuring process in the measurement modes 2 and 3 must be finished with the G33 command after the last layer.
- The measurement mode 5 (measure slot) considers the Y-value of the measuring position only.
- Software version 12.02.001 and higher: The coordinates for camera measuring commands can be programmed as absolute and incremental values.

Measurement mode	Method of measurement
1	Standard measurement
2	Measurement of a layer
3	Measurement of a layer
5	Consider Y-value (slot in X-direction)

Example

Note for the example:

- Two positions are measured in the G32 blocks at the beginning of the program.
- The correction values are calculated from the measured deviations.
- These values are considered during the execution of the bracketed program section (G31-G30).

Marginal condition	Setting
Format	5000

`%%5000`	
`X..Y..G32`	Measuring point P1

`X..Y..G32`	Measuring point P2
`G31`	Calculate correction values and activate corrective function
`( SECTION: T1 )`	
`X..Y..T1 M31 M31`	
`...`	
`XYM50`	
`X..Y..M50M30`	
`( SECTION: T2 )`	
`X..Y..T2 M31`	
`...`	
`XYM50`	
`X..Y..M50M30`	
`G30`	Deactivate corrective function
`( FINAL OFFSET )`	
`X123.45Y452.34 M50M30`	

Example: measure board and calculate correction values

The board is measured at 2 points. Any deviation from the axially parallel position is compensated automatically.

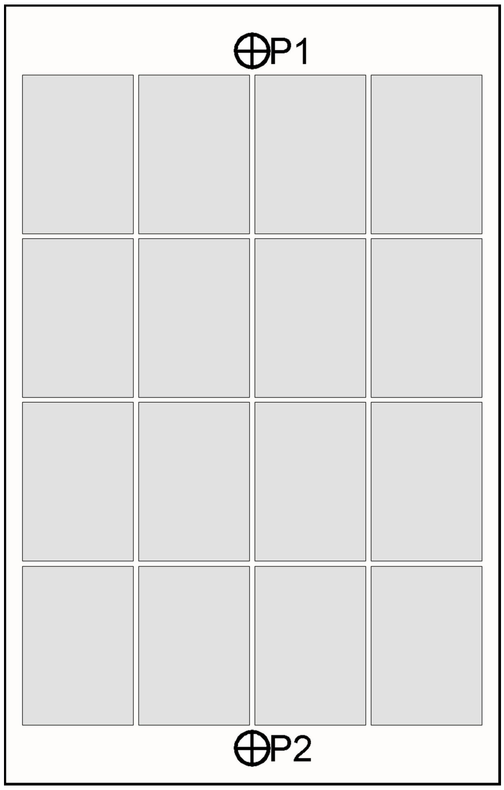


Fig. 74: Measure board and determine correction values

Related topics

G33: Run/Finish Measurement, page 130

G33: Run/Finish Measurement

1000	3000	5000
–	–	•

Command	Description
X*x*Y*y*G33	Run/finish measurement

A measurement is carried out at the XY-position with the G33 command.

- The effect of the command depends on the set measurement mode.
- In the measurement modes 2 and 3 the measuring cycle must be finished at a position with command G33, if several layers have been measured with the G32 command before.
- The measurement mode 5 (measure slot) only considers the X-value of the measuring position.
- Software version 12.02.001 and higher: The coordinates for camera measuring commands can be programmed as absolute and incremental values.

Measurement mode	Method of measurement
1	Default measurement (has the same effect as G32)
2	Finish measurement
3	Finish measurement
5	Consider X-value (slot in Y-direction)

Example

Note for the example:

- measurement mode = 5
- 2 slots into X-direction
- 2 slots into Y-direction
- The correction values are calculated from the measured deviations.
- These values are considered during the execution of the bracketed program section (G31-G30).

Marginal condition	Setting
Format	5000

`%%5000`	
`X..Y..G33`	Y-slot 1: X-value
`X..Y..G32`	X-slot 2: Y-value
`X..Y..G33`	Y-slot 3: X-value
`X..Y..G32`	X-slot 4: Y-value
`G31`	Calculate correction values and activate corrective function
`( SECTION: T1 )`	
`X..Y..T1 M31 M31`	
`...`	
`XYM50`	
`X..Y..M50M30`	
`( SECTION: T2 )`	

`X..Y..T2 M31`	
`...`	
`XYM50`	
`X..Y..M50M30`	
`G30`	Deactivate corrective function
`( FINAL OFFSET )`	
`X123.45Y452.34 M50M30`	

Related topics

G32: Run Measurement, page 128

G34: Deactivate Corrective Function and Clear Correction Values

1000	3000	5000
–	–	•

Command	Description
G30	Deactivate corrective function
G31	Calculate correction values and activate corrective function
G34	Deactivate Corrective Function and Clear Correction Values

The G31 to G30 commands are used for bracketing the program section in which the correction values are considered.

The correction values are cleared with the G34 command or via two consecutive G30 blocks.

G35: Send Byte to Camera Computer

1000	3000	5000
–	–	•

Command	Description
G35,*b*	Send byte to camera computer

Argument	Description
b	Byte (0 .. 255)

The G35 command is used for transmitting a value to the camera computer.

- The meaning of the value depends on the camera computer (e.g. number of a measuring tool etc.).
- The value and the command must be separated by a comma.
- For detailed information refer to the documentation of the machine manufacturer.

G36: Measure Offset for Single Point Correction

1000	3000	5000
–	–	•

Command	Description
X*x*Y*y*G36	Measure offset for single point correction

The G36 command sets the corrective function. The functionality of the G36 command depends on whether the correction values apply for a single point or for a pattern.

Settings in the CNC

CNC command	Explanation
NOCAMA	Single point correction and drilling ▸ First, a measuring cycle is carried out at the G36 position. ▸ Then, a hole is drilled under consideration of the calculated correction values.
CAMA	Board-pattern correction ▸ First, the complete board can be measured(G32 command). ▸ Then, up to 2 measuring points can be programmed in every pattern with the G36 command. ▸ Only one offset value is determined for these two measuring points. ▸ The used correction values are calculated with the sum of the measured values (G32 correction plus G36 offset correction).

- Software version 12.02.001 and higher: The coordinates for camera measuring commands can be programmed as absolute and incremental values.

Example: single point correction

First, a measuring cycle is carried out at every G36 position. Then, a drill stroke is executed.

Note

- The CNC command NOCAMA must be active.
- Every G36 position is measured and an offset is determined on the basis of the measurement result.
- This offset is considered during the following drilling operation.

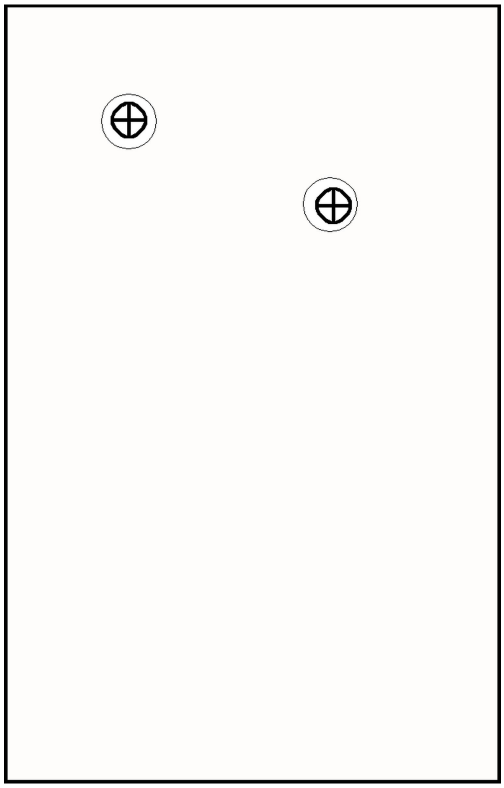


Fig. 75: Measure single point and drill

Marginal condition	Setting
Format	5000

`%%5000`	
`X..Y..T..M31 G36`	Determine offset and drill a hole
`X..Y.. G36`	Determine offset and drill a hole
`X..Y..`	All other positions are drilled without correction. t

`...`	
`XYM50`	
`X..Y..M50M30`	
`G30`	Deactivate corrective function

Example: board-pattern correction

Every pattern is measured individually. This allows individual alignment for every pattern.

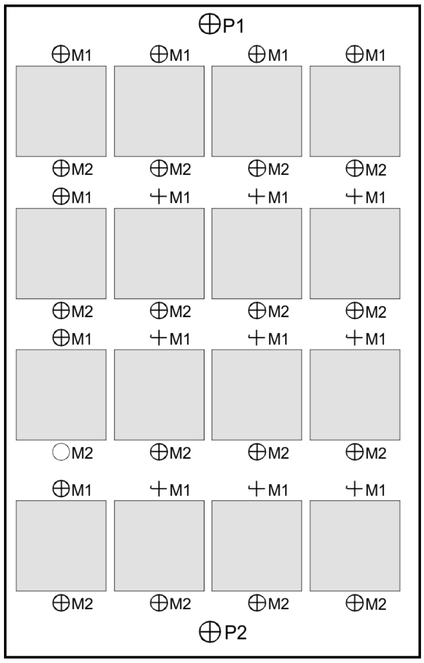


Fig. 76: Measure board and every pattern

Note

- The CNC command CAMA must be active.
- First, the correction values are determined for the complete board with the G32 commands.
- In every pattern two additional positions are measured with the G36 commands.
- The correction values are calculated with the sum of the measured values (G32 correction plus G36 correction).

Marginal condition	Setting
Format	5000

`%%5000`	
`X..Y..G32`	Program measuring point P1
`X..Y..G32`	Program measuring point P2
`G31`	Calculate correction values and activate corrective function
`M31`	
`X..Y..G36`	Point of pattern measurement M1
`X..Y..G36`	Point of pattern measurement M2
`X..Y..T..`	
`...`	
`X..Y..M50`	
`...`	
`X..Y..M50M30`	
`G30`	Deactivate corrective function

G39: Organize Correction Values

1000	3000	5000
–	–	•

Command	Description
G39,CLEAR	Clear correction values (memory location = STEP number)
G39,CLEAR*n*	Clear correction values (storage location= *n*)
G39,SAVE	Save correction values (memory location = STEP number)
G39,SAVE*n*	Save correction values (storage location = *n*)
G39,LOAD	Load correction values (memory location = STEP number)
G39,LOAD*n*	Load correction values (storage location = *n*)
G39,SPOP	Spindle-camera correction = calculated offset at work station 1
G39,SPOP1	Spindle-camera correction = calculated offset at work station 1
G39,SPOP2	Spindle-camera correction = calculated offset at work station 2
G39,SPISPI	Spindle-spindle correction = calculated offset (only for machines with spindle switch-over)
G39,REQCOMP	Get correction value from the camera computer (requirement: measurement mode = 3)
G39,DELALL	Clear all saved correction values
G39,SPOF*s*	Offset value for individual spindle = calculated offset

Argument	Description
n	Storage location
s	Spindle number (1 to 12, independent from the setting of the machine parameters Z.NUMBER and Z.SPINDLENUM by the machine manufacturer)

The G39 is used for organizing correction values.

- To prevent any unnecessary tool changes during the execution of part programs, correction values can be saved and loaded, if required.
- If no storage location is programmed, the CNC will use the current STEP value.
- Max. 1000 correction values can be saved (correction value = offset, rotation and scaling values).
- The correction values are automatically cleared, when starting an execution.
- The command must be separated from G39 by a comma.

Example

Note for the example:

- The CNC command NOCAMA must be active.
- Saving measuring values is useful, if they will be required later.

In the example 16 patterns are to be measured and manufactured after. To prevent time-consuming tool changes or measuring cycles, the correction values for every pattern are saved and loaded from the memory, if required.

Marginal condition	Setting
Format	5000

`%%5000`	
`M31`	
`G34`	Clear correction values
`X..Y..G32`	Measuring point P1
`X..Y..G32`	Measuring point P2

`G31`	Calculate correction values and activate corrective function
`G39,SAVE`	SAVE current correction values
`G30`	Delete table with the camera data
`X..Y..M50`	1st pattern
`...`	
`X..Y..M50M30`	16th pattern
`M31`	
`G39,LOAD`	LOAD current correction values
`X..Y..T1`	1st drilling position for T1
`...`	
`X..Y..`	Last drilling position for T1
`X..Y..M50`	1st pattern
`...`	
`X..Y..M50M30`	16th pattern
`G30`	Deactivate corrective function

M75: Send String to Camera Computer

1000	3000	5000
–	–	•

Command	Description
M75,*s*	Send string to camera computer

Argument	Description
s	Character string

The M75 command sends a character string to the camera computer.

- The meaning of the character string depends on the camera computer (e.g. coordinate value etc.).
- The character string and the command must be separated by a comma.
- The CNC command CAMM,J defines whether or not the CNC waits for a response after the transmission.
- The maximum waiting time is defined with the CNC command CATM.

For detailed information, refer to the documentation of the machine manufacturer.

10 Surface detection

This section describes commands used for programming surface detection in the part program. The CNC supports the following methods for SLM:

- Measure/Use the surface: The surface of the board or backup material is measured at programmed positions and saved in a surface memory. During the execution these values are loaded by means of commands from the memory and used for positioning the Z-axis. Only the surface value ahead the T0 block is considered for a complete routing contour. The function only works in combination with the depth control of the Z-axis.
- Surface detection in the grid: The surface of the board or backup material is measured at programmed positions, i.e. automatically, and saved in the surface memory. The interpolated values are used automatically for the execution. This method also allows considering modified surface values step-by-step within routing contours. The function only works in combination with the standard lowering method of the Z-axis.

Note

The command set of CNC 8x.00 listed in the following applies to software version 12.01.001 for SLM.

Please refer to the manual *CNC 8x.00 − Spot Facing* for the command set of CNC 8x.00 up to software version 11.17.006.

10.1 SLM

This section describes commands for programming surface detection (SLM) in the part program.

G70: Deactivate SLM

1000	3000	5000
–	–	•

Command	Description
G70	Deactivate SLM

The G70 command deactivates the function for detecting, using and storing a reference surface for depth control.

- All settings are kept.
- The G71 command activates the function again. The original settings will be used.

Note

If surface detection is programmed with Step-and-Repeat, it has to be activated and deactivated. Otherwise the error "'STEP+REPEAT' error + Usage of manual measurements (LH mode) not finished. (G70/G110 is missing)" (error 25/100) occurs.

Example: Measurement

Holes with an exact depth shall be drilled at different positions during the execution of a part program. Surface detection is not possible when a hole is drilled. Therefore, the surface of a board is scanned at several positions at the beginning of the program. The values are saved and called during part program execution.

Marginal condition	Setting
Format	5000

`%%5000`	
`( MEASURING )`	
`G71`	Activate SLM
`X165.Y15.G73L1`	1st surface scan; save measurement result in surface memory L1
`G75`	Retract measuring probe (optional)
`X110.Y10.G73L2`	2nd surface scan; save measurement result in surface memory L2
`G75`	Retract measuring probe (optional)
`( PRODUCTION )`	
`G71K.4`	Relative working plane = 0.4 mm
`L1`	Get measured value from surface memory L1
`X165.Y15.T1`	1st drill stroke, reference surface = L1 value, plunge depth = 0.4 mm
`X170.Y15.`	2nd drill stroke, reference surface = L1 value, plunge depth = 0.4 mm
`G71K.5`	Relative working plane = 0.5 mm
`L2`	Get measured value from surface memory L2
`X110.Y10.T0T5`	Beginning of routing
`X110.Y30.G1F1.5`	Routing process, reference surface = L2 value, plunge depth = 0.5 mm
`X110.Y30.T0`	End of routing
`G70`	Deactivate SLM
`X..Y..`	Program section without depth control
`$`	Program end
`T1D..S...`	
`T2D..S...`	
`$`	

Related topics

G71: Activate SLM and Define Relative Working Plane, page 137

G71: Activate SLM and Define Relative Working Plane

1000	3000	5000
–	–	•

Command	Description
G71	Activate SLM
G71K*k*	Define relative working plane

Argument	Description
k	Relative working plane, reference surface = current reference value ▸ positive value: plunge depth ▸ negative value: residual web

The G71 command activates SLM .

- Activating the function also activates depth control of the Z-axis. The feed rate for lowering the Z-axis is controlled that way that the axis does not overshoot when the working plane is reached.
- Optionally, the relative working plane can be programmed with the G71 command (G71K).
  - If the working plane is below the reference surface, the programmed value must be positive (e.g. G71K1.5).
  - If the working plane is above the reference surface, the programmed value must be negative (e.g. G71K−2.0).
- Note: The surface detection plane (detection of the reference surface) must be set with the CNC command BOTK or TOPK.

Note

If surface detection is programmed with Step-and-Repeat, it has to be activated and deactivated. Otherwise the error "'STEP+REPEAT' error + Usage of manual measurements (LH mode) not finished. (G70/G110 is missing)" (error 25/100) occurs.

Note

If an SLM probe is used, the following applies:

- (Up to software version 12.05.007) If a part program contains several consecutive G71 commands without any G70 command programmed in between, the Z-axis is retracted to the machine stop plane (Z.EMG) with every G71 command.
- (From software version 20.12.015 and higher) If a part program contains several consecutive G71 commands, without any G70 command programmed in between, the Z-axis is retracted to the machine stop plane (Z.EMG) with the first G71 command.

Example: Measurement

Holes with an exact depth shall be drilled at different positions during the execution of a part program. Surface detection is not possible when a hole is drilled. Therefore, the surface of a board is scanned at several positions at the beginning of the program. The values are saved and called during part program execution.

Marginal condition	Setting
Format	5000

Code	Comment
`%%5000`	
`( MEASURING )`	
`G71`	Activate SLM
`X165.Y15.G73L1`	1st surface scan; save measurement result in surface memory L1
`G75`	Retract measuring probe (optional)
`X110.Y10.G73L2`	2nd surface scan; save measurement result in surface memory L2
`G75`	Retract measuring probe (optional)

`( PRODUCTION )`	
`G71K.4`	Relative working plane = 0.4 mm
`L1`	Get measured value from surface memory L1
`X165.Y15.T1`	1st drill stroke, reference surface = L1 value, plunge depth = 0.4 mm
`X170.Y15.`	2nd drill stroke, reference surface = L1 value, plunge depth = 0.4 mm
`G71K.5`	Relative working plane = 0.5 mm
`L2`	Get measured value from surface memory L2
`X110.Y10.T0T5`	Beginning of routing
`X110.Y30.G1F1.5`	Routing process, reference surface = L2 value, plunge depth = 0.5 mm
`X110.Y30.T0`	End of routing
`G70`	Deactivate SLM
`X..Y..`	Program section without depth control
`$`	Program end
`T1D..S...`	
`T2D..S...`	
`$`	

Related topics

G70: Deactivate SLM, page 136

G73: Detect/Save Reference Surface, page 139

G72: Clear Surface Memory

1000	3000	5000
–	–	•

Command	Description
G72	Clear all values in the surface memory

The G72 command deletes all values from the surface memory.

Note

When a part program is loaded and at the beginning of a program run all values in the surface memory and in the average memory will be cleared.

Related Topics

G73: Detect/Save Reference Surface, page 139

G73: Detect/Save Reference Surface

1000	3000	5000
–	–	•

Command	Description
G73	Save the current measured value in the averaging area
G73L*n*	Save the current measured value in the surface memory L*n*
X*x*Y*y*G73	Measure surface at the XY-position and save the result in the averaging area
X*x*Y*y*G73L*n*	Measure surface at the XY-position and save the result in the surface memory L*n*

Argument	Description
n	Number of the storage location in the surface memory
x	X-coordinate of the measuring position
y	Y-coordinate of the measuring position

The G73 command allows measuring the surface and saving the current measured value in the surface memory.

Note

When executing the command for the first time, it will be checked if the SLM device is calibrated. If this is not the case, an error message appears.

Note

Detected surface values are saved and used in relation to the tool tip assuming that the tool is ideally clamped in the collet.

G73

The current measured value is saved in the averaging area. The measured value is for example the last surface detected during surface measurement. The averaging area has a memory for a max. 100 values. An average value is calculated with the G74 command.

G73L *n*

The current measured value is saved in the surface memory L*n* (n = number of the storage location).

X *x*Y*y*G73

The surface is measured at the XY-position and the measurement result is saved in the averaging area. The averaging area has a memory for max. 100 values.

X *x*Y*y*G73L*n*

The surface is measured at the XY-position and the measurement result is saved in the surface memory L*n*.

Note

If the No Tool function is active, a warning message appears when activating the command G73.

If no reference surface could be detected, an error message appears and the CNC stops the execution. The measurement must be repeated.

Note

To ensure correct execution of the function "Start with block", each program section has to begin with the command G71 and finished with G70.

Note

If the CNC command T0 is located in the line of the program command, the tool is dropped off.

Note

If the CNC command TD is located in the line of the program command, the dummy tool is picked up. Requirement: The machine is equipped with a dummy tool (see machine parameter DUMMY).

Example: Measurement

Holes with an exact depth shall be drilled at different positions during the execution of a part program. Surface detection is not possible when a hole is drilled. Therefore, the surface of a board is scanned at several positions at the beginning of the program. The values are saved and called during part program execution.

Marginal condition	Setting
Format	5000

`%%5000`	
`( MEASURING )`	
`G71`	Activate SLM
`X165.Y15.G73L1`	1st surface scan; save measurement result in surface memory L1
`G75`	Retract measuring probe (optional)
`X110.Y10.G73L2`	2nd surface scan; save measurement result in surface memory L2
`G75`	Retract measuring probe (optional)
`( PRODUCTION )`	
`G71K.4`	Relative working plane = 0.4 mm
`L1`	Get measured value from surface memory L1
`X165.Y15.T1`	1st drill stroke, reference surface = L1 value, plunge depth = 0.4 mm
`X170.Y15.`	2nd drill stroke, reference surface = L1 value, plunge depth = 0.4 mm
`G71K.5`	Relative working plane = 0.5 mm
`L2`	Get measured value from surface memory L2
`X110.Y10.T0T5`	Beginning of routing
`X110.Y30.G1F1.5`	Routing process, reference surface = L2 value, plunge depth = 0.5 mm
`X110.Y30.T0`	End of routing
`G70`	Deactivate SLM
`X..Y..`	Program section without depth control
`$`	Program end
`T1D..S...`	
`T2D..S...`	

```
$
```

Related Topics

[G70: Deactivate SLM, page 136]
[G71: Activate SLM and Define Relative Working Plane, page 137]
[G74: Calculate and Save Average Value, page 142]
[L: Space in Surface Memory, page 146]

G74: Calculate and Save Average Value

1000	3000	5000
–	–	•

Command	Description
G74L*n*	Calculate average value and save the value in the surface memory L*n*

Argument	Description
n	Number of the storage location in the surface memory

The G74 command calculates an average value and saves it in the surface memory.

- The average value is calculated on the basis of the values saved in the averaging area.
- The averaging area is filled with the command G73 (without L value).
- The calculation method can be configured:
  - Result = arithmetic average value. All measured values are added and divided by the number of measured values (CNC command SPAV,A).
  - Result = calculate average value from the median, values are added in ascending order. The averaging value for an odd number of measured values is the median; for an even number of measured values the two values in the middle are added and divided by 2 (CNC command SPAV,M).
  - Surface value = average value of the smallest and greatest measured value (CNC command SPAV,O).
  - Result = average value of all measured values, except from the smallest and largest value (CNC command SPAV,T).
- The calculated average value is saved in the storage location L*n* of the surface memory.
- Then, all entries are cleared from the averaging area.

Example: calculation of average value

An average value shall be calculated on the basis of several scanned surface values at different positions. The calculated average value will be saved in the surface memory.

Solution:

- All scanning processes are programmed with the G73 command (without L). This way the measurement result is saved in the averaging area.
- The G74 command (with L) calculates the average value and the result is saved in the surface memory. All entries are cleared from the averaging area.
- Note: The operator can save the content of the surface memory in an external file via the “Save” button.

Marginal condition	Setting
Format	5000

`%%5000`	
`( MEASURING )`	
`G71`	Activate SLM
`X130.Y120.G73`	Scanning process, save measurement result in averaging area
`X140.Y130.G73`	Scanning process, save measurement result in averaging area
`X120.Y120.G73`	Scanning process, save measurement result in averaging area
`X130.Y130.G73`	Scanning process, save measurement result in averaging area
`G74L1`	Calculate average value, save result in the surface memory L1
`( PRODUCTION )`	
`...`	
`L1`	Get measured value from surface memory L1
`G71Z.4`	Relative working plane = 0.4 mm
`X165.Y15.T1`	1st drill stroke, reference surface = L1 value, plunge depth = 0.4 mm
`X170.Y15.`	2nd drill stroke, reference surface = L1 value, plunge depth = 0.4 mm
`G71Z.5`	Relative working plane = 0.5 mm
`X125.Y125.T0T5`	Beginning of routing
`X135.Y135.G1F1.5`	Routing, plunge depth = 0.5 mm
`X135.Y135.T0`	End of routing
`L2`	Get measured value from surface memory L2
`...`	Reference surface = L2 value, plunge depth = 0.5 mm
`G70`	Deactivate SLM
`X..Y..`	Program section without depth control
`...`	...
`$`	Program end
`T1D..S...`	
`T2D..S...`	
`$`	

Example: Calculation of average value with protocol

An average value shall be calculated on the basis of the measured surface values according to the previous example. The measurement results are saved in the averaging area and in the surface memory. This allows checking, whether all required positions have been scanned correctly. However, only the calculated average value will be used for the execution.

Solution:

- All measurement processes are programmed with the G73 command (without L). This way the measured value is saved in the averaging area.

- In the next block the G73 command (with L, without coordinates) is used to additionally save the current value in the surface memory.
- The G74 command calculates the average value. In addition, all entries are cleared from the averaging area.
- Then, the operator can save the surface memory in an individual file.

Marginal condition	Setting
Format	5000

`%%5000`	
`( MEASURING )`	
`G71`	Activate SLM
`X130.Y120.G73`	Scanning process, save measurement result in averaging area
`G73L101`	Save measurement result additionally in surface memory L101 (log)
`X140.Y130.G73`	Scanning process, save measurement result in averaging area
`G73L102`	Save measurement result additionally in surface memory L102 (log)
`X120.Y120.G73`	Scanning process, save measurement result in averaging area
`G73L103`	Save measurement result additionally in surface memory L103 (log)
`X130.Y130.G73`	Scanning process, save measurement result in averaging area
`G73L104`	Save measurement result additionally in surface memory L104 (log)
`G74L1`	Calculate average value, save result in the surface memory L1
`( PRODUCTION )`	
`...`	
`L1`	Get measured value from surface memory L1
`G71Z.4`	Relative working plane = 0.4 mm
`X165.Y15.T1`	1st drill stroke, reference surface = L1 value, plunge depth = 0.4 mm
`X170.Y15.`	2nd drill stroke, reference surface = L1 value, plunge depth = 0.4 mm
`G71Z.5`	Relative working plane = 0.5 mm
`X125.Y125.T0T5`	Beginning of routing
`X135.Y135.G1F1.5`	Routing process, reference surface = L1 value, plunge depth = 0.5 mm
`X135.Y135.T0`	End of routing
`L2`	Get measured value from surface memory L2
`...`	Reference surface = L2 value, plunge depth = 0.5 mm
`G70`	Deactivate SLM

`X..Y..`	Program section without depth control
`...`	...
`$`	Program end
`T1D..S...`	
`T2D..S...`	
`$`	

Related Topics

L: Space in Surface Memory, page 146

G73: Detect/Save Reference Surface, page 139

G75: Retract Measuring Probe

1000	3000	5000
–	–	•

Command	Description
G75	Retract measuring probe

The G75 command retracts the measuring probe.

Note

The measuring probe is retracted automatically at the end of a program section with G73.

Example

After surface scanning the measuring probe shall be retracted via a function in the program.

Marginal condition	Setting
Format	5000

`%%5000`	
`( MEASURING )`	
`G71`	Activate SLM
`X10.Y120.G73L1`	Scanning process, save measurement result in surface memory L1
`X20.Y120.G73L2`	Scanning process, save measurement result in surface memory L2
`X30.Y120.G73L3`	Scanning process, save measurement result in surface memory L3
`G75`	Retract measuring probe (optional)
`( PRODUCTION )`	
`G71K.2`	Define relative working plane
`L1`	Get measured value from surface memory L1
`X12.Y120.T8`	Drill depth hole

`L2`	Get measured value from surface memory L2
`X22.Y120`	Drill depth hole
`L3`	Get measured value from surface memory L3
`X32.Y120`	Drill depth hole
`G70`	Deactivate SLM
`...`	...
`$`	Program end
`T1D..S...`	
`T2D..S...`	
`$`	

Related Topics

L: Space in Surface Memory, page 146

L: Space in Surface Memory

1000	3000	5000
–	–	•

Command	Description
L*n*	Number of the location in the surface memory

Argument	Description
n	Number of the storage location in the surface memory

The L command saves a value in the surface memory and loads it from the surface memory. The meaning of the command depends on the kind of programming:

- Save current measurement result in surface memory: The L command must be programmed in the same block with the G73 or G74 command.
- Get value from surface memory. The L command must be programmed in a separate block.

Note

If a value of the surface memory is used during the execution, the board surface is not checked when the Z-axis is lowered. This implies that the tool length is also not monitored (SUTO monitoring).

Example: Measurement

Holes with an exact depth shall be drilled at different positions during the execution of a part program. Surface detection is not possible when a hole is drilled. Therefore, the surface of a board is scanned at several positions at the beginning of the program. The values are saved and called during part program execution.

Marginal condition	Setting
Format	5000

```
%%5000
( MEASURING )
```

`G71`	Activate SLM
`X165.Y15.G73L1`	1st surface scan; save measurement result in surface memory L1
`G75`	Retract measuring probe (optional)
`X110.Y10.G73L2`	2nd surface scan; save measurement result in surface memory L2
`G75`	Retract measuring probe (optional)
`( PRODUCTION )`	
`G71K.4`	Relative working plane = 0.4 mm
`L1`	Get measured value from surface memory L1
`X165.Y15.T1`	1st drill stroke, reference surface = L1 value, plunge depth = 0.4 mm
`X170.Y15.`	2nd drill stroke, reference surface = L1 value, plunge depth = 0.4 mm
`G71K.5`	Relative working plane = 0.5 mm
`L2`	Get measured value from surface memory L2
`X110.Y10.T0T5`	Beginning of routing
`X110.Y30.G1F1.5`	Routing process, reference surface = L2 value, plunge depth = 0.5 mm
`X110.Y30.T0`	End of routing
`G70`	Deactivate SLM
`X..Y..`	Program section without depth control
`$`	Program end
`T1D..S...`	
`T2D..S...`	
`$`	

Related Topics

G73: Detect/Save Reference Surface, page 139

G74: Calculate and Save Average Value, page 142

10.2 Surface Detection in the Grid

This section describes commands used for programming grid measuring of the surface structure in the part program.

G78: Detect Reference Value and Clear Measured Grid Values

1000	3000	5000
–	–	•

Command	Description
XxYy G78	Detect reference value and clear measured grid values in surface memory

Note

This command is not required for SLM, but is still available for compatibility reasons. G78 works like G79.

Example

All in all, one measuring cycle each must be done at 120 XY-positions to detect the profile of the backup material (10 positions into X-direction and 12 positions into Y-direction). The XY-distance is 10.0 mm. The reference position is at the position "X10.Y5.". The allowed Z-deviation between the reference positions and the measuring positions is 1.0 mm.

Marginal condition	Setting
Format	5000

`M49,MAPT1.0`	Maximum admissible Z-deviation between the reference value and the current measurement result
`X10.Y5.T1G78`	Run measuring cycle and detect reference value
`XYG79M31M31`	Run measuring cycle and check/save the current measuring deviation
`XYM50V2`	1st column of the measuring positions
`X90.YM50W10M30`	Last column of the measuring positions (... in X-direction)
`XYM50V2`	1st row of the measuring positions
`XY110.M50W12M30`	Last row of the measuring positions (... in Y-direction)

Related Topics

G79: Run Measuring Cycle and Save Measuring Deviation, page 148

G79: Run Measuring Cycle and Save Measuring Deviation

1000	3000	5000
–	–	•

Command	Description
X*x*Y*y* G79	Run measuring cycle and save measuring deviation

The command G79 is used to run a measurement.

- After the measurement has been finished, the measurement result is compared with a reference value (for every Z-axis). The reference value for through-hole drilling is defined with the CNC command BOTR/TOPR. If no value is defined with BOTR/TOPR, the first measured value with G79 is saved as reference value.
- The difference between the reference value and the current measurement result must not exceed the admissible tolerance (CNC command BOTT,R/TOPT,R). Otherwise, an error message appears.
- Valid values are saved as measured grid values in the surface memory. There is one surface memory per work station.
- If a tool T.. is programmed in a block with G79 command, this tool has priority over the setting with the CNC command BOTL/TOPL.

Note

G78 works like G79.

Note

If the CNC command T0 is located in the line of the program command, the tool is dropped off.

Note

If the CNC command TD is located in the line of the program command, the dummy tool is picked up. Requirement: The machine is equipped with a dummy tool (see machine parameter DUMMY).

Whether a through hole stroke or depth hole stroke is carried out during measurement is defined by programming:

- For the depth in the SIEB & MEYER code: G83K..
- Exception: If BOTK/TOPK is defined, a depth hole stroke is always carried out.

Example

All in all, one measuring cycle each must be done at 120 XY-positions to detect the profile of the backup material (10 positions into X-direction and 12 positions into Y-direction). The XY-distance is 10.0 mm. The reference position is at the position "X10.Y5.". The allowed Z-deviation between the reference positions and the measuring positions is 1.0 mm.

Marginal condition	Setting
Format	5000

`M49,MAPT1.0`	Maximum admissible Z-deviation between the reference value and the current measurement result
`X10.Y5.T1G78`	Run measuring cycle and detect reference value
`XYG79M31M31`	Run measuring cycle and check/save the current measuring deviation
`XYM50V2`	1st column of the measuring positions
`X90.YM50W10M30`	Last column of the measuring positions (... in X-direction)
`XYM50V2`	1st row of the measuring positions
`XY110.M50W12M30`	Last row of the measuring positions (... in Y-direction)

Related Topics

G78: Detect Reference Value and Clear Measured Grid Values, page 147

11 Depth Control

The CNC distinguishes two control processes for the Z axis control.

- standard control
- depth control

Standard control

The table surface is the reference plane for the traveling plane and for the working plane.

- Every time an execution is started, the standard control is automatically activated.
- Usually, the machine manufacturer sets the parameters of the standard control for optimal speed.
- This may cause that the Z-axis marginally falls below the working plane.
- For further information refer to the documentation of the machine manufacturer.

Depth control

The board surface is the reference plane for the traveling plane and for the working plane.

- The functioning of the depth control depends on the configuration of the machine parameters.
- The G83 command activates the depth control.
- The depth control remains on until the execution is finished or until a G82 command is called.
- Usually, the exact plunge depth for the depth control is parameterized by the machine manufacturer.
- This ensures that the tool tip is lowered exactly up to the programmed plunge depth.
- For further information refer to the documentation of the machine manufacturer.

Example

Depth control can be flexibly activated/deactivated in the part program.

`T01`	Depth-controlled routing
`G00X01Y01`	
`M15`	
`M18Z2.`	Depth control = ON. Reference plane = board surface. The Z2. parameter defines a plunge depth of 2.0 mm below the board surface.
`G01X05`	
`M18Z1.5`	
`G01Y05`	
`M17`	Through-hole routing
`M19`	Depth control is deactivated
`G00X01Y01`	
`M15`	
`Z2.`	The CNC command PRGM,U must be active for correct interpretation of these Z-values.
`G01X05`	

```
Z1.5
G01Y05
M17
M30
```

G82: Deactivate Depth Control

1000	3000	5000
–	•	•

Command	Description
X*x*Y*y* G82	Deactivate depth control
X*x*Y*y* G82Z*z*	Deactivate depth control and define working plane
X*x*Y*y* G83	Activate depth control
X*x*Y*y* G83K*k*	Activate depth control and define plunge depth

Argument	Description
xy	The programmed coordinate is already executed with the function.
z	The Z-value is related to the table surface
k	The K-value is related to the board surface

Hardware requirements

To use depth control the machine must be equipped correspondingly. Examples:

- contact drilling module
- separate depth measuring system at each Z-axis and measuring switch for tool measurement

Deactivate depth control

The reference plane is the table surface.

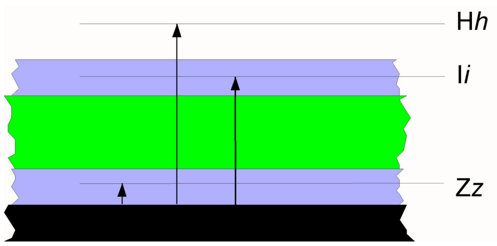


Fig. 77: Deactivate depth control

Reference plane = table surface (Z0)

- If no Z-value is programmed, the manually entered Z-value is used as working plane.
- Dependent on the machine type the Z0-plane is located above the table surface. For detailed information refer to the documentation of the machine manufacturer.All other Z-axes values refer to the Z0-plane:
  – Z = working plane
  – H = traveling plane
  – I = predrill plane during peck drilling (G81)

Example

The plunge depth is defined with the CNC command K.

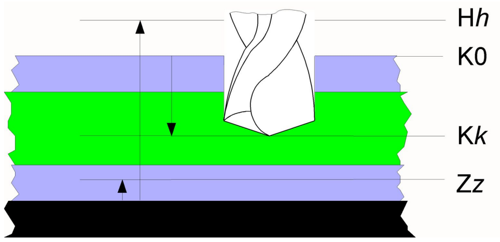


Fig. 78: Depth control activated

Depth control can be flexibly activated/deactivated in the part program.

Marginal condition	Setting
Format	5000

`X20.Y30.T5T0 Z3.`	Depth control = OFF (default). Reference plane = table surface. The Z-plane is defined with 3.0 mm above the table surface.
`X20.Y30.G1 F1.2`	Depth control = OFF. Rout straight line. During routing the tip of the cutter is lowered to the Z-plane Z3 (=3.0 mm).
`X40.Y30.`	Depth control = OFF. Rout straight line.
`X40.Y30.G83K2.`	Depth control = ON. Reference plane = board surface. The K2 parameter defines a plunge depth of 2.0 mm below the board surface (inclusive routing pad).
`X60.Y50.`	Depth control = ON. Rout straight line.
`X90.Y80.T0 G82`	Depth control = OFF. Reference plane = table surface. Thus, the Z-plane (3.0 mm) is used as working plane again. The K-value is kept.

Related topics

G83: Activate Depth Control, page 152

G83: Activate Depth Control

1000	3000	5000
–	•	•

Command	Description
X*x*Y*y* G82	Deactivate depth control
X*x*Y*y* G82Z*z*	Deactivate depth control and define working plane
X*x*Y*y* G83	Activate depth control
X*x*Y*y* G83K*k*	Activate depth control and define plunge depth

Argument	Description
xy	The programmed coordinate is already executed with the function.
z	The Z-value is related to the table surface
k	The K-value is related to the board surface

Hardware requirements

To use depth control the machine must be equipped correspondingly. Examples:

- contact drilling module

- separate depth measuring system at each Z-axis and measuring switch for tool measurement

Activate depth control

The reference plane is the board surface. The table surface is used as reference plane for the limits (H and Z).

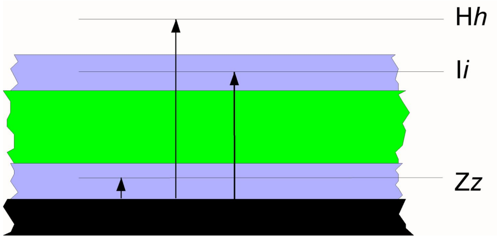


Fig. 79: Activate depth control

Reference plane = board surface

- The parameter K defines the plunge depth of the tool(e.g. for drilling blind hole and routing grooves).
- If no K-value is programmed, the manually entered K-value is used as plunge depth.
- The K-value is kept after deactivating and reactivating depth control.
- The following Z-axis values refer to the K0-plane (board surface):
  - K = working depth
  - QUIK = traveling plane, related to the board surface
- The following values still refer to the Z0-plane (table surface):
  - Z = lower limit for the Z-axis movement
  - H = upper limit for Z-axis movement

Note

CNC command DEPM,Z is active:

A program analysis automatically starts after a part program was loaded. To ensure validity of the values displayed in the K depth values column on the page “Tools (static 3)”, the program analysis must be finished.

The K-value must be repeated after a G82 command. Otherwise, the displayed values for the following tools are missing.

Example

The plunge depth is defined with the CNC command K.

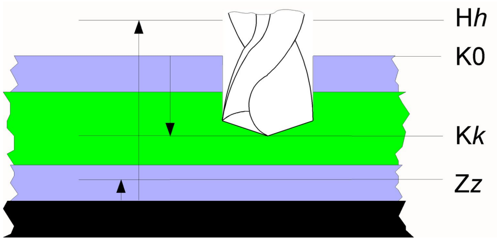


Fig. 80: Depth control activated

Depth control can be flexibly activated/deactivated in the part program.

Marginal condition	Setting
Format	5000

`X20.Y30.T5T0 Z3.`	Depth control = OFF (default). Reference plane = table surface. The Z-plane is defined with 3.0 mm above the table surface.
`X20.Y30.G1 F1.2`	Depth control = OFF. Rout straight line. During routing the tip of the cutter is lowered to the Z-plane Z3 (=3.0 mm).
`X40.Y30.`	Depth control = OFF. Rout straight line.
`X40.Y30.G83K2.`	Depth control = ON. Reference plane = board surface. The K2 parameter defines a plunge depth of 2.0 mm below the board surface (inclusive routing pad).
`X60.Y50.`	Depth control = ON. Rout straight line.
`X90.Y80.T0 G82`	Depth control = OFF. Reference plane = table surface. Thus, the Z-plane (3.0 mm) is used as working plane again. The K-value is kept.

Related topics

G82: Deactivate Depth Control, page 151

12 Appendix

A File Format

This section describes the structure of part program files.

A.1 SIEB & MEYER CNCs

There are different standards for the SIEB & MEYER CNC generations. The format setting includes the following information:

- Program structure: How part programs are saved on a data medium.
  - %% = beginning of the part program
  - $ = beginning of tool parameters etc.
- Interpretation of coordinates:
  - metric or inch
  - resolution (0.01 or 0.001 or 0.0001)
  - leading or trailing zeros
- Command set: The range of available programming commands (G1, M50, T1, etc.) and the corresponding programming rules.

A.2 ISO Code

The valid signs of a part program corresponds to the ISO code (DIN 66024). Since the ISO code is a subset of the 7-bit ASCII code, part programs can be programmed or edited on computers with ASCII-compatible text editor.

Note

The ISO code includes capital letters only!

A.3 Structure of a program line

Program lines have the following structure in all SIEB & MEYER formats:

- The X-coordinate stands at the beginning of a program line and
- is followed by the Y-coordinate.
- Then, a tool change is programmed. Note:The coordinates of this program line are already executed with the new tool.
- After the tool change, additional commands and parameters follow (routing commands, rows of holes, repetitions, offsets etc.) If commands are programmed twice in one line, the last entry is considered. Command that include plain text must be programmed at the end of the program line. Otherwise, other commands are interpreted as plain text.
- All command lines are finished with a line break.

`%%5000`	Code: beginning of program (Format 5000)
`...`	
`X12.345Y34.567T1M31`	Left bracket for tool change and Step-and-Repeat

`...`	
`X50.Y76.T0`	Routing starts
`X55.Y76.G1F1.2`	Routing condition
`X55.Y76.T0`	End of routing
`...`	
`X100.Y300.T7M97,ABC`	Tool change and plain text drilling
`...`	
`X34567Y45678M50M30`	Right bracket for offset command and Step-and-Repeat
`$`	Code: tool parameters
`...`	

A.4 End of Line (Line Break)

Every program line must be finished with a line break! Depending on the operating system, application or editor you can use the following Hex codes as (0x = the following hexadecimal signs are numerical values).

- 0x0A (line feed)
- 0x0D (carriage return)
- 0x0A0D (line feed, carriage return)
- 0x0D0A (carriage return, line feed)

When in doubt just press the return key in the editor. This a line break.

A.5 Format 1000

Format 1000 was developed for the CNC 25.05. The available commands and the corresponding programming rules are described in the programming manual of the CNC 25.05.

A.5.1 Definitions for Format 1000

Format 1000 was developed for the CNC 25.05.

Comment

Any comment can be programmed at the beginning of a file.

- The max. number of characters in a comment line is 64. Every comment line must be finished with a line break.
- How comments are processed depends on the used CNC.
  - CNC 25.05: When a program is loaded comments are listed on the screen and are not adapted into the program memory.
  - CNC 35.00: When a punched tape is loaded comments are listed on the screen and are not adapted into the program memory.
  - CNC 4x.00 and higher: Comments are ignored when the program is loaded.

`PROGRAM EXAMPLE`	Comment

`CREATED 18.07.89`	Comment
`MODIFIED 20.07.89`	Comment
`%%`	Code: beginning of program
`...`	

Beginning of program

The beginning of a program is marked by two percentage signs (%%), followed by a line break.

`...`	Comment
`%%`	Code: Beginning of program
`...`	

Program Line

The program lines are the actual part program.

- The max. length of a program is 64 signs.
- Every program line must be finished with a line break.
- In the manuals a program line is also referred to as "block".
- For detailed information refer to the programming manual of the SIEB & MEYER CNC 25.05.

`%%`	Code: Beginning of program
`...`	
`X..Y..`	
`...`	
`$`	Code: tool parameters

Tool Parameters

Tool parameters are programmed at the end of the part program.

- The section for tool parameters starts with a dollar sign ($).
- Format 1000 allows max. 15 tool parameters at the end of the program.
- The parameters must be defined for all (15) tool numbers.
- Each parameter value is closed with the "$" sign.
- If a parameter value is zero, only the $ sign is given out.
- The order of tool parameters is fixed.
- Magazine assignments can not be modified. The following default assignment applies: T1 in M1, T2 in M2, T3 in M3 etc.

Parameter	Unit	Description	Example
D	0.01 mm	Tool diameter	80$ = 0.8 mm
S	1000 rpm	Spindle speed	55$ = 55000 rpm
F	0.1 m/min	Feed rate	12$ = 1.2 m/min
R	0.1 m/min	Retract feedrate	99$ = 9.9 m/min
N	100 strokes	Preset tool life	30$ = 3000 strokes

Parameter	Unit	Description	Example
A	1 ms	Dwell	1000$ = 1 s

The number of all "$" given out must always be 91!

1	$ sign as start sign
90	$ sign for 15 tool numbers with 6 parameter values each

`$80$55$12$99$30$300$200$40$22$99$30$ $..$`	Tool parameters (without line break)

A.5.2 Example for Format 1000

The meaning of the program lines is described in the comments of the example.

`PROGRAM EXAMPLE`	Max. 64 signs without % sign
`CREATED 18.07.89`	Max. 64 signs without % sign
`MODIFIED 22.07.89`	Max. 64 signs without % sign
`%%`	Code: Beginning of program
`X1000Y1000T1`	Tool change to T1, drill hole at X=10.0 mm/ Y10.0 mm
`X3456Y4567`	Drill hole at X=34.56 mm/Y45.67 mm
`...`	
`X12345Y450`	Drill hole at X=123.45 mm/.4.5 mm
`X4000Y4000T2T0`	Tool change and positioning to the start position for routing X=40.0 mm/Y=40.0 mm
`X6000Y4000G1F15`	1st routing track to X=60.0 mm/Y=40.0 mm
`X6000Y7000G1`	2nd routing track to X=60.0 mm/Y=70.0 mm
`X7000Y6000T0`	End of routing and retraction of the cutter
`...`	
`$80$50$20$99$30$300$200$40$22$99$30$ $..$`	Tool parameters (without line break)

The tools parameters are saved at the end of the part program. The first two tool numbers and their parameters are defined as follows:

Tool number	T1	T2
Diameter	80$ = 0.8 mm	200$ = 2.0 mm
Speed	50$ = 50000 rpm	40$ = 40000 rpm
Feed rate	20$ = 2.0 m/min	22$ = 2.2 m/min
Retract feedrate	99$ = 9.9 m/min	99$ = 9.9 m/min
Tool life	30$ = 3000 strokes	30$ = 3000 strokes
Dwell	300$ = 300 ms	$ = 0 ms

The magazine assignment in Format 1000 is fixed:

- T1 is in M1.
- T2 is in M2 etc.

A.6 Format 3000

Format 3000 was developed for the CNC 35.00. The available commands and the corresponding programming rules are described in the programming manual of the CNC 35.00. Format 3000 is compatible to Format 1000.

A.6.1 Definitions for Format 3000

Comment

Any comment can be programmed at the beginning of a file.

- The max. number of characters in a comment line is 64. Every comment line must be finished with a line break.
- How comments are processed depends on the used CNC and on the storage medium.
  - CNC 35.00: When a punched tape is loaded comments are listed on the screen and are not adapted into the program memory.
  - CNC 4x.00 and higher: Comments are ignored when the program is loaded.

`PROGRAM EXAMPLE`	Comment
`CREATED 18.07.89`	Comment
`MODIFIED 20.07.89`	Comment
`%%3000`	Code: beginning of the program
`...`	

Beginning of the Program

The beginning of a program is marked by two percentage signs (%%),

- The four-digit format setting follows the percentage sign. 3000. When the program is loaded another format than the default format is used for converting the format-dependent numerical values.
- The line must be finished with a line break. When the program is loaded all following lines (up to the $ sign) are loaded into the program memory.

`...`	Comment
`%%3000`	Code: beginning of the program
`...`	

Program Line

The program lines are the actual part program.

- The max. length of a program is 64 signs.
- Every program line must be finished with a line break.
- In the manuals a program line is also referred to as "block".
- When the program is loaded all program lines are added to the program memory. This allows editing the program with the CNC editor and displaying it in the graph.
- For detailed information refer to the manual SIEB & MEYER - Command Set SIEB & MEYER.

`%%3000`	Code: beginning of the program

Code	Comment
`...`	
`X..Y..`	
`...`	
`$`	Code: tool parameters

Tool Parameters

Tool parameters are programmed at the end of the part program.

- The section for tool parameters starts with a dollar sign ($).
- Format 3000 allows max. 30 tool parameters at the end of the program.
- Only parameters of the tool numbers programmed in the part program are given out. Tool data sets can consist of several lines, each of which finished with a line break.
  - 1st line = tool data
  - 2nd line and following: magazine list (optional)
- Tool data include the tool number T and the tool parameters (D, S, F, R etc.).
  - If identical code letters with different values are given out for one tool number, the CNC uses the last value.
  - If only the code letter is saved for a parameter, the corresponding value in the tool table is reset to zero, when the program is loaded into the CNC.
  - If no code letter is saved for a parameter, the corresponding value in the tool table remains unchanged, when the program is loaded into the CNC.
- The magazine list includes the magazine numbers that according to the machine.
  - Every magazine number is labeled with M.
  - Ten magazines in the list and the end of the magazine list are followed by a line break.
- A $ sign $ closes the section tools parameters.
- When programs are loaded tool parameters are applied to the tool table (default setting). However, the actual behavior depends on the configuration of the CNC (e.g. the setting of the CNC commands OT, OD, etc.).

Parameter	Unit	Description	Example
T	1 to 30	Tool number	T1 = tool number T1
D	0.01 mm	Tool diameter	D80 = 0.8 mm
S	1000 rpm	Spindle speed	S55 = 55000 rpm
F	0.1 m/min	Feed rate	F12 = 1.2 m/min
R	0.1 m/min	Retract feedrate	R99 = 9.9 m/min
N	100 strokes	Preset tool life	N30 = 3000 strokes
A	1 ms	Dwell	A1000 = 1 s
M	1 to 30	Magazine number	M1M2M3M4 or M1 to M4

If a parameter value is zero only the code letter must be saved.

Code	Comment
`$`	Code: tool parameters
`T1D80S50F20R99N10A1000`	Tool parameter T1
`M1M2M3`	Magazine assignment for T1
`T2D200S40F22R99N30`	Tool parameter T2
`M4M5M6M7M8M9M10M11M12M13`	Line break after every 10 magazines
`M14M15M16`	Remaining magazines for T1

`...`	Other tool parameters
`$`	End of the tool table

A.6.2 Example for Format 3000

The meaning of the program lines is described in the comments of the example.

`PROGRAM EXAMPLE`	Max. 64 signs without % sign
`CREATED 18.07.89`	Max. 64 signs without % sign
`MODIFIED 22.07.89`	Max. 64 signs without % sign
`%%3000`	Code: beginning of program (Format 3000)
`X10.Y10.T1`	Tool change to T1, drill hole at X=10.0 mm/Y=10.0 mm
`X.3Y.45`	Drill hole at X=0.3 mm/Y=0.45 mm
`...`	
`X123.45Y.451`	Drill hole at X=123.45 mm/Y=0.451 mm
`X40.Y40.T1T0`	Tool change and positioning to the start position for routing X=40.0 mm/Y=40.0 mm
`X60.Y40.G1F15`	1st routing track to X=60.0 mm/Y=40.0 mm
`X60.Y70.`	2nd routing track to X=60.0 mm/Y=70.0 mm
`X60.Y70.T0`	End of routing and retraction of the cutter
`$`	Code: tool parameters
`T1D80S50F20R150N10A300`	
`M1M2M3`	
`T2D200S40F22R150N30`	
`M4M5M6M7M8M9M10M11M12M13`	
`M14M15M16`	
`$`	

The tools parameters are saved at the end of the part program. The first two tool numbers and their parameters are defined as follows:

Tool number	T1	T2
Diameter	D80 = 0.8 mm	D200 = 2.0 mm
Speed	S50 = 50000 rpm	S40 = 40000 rpm
Feed rate	F20 = 2.0 m/min	F22 = 2.2 m/min
Retract feedrate	R150 = 15.0 m/min	R150 = 15.0 m/min
Tool life	N30 = 3000 strokes	N30 = 3000 strokes
Dwell	A300 = 300 ms	
Magazine assignment	m..= magazines 1 to 3	M.. = magazines 4 to 16

A.7 Format 5000

Format 5000 was developed for the series CNC 4x.00 and is also used for the series CNC 8x.00. Format 5000 is compatible to Format 1000 and 3000. For information about the available commands and the corresponding programming rules refer to the manual SIEB & MEYER - Command Set.

A.7.1 Definitions for Format 5000

Comment

Any comment can be programmed at the beginning of a file.

- The max. number of characters in a comment line is 64. Every comment line must be finished with a line break.
- Comments are ignored when the program is loaded.

`PROGRAM EXAMPLE`	Comment
`CREATED 18.07.89`	Comment
`MODIFIED 20.07.89`	Comment
`%%5000`	Code: Beginning of program
`...`	

Beginning of program

The beginning of a program is marked by two percentage signs (%%),

- The four-digit format setting follows the percentage sign. 5000. When the program is loaded another format than the default format is used for converting the format-dependent numerical values.
- The line must be finished with a line break. When the program is loaded all following lines (up to the $ sign) are loaded into the program memory.

`...`	Comment
`%%5000`	Code: Beginning of program
`...`	

Program Line

The program lines are the actual part program.

- The max. length of a program is 64 signs.
- Every program line must be finished with a line break.
- In the manuals a program line is also referred to as "block".
- When the program is loaded all program lines are added to the program memory. This allows editing the program with the CNC editor and displaying it in the graph.
- For detailed information refer to the manual SIEB & MEYER - Command Set SIEB & MEYER

`%%5000`	Code: Beginning of program
`...`	
`X..Y..`	

`...`	
`$`	Code: tool parameters

Tool Parameters

Tool parameters are programmed at the end of the part program.

- The section for tool parameters starts with a dollar sign ($).
- Format 5000 allows max. 99 tool parameters at the end of the program.
- Only parameters of the tool numbers programmed in the part program are given out. Tool data sets can consist of several lines, each of which finished with a line break.
  - 1st line = tool data
  - 2nd line and following: magazine list (optional)
- Tool data include the tool number T and the tool parameters (D, S, F, R etc.).
  - If identical code letters with different values are given out for one tool number, the CNC uses the last value.
  - If only the code letter is saved for a parameter, the corresponding value in the tool table is reset to zero, when the program is loaded into the CNC.
  - If no code letter is saved for a parameter, the corresponding value in the tool table remains unchanged, when the program is loaded into the CNC.
- The magazine list includes the magazine numbers that according to the machine.
  - Every magazine number is labeled with M.
  - Ten magazines in the list and the end of the magazine list are followed by a line break.
- A $ sign $ closes the section tools parameters.
- When programs are loaded tool parameters are applied to the tool table (default setting). However, the actual behavior depends on the configuration of the CNC (e.g. the setting of the CNC commands OT, OD, etc.).

Parameter	Unit	Description	Example
T	1 to 99	Tool number	T1 = tool number T1
D	0.01 mm	Tool diameter	D.8 = 0.8 mm
S	1000 rpm	Spindle speed	S55 = 55000 rpm
F	0.1 m/min	Feed rate	F1.2 = 1.2 m/min
R	0.1 m/min	Retract feedrate	R9.9 = 9.9 m/min
N	1 stroke	Preset tool life	N3000 = 3000 strokes
A	1 ms	Dwell	A1000 = 1 s
Z	0.001 mm	Z-offset (addend to absolute working plane = Z-value)	Z.01 = 0.01 mm
M	1 to 30	Magazine number	M1M2M3M4 or M1 to M4

If a parameter value is zero only the code letter must be saved.

`$`	Code: tool parameters
`T1D8.S50.F2.R9.9N3000A1000`	Tool parameter T1
`M1M2M3`	Magazine assignment for T1
`T2D2.S40.F2.2R9.9N3000`	Tool parameter T2
`M4M5M6M7M8M9M10M11M12M13`	Line break after every 10 magazines
`M14M15M16`	Remaining magazines for T1
`...`	Other tool parameters

`$`	End of the tool table

A.7.2 Example for Format 5000

The meaning of the program lines is described in the comments of the example.

`PROGRAM EXAMPLE`	Max. 64 signs without % sign
`CREATED 18.07.89`	Max. 64 signs without % sign
`MODIFIED 22.07.89`	Max. 64 signs without % sign
`%%3000`	Code: beginning of program (Format 3000)
`X10.Y10.T1`	Tool change to T1, drill hole at X=10.0 mm/Y=10.0 mm
`X.3Y.45`	Drill hole at X=0.3 mm/Y=0.45 mm
`...`	
`X123.45Y.451`	Drill hole at X=123.45 mm/Y=0.451 mm
`X40.Y40.T1T0`	Tool change and positioning to the start position for routing X=40.0 mm/Y=40.0 mm
`X60.Y40.G1F15`	1st routing track to X=60.0 mm/Y=40.0 mm
`X60.Y70.`	2nd routing track to X=60.0 mm/Y=70.0 mm
`X60.Y70.T0`	End of routing and retraction of the cutter
`$`	Code: tool parameters
`T1D.8S50.F2.R25.N3000A.3`	
`M1M2M3`	
`T2D2.S40.F2.2R25.N3000Z.3A.3`	
`M4M5M6M7M8M9M10M11M12M13`	
`M14M15M16`	
`$`	

The tools parameters are saved at the end of the part program. The first two tool numbers and their parameters are defined as follows:

Tool number	T1	T2
Diameter	D.8 = 0.8 mm	D2 = 2.0 mm
Speed	S50 = 50000 rpm	S40 = 40000 rpm
Feed rate	F2. = 2.0 m/min	F2.2 = 2.2 m/min
Retract feedrate	R25. = 15.0 m/min	R25. = 15.0 m/min
Tool life	N3000 = 3000 strokes	N3000 = 3000 strokes
	Z = 0.0 mm	Z.3 = 0.3 mm
Dwell	A.3 = 300 ms	A.3 = 300 ms
Magazine assignment	M.. = magazines 1 to 3	M.. = magazines 4 to 16

B SIEB & MEYER Formats

The table compares the SIEB & MEYER formats.

- The second table shows the assignment of the formats to the SIEB & MEYER CNCs. *Note: All CNCs are downwards compatible!*
  - CNC 25.05: format 1000
  - CNC 35.00: formats 1000 and 3000
  - CNC 4x.00: formats 1000, 3000 and 5000 (except for spot facing commands)
  - CNC 8x.00: formats 1000, 3000 and 5000.

Note

The actually available functions always depend on the CNC and the machine equipment used. For detailed information refer to the documentation of the machine manufacturer.

Note

The units of numerical values in the table only apply for the Format 1000, 3000 and 5000! Other units are special cases and not considered in this manual (Formats *xx*01 to *xx*10).

Command	Explanation	Format 1000	Format 3000	Format 5000
D*d*	Round edge	–	● 0.001 mm	● 0.001 mm
F*f*	Routing feed rate	● F1 = 0.1 m/min No decimal point allowed!	● F1 = 0.1 m/min No decimal point allowed!	● F1 = 0.001 m/min Decimal point is allowed!
G1	Rout straight line	● G1 must be programmed in every block.	●	●
G2 **G3**	Rout circular arc	● G2 and G3 must be programmed in every block! Only for every quadrant. Radius definition only with I and J.	● Radius can be defined with R or I/J.	● Radius can be defined with R or I/J.
G6	Rout ramp	–	–	●
G11	Finish-routing function	●	●	●
G30	Deactivate corrective function (camera)	–	–	●
G31	Activate corrective function (camera)	–	–	●
G32	Carry out measurement (camera)	–	–	●
G33	Carry out measurement (camera)	–	–	●
G34	Deactivate corrective function and clear correction values (camera)	–	–	●
G35	Send byte to camera system (camera)	–	–	●
G36	Measure offset (camera)	–	–	●
G37	Reserved	–	–	–
G38	Reserved	–	–	–
G39	Organize correction values (camera)	–	–	●

Command	Explanation	Format 1000	Format 3000	Format 5000
G40	Deactivate cutter radius compensation	–	●	●
G41	Activate cutter radius compensation	●	●	●
G42				
G43	Cut routing contour completely Only in combination with the commands G45, G46, G49 and G50.	–	–	●
G45	Rout out full circle counterclockwise	●	●	●
G46	Rout out full circle clockwise	●	●	●
G47	Rout out disk counterclockwise	–	●	●
G48	Rout out disk clockwise	–	●	●
G49	Rout out rectangle counterclockwise	●	●	●
G50	Rout out rectangle clockwise	●	●	●
G70	Deactivate SLM	–	–	●
G71	Activate SLM	–	–	●
G72	Clear surface memory	–	–	●
G73	Detect/Save reference surface	–	–	●
G74	Detect/Save reference surface	–	–	●
G75	Retract measuring probe	–	–	●
G76	Measure depth/residual web and record deviation	–	–	●
G77	Deactivate surface detection for one block	–	–	●
G78	Detect reference value and clear measured grid values	–	–	●
G79	Run measuring cycle and save measuring deviation	–	–	●
G80	Deactivate peck drilling	–	–	●
G81	Activate peck drilling I*i* = lowering value P*p* = %-factor for the plunge rate J*j* = height of a partial strokes W*w* = %-factor for partial strokes	–	–	●
G82	Reference plane = table surface Z*z* = absolute working plane	–	●	●
G83	Reference plane = board surface see K*k*	–	● The machine must be equipped with a surface detection unit.	● The machine must be equipped with a surface detection unit.
G84	Nibble circle	–	●	●
G85	Nibble slot	–	●	●
G88	Activate pulse drilling	–	–	●
G89	Deactivate pulse drilling	–	–	●
G90	XY-coordinates are interpreted as absolute values	●	●	●
G91	XY-coordinates are interpreted as incremental values	● G91 must be programmed in every block.	●	●
H*h*	Absolute traveling plane Reference plane = table surface	–	● H1 = 0.01 mm	● H1 = 0.001 mm

Command	Explanation	Format 1000	Format 3000	Format 5000
I*i*	Routing parameters: ▸ G2/G3/G45 to G48: X-component of the radius ▸ G49/G50: X side length	● 0.01 mm Circular arc: quadrant passage not allowed!	● I1 = 0.001 mm	● I1 = 0.001 mm
I*i*	Peck drill parameter (G81): lowering value	–	–	●
J*j*	Routing parameters: ▸ G2/G3/G45 to G48: X-component of the radius ▸ G49/G50: X side length	● 0.01 mm Quadrant passage not allowed!	● J1 = 0.001 mm	● J1 = 0.001 mm
J*j*	Peck drilling parameter (G81): Start height of a partial stroke (see peck drill parameter W)	–	–	●
K*k*	Relative working plane Reference plane = board surface	–	● K1 = 0.01 mm	● K1 = 0.001 mm
L	Space in surface memory	–	–	●
M20	Optional stop	●	●	●
M21	Auxiliary function	●	●	●
M22 to M27	Machine-specific help functions The working mode of M-functions is set by the machine manufacturer (refer to the relevant documentation).	–	–	●
M34 to M45		–	–	●
M52 to M55		–	–	●
M28	Move XY-axes to calibration point	–	●	●
M29	XY-axes to park position	–	●	●
M30	End of a program section (right Step-and-Repeat bracket)	●	●	●
M31	Beginning of a program section (left Step-and-Repeat bracket)	–	●	●
M47	User message	–	●	●
M49	Execute CNC command	–	– M49 is interpreted as M47.	●
M50	Single offset (no rotation/mirroring)	●	●	●
M56	Define check area in X-direction (condition: multiple-area function is on)	–	–	●
M57	Define check area in Y-direction (condition: multiple-area function is on)	–	–	●
M58	Define Forbidden Areas	–	–	●
M60	Offset with rotation by 90°	●	●	●
M70	Offset with mirroring around the Y-axis	●	●	●
M75	Send string to camera system	–	–	●
M76	Transmit string to M76 device (see documentation of the machine manufacturer)	–	–	●
M80	Offset with mirroring around the X-axis	●	●	●
M90	Offset with rotation by 180°	●	●	●
M92	Deactivate scaling function	–	–	●
M93	Activate scaling function	–	–	●
M94	Deactivate broken tool monitoring	–	–	●
M95	Activate broken tool monitoring	–	–	●
M97	Drill plain text in the X-direction	–	●	●
M98	Drill plain text in the Y-direction	–	●	●

Command	Explanation	Format 1000	Format 3000	Format 5000
M99	Call subprogram	–	–	●
P	Peck drill parameter (G81): reduce feed rate for peck drilling	–	–	●
R	Routing parameters: ▸ G2/G3: radius of a circular arc ▸ G49/G50: side length of a square	–	● R1 = 0.001 mm	● R1 = 0.001 mm
T*t*	Tool change	●	●	●
V1	Drill dual row of holes (dual-in-line)	●	●	●
V2	Drill single row of holes	●	●	●
V2	Create pattern row	–	●	●
V3	Drill quadruple row of holes	–	–	●
V4	Drill circular row of holes	–	–	●
W*w*	Number of repetitions in combination with the commands V1, V2, V3 and V4.	● Only V1 and V2	●	●
W*w*	Peck drill parameter (G81): factor for reducing the height of the partial strokes of a peck drill hole	–	–	●
X*x*	Drilling, routing and offset coordinates in X-direction	● X1 = 0.01 mm	● X1 = 0.001 mm	● X1 = 0.001 mm
Y*y*	Drilling, routing and offset coordinates in Y-direction	● Y1 = 0.01 mm	● Y1 = 0.001 mm	● Y1 = 0.001 mm
Z*z*	Absolute working plane Reference plane = table surface	–	● Z1 = 0.01 mm	● Z1 = 0.001 mm
@	Define a subprogram Only allowed in subprogram files!	–	–	●
(	Insert comment	–	●	●
/	Label program line If BLKD is active, this line is ignored.	–	●	●

C Glossary

A

Averaging area

In surface detection the averaging area is the storage, in which measured surfaces are saved for the calculation of the average value.

C

Cutter radius compensation

Cutter radius compensation means that the programmed contour is corrected by the radius of the cutter. The compensated (corrected) routing path runs in parallel to the programmed contour. The distance corresponds to the radius of the cutter.

D

Depth control

Depth control stands for depth-controlled drilling/routing(see *depth-controlled drilling/routing*).

This term is always used when the working plane refers to the surface of the PCB.

This function requires a machine being equipped with a measuring device for surface detection.

K

K-value

Distance between PCB surface K_0 and (relative) working plane K (plunge depth).

N

Nested Step-and-Repeat

Step-and-Repeat used within a Step-and-Repeat section is referred to as nested Step-and-Repeat. This way patterns and their repetitions are repeated as a whole. The number of original patterns multiplies with every repetition.

Nibbling

Nibbling is a production method for producing a slot or hole by means of a drill bit (instead of a routing tool).

- The result is reached by drill holes arranged close to each other.
- Nibbling is for example used to execute short routing tracks on a drilling machine without routing functions.
- Two nibbling processes are distinguished.

Alternating nibbling

When alternating nibbling is used, the outer holes are drilled first.

- The following drilling runs produce drill holes between already drilled holes.
- This results in neutral material crowding for every drill stroke. hole order: 1 6 4 7 3 8 5 9 2.

Sequential nibbling

Sequential nibbling corresponds to a row of holes with a hole center distance of 0.381 mm.

- hole order: 1 2 3 4 5 6 7...

P

Part program

A program describing the operational procedures of the CNC control system. During the program run a sequence of orders is executed that were defined for the production (drilling, routing).

Peck drilling

Peck drilling divides a drill stroke into several single strokes.

- The single strokes penetrate deeper and deeper into the material.
- For thin drill bits this has the effect that the chip brakes and ensures that the tool does not brake.

Plain text

Drilled text.

- Appropriate programming allows aligning the text in parallel to the X-axis or Y-axis.
- Depending on the configuration the text is drilled in correct position or legibly.

Pulse drilling

Pulse drilling divides a drill stroke into several single strokes.

- The single strokes penetrate deeper and deeper into the material.
- Contrary to peck drilling the tool is **not** retracted after every single drill stroke.

R

Run

A program run describes the phase from starting a part program up to the end of the program.

S

SLM probe

The SLM probe measures the depths while the Z-axis is lowered. The SLM probe is mounted in parallel to the Z-axis and as close as possible to the drilling/routing spindle and requires individual calibration. The probe is equipped with an own measuring system evaluated for surface detection.

Surface memory

In surface detection the averaging area is the storage for saving average values. Depending on the kind of measurement the values are saved as measured LH values or measured grid values in the surface memory. The values are saved in the surface memory via an index (L or H-index) and loaded via this index for drilling/routing. In this case the values are referred to as "measured LH values". If the values have been measured in the grid, the values are saved via the XY-coordinates of the corresponding measurement in the surface memory and loaded for drilling/routing. In this case the values are referred to as "measured grid values".

13 Index