LNC-Lathe Series

Programming Manual

2020/06 Ver：V04.00.004(LN4408110038)

Enable intelligent machines

寶元數控股份有限公司

LNC Technology Co., Ltd.

Table of Content

1 G-Code Function Table

Function	Group	**TYPE A**	**TYPE B**	**TYPE C**
Positioning in rapid	01	G00	G00	G00
Linear interpolation	01	G01	G01	G01
Circular interpolation (cw.)	01	G02	G02	G02
Circular interpolation (ccw.)	01	G03	G03	G03
Dwell	00	G04	G04	G04
Exact stop	00	G09	G09	G09
Data setting	00	G10	G10	G10
ARC Plane Setting	02	G17~19	G17~19	G17~19
input in inch	06	G20	G20	G70
input in mm	06	G21	G21	G71
Stored stroke check ON	09	G22	G22	G22
Stored stroke check OFF	09	G23	G23	G23
Reference position return check	00	G27	G27	G27
1st reference position return	00	G28，G29	G28，G29	G28，G29
2nd,3rd,4th, reference position return	00	G30	G30	G30
Skip function	00	G31	G31	G31
Thread cutting	01	G32	G33	G33
Variable Thread Pith Threading	01	G34	G34	G34
Tool nose radius compensation Cancel	07	G40	G40	G40
Tool nose radius compensation Left	07	G41	G41	G41
Tool nose radius compensation Right	07	G42	G42	G42
Machine coordinate system	00	G53	G53	G53
Selection of work coordinate system	14	G54 ~G59	G54 ~ G59	G54 ~ G59
Exact stop mode	15	G61	G61	G61
Cutting mode	15	G64	G64	G64
User macro simple call	00	G65	G65	G65
User macro modal call	12	G66	G66	G66
User macro modal call cancel	12	G67	G67	G67
Mirror image for double turrets ON	16	G68	G68	G68
Mirror image for double turrets OFF	16	G69	G69	G69

Function	Group	**TYPE A**	**TYPE B**	**TYPE C**
Finishing Cycle	00	G70	G70	G72
Stock removal in turning	00	G71	G71	G73
Stock removal in facing	00	G72	G72	G74
Pattern repeating	00	G73	G73	G75
End face peck drilling cycle (Z axis)	00	G74	G74	G76
Outer diameter / Internal diameter drilling cycle (X axis)	00	G75	G75	G77
Multiple thread cutting cycle	00	G76	G76	G78
Outer diameter/internal diameter cutting cycle	01	G90	G77	G20
Taper thread cutting cycle	01	G92	G78	G21
End face turning cycle	01	G94	G79	G24
Canned cycle for drilling cancel	10	G80	G80	G80
Face drilling cycle	10	G83	G83	G83
Face tapping cycle	10	G84	G84	G84
Face boring cycle	10	G85	G85	G85
Side drilling cycle (X axis)	10	G87	G87	G87
Side tapping cycle (X axis)	10	G88	G88	G88
Side boring cycle (X axis)	10	G89	G89	G89
Absolute programming	03	-	G90	G90
Incremental programming	03	-	G91	G91
Coordinate system setting or max. spindle speed setting	00	G50	G92	G92
Feed per minute (mm/min)	05	G98	G94	G94
Feed per revolution (mm/rev)	05	G99	G95	G95
Constant surface speed control ON	02	G96	G96	G96
Constant surface speed control OFF	02	G97	G97	G97
Intial point return	11	-	G98	G98
R point return	11	-	G99	G99
Side drilling cycle (Y axis)	10	187	187	187
Side tapping cycle (Y axis)	10	188	188	188
Side boring cycle (Y axis)	10	189	189	189

（Note）The TYPE　is A, B or C to decide on the Pr153, default is TYPE B.

2 General M-Code Function Table

M Code	Function	Remark
M00	Program stop	CNC
M01	Optional stop	CNC
M02	End of program	CNC
M03	Spindle CW	
M04	Spindle CCW	
M05	Spindle stop	
Txx	Auto tool change	xx：Tool no.
M08	Coolant ON	
M09	Coolant OFF	
M10	Chuck clamp	
M11	Chuck unclamp	
M30	Program rewind	CNC
M98	Calling of subprogram	CNC
M99	End of subprogram	CNC

3 Syntax of G code

G00: Positioning in rapid

Format:

G00 X___ Y___ Z___ C___;

Argument:

X__ Y__ Z__ : For G90, the coordinate of an end point in absolute command.

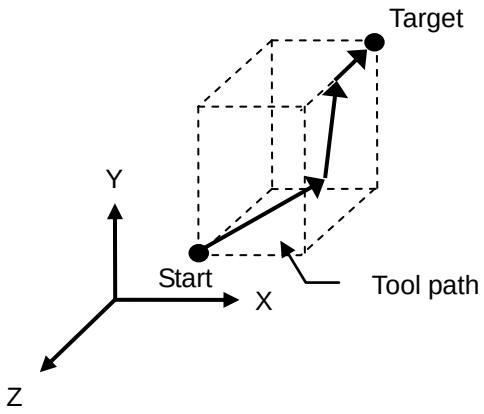
C__ For G91, the coordinate of an end point in Incremental command.

Action:

The function of G00 command is to make the tool move to the position of the specified coordinate rapidly.

When using G00, the speed of moving is not descided by the format of F__, but by setting values of parameter 1000 ~ 1003. Meanwhile, the rapid traverse adjustment knob can be used to adjust the percentage of speed. (F0, 25%, 50%, 100%)

Illustration:



Note:

Regarding G00 movement command, the movement of each servo axis is independent. The movement speed of each axis is specified by parameters respectively. Operators should be especially careful lest the tool may collide with the workpiece.

Methods of determing G00 simultaneously interpolated feed rate

	G00 command or commands with same function under MEM, MDI modes	G00, G53 command of PMC axis function
None dry run mechanism	Moving speed of each axis does not exceed respectively set G00 speed (Remark 1)	Moving speed of each axis does not exceed respectively set G00 speed
Dry run mechanism paremeter 0083 is 0	Moving speed of each axis does not exceed respectively set JOG speed (Remark 2)	C23 is OFF: Moving speed of each axis does not exceed respectively set JOG speed C23 is ON: Moving speed of each axis does not exceed respectively set G00 speed
Dry run mechanism paremeter 0083 is 1	Moving speed of each axis does not exceed respectively set G00 speed	Moving speed of each axis does not exceed respectively set G00 speed

Remark 1 Under this condition, Override depends on rapid traverse percentage.
Remark 2 Under this condition, Override depends on cutting feed percentage.

G01: Linear interpolation

Format:

G01 X(U)___ Z(W)___ F___;

Argument:

X___, Z___ : For G90, the coordinate of an end point in absolute command.
For G91, the coordinate of an end point in Incremental command.

U___, W___ : For G90/G91, the coordinate of an end point in Incremental command.

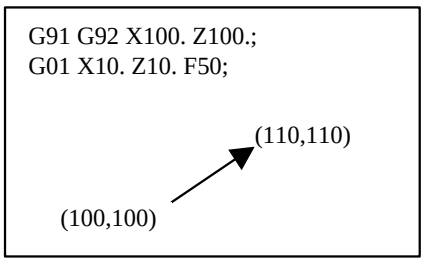
F___ : Feedrate.

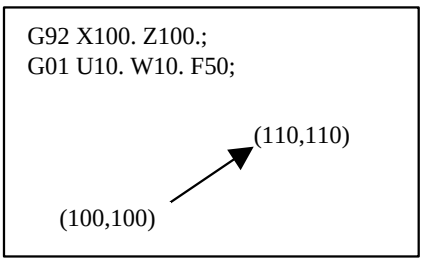
Action:

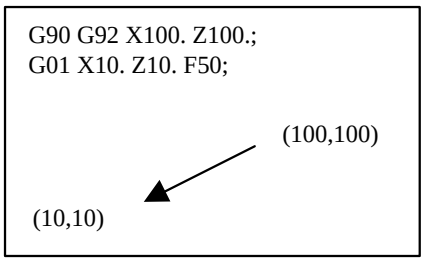
G01 depends on the interpolation feed rate specified by F code, starts from current tool position, to cut in a linear path to the end. Axes which are not specified do not move. Actual cutting feed can be adjusted by the continuous feed rate adjustment knob at any time (0%-150%).

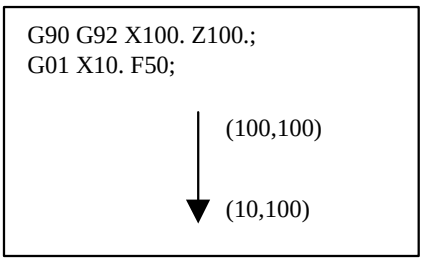
Max interpolation feed rate of G01 is specified by system parameter 1004. Acc/Dec time of G01 is specified by system parameter 0014.

Illustration:









G02, G03: Circular interpolation (cw./ccw.)

Format:

$$\begin{bmatrix} G02 \\ G03 \end{bmatrix} X(U)\_\_ \; Z(W)\_\_ \begin{bmatrix} I\_\_K\_\_ \\ R\_\_ \end{bmatrix} F\_\_;$$

Argument:

X___, Z___	:	For G90, the coordinate of an end point in absolute command. For G91, the coordinate of an end point in Incremental command.
U___, W___	:	For G90/G91, the coordinate of an end point in Incremental command.
R___	:	Arc radius. (R>0, Arc<=180°. R<0. Arc>180°)
I___	:	Xp axis distance from start point to the center of an arc. Xp is a component in X direction.
K___	:	Zp axis distance from start point to the center of an arc. Zp is a component in Z direction.
F___	:	Feedrate.

Action:

Pay attention to current tool position. The end point and the center of circle should be in the same circle. If not, the controller emits an error signal INT 132. When R___, I___, and K___ of the program are all written-in, system will depends only on the setting of R___.

System parameter 132 (XRC) defines if X (U) position is radius-specified or diameter-specified. Radius-specified (XRC=1) and diameter-specified (XRC=0) differ in 2 times of the actual movement amount of X axis. E.g. The movement amount of radius-specified U-10 is equal to diameter-specified U-20.

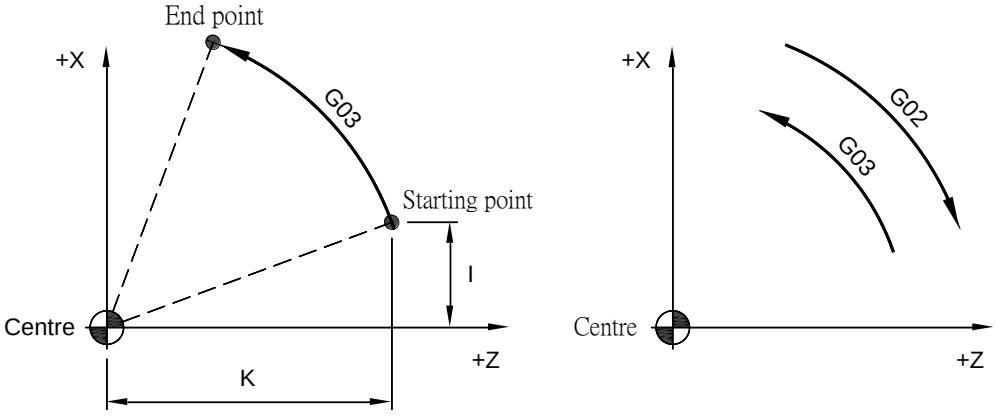
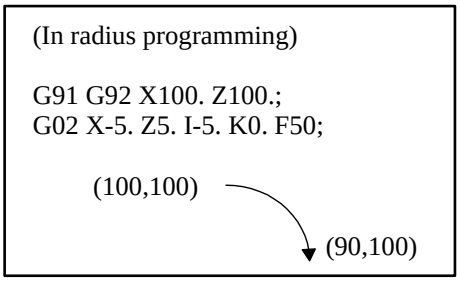
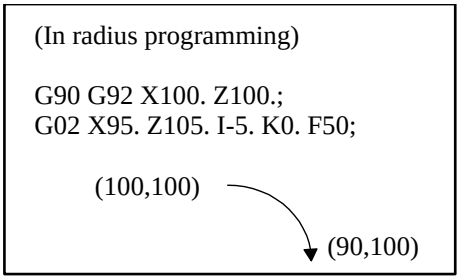
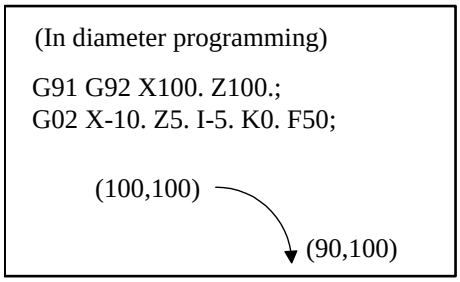


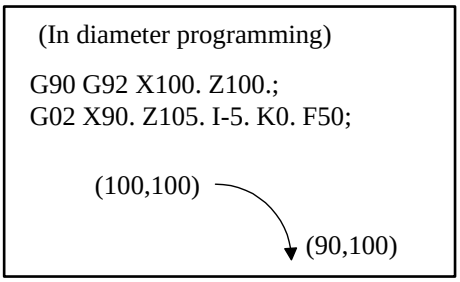
Illustration:

The following 4 figures have the same cutting path of a clockwise cutting of a 1/4 circle, and the radius is 5.









Direct pattern making

In order to make the manufacturing of workpieces easier, the controller provides functions of making corner chamfering (**,C**_), corner rounding (**,R**_) and angle of straight line (**A**_).

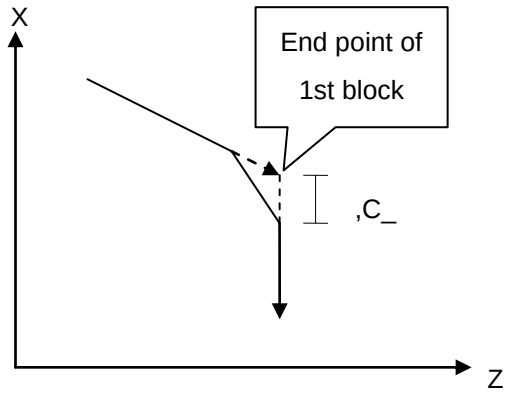
1. Chamfering ",C_"

Continuous 2 blocks of command. In the 1st block, **C**_ sets up chamfer length between the 2 blocks. It is applicable also when the previous and next blocks are circular commands.

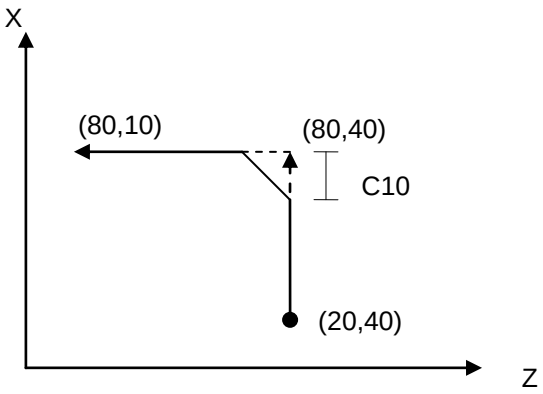
Format:

$$\begin{bmatrix} G01 \\ G02 \\ G03 \end{bmatrix} X\_\_\ Z\_\_\ , C\_\_;$$

$$\begin{bmatrix} G01 \\ G02 \\ G03 \end{bmatrix} X\_\_\ Z\_\_;$$



Example:



G00 X20.0 Z40.0

G01 X80.0,**C10.0** F100.0

G01 Z10.0

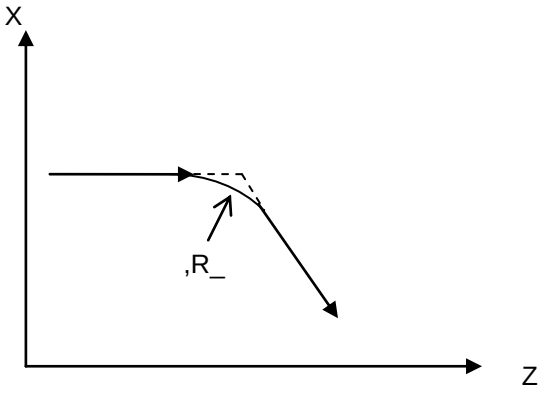
2. Corner Rounding ",R_"

Continuous 2 blocks of command. In the 1st block,,**R**_ sets up the radius of the rounding corner linking the 2 blocks. It is applicable also when the previous and next blocks are circular commands.

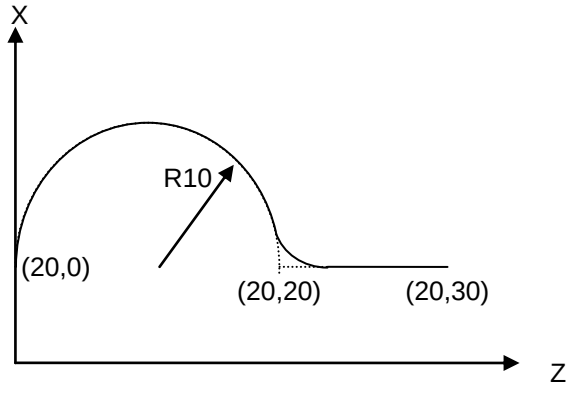
Format:

$$\begin{bmatrix} G01 \\ G02 \\ G03 \end{bmatrix} X\_\_\ Z\_\_\ ,\ R\_\_;$$

$$\begin{bmatrix} G01 \\ G02 \\ G03 \end{bmatrix} X\_\_\ Z\_\_;$$



Example:



G00 X20.0 Z0.0

G02 X20.0 R10.0,**R3.0** F100.0

G01 Z30.0

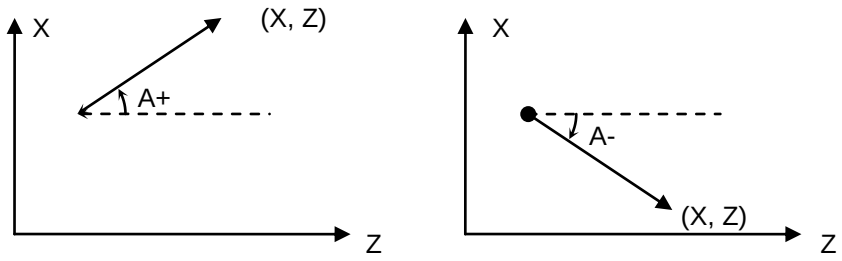
3. Angle of Straight Line ”A_”

When applying linear interpolation command G01, only the positions of X_ or Z_ and the angle of the line **A**_ can be specified. The actual position of the corresponding end point will be calculated by the controller. It is especially convenient when the manufacturing drawing provides only coordinates of X or Z directions and angles.

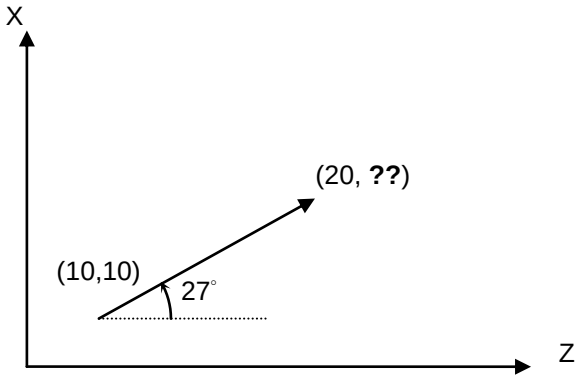
Format:

$$G01\begin{bmatrix} X\_\_ \\ Z\_\_ \end{bmatrix} A\_\_;$$

Wherein A_ angle is calculated from the horizontal direction (Z axis). A positive value of A represents a counterclockwise direction, and vice versa.



Example:



G00 X10.0 Z10.0

G01 X20.0 **A27.0** F100.0

4. Geometric input function

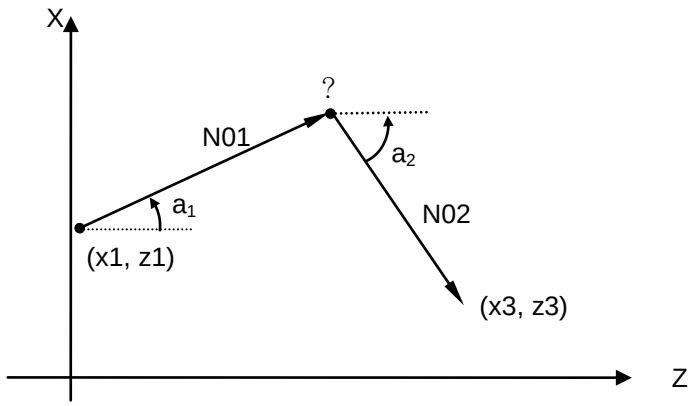
In a manufacturing drawing, it is often that angles are provided, but not correct cutting positions; or that only the size of corner or round corner is provided when making a corner. Therefore the calculation of the coordinates of the intermediate point can be inconvenient when transforming the data to linear and circular cutting command while compiling a manufacturing program. This function makes compiling programs more easily, and help avoid calculation errors.

- **Type 1**

  Intersection point of 1st and 2nd blocks is unknown
  Angles of 1st and 2nd blocks are known
  End point coordinates are known

Format:

N01 G01 A$\underline{a_1}$
N02 G01 X$\underline{x_3}$ Z$\underline{z_3}$ A$\underline{a_2}$

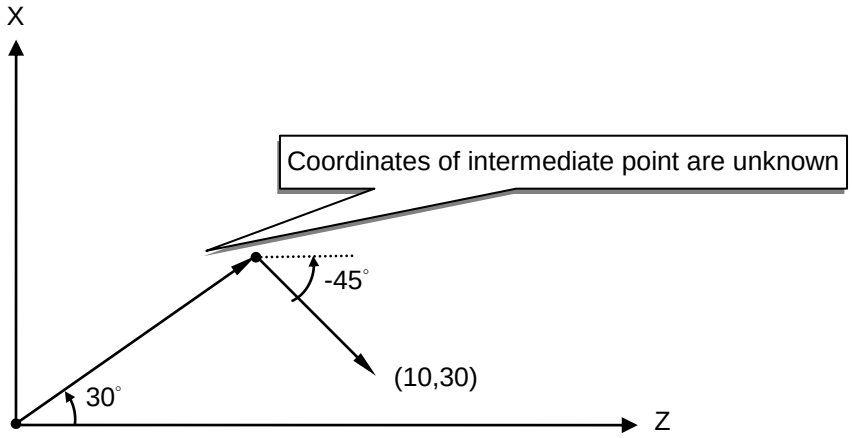


Example:

G00 X0.0 Z0.0

G01 A30.0

G01 X10.0 Z30.0 A-45.0



- **Type 2**

Intersection point of 1st and 2nd blocks is unknown

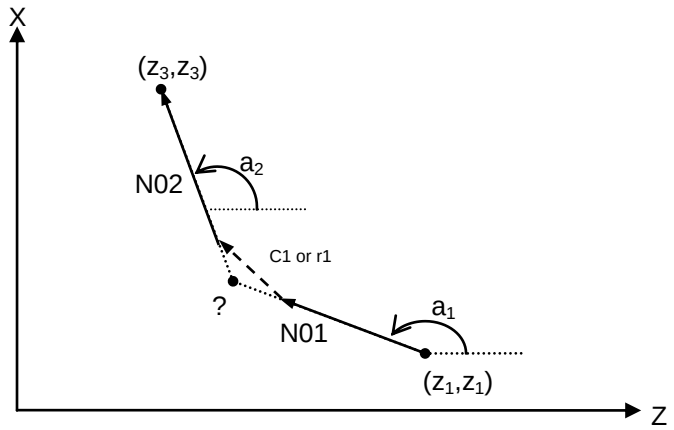
Angles of 1st and 2nd blocks are known

End point coordinates are known

Corner of 1st and 2nd blocks is chamfer or fillet

N01 G01 Aa1,Cc1 (,Rr1)
N02 G01 Xx3 Zz3 Aa2

Format:



- **Type 3**

  Either X_ or Z_ of the intersection point of 1st and 2nd blocks is known

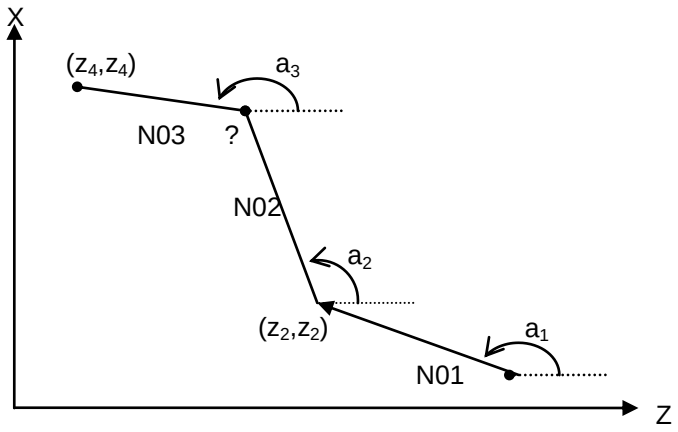
  Intersection point of 2nd and 3rd blocks is unknown

  Coordinates of end point of 3rd block are known

  Angles of 1st, 2nd and 3rd blocks are known

N01 G01 Xx2(Zz2) Aa1
N02 G01 Aa2
N03 G01 Xx4 Zz4 Aa3

Format:



- **Tyep 4**

  Intersection point of 1st and 2nd blocks is known

  Intersection point of 2nd and 3rd blocks is unknown

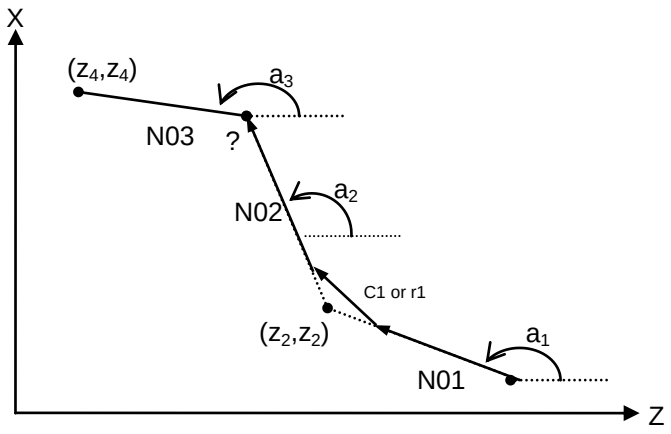
  Coordinates of end point of 3rd block is known

  Angles of 1st, 2nd and 3rd blocks are known

  Corner of 1st and 2nd blocks is chamfer or fillet

N01 G01 X$\underline{x_2}$(Z$\underline{z_2}$) A$\underline{a_1}$
N02 G01 A$\underline{a_2}$
N03 G01 X$\underline{x_4}$ Z$\underline{z_4}$ A$\underline{a_3}$

Format:



- **Type 5**

  Intersection point of 1st and 2nd blocks is unknown

  Intersection point of 2nd and 3rd blocks is known

  Coordinates of end point of 3rd block are known

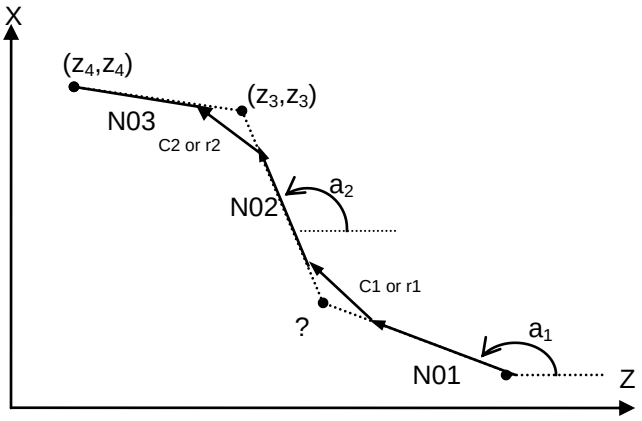
  Angles of 1st and 2nd blocks are known

  Corner of 1st and 2nd blocks is chamfer or fillet

  Corner of 2nd and 3rd blocks is chamfer or fillet

N01 G01 Xx_2 Zz_2 ,Cc_1 (,Rc_1)
N02 G01 Aa_2
N03 G01 Xx_3 Zz_3 Aa_1

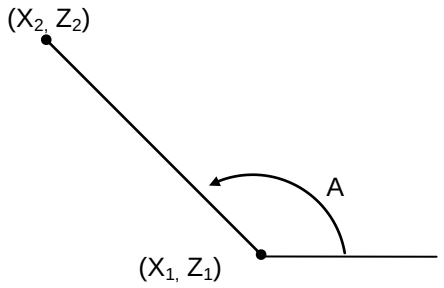
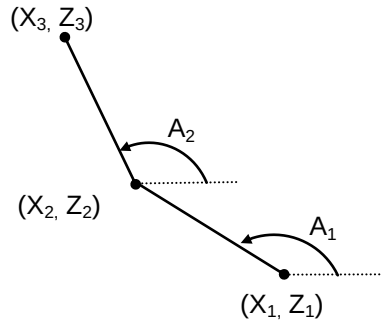
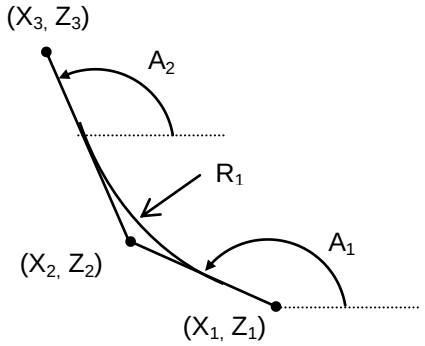
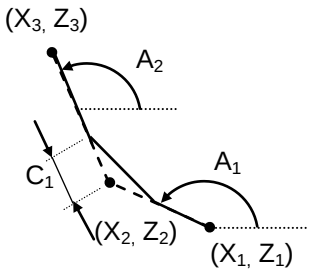
Format:

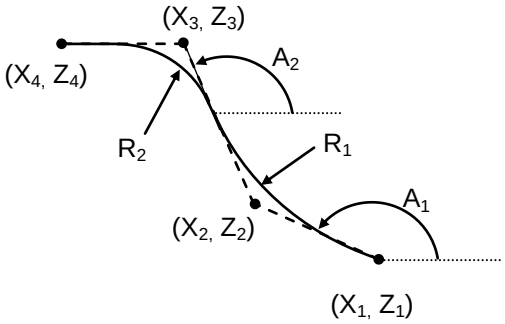
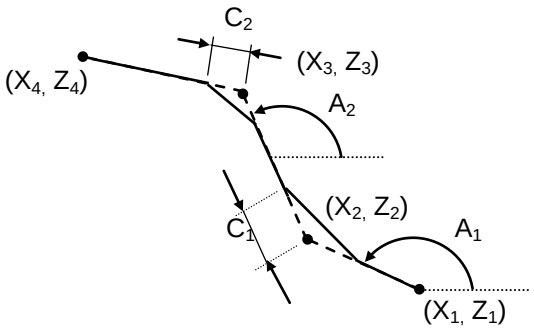
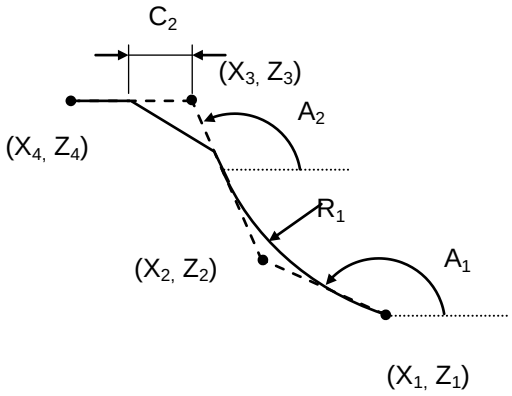
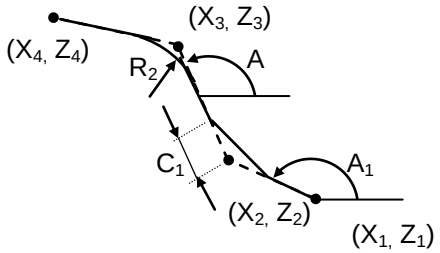


Note:

1. The following G codes can not be in the same block with geometric input commands, or be used to input pattern size of continuous shapes.
   a. G codes of Group 00 (excluding G04)
   b. G02, G03, G90, G92, G94 of Group 01
2. Only be effective under MEM Mode
3. Fillet command can not be used in thread-cutting blocks.
4. When applying G01 X_ A_, if angle value is 0∘±1, 180∘±1, then the command is ineffective.
5. When applying G01 Z_ A_, if angle value is 90∘±1, 270∘±1, then the command is ineffective.
6. If the angle between 2 lines is within +1∘, chamfering and filleting will be ignored.

Table of usages of geometric commands

	Command	Illustration
1	X_2_(Z_2_) A_	
2	A_1_ X_3_ Z_3_ A_2_	
3	X_2_ Z_2_,R_1_ X_3_ Z_3_ or A_1_,R_1_ X_3_ Z_3_ A_2_	
4	X_2_ Z_2_,C_1_ X_3_ Z_3_ or A_1_,C_1_ X_3_ Z_3_ A_2_	

5	X_2_ Z_2_,R_1_ X_3_ Z_3_,R_2_ X_4_ Z_4_ or A_1_,R_1_ X_3_ Z_3_ A_2_,R_2_ X_4_ Z_4_	
6	X_2_ Z_2_,C_1_ X_3_ Z_3_,C_2_ X_4_ Z_4_ or A_1_,C_1_ X_3_ Z_3_ A_2_,C_2_ X_4_ Z_4_	
7	X_2_ Z_2_,R_1_ X_3_ Z_3_,C_2_ X_4_ Z_4_ or A_1_,R_1_ X_3_ Z_3_ A_2_,C_2_ X_4_ Z_4_	
8	X_2_ Z_2_,C_1_ X_3_ Z_3_,R_2_ X_4_ Z_4_ or A_1_,C_1_ X_3_ Z_3_ A_2_,R_2_ X_4_ Z_4_	

G04: Dwell

Format:

```
G04 X___;
G04 P___;
```

Argument:

X___ : Specify a time. Unit: sec. Range: 0.001 ~ 99999.999.

P___ : Specify a time. Unit: ms. No decimal poiint. Range: 1 ~ 99999999.

Action:

Dwell action; set up dwell time after G04; when the time is over, next block will be executed automatically.

Example:

G04 X100.; ---------- 100 sec

G04 P100; ---------- 0.1 sec

G04; ---------- Exact stop (G09)

G09: Exact stop

Format:

```
G09 [G01 __]
    [G02 __] ;
    [G03 __]
```

Argument:

G09 is a command used along with the exact stop of cutting. When using G09, system checks positioning degrees after executing every positioning command. After making sure statuses of positioning comply with settings, system continues executing next block. Therefore, if cutting positioning exists between blocks, there might be a little interruption due to the demanding of the precision of positioning point. Speed is sacrificed for a higher shape precision. The degree of precision is specified by parameters 0006 ~ 0009. The function of G09 only takes effect within its block.

Example:

G91 G09 G01 Y100. F200.; ---------- (1)

G01 X100.; ---------- (2)

Illustration:

G10: Data setting

Format 1:

G10 P 1~30 X_ (U_) Z_ (W_) R_ (C_) Q_;

Format 2:

G10 P 101~130 X_ (U_) Z_ (W_) R_ (C_) Q_;

Format 3:

G10 P 153~159 X__ Z__

Format 4:

G10 L20 P 1~300 X__ Z__

Argument of format 1:

P___	:	No. of compensation. P1~30 are the values of no. 1~30 tool wear compensation.
X___	:	The value of tool wear compensation of X axis. (Absolute)
Z___	:	The value of tool wear compensation of Z axis. (Absolute)
U___	:	The value of tool wear compensation of X axis. (Increment)
W___	:	The value of tool wear compensation of Z axis. (Increment)
Q___	:	Tool type. Types are shown below figure.
R___	:	The value of tool nose wear compensation. (Absolute)
C___	:	The value of tool nose wear compensation. (Increment)

Argument of format 2:

P___	:	No. of compensation. P101~130 are the values of no. 1~30 tool length compensation.
X___	:	The value of tool length compensation of X axis. (Absolute)
Z___	:	The value of tool length compensation of Z axis. (Absolute)
U___	:	The value of tool length compensation of X axis. (Increment)
W___	:	The value of tool length compensation of Z axis. (Increment)
Q___	:	Tool type. Types are shown below figure.
R___	:	The value of tool nose wear compensation. (Absolute)
C___	:	The value of tool nose wear compensation. (Increment)

Argument of format 3:

P___ : No. of compensation.

Coordinate, range 153 ~ 159, relative to 00 and G54 ~ G59.

X___ : The value of X axis coordinate.

Z___ : The value of Z axis coordinate.

Argument of format 4:

L20 : Extension coordinate mode.

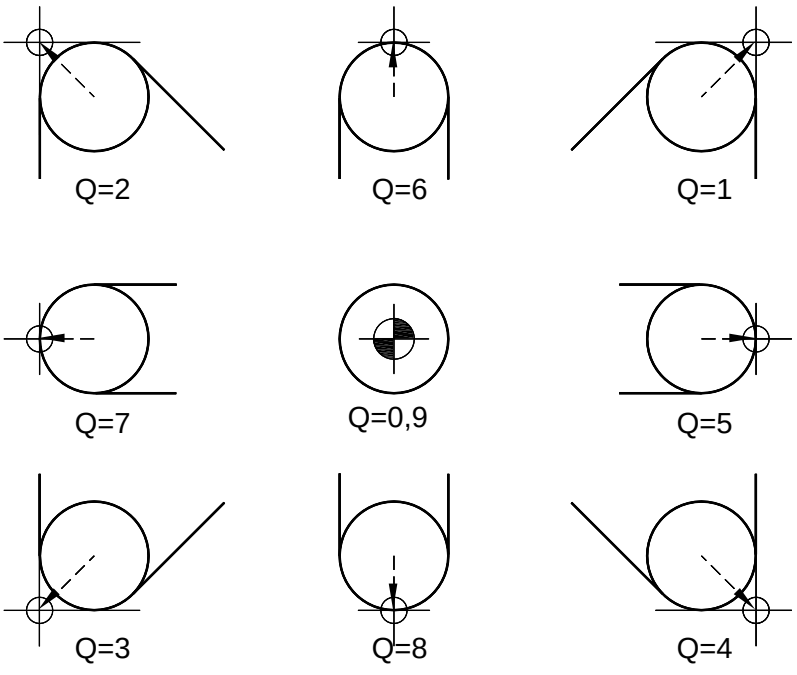
P___ : No. of compensation.

Extension coordinate, range 1 ~ 300, relative to G54 P1~P300.

X___ : The value of X axis coordinate.

Z___ : The value of Z axis coordinate.

Illustration:



Tool type(Q__)

G20, G21: Inch \ mm input

Format:

```
G20;
G21;
```

Argument:

G20 : Inch input, minimum 0.0001inch.

G21 : mm input, minimum 0.001mm.

These commands should be used alone in their respective block without any other command, and they have to be at the beginning of programs, i.e. before the setting of coordinate system.

When converting unit, pay attention to the following items:

(1) Recover workpiece coordinates to basic system.

(2) Cancel tool compensation.

(3) System-related parameters should be meanwhile adjusted to be in accordance with new unit.

G22, G23: Stored stroke check ON / OFF

Format:

G22 X___ Z___ I___ K___;
G23;

Argument:

X___ Z___andI___ K___: Marks travel range. Machine coordinates. Refer to example.

Action:

G23 is used to cancel tool-stored travel check.

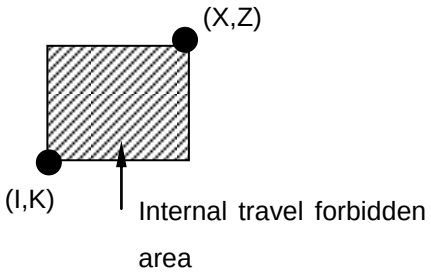
Execute G22 command after manual home returing. Once it's set, the tool can not enter travel-forbidden area specified by G22, or a system warning will occur.

"MOT 9009 X axis over G22 soft limit(+)"
"MOT 9010 X axis over G22 soft limit(-)"
"MOT 9013 Z axis over G22 soft limit(+)"
"MOT 9014 Z axis over G22 soft limit(+)"

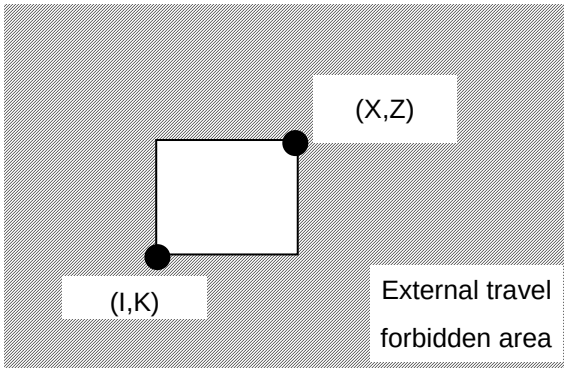
In manual mode, users can move spindle in the opposite direction to cancel warning. In auto mode, besides the aforementioned warning, system warning "MOT 4058 Over soft limit" will also be emitted, and NC stops moving, then users have to click RESET button to cancel warning status.

G22-specified forbidden area can be either internal ro external, and it can be specified by system parameter 0071.

Illustration:



P0071 be set to 1.



P0071 be set to 0.

G27: Reference position return check

Format:

G27 X(U)___ Z(W)___;

Argument:

X___, Z___ : For G90, the reference position in absolute command.
For G91, the reference position in Incremental command.

U___, W___ : For G90/G91, the reference position in Incremental command.

Action:

When program finishes an execution cycle, and is at the end point or back to the reference position, users can perform a position return check in order to make sure the correctness of current real position. This command can check if system returns to the reference position. If it is back to the reference position, the reference position indication light will be alight, and the next block will be executed. If it is not at the reference position, system will emit a warning signal “MOT 4046 reference position return failure”.

If X__ or U__ axis is specified in command, then X axis will perform return and check. If it is not specified, X axis will not move. It is the same as with Z or W axes.

Cancel all compensations before using command G27.

Illustration:

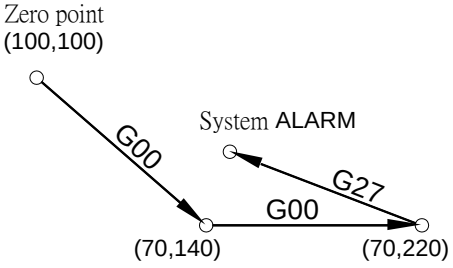
◎Failure

(Turn on the power, then return reference position)
G92 X100. Z100.;
(Reference position be set to (100,100))
G00 U-30. W40.;
G00 W80.;
G27 U20. W-70.;
(ALARM)

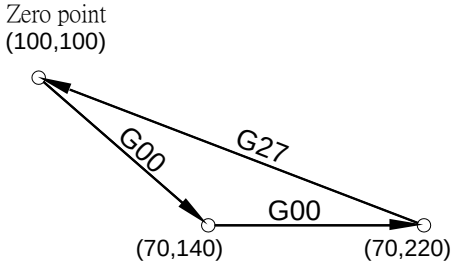
◎Success

G92 X100. Z100.;

G00 U-30. W40.;
G00 W80.;
G27 U30. W-120.;
(Normal)



Failure



Success

G28: 1st reference position return

Format:

G28 X(U)___ Z(W)___;

Argument:

X___Z___ : For G90, the intermediate position in absolute command.
For G91, the intermediate position in Incremental command.

U___W___ : For G90/G91, the intermediate position in Incremental command.

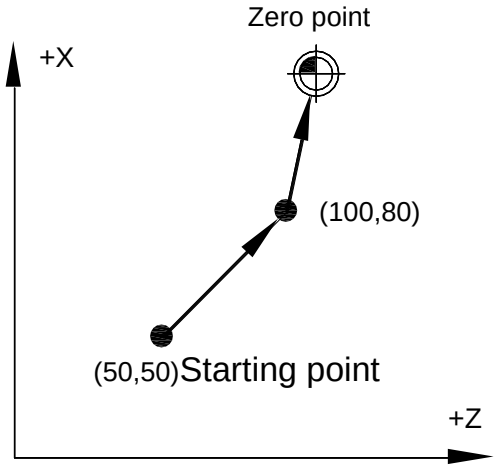
Action:

System reserves G28-specified coordinates of the intermediate point for later G29 to use.

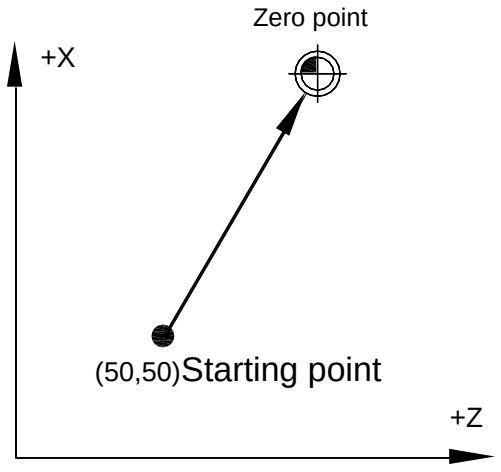
In manufacturing program, use G28 command to control tool to travel through specified intermediate point, and then automatically return to 1st reference position (machine home). Before executing G28, users have to manually perform home return process lest system warning"MOT 4018 no home return after system start" will be emitted.

When argument X___ is not specified, X axis does not perform the process of 1st reference position return, and so do the other axes. However, if there is not any argument of axis direction specified, all axes will perform the process of 1st reference position return.

Illustration:



G90 G28 X100. Z80.;



G91 G28 X0. Z0.;(no passing through intermediate point)

G29: From 1st reference position return

Format:

G29 X(U)___ Z(W)___;

Argument:

X___Z___ : For G90, the target position in absolute command.
For G91, the target position in Incremental command.

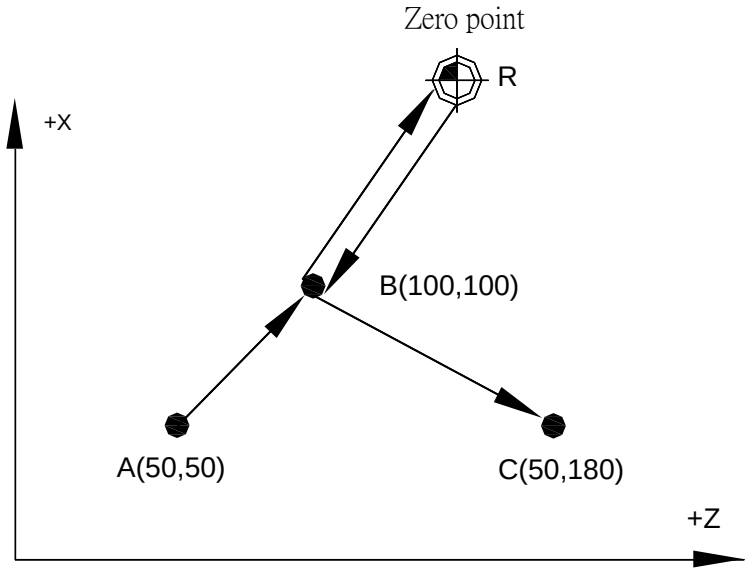
U___W___ : For G90/G91, the target position in Incremental command.

Action:

G29 command is used only after G28. After executing G28, tool stops on 1st reference position, and then G29 can control tool to move from 1st reference position through G28-specified intermediate point to destination position.

Illustration:

G00 X50. Z50.; ---------- (A)
G90 G28 X100. Z100.;---------- (A→B→R)
G29 X50. Z180.;---------- (R→B→C)



G30: 2nd,3rd,4th, reference position return

Format:

$$G30\ P\begin{bmatrix}2\\3\\4\end{bmatrix}\ X(U)\_\_\ Z(W)\_\_;$$

Argument:

P__	:	P2~4: 2nd, 3rd, 4th reference position return.
X___, Z___	:	For G90, the intermediate position in absolute command.
		For G91, the intermediate position in Incremental command.
U___, W___	:	For G90/G91, the intermediate position in Incremental command.

Action:

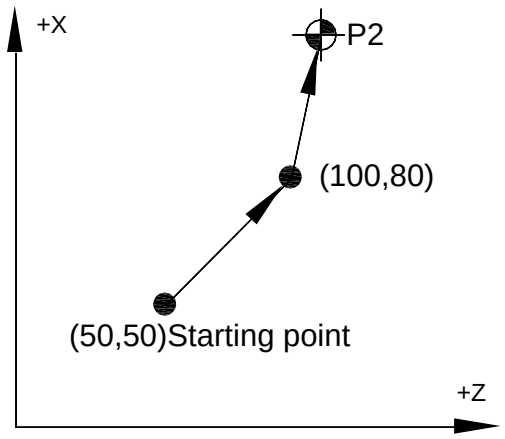
This command is used to perform 2nd, 3rd and 4th reference position return process. The tool will move from current position through specified intermediate point to 2nd, 3rd and 4th reference positions.

The offset amount between 2nd reference position and machine home point can be specified by parameter 1022 ~ 1025; offset amount between 3rd reference position and machine home point can be specified by parameter 1026 ~ 1029; and offset amount between 4th reference position and machine home point can be specified by parameter 1030 ~ 1033.

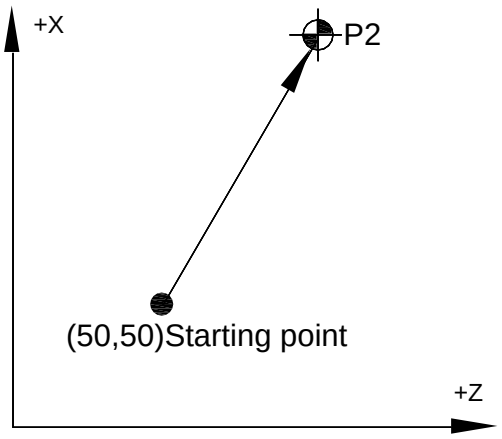
Before executing G30, users have to manually perform home return process lest system warning”MOT 4018 no home return after system start” will be emitted..

When argument X___ is not specified, X axis does not perform the process of reference position return, and so do the other axes. However, if there is not any argument of axis direction specified, all axes will perform the process of reference position return.

Illustration:



G90 G30 P2 X100. Z80.;



G91 G30 P2 X0. Z0.;(no passing through intermediate point)

G31: Skip function

Format:

G31 X(U)___ Z(W)___ P___ F___;

Argument:

X___, Z___ : For G90, the coordinate of an end point in absolute command.
For G91, the coordinate of an end point in Incremental command.

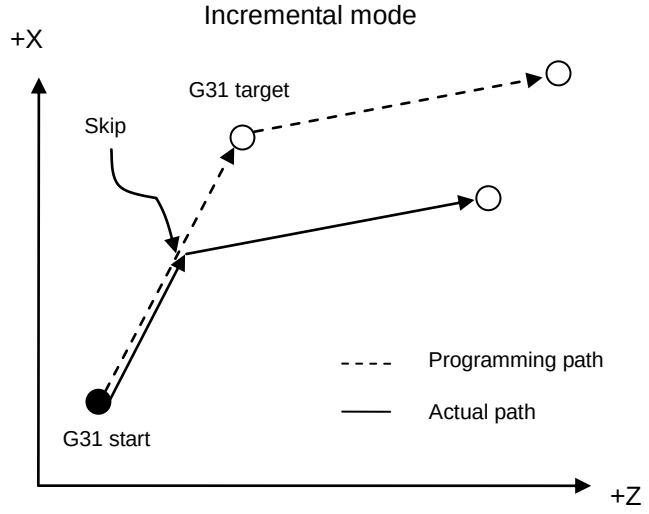
U___, W___ : For G90/G91, the coordinate of an end point in Incremental command.

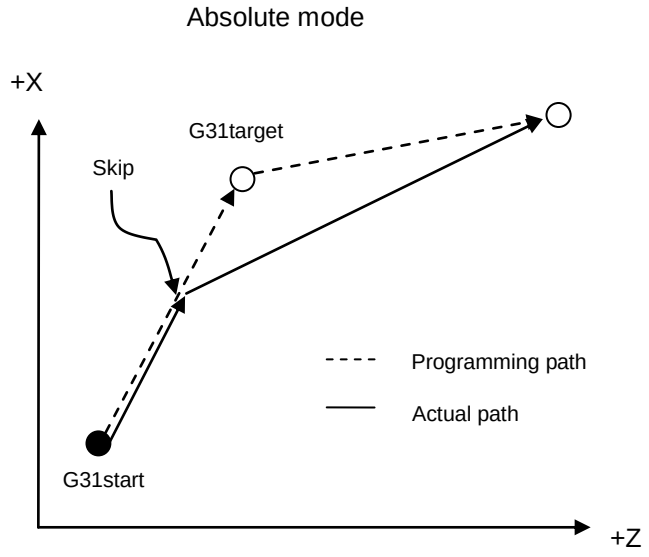
P___ P1~P4.designate skipsignal source. If P_ is not specified, the default value is P1.

F___ : Feed rate of G31 block. The specified is only effective in this block. If it's not specified, the value of parameter 1042 will be the feed rate of the block.

Action:

This command has the same function of G01. But if a skip signal is triggered duting the execution, the block will cease working, and program will move to the next block.





Note:

1. When G31 skip signal is triggered, system saves coordinate value of the break point to system macro variables, as the following table shows. But before G31 skip signal is triggered, these variables are destination position coordinates of G31 command. Besides, if absolute coordinate was once used to redesignate absolute coordinate, e.g. G92 (G50 in lathe A type), the saved absolute coordinate will not include the offset made by G92.

Lathe		X	Y	Z	4th axis
P1	Absolute coordinate of skip point	$140	$141	$142	$143
	Machine coordinate of skip point	$144	$145	$146	$147
P2	Absolute coordinate of skip point	$148	$149	$150	$151
	Machine coordinate of skip point	$152	$153	$154	$155
P3	Absolute coordinate of skip point	$156	$157	$158	$159
	Machine coordinate of skip point	$160	$161	$162	$163
P4	Absolute coordinate of skip point	$164	$165	$166	$167
	Machine coordinate of skip point	$168	$169	$170	$171

2. The lock feature of P1~P4 break position in G31 can be specified by parameter G31 P1~4 signal source Local signal point serial numbers. Specified as 1~2: corresponding to 1st axis card's 1~2 Local Input, will save values in absolute position value recorder of each axis, and the coordinate of break point can be obtained accurately. Specified as 3~8: corresponding to 1st axis card's 3~8 Local Input, for using software-specified lock action.

G33: Thread cutting

Format:

G33 X(U)__ Z(W)__ F__; (Constant-lead thread cutting)
G33 X(U)__ Z(W)__ F__ Q__; (Mulitiple-thread cutting)

Argument:

X(U)___ : End point of X axis.(mm)

Z(W)___ : End point of Z axis.(mm)

F___ : Lead in longitudinal direction.(mm/rev)

Example:

G33 X_ F_ ; Lead in direction of X axis.

G33 X_ Z_ F_; Lead in direction of X axis.(X_ > Z_)

G33 X_ Z_ F_; Lead in direction of Z axis. (X_ < Z_)

Q___ : Threading start angle. (Unit: 0.001deg. No decimal poiint. Range: 0 ~ 360000)

Action:

1. F__ and Q__ are both norms. Once specified, there will be no need to input in later blocks.
2. In continuous G33 blocks, only the specified Q__ of the 1st G33 block is effective.
   E.g.:
   G33 W-10 F1 Q18000; ---------- Start angle is 180 degree
   U-5 W-5 Q270000; ----- Due to continuous threading, the specified 270 degree is not effective
3. In G33 threading process, feed rate adjustment knob is not effective (fixed to 100%).
   In continuous G33 blocks, only the spindle turn around signal in the 1st block will be searched, the signal will not be waited in later blocks.
   E.g.:
   G33 W-10 F1; ---------- Wait for a turn around signal
   U-5 W-10;---------- Do not wait
   U-10;---------- Do not wait
4. During threading, spindle speed can be adjusted, but it will result in threading error.
5. Because servo system follows the error, when threading, incomplete threads will occur at the stard and end positions. In order to improve it, when perform threading, designate the length of thread to be longer than actual necessity.

Example:

Self-defined departure angle and departure speed cutting (peroform cutting only once)

T0707; ---------- Call for #7 tool

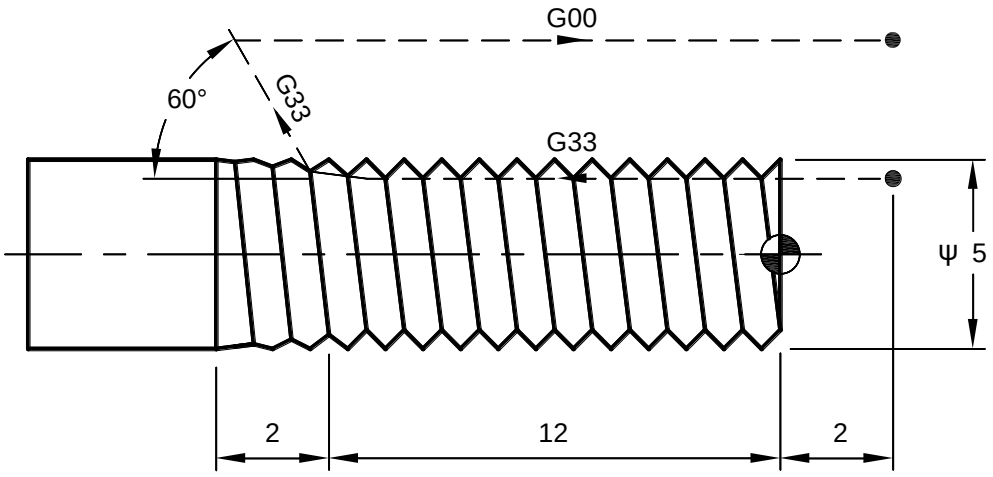
G00 X4.5 Z2.; ---------- X axis, Z axis rapidly move to cutting start position

G33 Z-12. F1; ----------1st section of threading, thread pith 1mm

G33 X11.43 Z-14; ----------2nd section of threading, departure angle 60 degrees

G00 Z2; ----------Z axis rapid movement

M30; ---------- Program ends



G34: Variable Thread Pith Threading

Format:

G34 X(U)___Z(W)___ F___ Q___ K___

Argument:

X(U)___	:	End point of X axis. (mm)
Z(W)___	:	End point of Z axis. (mm)
F___	:	Lead in longitudinal direction.(mm/rev)
Q___	:	Threading start angle. (Unit: 0.001deg, no decimal poiint. Range: 0 ~ 360000)
K___	:	Lead per spindle revolution. (mm/rev) Increment in positive, and decrement in negative.

Action:

G34 has the same usage as G33 (excluding K___).

In continuous G34, K can be specified repeatly in every block (E.g. A lead which originally increase gradually can turn to decrease gradually in the next block).

Example:

main program

T0707; ---------- call for #7 tool

G0 X7.; ---------- move to start position

Z-5.;

G66 P0342 K0.1;---------- Use G66 call for self-defined Macro O0342

X6.5;----------1st cut

X6.25;---------- 2nd cut

X6.04;---------- 3rd cut

X5.9;----------4th cut

X5.8;----------5th cut

G67; ---------- End self-defined Macro

G0 X20;

Z30;

M30;

subprogram

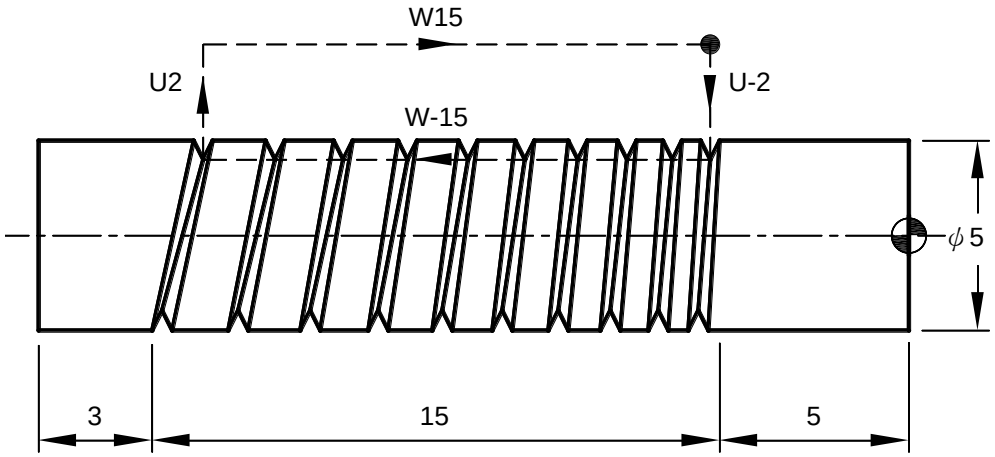
G33 U-2. F2;Use G33 thread pitch 4mm for approach (F=2mm is used to increase approach speed)

G34 W-15. K#11 F1; ---------- variable lead thread cutting (K=0.1)

G33 U2 F4;Use G33 thread pitch 8mm for departure (F=4mmis used to increase departure speed)

G0 W15;----------use G00 to rapidly return to start position

M99;----------subprogram ends



Continuous Threading

command format 1: (continuous G33)

G33 X(U)__ Z(W)__ F__ Q__;
G33 X(U)__ Z(W)__ F__;
G33 X(U)__ Z(W)__ F__;

Action:

It is used to cut continuous threads. At the meantime, Z axis direction cutting amount or X axis direction cutting amount both follow spindle revolve amount (please refer to the instruction on G33 argument F__).

Note:

1. Specified threading approach angle Q__ is only effective in the 1st block of continuous G33 blocks.
2. Thread pitch F__ can be specified in every block, and sothreads with variable thread pitch can be made.

command format 2: (continuous G34)

G34 X(U)__ Z(W)__ F__ Q__ K__;
G34 X(U)__ Z(W)__ K__;
G34 X(U)__ Z(W)__ K__;

Action:

It is used for cutting continuous variable thread.

Note:

1. Designation of threading approach angle Q__ is only effective in the 1st block of continuous G34 blocks.
2. Designation of thread pitch F__ is only effective in the 1st block of continuous G34 blocks.
3. Thread increment K__ can be specified in every block.

command format 3: (combined G33, G34)

G33 X(U)__ Z(W)__ F__ Q__
G34 X(U)__ Z(W)__ F__ K__
G34 X(U)__ Z(W)__ K__
G33 X(U)__ Z(W)__ F__

Action:

Achieve special thread cutting with features of continuous G33 and G34. G33 can change pitch in continuous blocks, and G34 can change thread increment in continuous blocks, but start angle Q__ can still be specified in the 1st block only.

Example:

continuous thread cutting

main program

T0707; ---------- call for #7 tool
G0 X12.; ---------- X axis rapidly traverses to cutting start position
Z-5.; ----------Z axis rapidly traverses to cutting start position
G66 P0332;---------- continuous call for subprogram
X11.6; ----------thread cutting 1st cut
X11.4; ---------- thread cutting 2nd cut
X11.1; ----------thread cutting 3rd cut
X10.9; ----------thread cutting 4th cut
X10.8; ----------thread cutting 5th cut
G67; ----------cancel call for subprogram
G0 X20;---------- X axis rapidly traverses
Z30;---------- Z axis rapidly traverses
M30; ---------- program ends

subprogram

G33 U-7 F1; ----------thread cutting approach thread pitch 1 mm
W-5; ---------- thread cutting 1st section
U5 W-10; ---------- thread cutting 2nd section (taper section)
W-5; ---------- thread cutting 3rd section
U2; ---------- thread cutting departure
G0 W20; ----------Z axisrapidly returns tostart position
M99; ----------Return from subprogram to main program

multi-line continuous thread cutting

main program

T0707; ---------- call for #7 tool

G0 X12.;---------- X axis rapidly traverses to cutting start position

Z-5.; ----------Z axis rapidly traverses to cutting start position

G66 P0332 A0; ---------- continuous call for subprogram

X11.6; ---------- thread cutting 1st cut

X11.4; ---------- thread cutting 2nd cut

X11.1; ----------thread cutting 3rd cut

X10.9;----------thread cutting 4th cut

X10.8;----------thread cutting 5th cut

G67; ----------cancelcall for subprogram

G66 P0332 A180000; -- continuous call for subprogram, 2nd line of thread (start angle 180 degree)

X11.6; ---------- thread cutting 1st cut

X11.4; ---------- thread cutting 2nd cut

X11.1; ----------thread cutting 3rd cut

X10.9;----------thread cutting 4th cut

X10.8;----------thread cutting 5th cut

G67; ----------cancel call for subprogram

G0 X20;----------X axis rapidly traverses

Z30;----------Z axis rapidly traverses

M30;---------- program ends

subprogram

G33 U-7 F1 Q#1; approach, thread pitch 1 mm (#1 is thread approach position angle, inserted from A__ to subprogram)

W-5; ---------- thread cutting 1st section

U5 W-10;----------thread cutting 2nd section

W-5; ---------- thread cutting 3rd section

U2;----------tool depart

G0 W20;---------- Z axis rapidly traverses (increment coordinate)

M99;----------return from subprogram to main program

sectional variable lead continuous thread cutting

main program

T0707; ---------- call for #7 tool

G0 X12.;---------- X axis rapidly traverses to cutting start position

Z-5.; ----------Z axis rapidly traverses to cutting start position

G66 P0332;---------- continuous call for subprogram

X11.6; ---------- thread cutting 1st cut

X11.4; ---------- thread cutting 2nd cut

X11.1; ----------thread cutting 3rd cut

X10.9;----------thread cutting 4th cut

X10.8;----------thread cutting 5th cut

G67; ----------cancelcall for subprogram

G0 X20;---------- X axis rapidly traverses

Z30;---------- Z axis rapidly traverses

M30;---------- program ends

subprogram

G33 U-7. F2.; ----------approach, thread pitch 2 mm

W-5. F1.25;----------thread cutting 1st section, thread pitch 1.25 mm

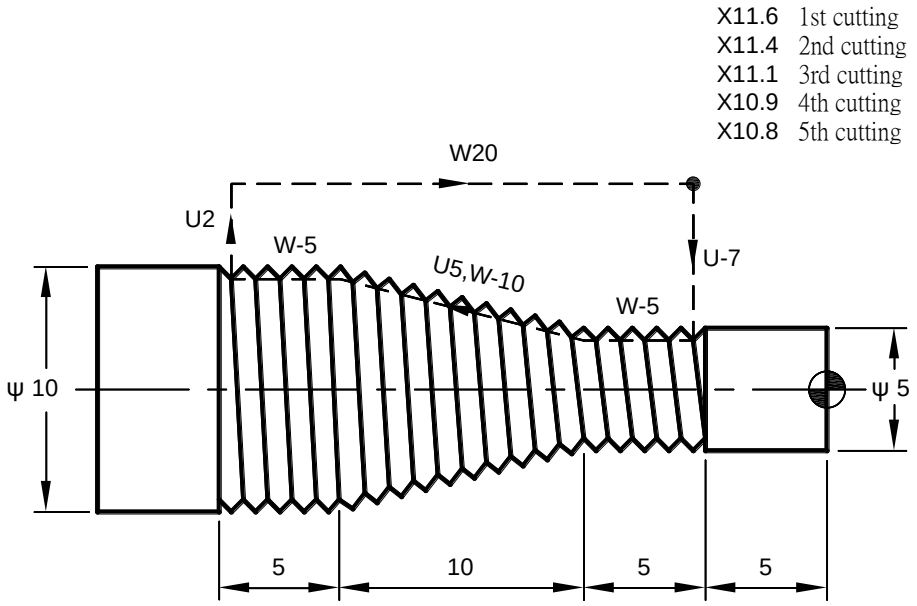
U5. W-10. F1.5;----------thread cutting 2nd section, thread pitch 1.5 mm

W-5. F1.75; ----------thread cutting 3rd section, thread pitch 1.75 mm

U2. F2.;----------departure, thread pitch 2 mm

G0 W20.;---------- Z axis rapidly returns to start position

M99;---------- return from subprogram to main program


X11.6 1st cutting
X11.4 2nd cutting
X11.1 3rd cutting
X10.9 4th cutting
X10.8 5th cutting
W20
U2
W-5
U5,W-10
U-7
W-5
ψ 10
ψ 5
5
10
5
5

G40, G41, G42: Tool nose radius compensation

Format:

$$\begin{bmatrix} G17 \\ G18 \\ G19 \end{bmatrix} \begin{bmatrix} G41 \\ G42 \end{bmatrix}$$
G40;

Argument:

G40	:	Tool nose radius compensation. (Cancel)
G41	:	Tool nose radius compensation. (Left)
G42:	:	Tool nose radius compensation. (Right)

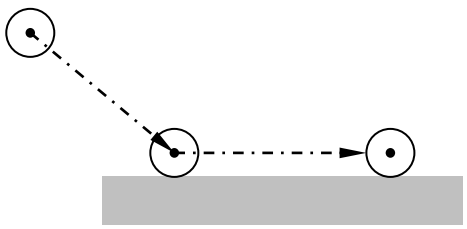
Action:

Blocks in which tool radius compensation value begins and cancels must be linear command (G00 or G01), not arc command (G02 or G03).

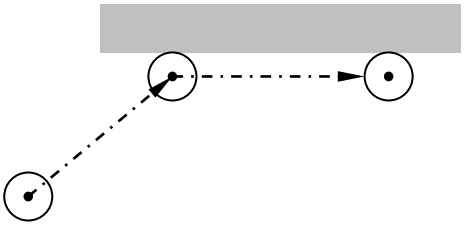
Tool radius compensation can be divided into Type A and Type B, which is decided by parameter 0131.

Illustration:

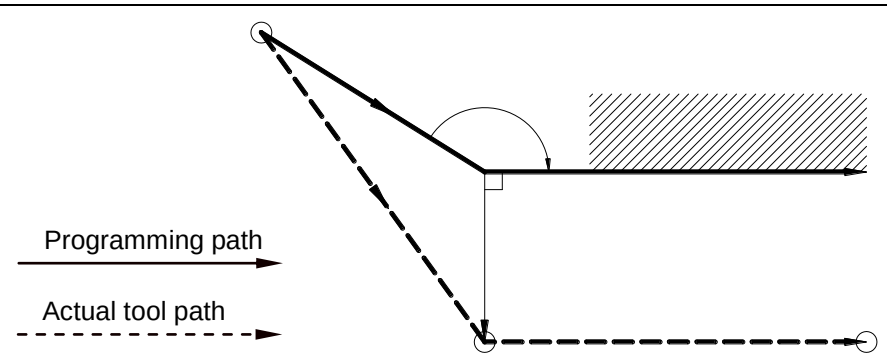
G41: When face to tool movement direction, tool shifts to the left for a radius.

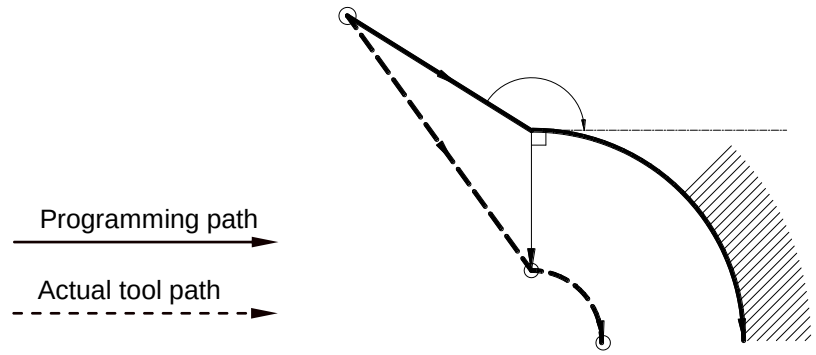


G42: When face to tool movement direction, tool shifts to the right for a radius.

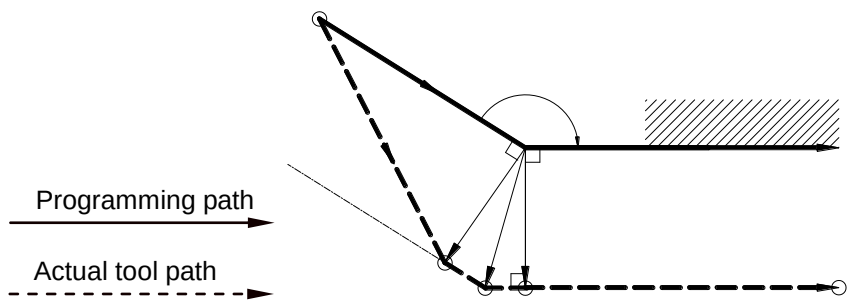


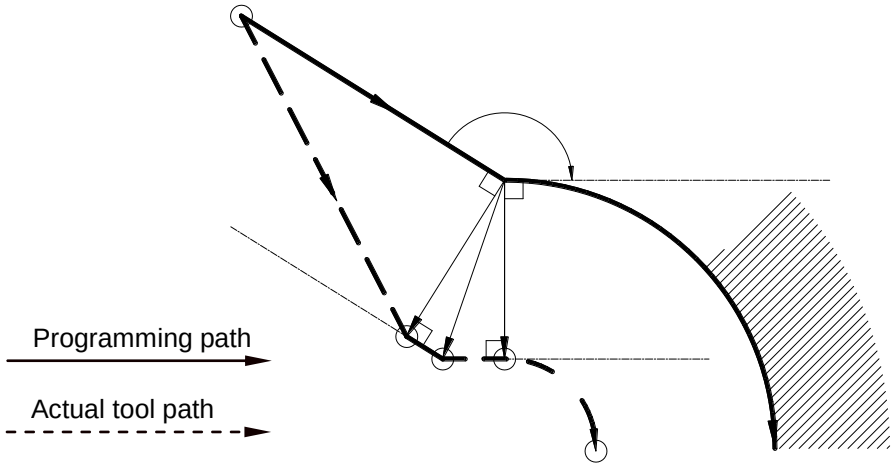
TYPE A


Programming path
Actual tool path


Programming path
Actual tool path

TYPE B


Programming path
Actual tool path


Programming path
Actual tool path

G53: Machine coordinate system

Format:

G53 X__ Z__;

Argument:

X___ : Tool traverses to machine coordinate of X axis.

Z___ : Tool traverses to machine coordinate of Z axis.

Action:

Machine home point is the fixed original position specified by a machine factory while producing CNC machines. This coordinate system is fixed, and can not be changed. When designating G53 command and coordinate commands, tool traverses to the specified position on the basic machine coordinate system. When tool returns to machine home point(0,0), the position of G53 is the original position of the machine coordinate system.

G53 machine coordinate system is also called 00 coordinate system.

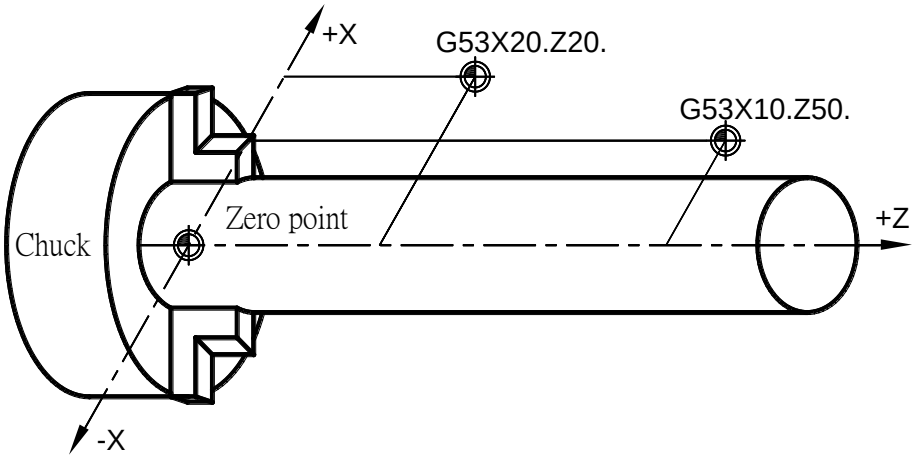
Note:

1. G53 command is only effective in the specified block.
2. G53 is only effective in absolute value mode, not in incremental value mode.
3. Before G53 is specified, erase relevant tool radius, length or position compensation.
4. Befoire using G53 to set up coordinate system, manually set up a coordinate system based on the returned position of reference position.
5. If G53 coordinate system has a set value, when executing G54~G59 coordinate system, an offset occurs to G53 coordinate system's set value.

Program Example :

G53 X20. Z20. ; ---------- (Move to machine coordinate)

G53 X10. Z50. ; ---------- (Move to machine coordinate)


+X
G53X20.Z20.
G53X10.Z50.
Chuck
Zero point
+Z
-X

G54 ~ G59: Selection of work coordinate system

Format:

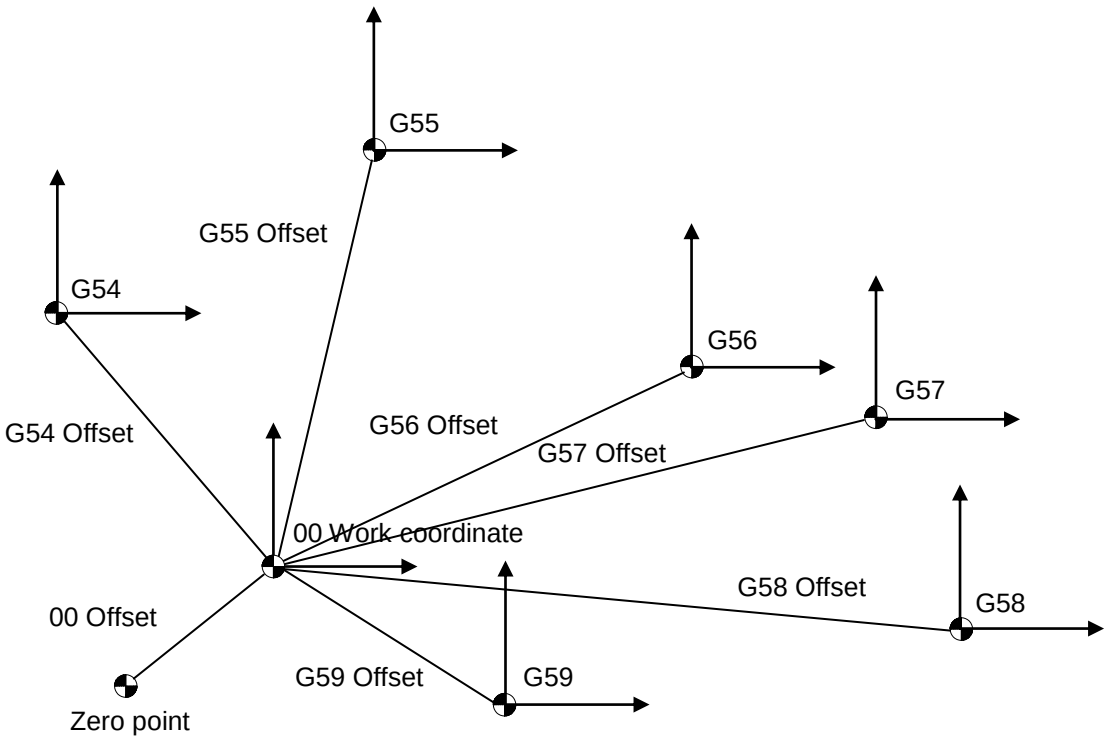
```
[G54;
 G55;
 G56;
 G57;
 G58;
 G59;]
```

Action:

The workpiece coordinate system adopts G54~G59 to represent 6 different coordinate systems. Users can select among them according to manufacturing needs.

Each coordinate system's original position offset can be set in 〈OFFSET〉→〈coordinate system setting〉; refer to operation manual for detailed instructions. Besides, it can also be set by G10 command, and please refer to G10 command for detailed instructions.

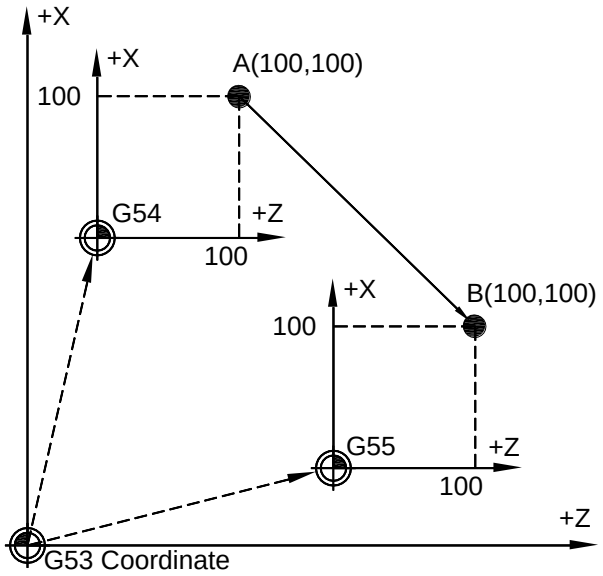
The relationship between each coordinate system is as the following: (default coordinate system when system starts is G54 coordinate system)



Example:

G90 G54 G00 X100. Z100.;

G55 X100. Z100.; ----------A→B



G61, G64: Exact stop mode / Cutting mode

Format:

```
G61;
G64;
```

Argument:

G61 : Exact stop mode

G64 : Cutting mode

Action:

G61's function is the same as G09, while G09 is only effective in its block, G61 is effective ever since it's executed until G64 (general cutting) is executed. G64 is the default system mode, unless G61 is executed, system stays in G64 mode.

To cutting commands (G01/G02/G03), each axis' positioning precision is set by parameter 0006 ~ 0009; to rapid traverse (G00), each axis' positioning precision is set by parameter 0800 ~ 0803. Moreover, use parameter 0043 to define whether each axis' correct positioning function is enabled.

Illustration:

Example:

```
G61 G91 G01 X100. F200.; ---------- Exact stop mode
Z100.; ---------- Exact stop mode
G64; ---------- Cutting mode
```

G65: User macro simple call

Format:

G65 P__ L__ <Argument…>;

Argument:

P__ : Number of the program to call. P9010 call file name O9010. If P_ is inputted vacant, controller will have an alarm “INT 3111 Lack of file name”.

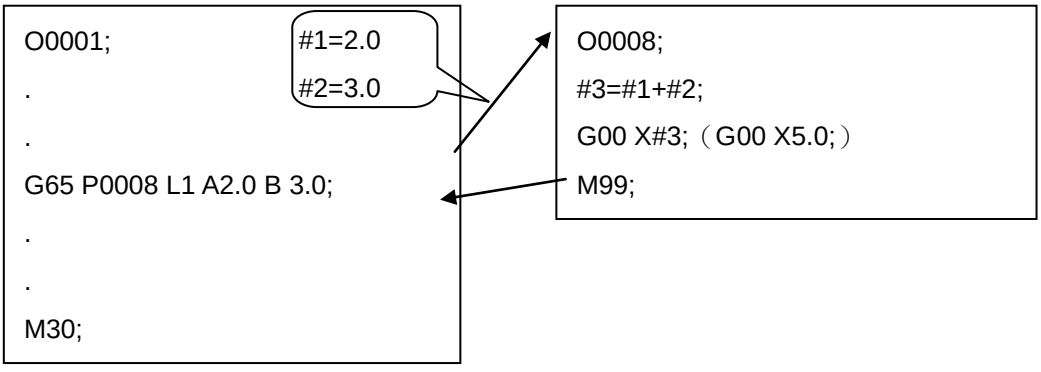
L__ : Repetition count. (1 by default)

Besides P and L arguments, users can use other NC addresses (English letters excluding G, L, N, O, P) to lead in arguments. The order of sequence does not matter. These argument values are corresponding to local variables in called macro programs as the following charts:

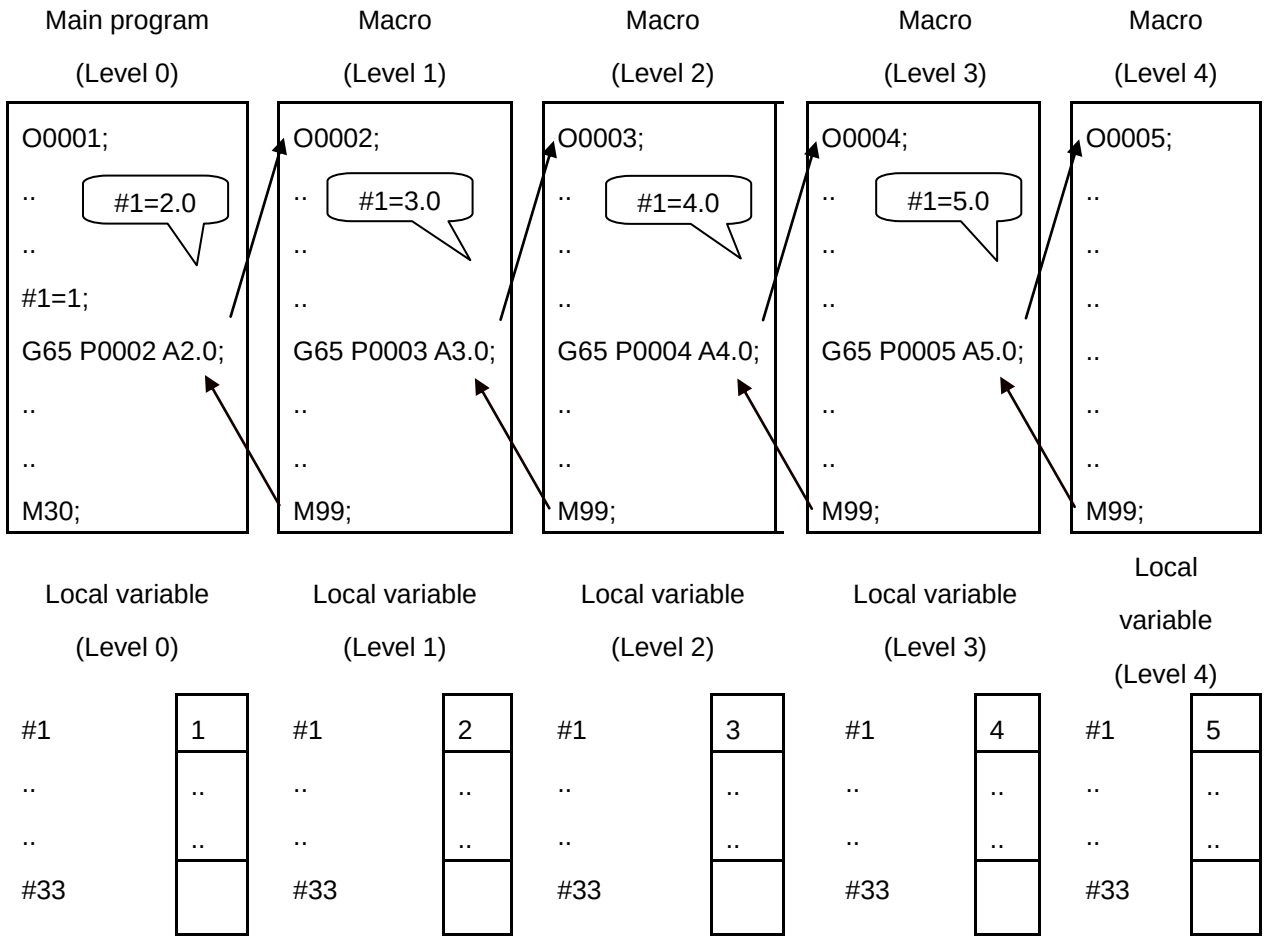
Address	Local variable
A	#1
B	#2
C	#3
D	#4
E	#5
F	#6
H	#8

Address	Local variable
I	#9
J	#10
K	#11
M	#13
Q	#17
R	#18
S	#19

Address	Local variable
T	#20
U	#21
V	#22
W	#23
X	#24
Y	#25
Z	#26



In a G65 block, G65 must be prior to all arguments. G65 can do nested calls, the combination of G65 and G66 can be up to the 4th level (excluding main program which is the 0th level), and each level has its own local variables, as shown in the following charts:



G66: User macro modal call

Format:

G66 P__ L__ <argument…>;

Argument:

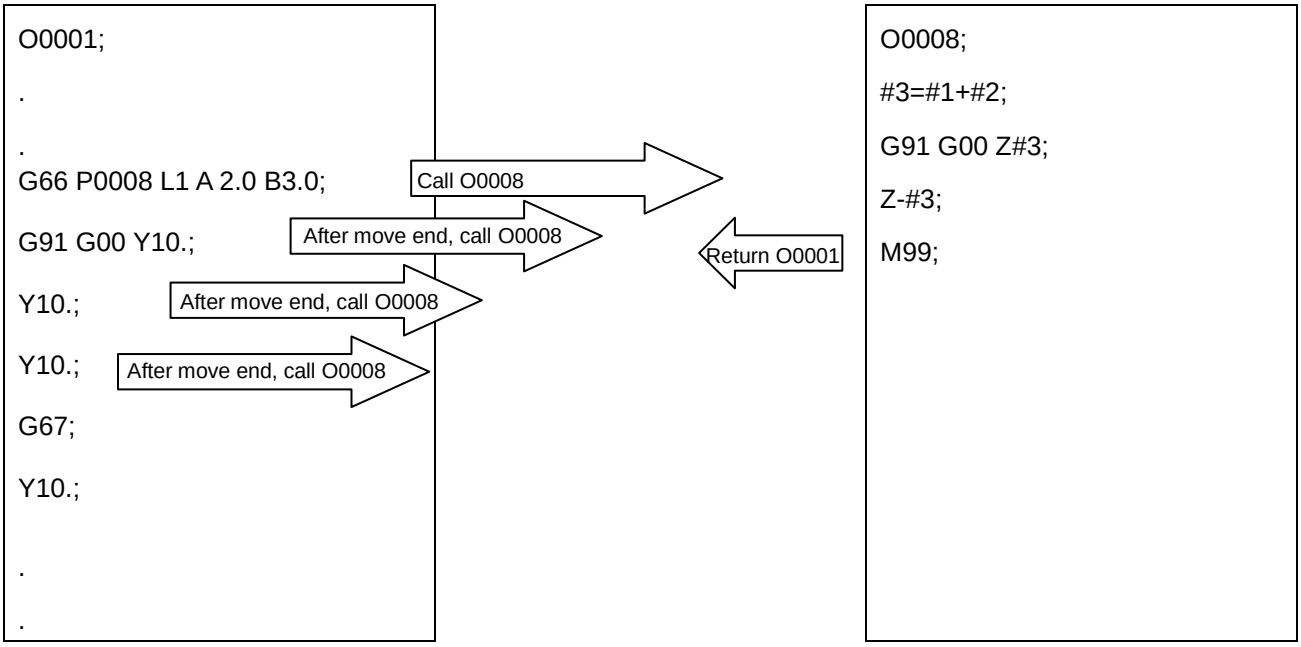
P__ : Number of the program to call. P9010 call file name O9010. If P_ is inputted vacant, controller will have an alarm “INT 3111 Lack of file name”.

L__ : Repetition count. (1 by default)

Besides P and L arguments, users can use other NC addresses (English letters excluding G, L, N, O, P) to lead in arguments. The order of sequence does not matter. These argument values are corresponding to local variables in called macro programs, and please refer to G65 for relevant instructions.

Action:

The only difference between G66 and G65 is that G65 calls a macro program which is used for only one time, but the macro programs called by G66 will be called every time when a motion block finishes until G67 is used to cancel this mode.



In a G66 block, G66 must be prior to all arguments. G66 can do nested calls. The combination of G65 and G66 can be up to the 4^{th} level (excluding main program which is the 0^{th} level), but G66’s argument (corresponding to macro program’s local variables) is only specified once in the block of G66, and it won’t be respecified in following mode calls.

G67: User macro modal call cancel

Format:

G67;

Action:

G67 is applied to cancel the calling for G66 macro program mode.

G68: Mirror image for double turrets ON

G69: Mirror image for double turrets OFF

Format:

G68;
G69;

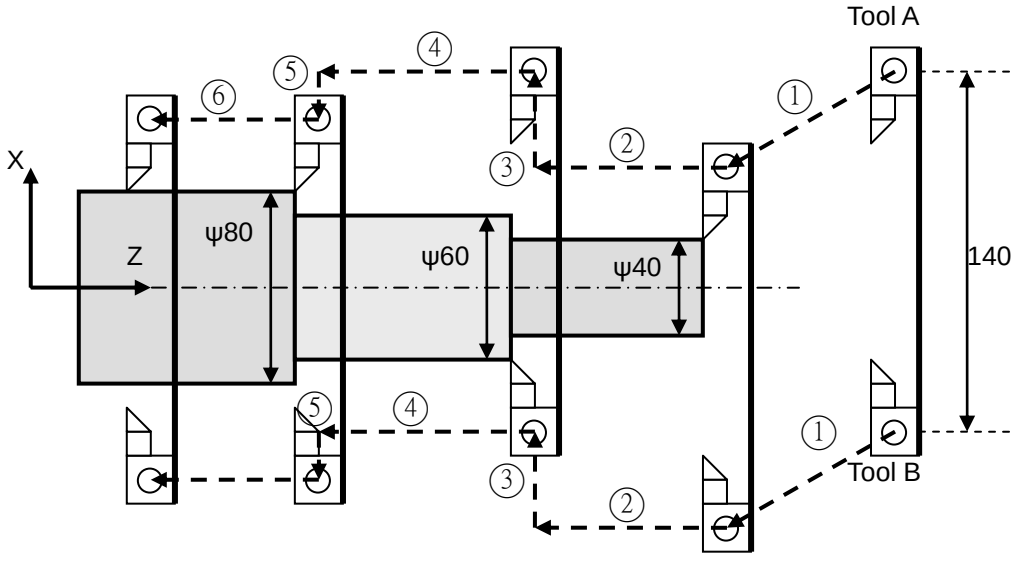
Action:

This G CODE group mode can be obtained through system variable $16.

Once enter G68 mode, users can only cancel it by G69, i.e. clicking RESET button will not change this mode.

When users execute G68 command, NC will at first make an offset to the coordinate system of the corresponding tool on the other side (the distance bwtween the two paired tools is specified by system parameter #1099); latter on, the specified X axis command amount in the manufacturing program will take Z axis as the center axis and be mirrored to the other side.

Illustration:



…

T0101; ----------Tool A

G00X40.Z150.; ---------- Path 1

G01Z120.F0.1; ---------- Path 2

G68; ----------Mirror image of X axis ON. B(-140mm)

T0202; ---------- Tool B

G00X60. ---------- Path 3

G01Z100.; ---------- Path 4

G69; ----------Mirror image of X axis OFF. A(140mm)

T0101; ----------Toll A

G00X80.; ---------- Path 5

G01Z50.; ---------- Path 6

…

G70: Finishing Cycle

Format:

G70 P__ Q__;

Argument:

P__ : Sequence number of the first block for the program of finishing shape.

Q__ : Sequence number of the last block for the program of finishing shape.

Action:

It is used after G71, G72, G73 rough cutting cycles along with G70 command to perform precise cutting to obtain desired size.

Please refer to G71 for examples.

G71: Stock removal in turning

Format:

G71 U d R e ;

G71 P ns Q nf U___ W___ F___ S___ T___;

Argument:

U d : Depth of cut in X axis direction. (Radius designation) This value can be specified by the parameter (P1081).

R e : Escaping amount. This value can be specified by the parameter (P1082).

P ns : Sequence number of the first block for the program of finishing shape.

Q nf : Sequence number of the last block for the program of finishing shape.

U___ : Distance and direction of finishing allowance in X axis direction.

W___ : Distance and direction of finishing allowance in Z axis direction.

F___ : Feedrate.

T___ : Tool number.

S___ : Spindle speed.

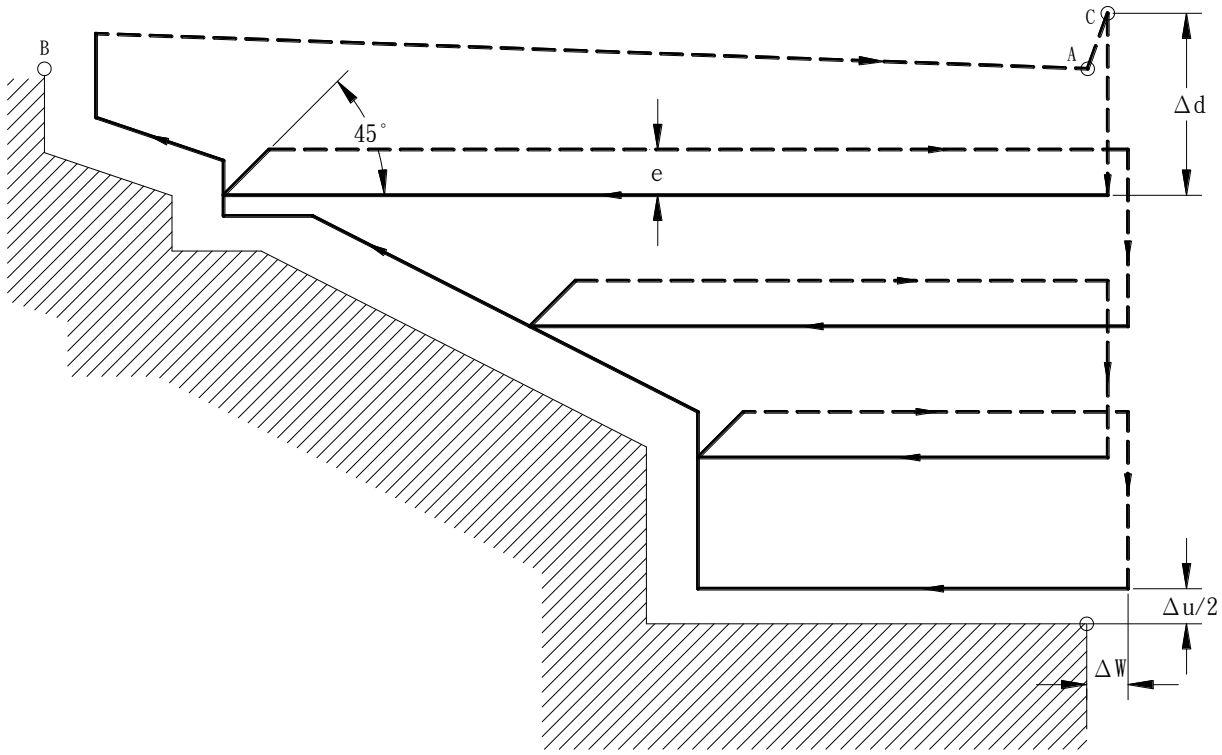
Action:

1. Rapidly traverse (G00) to A position (start position) before cycle begins;
2. After executing G71 command, tool takes the set preserved precise cutting amount (X axis is U/2, Z axis is W) as the offset amount;
3. Tool again traverses to Z axis for a distance of U(d), and traverse to outline surface;
4. Then depart in Z axis direction for a distance of e at 45◦, and X axis traverse in the opposite direction until the adjacent position which is parallel to the start position;
5. Then depart in Z axis direction for a distance of U(d) to continue the next repeated cycle ;
6. As the last cycle ends, tool lathe cutting once along outline A'→B;
7. When finishes, tool rapidly traverses to A position, and waits for next cycle to start.

Note:

1. Outline path is specified by blocks between ns and nf, ranges from A position to A' position then to B position.
2. F, S, T commands specified in blocks between ns→nf are not effective, they can be effective only when written in blocks of rough lathing cycle (G71).
3. Blocks between ns→nf can not do subprogram call .

Illustration:



G72: Stock removal in facing

Format:

G72 W d R e ;

G72 P ns Q nf U___ W___ F___ S___ T___;

Argument:

W d : Depth of cut in Z axis direction. This value can be specified by the parameter (P1081).

R e : Escaping amount. This value can be specified by the parameter (P1082).

P ns : Sequence number of the first block for the program of finishing shape.

Q nf : Sequence number of the last block for the program of finishing shape.

U___ : Distance and direction of finishing allowance in X axis direction.

W___ : Distance and direction of finishing allowance in Z axis direction.

F___ : Feedrate.

T___ : Tool number.

S___ : Spindle speed.

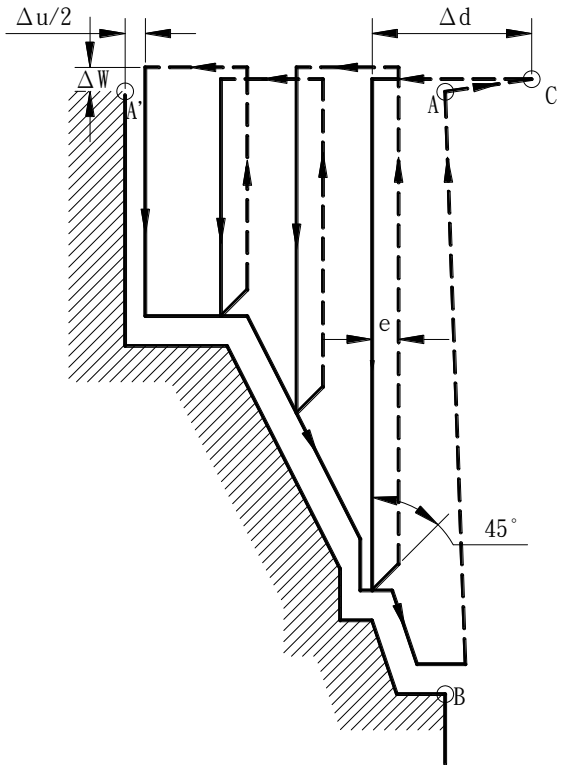
Action:

1. Rapidly traverse (G00) to A position (start position) before cycle begins;
2. After executing G72 command, tool takes the set preserved precise cutting amount (X axis is U/2, Z axis is W) as the offset amount;
3. Tool again traverses to Z axis for a distance of U(d), and traverse to outline surface;
4. Then depart in Z axis direction for a distance of R(e) at 45∘, and X axis traverse in the opposite direction until the adjacent position which is parallel to the start position;
5. Then depart in Z axis direction for a distance of U(d) to continue the next repeated cycle ;
6. As the last cycle ends, tool lathe cutting once along outline A'→B;
7. When finishes, tool rapidly traverses to A position, and waits for next cycle to start.

Note:

1. Outline path is specified by blocks between ns and nf, ranges from A position to A' position then to B position.
2. F, S, T commands specified in blocks between ns→nf are not effective, they can be effective only when written in blocks of rough lathing cycle (G72).
3. Blocks between ns→nf can not do subprogram call .

Illustration:



G73: Pattern repeating

Format:

G73 U i W k R d ;

G73 P ns Q nf U___ W___ F___ S___ T___;

Argument:

U i : Distance and direction of relief in X axis direction. (Radius designation) This value can be specified by the parameter (P1083).

W k : Distance and direction of relief in Z axis direction. This value can be specified by the parameter (P1084).

R d : The number of division. This value can be specified by the parameter (P1085).

P ns : Sequence number of the first block for the program of finishing shape.

Q nf : Sequence number of the last block for the program of finishing shape.

U___ : Distance and direction of finishing allowance in X axis direction.

W___ : Distance and direction of finishing allowance in Z axis direction.

F___ : Feedrate.

S___ : Tool number.

T___ : Spindle speed.

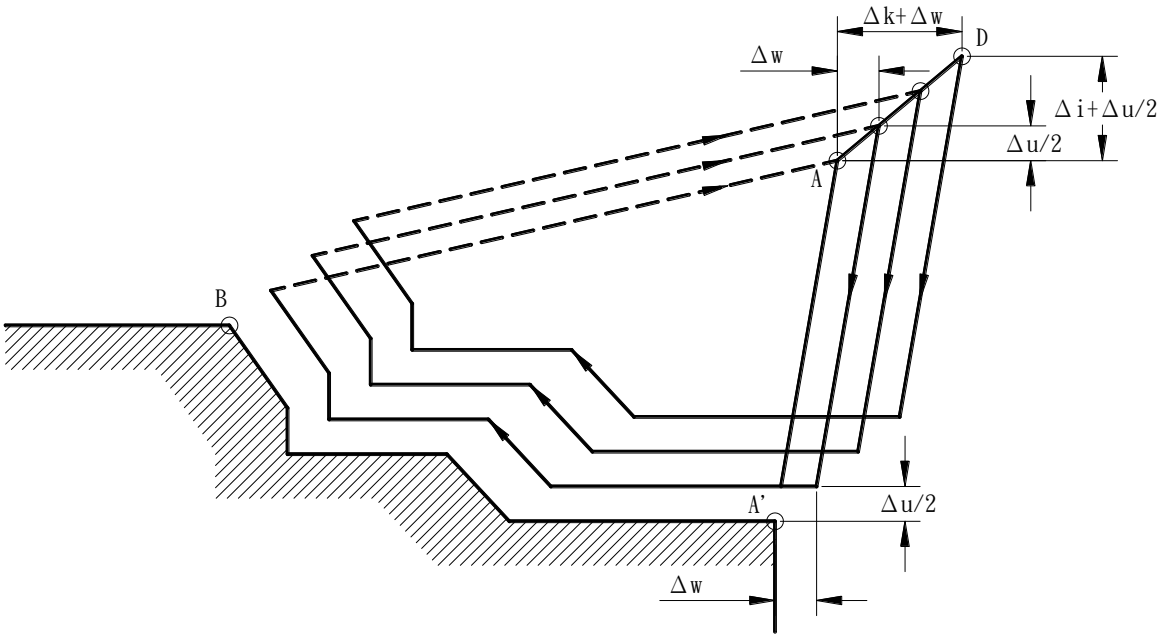
Action:

G73 command is an outline forming rough lathing cycle which is used when the workpiece is a welding or casting product with preliminary shapes, and with a size slightly bigger than precise manufacturing, hence if using G71, G72 lathe command will result in generating many unnecessary cutting path and the waste of time. Therefore, G73 (mixed-type outline rough cutting fixed cycle) is used to make repeated lathing which can move along workpiece's indigenous outline, repeat cutting for necessary times, and move for a suitable distance and depth in each cycle.

1. Traverse tool to A position (start position) before cycle begins;
2. After executing G73 command, tool takes the set preserved precise cutting amount (X axis is U/2, Z axis is W) plus cutting amount (X axis is i, Z axis is k) as the offset amount, and then traverse to C position;
3. Tool lathes along program path A→A'→B, and according to feed and cutting times to finish cycle-type manufacturing;

4. When last cycle finishes, tool automatically returns to A position, and waits for next lathe cycle.

Illustration:



Note:

1. Outline path is specified by blocks between ns and nf, ranges from A position to A' position then to B position.
2. F, S, T commands specified in blocks between ns→nf are not effective, they can be effective only when written in blocks of rough lathing cycle (G73).
3. Blocks between ns→nf can not do subprogram call .

G74: End face peck drilling cycle (Z axis)

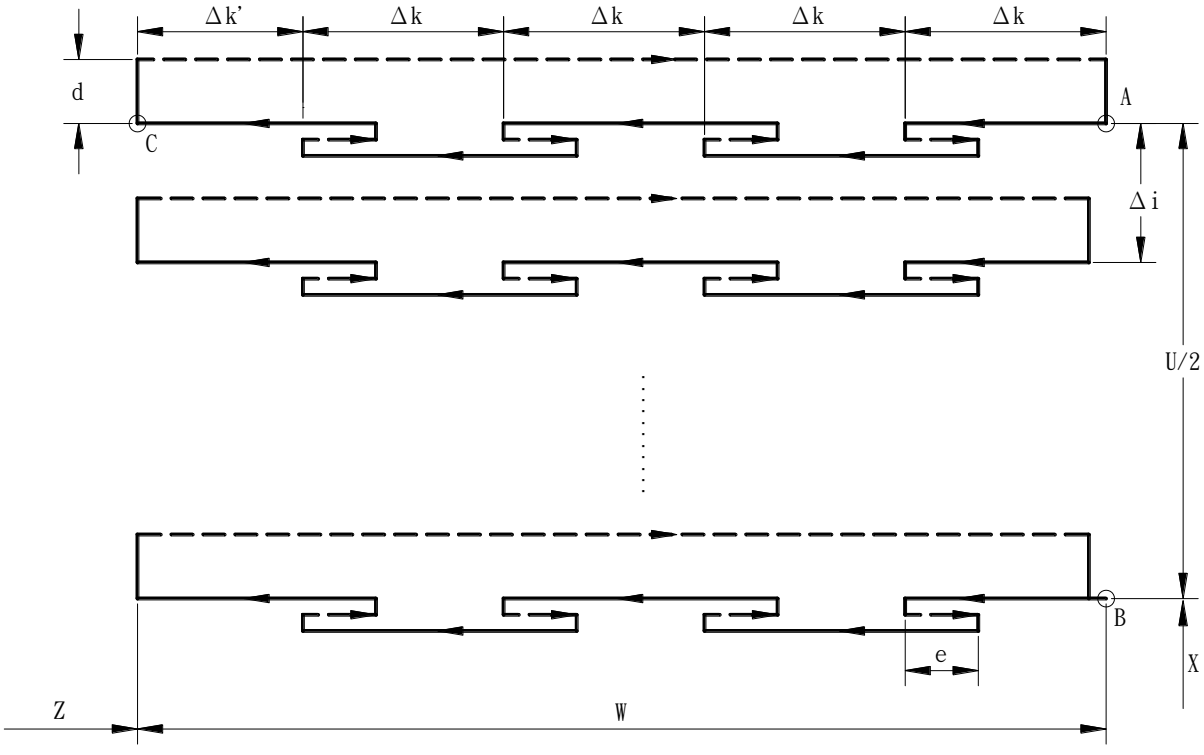
Format:

```
G74 R e   P   ;
G74 X(U)   Z(W)   P  i  Q  k  R d  F   ;
```

Argument:

R e	:	Return amount. This value can be specified by the parameter (P1086).
P___	:	Dwell time at the cutting bottom.
X___	:	X component of point B.
Z___	:	Z component of point C.
U___	:	Incremental amount from A to B.
W___	:	Incremental amount from A to C.
P i	:	Movement amount of X axisdirection. No decimal poiint.
Q k	:	Depth of cut in Z axisdirection. No decimal poiint.
R d	:	Relief amount of the tool at the ccutting bottom in X axis direction.
F___	:	Feedrate.

Illustration:



G75: Outer diameter / Internal diameter drilling cycle (X axis)

Format:

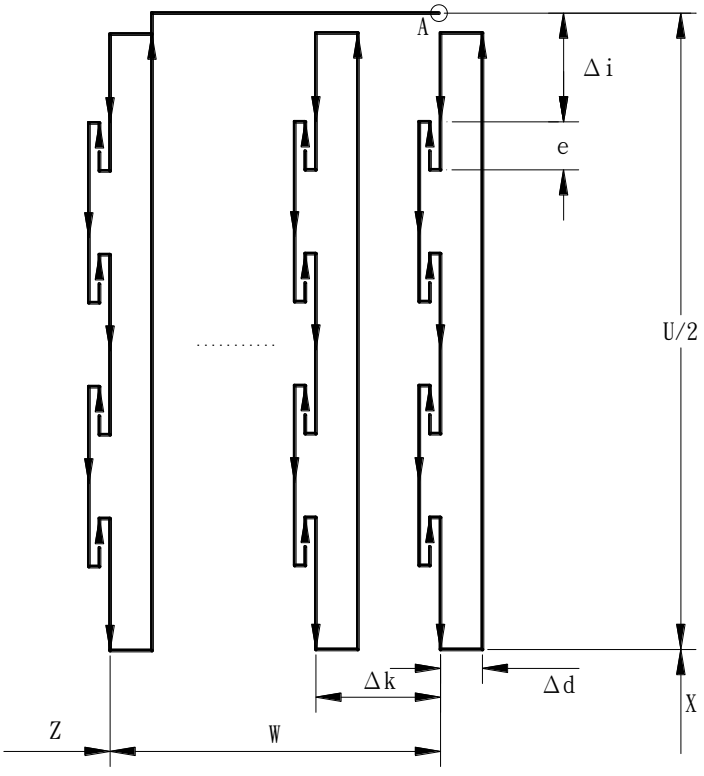
G75 R e P ;

G75 X(U) Z(W) P i Q k R d F ;

Argument:

R e	:	Return amount in X axis direction. This value can be specified by the parameter (P1086).
P	:	Dwell time at the cutting bottom.
X	:	The coordinate of X axis.
Z	:	The coordinate of Z axis.
U	:	Incremental amount in X axis direction.
W	:	Incremental amount in Z axis direction.
P i	:	Depth of cut in X axisdirection. No decimal poiint.
Q k	:	Movement amount of Z axisdirection. No decimal poiint.
R d	:	Relief amount of the tool at the ccutting bottom in Z axis direction.
F	:	Feedrate.

Illustration:



G76: Multiple thread cutting cycle

command format 1:

G76 P m r a Q___ R___;

G76 X(U)___ Z(W)___ R i P k Q d F___;

command format 2:

G76 P m r a Q___ R___;

G76 X(U)___ Z(W)___ R i P k Q d E___;

Argument:

P m r a : m: Repetitive count in finishing.

r: Chamfering amount.

a: Agnle of tool tip.

m and r can be specified by the parameter(P1087).

a can be specified by the parameter (P1088).

a can be set to $0°$ for acme thread or square thread.

$(\Delta d\sqrt{n} - \Delta d\sqrt{n-1})$: Depth of cut in nth cut.

Δd : Depth of cut in 1st cut.

Q___ : Minumum cutting depth $(\Delta d\sqrt{n} - \Delta d\sqrt{n-1}) < Q$. This value can be specified by the parameter (P1089).

R___ : Finishing allowance (specified by the radius value). This value can be specified by the parameter (P1090).

X(U)___ : Distance from A to C in X axis direction.

Z(W)___ : Distance from C to D in Z axis direction.

R i : Difference of thread radius.

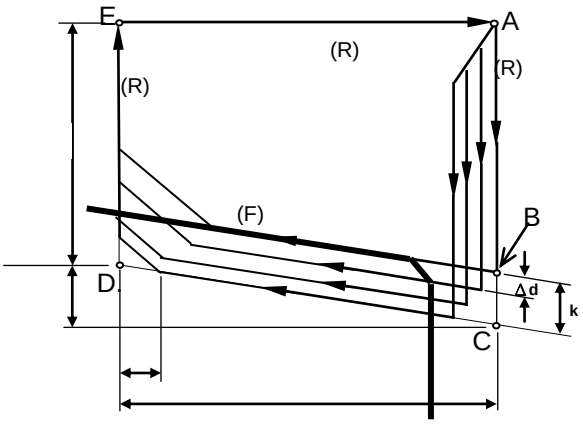
P k : Height of thread. No decimal point.

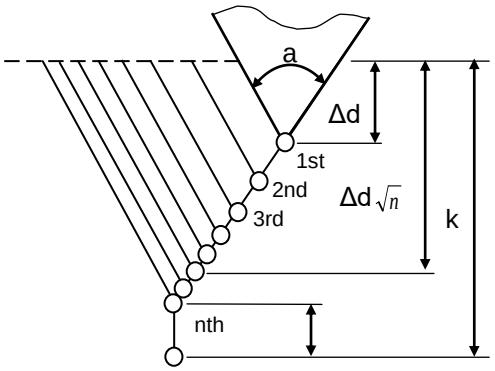
Q d : Depth of cut in 1st cut. No decimal point.

F___ : Lead of thread. (mm/rev)

E___ : Inch thread. (threads/inch)

Illustration:





G77: Outer diameter/internal diameter cutting cycle

Format:

G77 X(U)___ Z(W)___ R___ F___;

Argument:

X___ Z___ : X=U+(The coordinate of X axis at start point)

Z=W+(The coordinate of Z axis at start point)

U___W___R___ : Definitions are shown in the figure below.

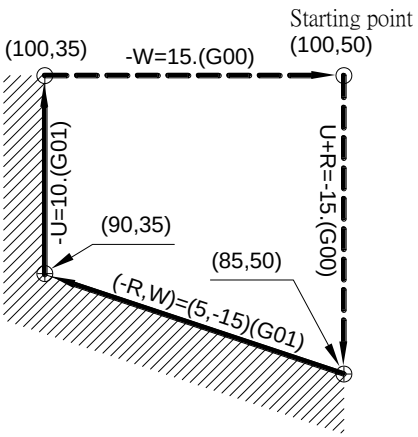
F___ : Feedrate.

Illustration:

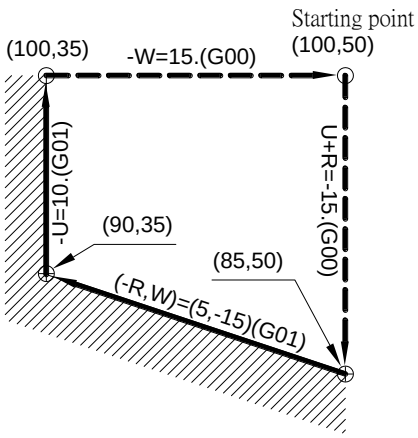
The following two examples generat the same cutting path,

U= -10. W= -15. R= -5.

G77 U-10. W-15. R-5. F50;



G92 G90 X100. Z50.;
G77 X90, Z35. R-5. F50;



G78: Taper thread cutting cycle

Format:

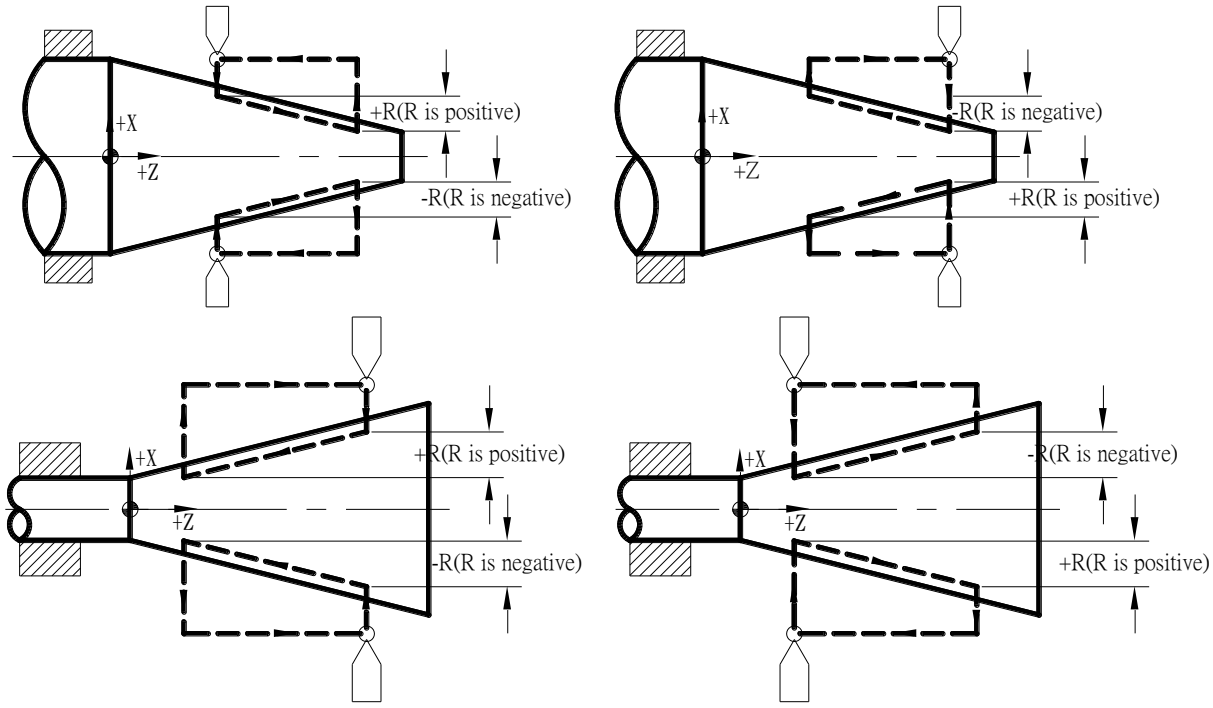
G78 X(U)__ Z(W)__ R__ H__ [F__ / E__];

Argument:

X(U)___	:	The coordinate of X axis at end point.
Z(W)___	:	The coordinate of Z axis at end point .
R___	:	Increment / decrement taper. (mm)
H___	:	The thread line number of multiple thread.
F___	:	Lead of thread. (mm/rev)
E___	:	Inch thread. (threads/inch)

Illustration:

Positive or negative of R value



Example:

Single-line thread

T0707; ---------- call for #7 tool

G0 X6.; ---------- X axis rapidly traverses to cutting start position

Z-5.; ----------Z axis rapidly traverses to cutting start position

G78 X4.5 Z-15. F1; ---------- cutting speed of 1st cut of thread cutting, pitch 1 mm

X4.25; ---------- thread cutting 2nd cut

X4.04; ----------thread cutting 3rd cut

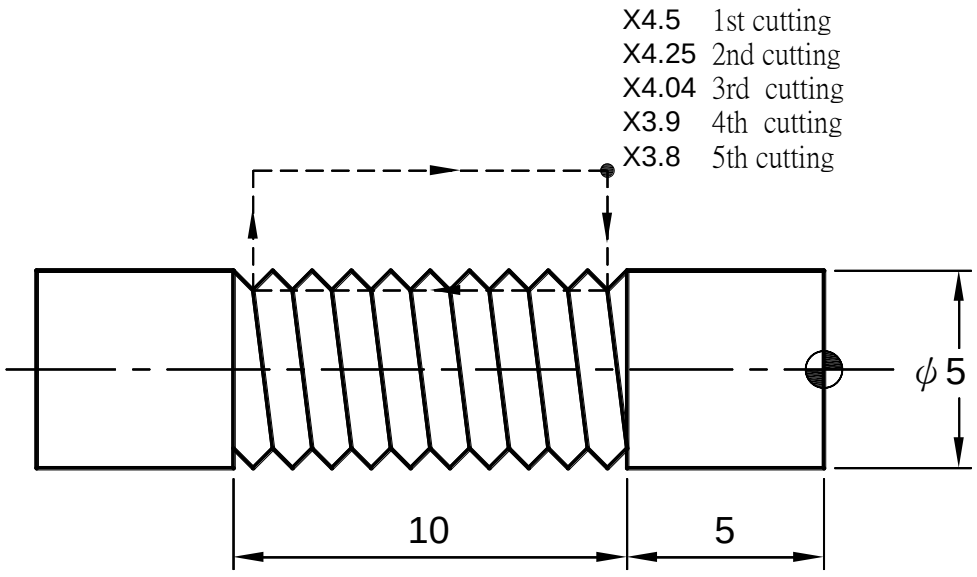
X3.9;----------thread cutting 4th cut

X3.8;----------thread cutting 5th cut

G0 X20;----------X axis rapidly traverses

Z30;----------Z axis rapidly traverses

M30; ---------- program ends



Multi-line thread

T0707; ---------- call for #7 tool

G0 X6.; ---------- X axis rapidly traverses to cutting start position

Z-5.; ----------Z axis rapidly traverses to cutting start position

G78 X4.5 Z-15. H5 F1; ---- cutting speed of 1st cut of 5-line thread cutting, pitch 1 mm, cut 5 times

X4.25; ----------thread cutting 2nd cut, cut 5 times

X4.04; ----------thread cutting 3rd cut, cut 5 times

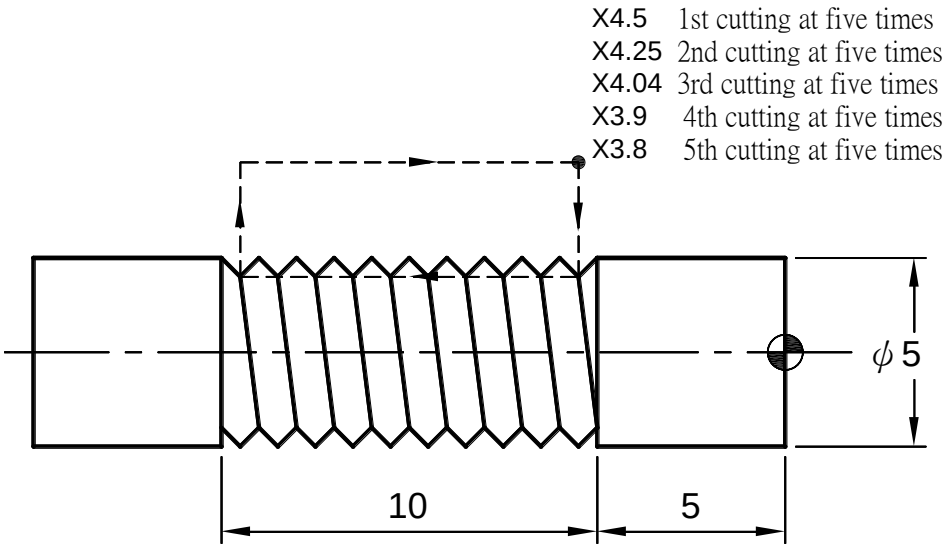
X3.9;----------thread cutting 4th cut, cut 5 times

X3.8;----------thread cutting 5th cut, cut 5 times

G0 X20;----------X axis rapidly traverses

Z30;----------Z axis rapidly traverses

M30; ---------- program ends

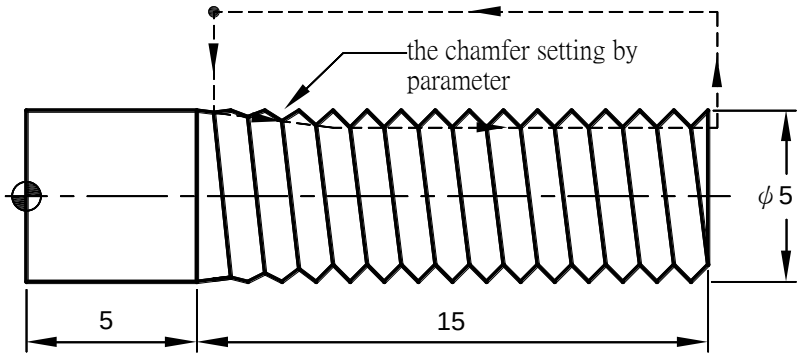


Approach/departure chamfer thread cutting (system parameterP196, P197, P198, 199 setting)

system parameter P196 (when set as 40, cutting chamfer thread number is 4)
P197 (when set as 10, cutting chamfer is 10 degree)

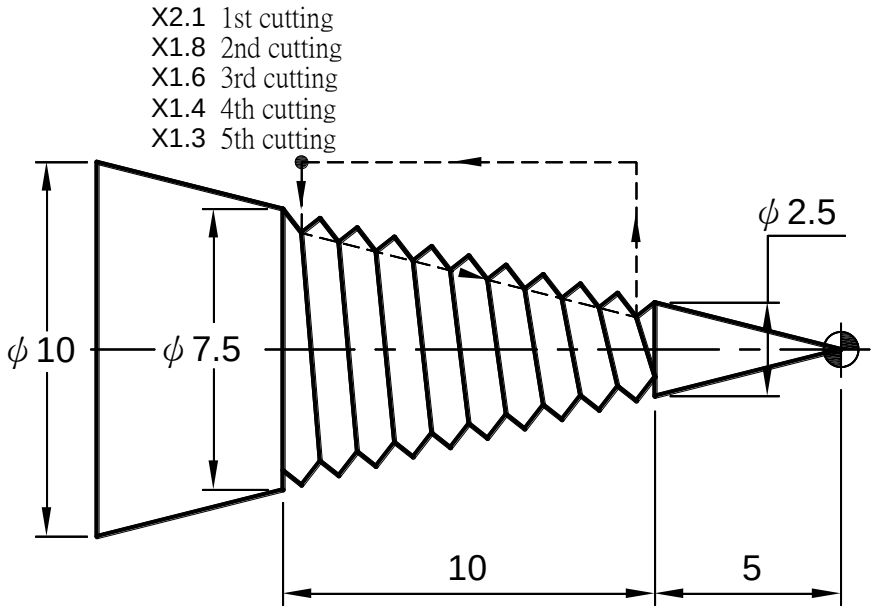
T0707; ---------- call for #7 tool
G0 X6.; ---------- X axis rapidly traverses to cutting start position
Z5.; ----------Z axis rapidly traverses to cutting start position
G78 X4.5 Z20. H5 F1; ----------cutting speed of 1st cut of thread cutting, pitch 1 mm
X4.25;---------- thread cutting 2nd cut
X4.04;----------thread cutting 3rd cut
X3.9;----------thread cutting 4th cut
X3.8;----------thread cutting 5th cut
G0 X20;---------- X axis rapidly traverses
Z30;---------- Z axis rapidly traverses
M30;---------- program ends

X4.5 1st cutting
X4.25 2nd cutting
X4.04 3rd cutting
X3.9 4th cutting
X3.8 5th cutting



Cone thread cutting

```
T0707; ---------- call for #7 tool
G0 X9.5;---------- X axis rapidly traverses to cutting start position
Z-15.;----------Z axis rapidly traverses to cutting start position
G78 X2.1 Z-5. R2.5 F1;---------- 1st cut of thread cutting, pitch 1mm, radius of rake change is 2.5
X1.8;---------- thread cutting 2nd cut
X1.6;----------thread cutting 3rd cut
X1.4;----------thread cutting 4th cut
X1.3;----------thread cutting 5th cut
G0 X20;---------- X axis rapidly traverses
Z30;---------- Z axis rapidly traverses
M30; ---------- program ends
```



G79: End face turning cycle

Format:

G79 X(U)___ Z(W)___ R___ F___;

Argument:

X___ Z___ : X=U+(The coordinate of X axis at start point)

Z=W+(The coordinate of Z axis at start point)

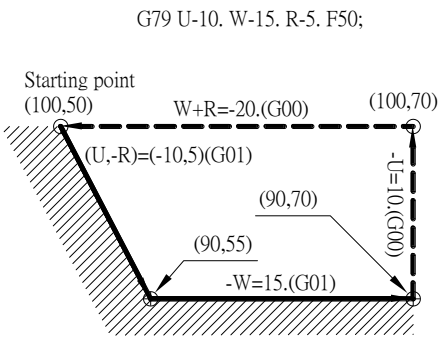
U___ W___ R___ : Definitions are shown in figure below.

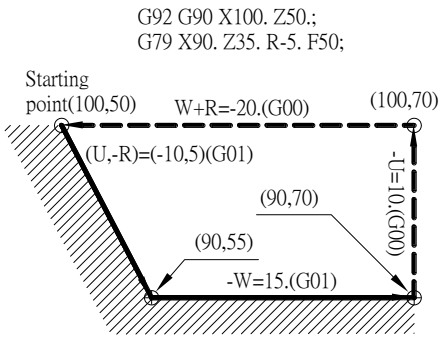
F___ : Feedrate.

Illustration:

The following two example generate the same cutting path.

U= -10. W= -15. R= -5.





G80: Canned cycle for drilling cancel

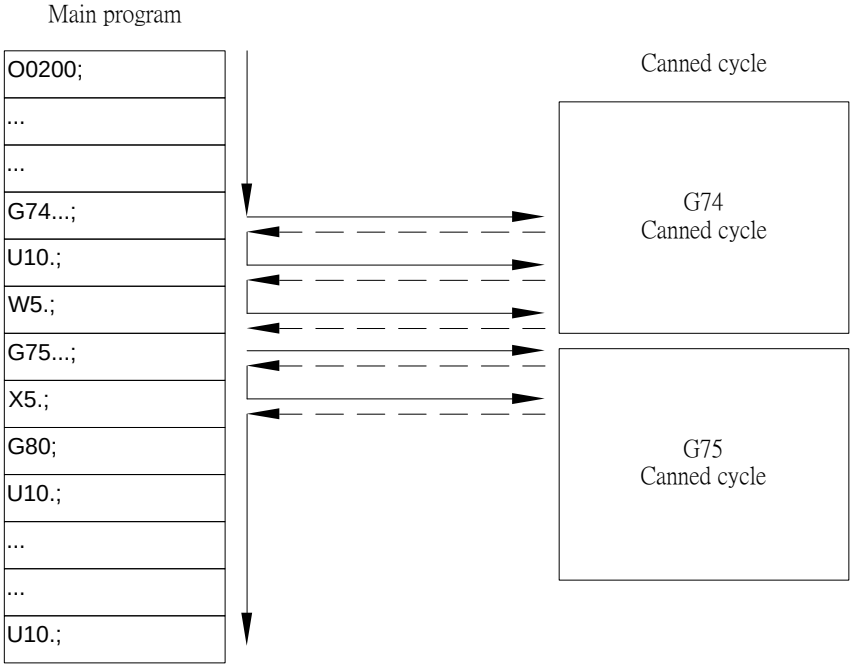
Format:

G80;

Action:

Fixed cycle (G70~G89) once be set will keeps being effective. When command (X/U/Z/W) is met, the fixed cycle will be called to work until it's replaced by another fixed cycle, or G80 command, or a command of group 01 (GROUP01)--G00/G01/G02/G03/G33 appears.

Illustration:



G83: Face drilling cycle

Format:

G83 X(U)__ C(H)__ Z(W)__ R__ Q__ P__

D__ F__ K__ M__ ;

Argument:

X (U)___, C (H)___	:	Hole position data.
Z___	:	For G90, coordinate of Z axis at the hole bottom.
		For G91, distance from R to the hole bottom.
W___	:	Distance from R to the hole bottom.
R___	:	For G90, coordinate of Z axis at R.
		For G91, distance from start point to R.
Q___	:	Depth of cut for each cutting feed. No decimal poiint.
P___	:	Dwell time at the bottom of a hole. Unit: ms
D___	:	Retraction distance.
F___	:	Feedrate.
K___	:	Number of repeats. (Default by 1)
		For G90, drill K times at the same position.
		For G91, drill K holes in X Y direction.
M___	:	M code for spindle clamp.
		M code + 1: M code for spindle unclamp.

Illustration:

Parameter description :

TYPE I : High speed mode. (Parameter P1044 =0)

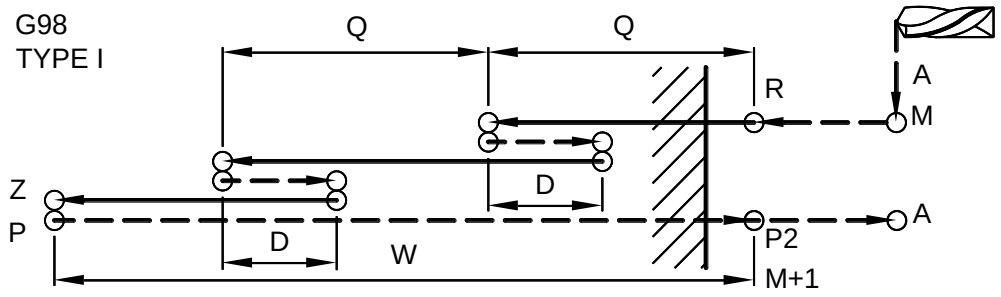
A : Start point.

M : M code for spindle clamp.

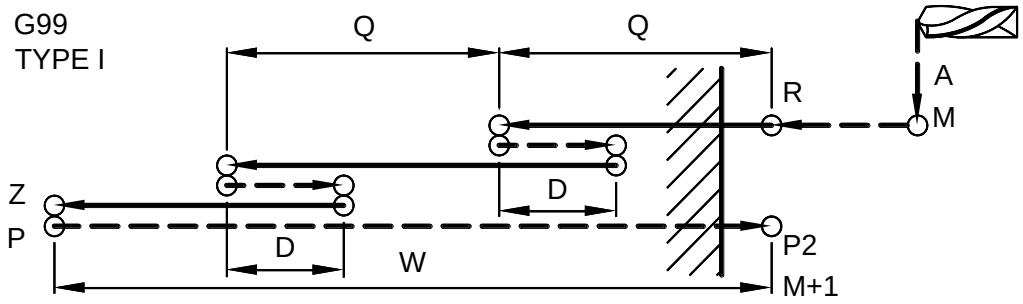
M+1 : M code for spindle unclamp.

D : Retraction distance specified in parameter P0150.

P2 : Dwell time specified in parameter P1045.



For G98, TYPE I (Parameter P1044=0)



For G99, TYPE I (Parameter P1044=0)

Parameter description :

TYPE II: Normal speed mode. (Parameter P1044 =1)

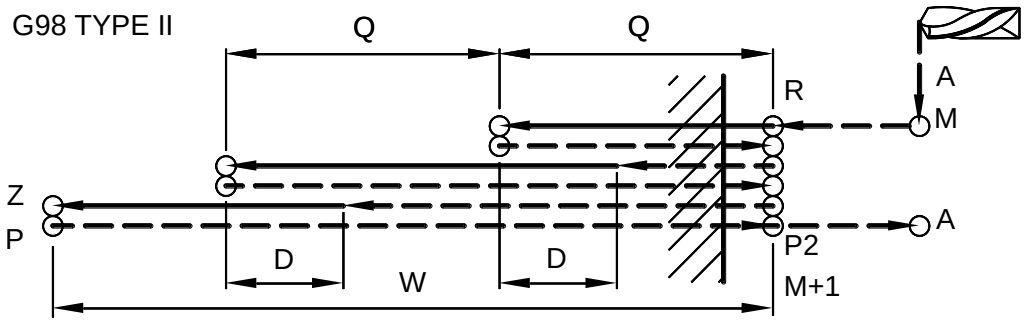
A : Start point.

M : M code for spindle clamp.

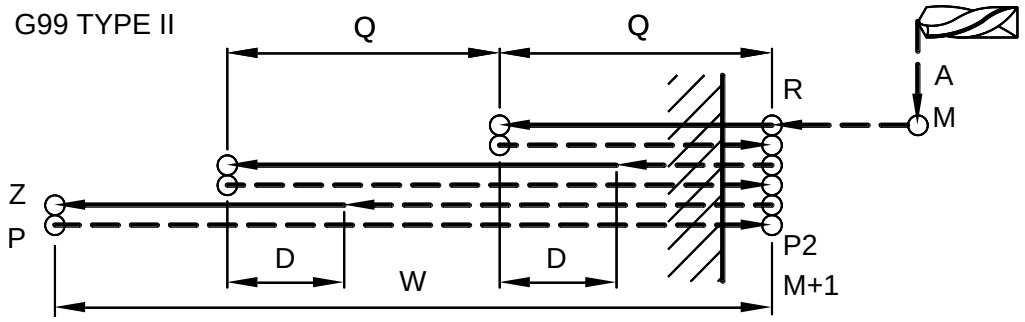
M+1 : M code for spindle unclamp.

D : Retraction distance specified in parameter P0150.

P2 : Dwell time specified in parameter P1045.



For G98, TYPE II (Parameter P1044=1)



For G99, TYPE II (Parameter P1044=1)

Parameter description :

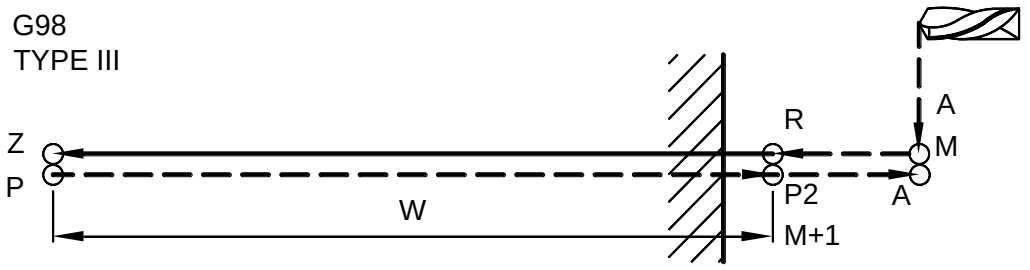
TYPE III : Q__ vacant

A : Start point.

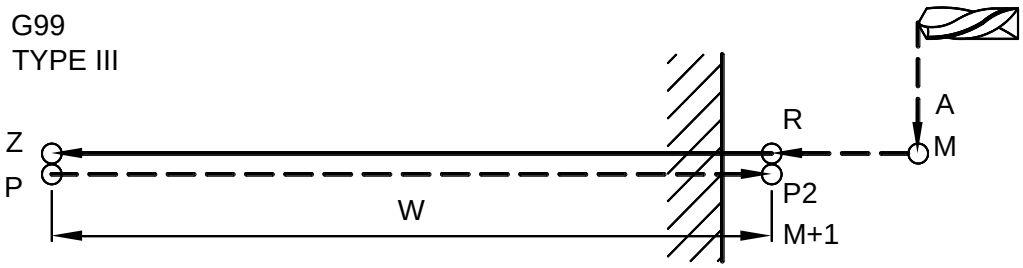
M : M code for spindle clamp.

M+1 : M code for spindle unclamp.

P2 : Dwell time specified in parameter P1045.



For G98, TYPE III (Q__ vacant)



For G99, TYPE III (Q__ vacant)

G84: Face tapping cycle

Format:

G84 X(U)__ C(H)__ Z(W)__ R__ P__ F__

K__ M__ D__ ;

Argument:

X (U)___, C (H)___	:	Hole position data.
Z___	:	For G90, coordinate of Z axis at the hole bottom.
		For G91, distance from R to the hole bottom.
W___	:	Distance from R to the hole bottom.
R___	:	For G90, coordinate of Z axis at R.
		For G91, distance from start point to R.
P___	:	Dwell time at the bottom of a hole. Unit: ms
F___	:	Feedrate.
K___	:	Number of repeats. (Default by 1)
		For G90, tap K times at the same position.
		For G91, tap K holes in X Y direction.
M___	:	M code for spindle clamp.
		M code + 1: M code for spindle unclamp.
D___	:	D_ vacant: Tapping using M03.
		D0 : Tapping using M03.
		D1 : Tapping using M04.

(Remark)D__ has a feature of succession:

Once position direction tapping is used, even though later command does not specify D value, it will still be a positive direction tapping; Once negative direction tapping is used, even though later command does not specify D value, it will still be a negative direction tapping. The feature of succession is inactive after next G80 commnad (cancel fixed cycle).

Example:

Direction of tapping

…

G84…; ---------- Tapping using M03.

G84…; ---------- Tapping using M03.

G84…D1; ---------- Tapping using M04.

G84…; ---------- Tapping using M04. (Inheritance)

G84…; ---------- Tapping using M04. (Inheritance)

G84…D0; ---------- Tapping using M03.

…

Illustration:

Parameter description :

A : Start point.

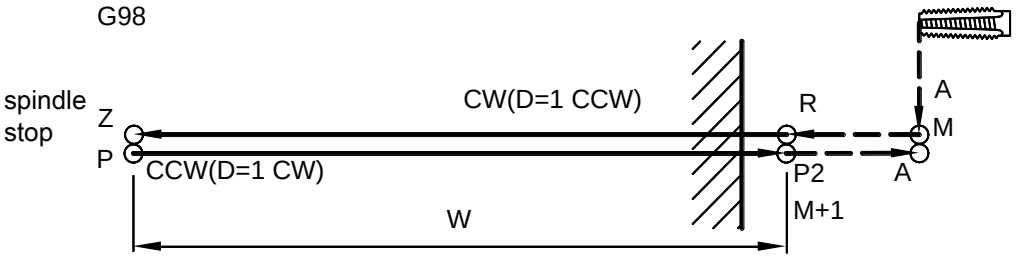
M : M code for spindle clamp.

M+1 : M code for spindle unclamp.

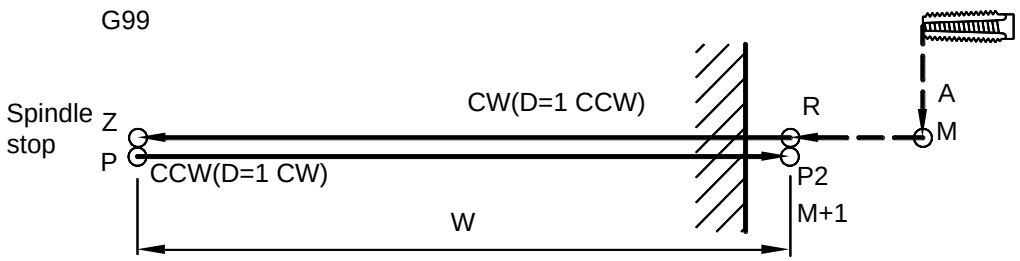
P2 : Dwell time specified in parameter P1045.

CW : M03

CCW : M04



G98 mode



G99 mode

G85: Face boring cycle

Format:

G85 X(U)__ C(H) __ Z(W)__ R__ P__ F__

K__ M__

Argument:

X (U)___, C (H)___	:	Hole position data.
Z___	:	For G90, coordinate of Z axis at the hole bottom.
		For G91, distance from R to the hole bottom.
W___	:	Distance from R to the hole bottom.
R___	:	For G90, coordinate of Z axis at R.
		For G91, distance from start point to R.
P___	:	Dwell time at the bottom of a hole. Unit: ms
F___	:	Feedrate.
K___	:	Number of repeats. (Default by 1)
		For G90, bore K times at the same position.
		For G91, bore K holes in X Y direction.
M___	:	M code for spindle clamp.
		M code + 1: M code for spindle unclamp.

Illustration:

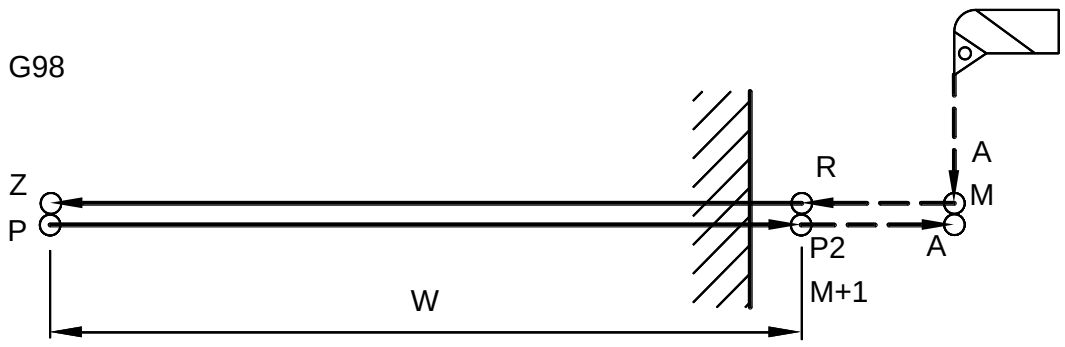
Parameter description :

A : Start point.

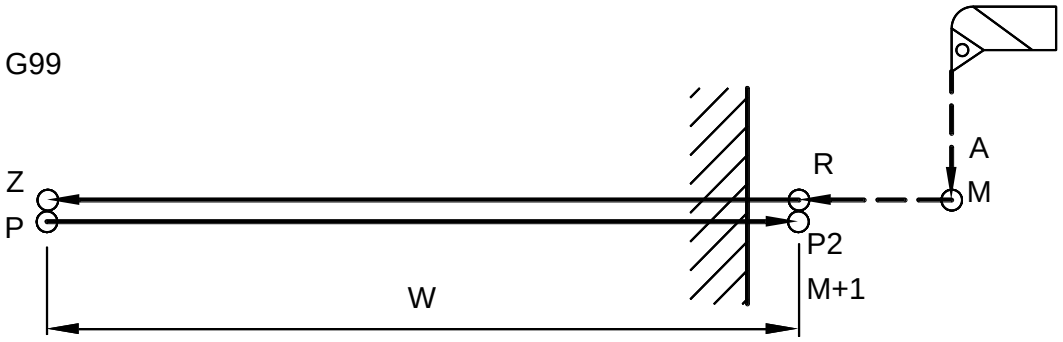
M : M code for spindle clamp.

M+1 : M code for spindle unclamp.

P2 : Dwell time specified in parameter P1045.



G98 mode



G99 mode

G87: Side drilling cycle (X axis)

Format:

G87 Z(W)__ C(H)__ X(U)__ R__ Q__ P__

D__ F__ K__ M__;

Argument:

Z (W)___, C (H)___	:	Hole position data.
X___	:	For G90, coordinate of X axis at the hole bottom.
		For G91, distance from R to the hole bottom.
U___	:	Distance from R to the hole bottom.
R___	:	For G90, coordinate of X axis at R.
		For G91, distance from start point to R.
Q___	:	Depth of cut for each cutting feed. No decimal poiint.
P___	:	Dwell time at the bottom of a hole. Unit: ms
D___	:	Retraction distance.
F___	:	Feedrate.
K___	:	Number of repeats. (Default by 1)
		For G90, drill K times at the same position.
		For G91, drill K holes in Y Z direction.
M___	:	M code for spindle clamp.
		M code + 1: M code for spindle unclamp.

Action:

While using G87, lathe spindle is locked, and tool revolves to cut workpiece.

Illustration:

Parameter description :

TYPE I : High speed mode. (Parameter P1044 =0)

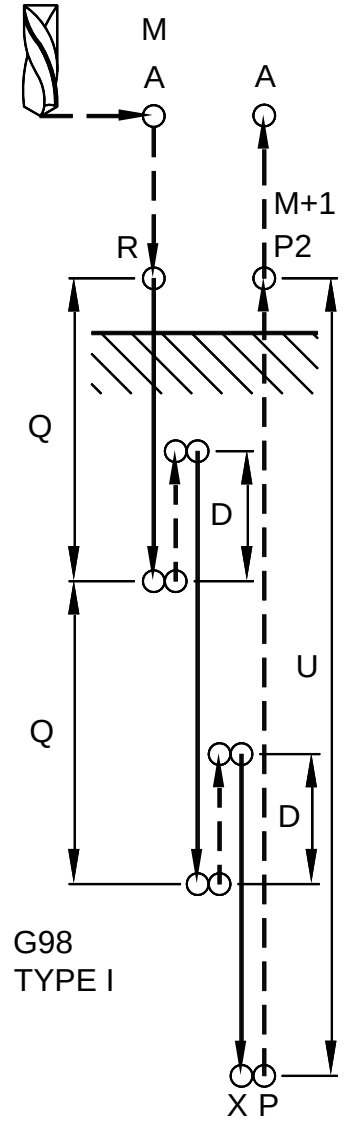
A : Start point.

M : M code for spindle clamp.

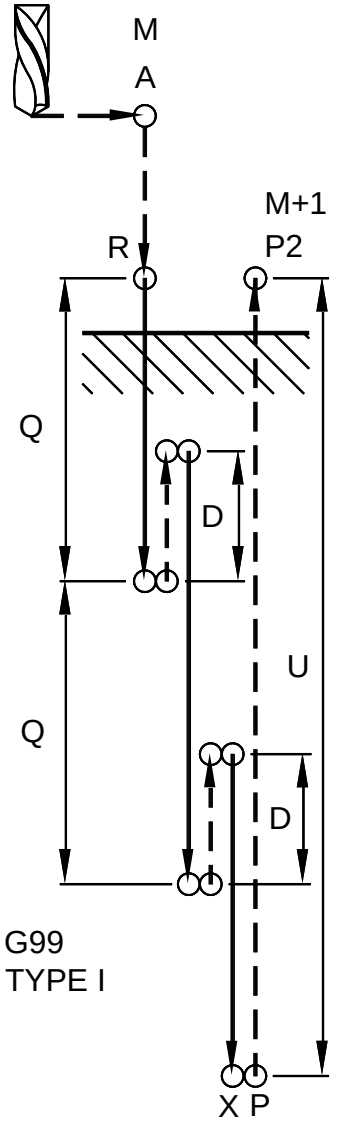
M+1 : M code for spindle unclamp.

D : Retraction distance specified in parameter P0150.

P2 : Dwell time specified in parameter P1045.



For G98, TYPE I (Parameter P1044=0)



For G99, TYPE I (Parameter P1044=0)

Parameter description :

TYPE II: Normal speed mode. (Parameter P1044 =1)

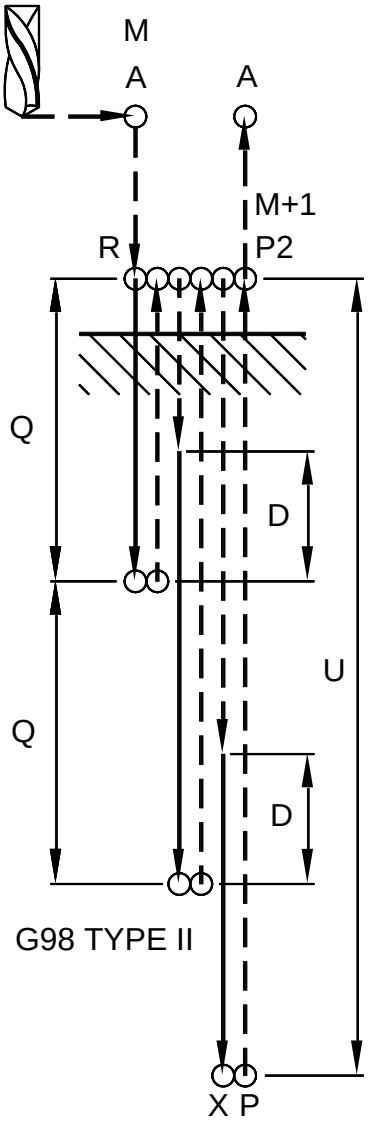
A : Start point.

M : M code for spindle clamp.

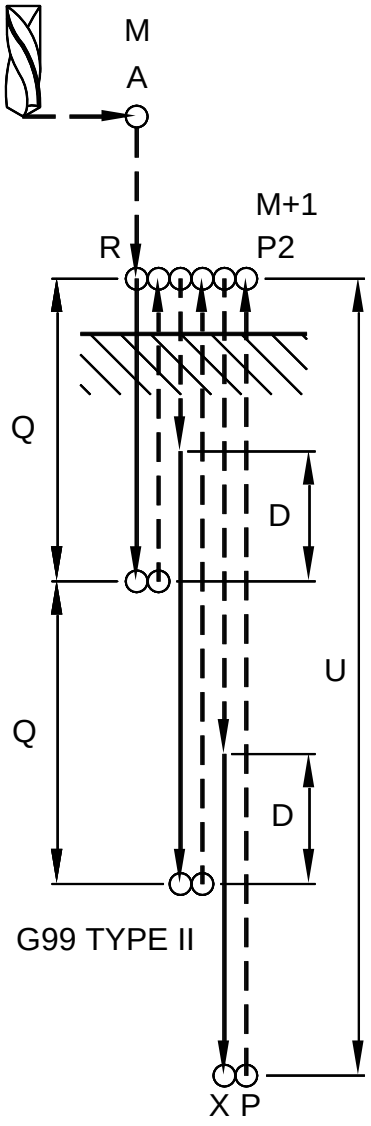
M+1 : M code for spindle unclamp.

D : Retraction distance specified in parameter P0150.

P2 : Dwell time specified in parameter P1045.



For G98, TYPE II (Parameter P1044=1)



For G99, TYPE II (Parameter P1044=1)

Parameter description :

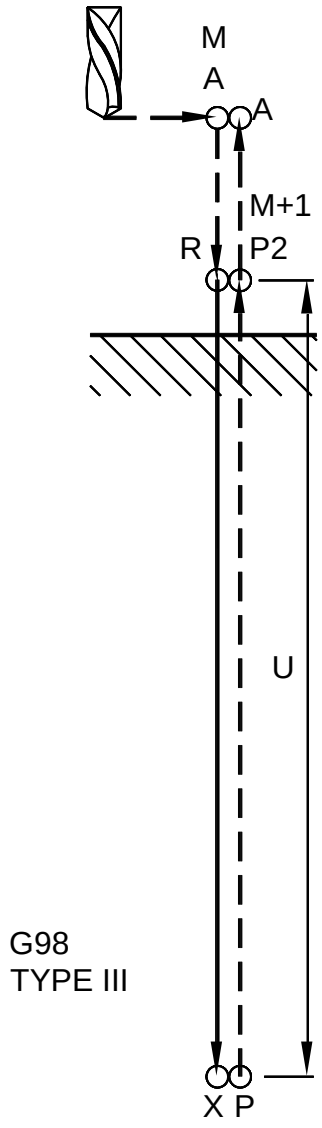
TYPE III : Q__ vacant

A : Start point.

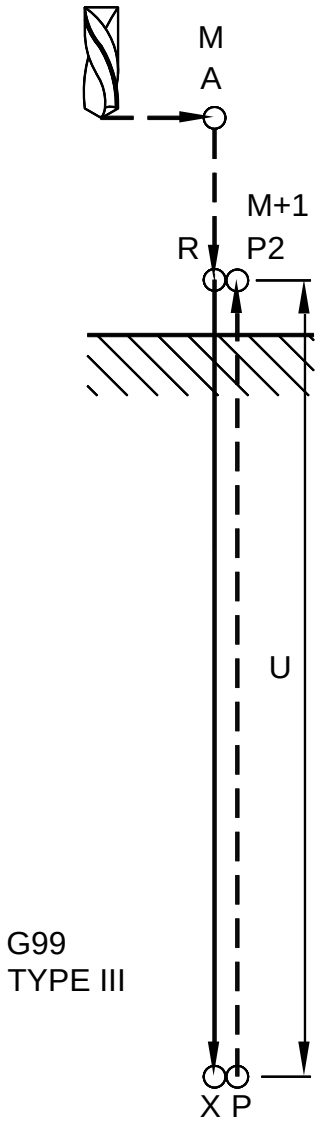
M : M code for spindle clamp.

M+1 : M code for spindle unclamp.

P2 : Dwell time specified in parameter P1045.



For G98, TYPE III (Q__ vacant)



For G99, TYPE III (Q__ vacant)

G88: Side tapping cycle (X axis)

Format:

G88 Z(W)__ C(H)__ X(U)__ R__ P__ F__

K__ M__ D__;

Argument:

Z (W)___, C (H)___	:	Hole position data.
X___	:	For G90, coordinate of X axis at the hole bottom.
		For G91, distance from R to the hole bottom.
U___	:	Distance from R to the hole bottom.
R___	:	For G90, coordinate of X axis at R.
		For G91, distance from start point to R.
P___	:	Dwell time at the bottom of a hole. Unit: ms
F___	:	Feedrate.
K___	:	Number of repeats. (Default by 1)
		For G90, tap K times at the same position.
		For G91, tap K holes in Y Z direction.
M___	:	M code for spindle clamp.
		M code + 1: M code for spindle unclamp.
D___	:	D_ vacant: Tapping using M03.
		D0 : Tapping using M03.
		D1 : Tapping using M04.

(Remark)D__ has a feature of succession:

Once position direction tapping is used, even though later command does not specify D value, it will still be a positive direction tapping; Once negative direction tapping is used, even though later command does not specify D value, it will still be a negative direction tapping. The feature of succession is inactive after next G80 commnad (cancel fixed cycle).

Example:

tappingdirection

…

G88…; ---------- Tapping using M03.

G88…; ---------- Tapping using M03.

G88…D1; ---------- Tapping using M04.

G88…; ---------- Tapping using M04. (Inheritance)

G88…; ---------- Tapping using M04. (Inheritance)

G88…D0; ---------- Tapping using M03.

…

Illustration:

Parameter description :

A : Start point.

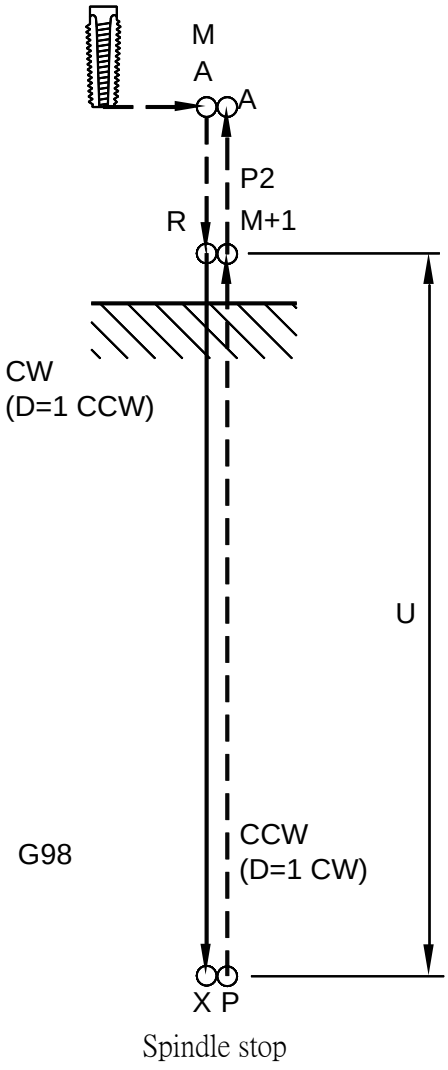
M : M code for spindle clamp.

M+1 : M code for spindle unclamp.

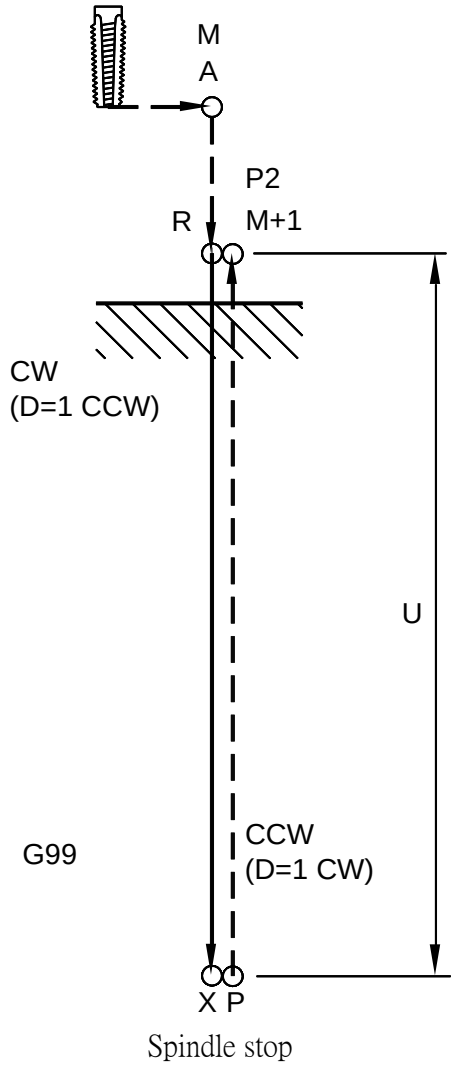
P2 : Dwell time specified in parameter P1045.

CW : M03

CCW : M04



G98 mode



G99 mode

G89: Side boring cycle (X axis)

Format:

G89 Z(W)__ C(H)__ X(U)__ R__ P__ F__

K__ M__ ;

Argument:

Z (W)___, C (H)___	:	Hole position data.
X___	:	For G90, coordinate of X axis at the hole bottom.
		For G91, distance from R to the hole bottom.
U___	:	Distance from R to the hole bottom.
R___	:	For G90, coordinate of X axis at R.
		For G91, distance from start point to R.
P___	:	Dwell time at the bottom of a hole. Unit: ms
F___	:	Feedrate.
K___	:	Number of repeats. (Default by 1)
		For G90, bore K times at the same position.
		For G91, bore K holes in Y Z direction.
M___	:	M code for spindle clamp.
		M code + 1: M code for spindle unclamp.

Illustration:

Parameter description :

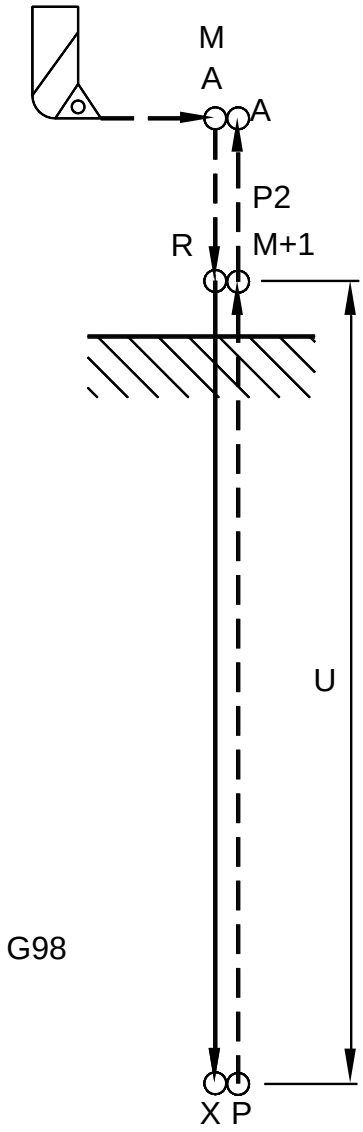
A : Start point.

M : M code for spindle clamp.

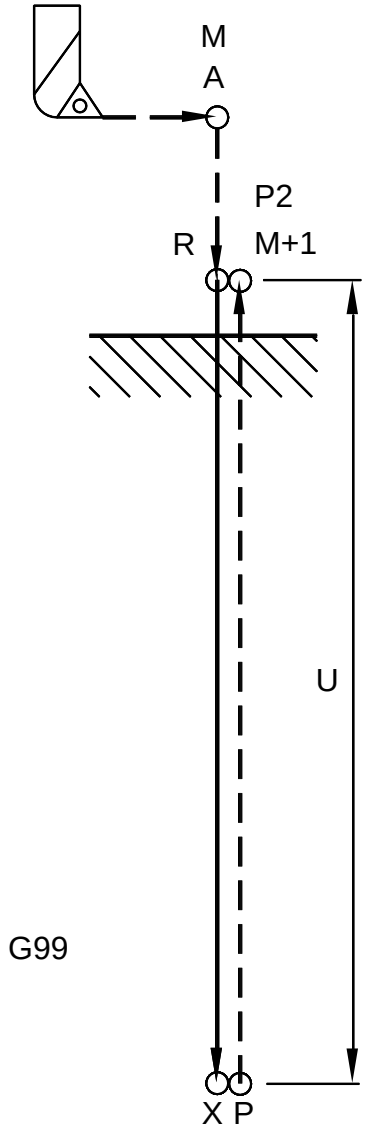
M+1 : M code for spindle unclamp.

D : Retraction distance specified in parameter P0150.

P2 : Dwell time specified in parameter P1045.



G98 mode



G99 mode

G187: Side drilling cycle (Y axis)

Format:

```
G187 X(U)__ Z(W)__ C(H)__ Y(V)__ R__ Q__ P__ D__ F__
     K__ M__;
```

Argument:

X (U)___, Z (W)___,	:	Hole position data.
C (H)___		
Y___	:	For G90, coordinate of Y axis at the hole bottom.
		For G91, distance from R to the hole bottom.
V___	:	Distance from R to the hole bottom.
R___	:	For G90, coordinate of Y axis at R.
		For G91, distance from start point to R.
Q___	:	Depth of cut for each cutting feed. No decimal poiint.
P___	:	Dwell time at the bottom of a hole. Unit: ms
D___	:	Retraction distance.
F___	:	Feedrate.
K___	:	Number of repeats. (Default by 1)
		For G90, drill K times at the same position.
		For G91, drill K holes in XZ direction.
M___	:	M code for spindle clamp.
		M code + 1: M code for spindle unclamp.

Parameter description :

TYPE I : High speed mode. (Parameter P1044 =0)

A : Start point.

M : M code for spindle clamp.

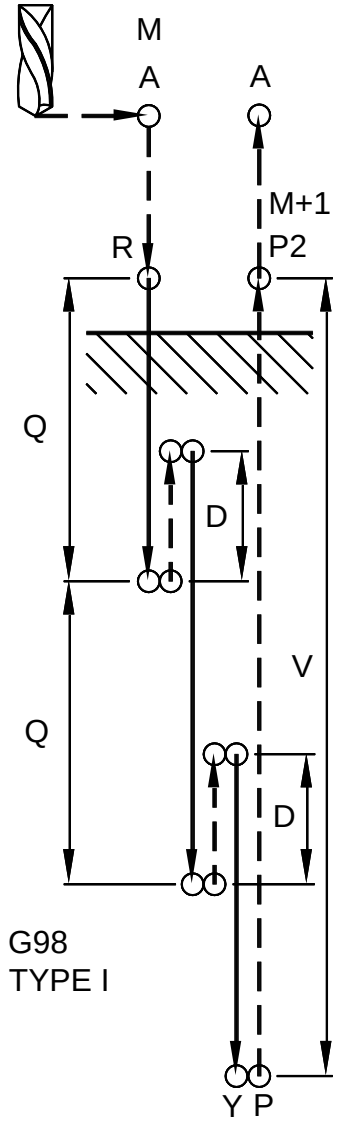
M+1 : M code for spindle unclamp.

D : Retraction distance specified in parameter P0150.

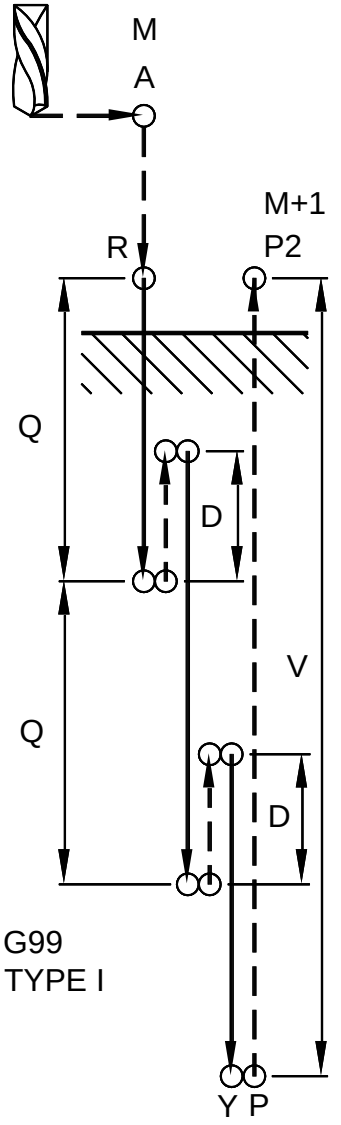
P2 : Dwell time specified in parameter P1045.

Illustration:

Action: When using G187, lathe spindle is locked, and tool revolves to cut workpiece.



For G98, TYPE I (Parameter P1044=0)



For G99, TYPE I (Parameter P1044=0)

Parameter description :

TYPE II: Normal speed mode. (Parameter P1044 =1)

A : Start point.

M : M code for spindle clamp.

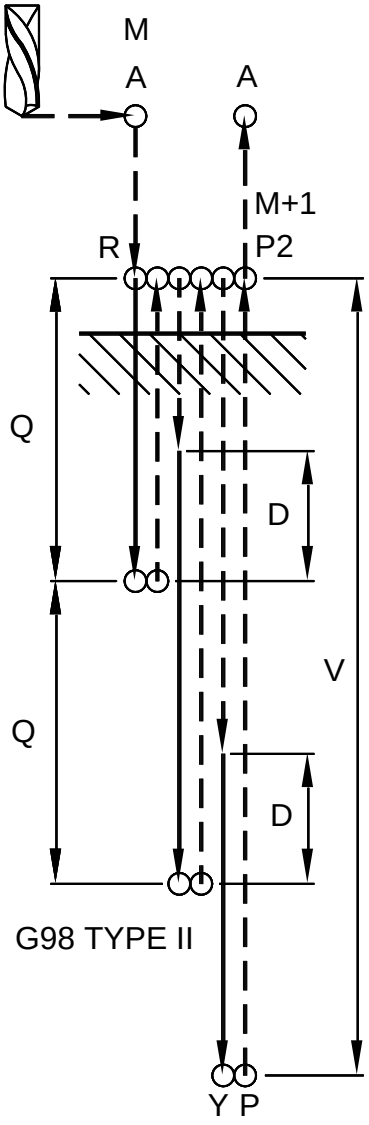
M+1 : M code for spindle unclamp.

D : Retraction distance specified in parameter P0150.

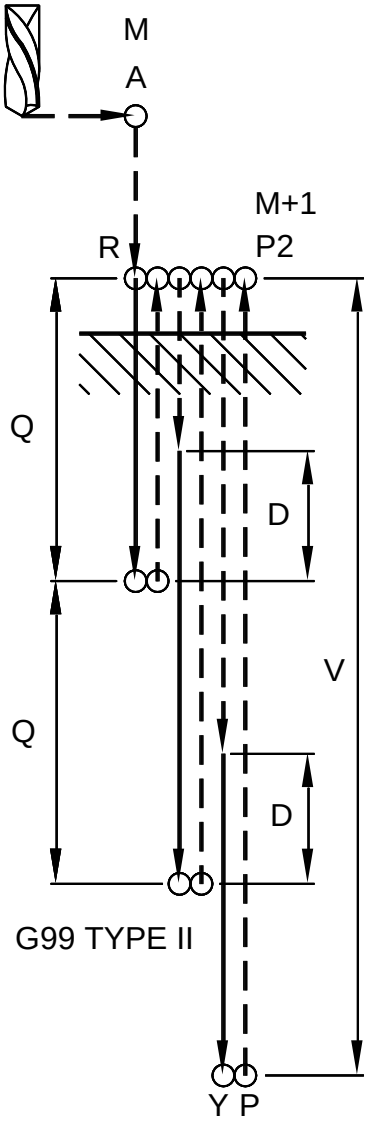
P2 : Dwell time specified in parameter P1045.

Illustration:

Action: When using G187, lathe spindle is locked, and tool revolves to cut workpiece.



For G98, TYPE II (Parameter P1044=1)



For G99, TYPE II (Parameter P1044=1)

Parameter description :

TYPE III : Q__ vacant

A : Start point.

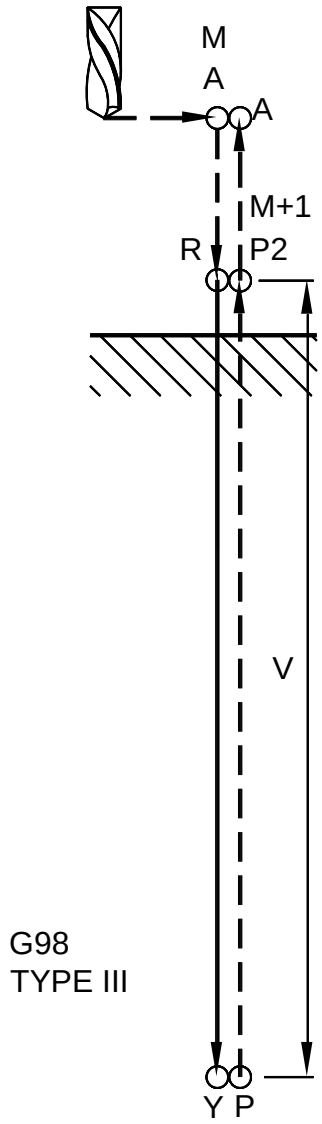
M : M code for spindle clamp.

M+1 : M code for spindle unclamp.

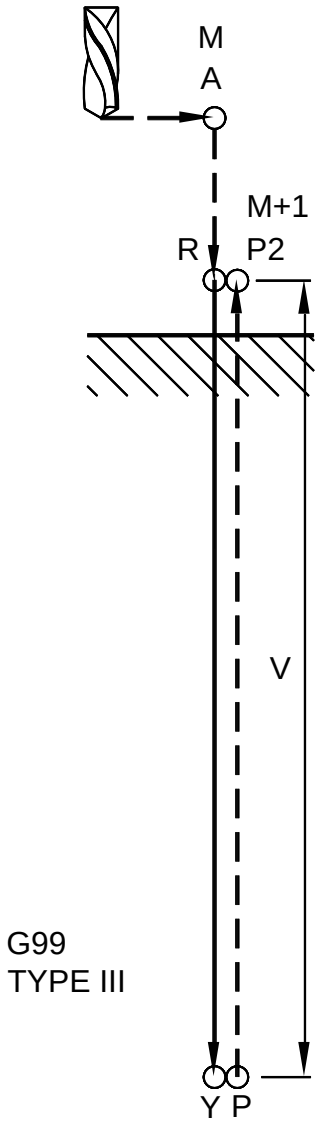
P2 : Dwell time specified in parameter P1045.

Illustration:

Action: When using G187, lathe spindle is locked, and tool revolves to cut workpiece.



For G98, TYPE III (Q__ vacant)



For G99, TYPE III (Q__ vacant)

G188: Side tapping cycle (Y axis)

Format:

```
G188 X(U)__ Z(W)__ C(H)__ Y(V)__ R__ P__ F__ K__ M__
     D__;
```

Argument:

X (U)___, Z (W)___,	:	Hole position data.
C (H)___		
Y___	:	For G90, coordinate of Y axis at the hole bottom.
		For G91, distance from R to the hole bottom.
V___	:	Distance from R to the hole bottom.
R___	:	For G90, coordinate of Y axis at R.
		For G91, distance from start point to R.
P___	:	Dwell time at the bottom of a hole. Unit: ms
F___	:	Feedrate.
K___	:	Number of repeats. (Default by 1)
		For G90, tap K times at the same position.
		For G91, tap K holes in XZ direction.
M___	:	M code for spindle clamp.
		M code + 1: M code for spindle unclamp.
D___	:	D_ vacant: Tapping using M03.
		D0 : Tapping using M03.
		D1 : Tapping using M04.

Parameter description :

A : Start point.

M : M code for spindle clamp.

M+1 : M code for spindle unclamp.

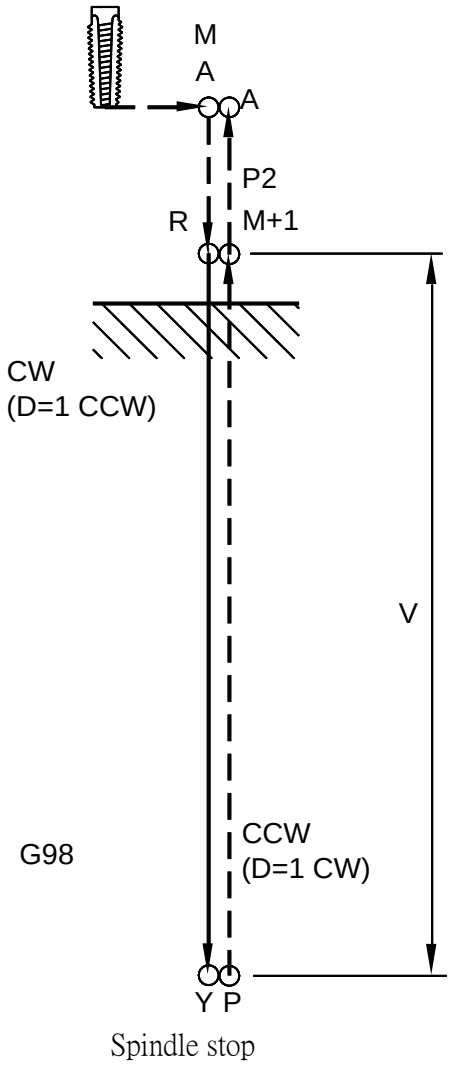
P2 : Dwell time specified in parameter P1045.

CW : M03

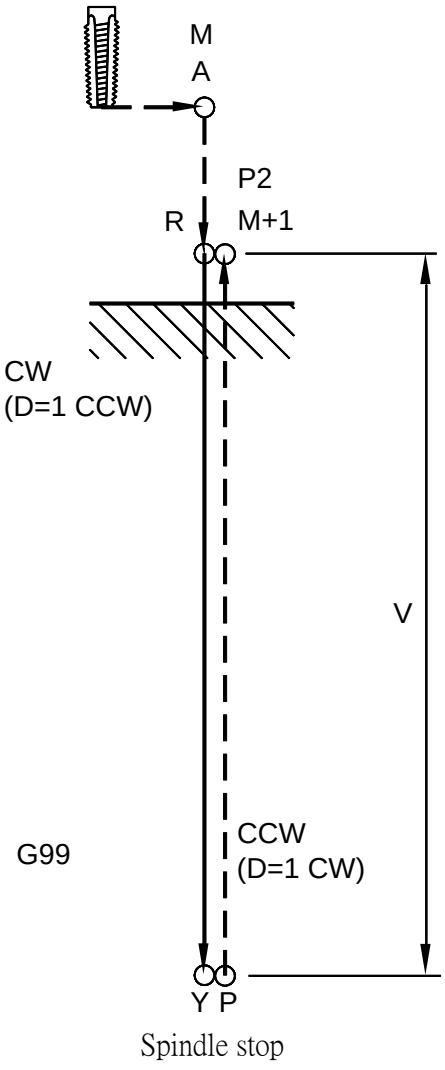
CCW : M04

Illustration:

Action: When using G188, lathe spindle is locked, and tool revolves to cut workpiece.



G98 mode



G99 mode

G189: Side boring cycle (Y axis)

Format:

G189 X(U)__ Z(W)__ C(H)__ Y(V)__ R__ P__ F__ K__ M__;

Argument:

X (U)___, Z (W)___,	:	Hole position data.
C (H)___		
Y___	:	For G90, coordinate of Y axis at the hole bottom.
		For G91, distance from R to the hole bottom.
V___	:	Distance from R to the hole bottom.
R___	:	For G90, coordinate of Y axis at R.
		For G91, distance from start point to R.
P___	:	Dwell time at the bottom of a hole. Unit: ms
F___	:	Feedrate.
K___	:	Number of repeats. (Default by 1)
		For G90, bore K times at the same position.
		For G91, bore K holes in XZ direction.
M___	:	M code for spindle clamp.
		M code + 1: M code for spindle unclamp.

Parameter description :

A : Start point.

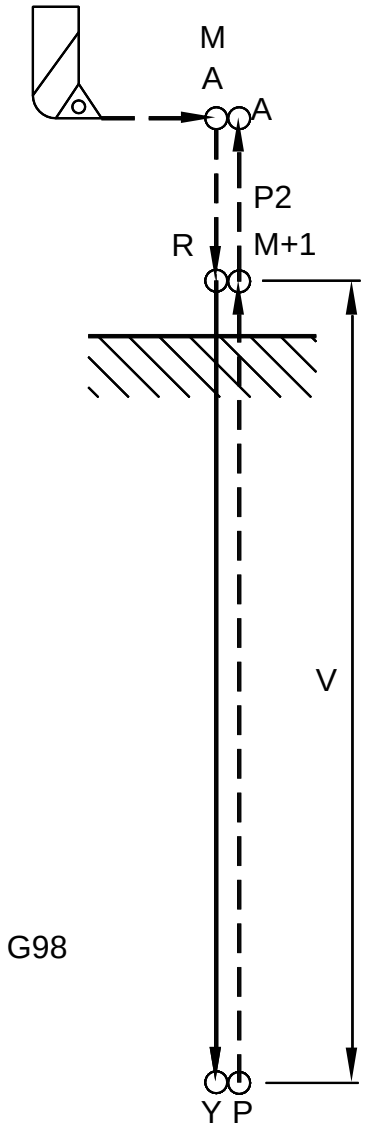
M : M code for spindle clamp.

M+1 : M code for spindle unclamp.

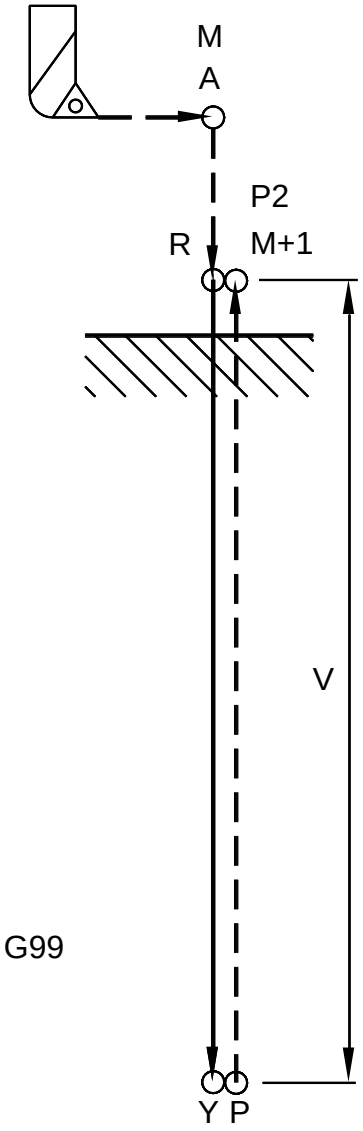
P2 : Dwell time specified in parameter P0145.

Illustration:

Action: When using G189, lathe spindle is locked, and tool revolves to cut workpiece.



G98 mode



G99 mode

G90, G91: Absolute / Incremental programming

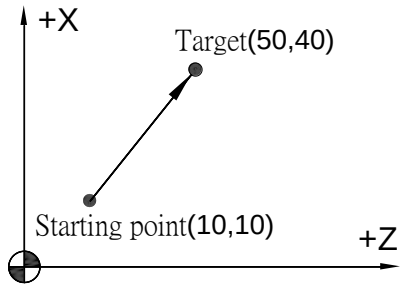
Format:

G90;
G91;

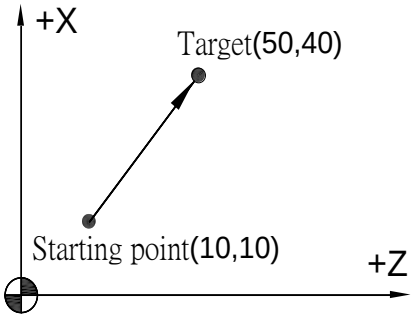
Argument:

G90	:	Absolute command.
G91	:	Incremental command.
X__ Z__	:	In the absolute command, coordinate value of the end position is programmed.
U__ W__	:	In the incremental command, move distance of the position itself is programmed.

Example 1:

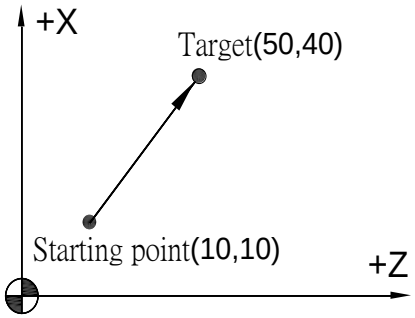


G90 X50. Z40.;

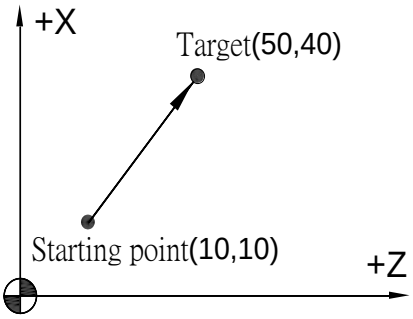


G91 X40. Z30.;

Example 2:



G90 U40. W30.;



G91 U40. W30.;

G92: Coordinate system setting or max. spindle speed setting

Format:

G92 X__ Z__ S__;

Argument:

X__ Z__ : Coordinate value of current tool position.

S__ : Maximum spindle speed. If S_ is set, it will replace to the value of parameter P1096.

Action:

G92 command is used to set current position to specified coordinate values, and the offset betweenthe new coordinate value and old coordinate value will effect all coordinate systems G54 ~ G59. Once G92 is set, motion commands in absolute mode will refre to the coordinate system after the offset to proceed calculation. In order to cancel coordinate offset made by G92, redo manual original position return process.

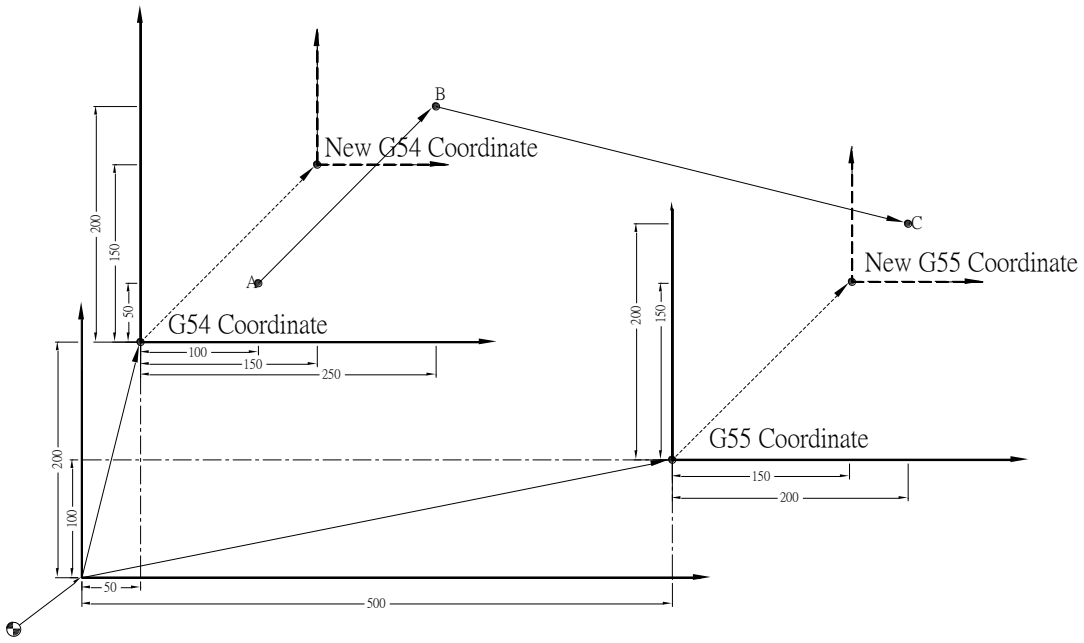
Example:

G90 G54 G00 X100. Y50.;---------- Point A

G92 X-50. Y-100.;---------- Coordinate offset (150,150)

X100. Y50.;----------A→B

G55 G00 X50. Y50.;---------- B→C



G94, G95: Feed per minute (mm/min), Feed per revolution (mm/rev)

Format:

G94 F___;
G95 F___;

Argument:

G94 : Feed per minute. Unit: mm/min or inch/min.

G95 : Feed per revolution. Unit: mm/rev or inch/rev.

Action:

Feedrate of G01/G02/G03 are commanded with numbers after F code.

G96, G97: Constant surface speed control ON / OFF

Format:

G96 S___;
G97 S___;

Argument:

S___ : For G96, surface speed. Unit: m/min or feet/min.
For G97, spindle speed. Unit: RPM.

Action:

G96: Constant surface speed control ON.

G97: Constant surface speed control OFF.

When lathe a workpiece, and surface speed has to be constant, use G96 to specify the surface speed. Spindle speed will change according to the absolute coordinate of X axis (i.e. spindle speed changes along with the distance between tool tip and the rotation center of workpiecere, Remark), and the relationship between surface speed, spindle speed and X axis coordinate is as the following function:

$$V = \pi \frac{D}{1000} N$$

V: surface speed, specified by G96 S__, unit: m/minor feet/min
D: distance between tool tip to workpiece rerotation center (X axis absolute coordinate)
N: spindle speed, unit: rpm

Typically when use constant surface cutting speed function, it is necessary to use G92 S___ to limit spindle speed at the same time lest spindle speed is too high and make the workpiece fall off.

(Remark) Whenuse constant surface cutting speed function, it is necessary to build up a coordinate system based on workpiece's rotation center, i.e. when move the toolto workpiece rotation center, its X axis absolute coordinate has to be 0.

G98, G99: Intial point return / R point return

Format:

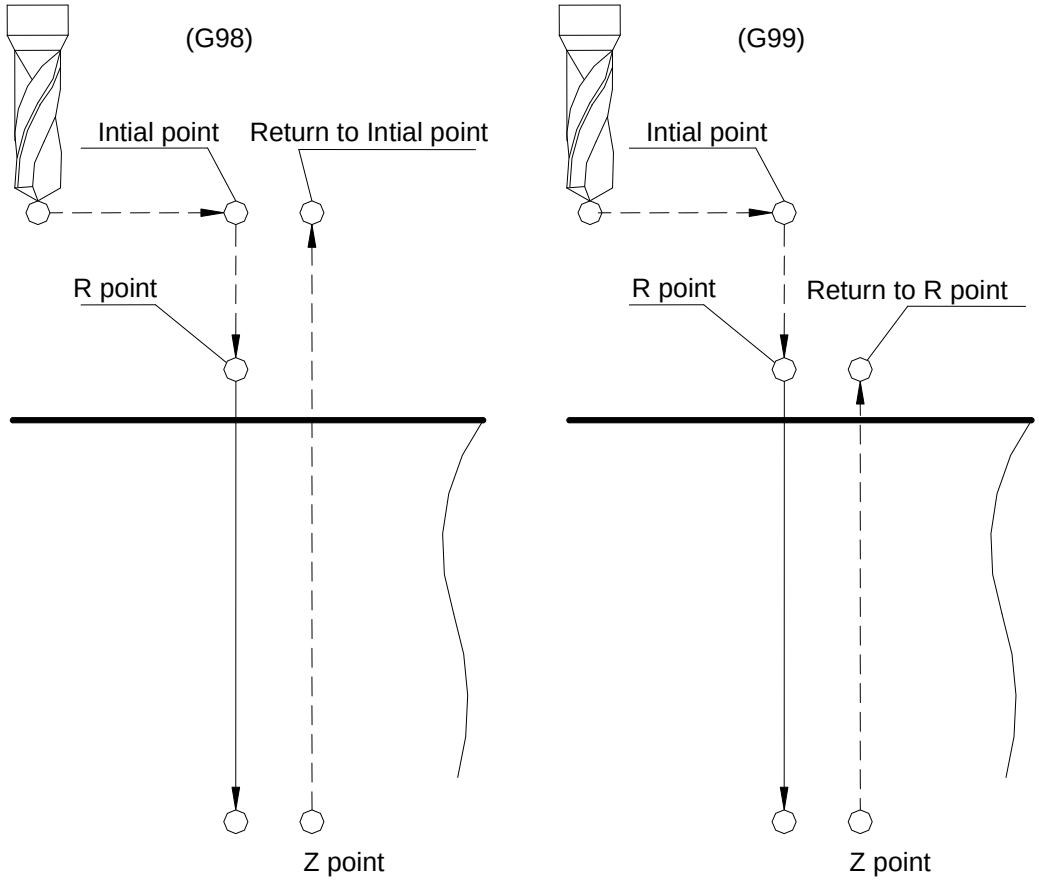
```
G98;
G99;
```

Argument:

G98 : Intial point return in canned cycle.

G99 : R point return in canned cycle.

Illustration:



4 Auxiliary function (M code)

Auxiliary functions are used to control the ON and OFF machines functions. The command's format is that M CODE followed by a number of 1 or 2 digits. The following introduces M CODEs which are stored in NC with fixed functions, and not designed by machine manufacturers. This type of M CODE includes M00, M01, M02, M30, M98, and M99. I.e. these functions have nothing to do with LADDER program compilation.

（1） M00: Program stop

When CNC executes M00 command, it pauses executing the program, and then users can proceed size inspection and compensation adjustment; please click 〈CYCLE START〉again to restart the program.

（2） M01: Option stop

M01's function is like that of M00; but M01 is controlled by〈Selective Pause〉button on the panel: when indication light is ON, program pauses when M01 is executed; when indication light is OFF, M01 is inactive.

（3） M02: End of program

When CNC executes this command, manufacturing mode ends. Click 〈RESET〉button and then click〈CYCLE START〉, if want to rerun the program.

（4） M30: Program end and return to the start of program.

Program ends, function is the same as M02, but cursor on program check page returns to program's initial position.

（5） M98: Calling of subprogram

Format:

M98 P__ L__;

Argument:

P__ : Number of the program to call. P9010 call file name O9010. If P_ is inputted vacant, controller will have an alarm"INT 3111 Lack of file name".

L__ : Number of repeats.

Refer to instruction of 5.3 for the difference between macro program call and general subprogram call.

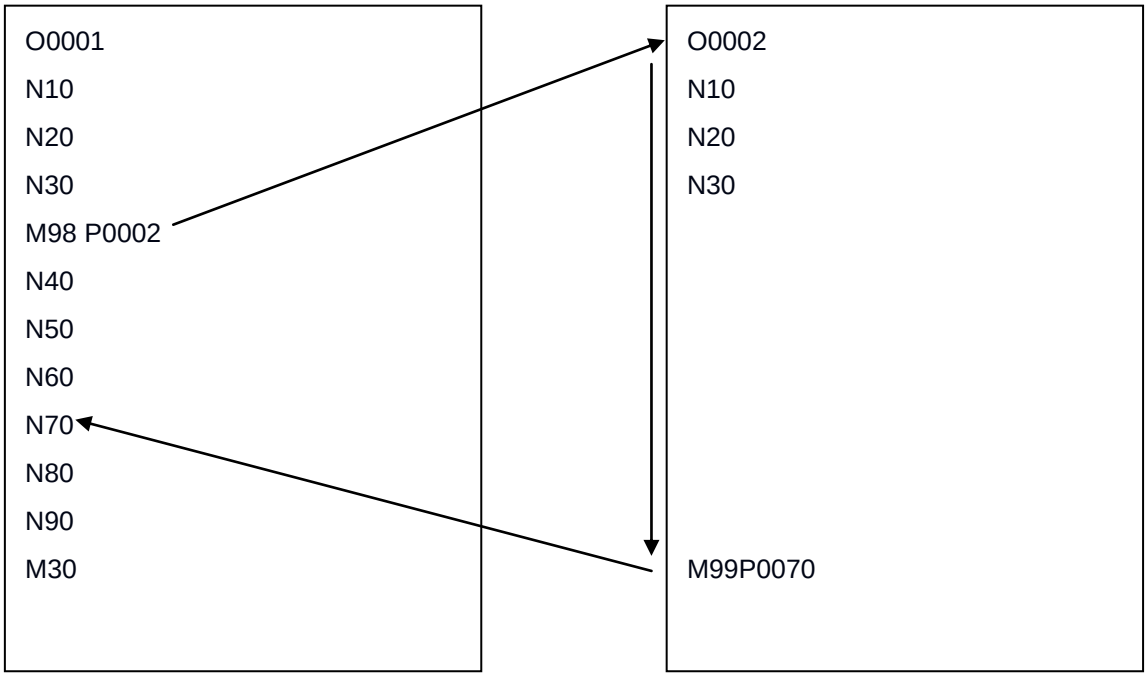
（6） M99: End of subgrogram

1. When NC executes M99 in main program, it returns to the initial of program and reruns the program. In subprograms, it has to use M99 as the end of program, and let program execution returns to main program.
2. **Format:** M99 P__ ;

   P__: specified sequence number of the line to return to

   If use M99 P__ in main program, system looks for M99-specified line, and execute since there after.

If use M99 P__ in the end of subprogram, when subprogram ends and system returns to main program, system continues executing from the line specified by M99.



The following is a chart of M CODEs. Those M codes in the chart, excluding M00, M01, M02, M30, M98, and M99, are fixed functions designed by LADDER program, and their functions are not specified by the system, therefore they may differ between machines. Users have to check command specifications of the machine. (Functions listed in the chart belong to standard published LADDER version.)

M Code	Functions	Note
M00	Program stop	CNC
M01	Optional stop	CNC
M02	End of program	CNC
M03	Spindle CW	
M04	Spindle CCW	
M05	Spindle stop	
Txx	Auto tool change	xx: Tool no.
M08	Coolant ON	
M09	Coolant OFF	
M10	Chuck clamp	
M11	Chuck unclamp	
M30	Program rewind	CNC
M98	Calling of subprogram	CNC
M99	End of subprogram	CNC

（7） Waiting M code (For T800)

When dual-system lathe is working, with the control of waiting M CODEs, one of the system manufacturing processes can wait for the other system manufacturing process. The range of the waiting M code is specified by parameters.

- **Manufacturing Program Example**

  Example and instruction for waiting functions of dual-system tool rack:

  In system1's parameter1096 and parameter1097, set M200 to M203 as waiting M CODEs.

  Parameter setting:

  parameter1096=200 (minimum wait M CODE: M200)

  parameter1097=203 (maximum wait M CODE: M203)

System 1 manufacturing program	System 2 manufacturing program	Instruction
G00 X0. Z0. F500 T0 T0101 S1000M03	G00 X0. Z0. F500 T0 T0202 S1000M03	
M200	**M200**	**Both two system execute M200, and then continue executing.**
N1100 G01 X5. Z5. F500 …… …… …… …… …… N1199 ……	N2100 G01 X10. Z10. F500 …… N2199	System 1's manufacturing program (N1100 to N1119) and system 2's manufacturing program (N2100 to N2199) work independantly at the same time
M201	**M201**	**Wait**
	N2200 G01 X10. Z10. F500 …… …… ……	Only system 2 executes (N2200 to N2299) System 1 executes to M202 Wait system to execute to M202
M202	**M202**	**Wait**
N1300 G00 X30. Z30. F500 T0505 …… …… …… N1399	N2300 G00 X30. Z30. F500 T0707 …… …… …… N2399	System 1 (N1300 to N1319) System 2(N2300 to N2399) Manufacture independantly at the same time
M203	**M203**	**Wait**
M30	M30	program ends

- **Enable and Disable of waiting M CODE function**

  Use C Bit 301 and 801 to cancel system 1 and system 2's waiting M CODE function. Even when already in waiting condition, once this C Bit is enabled, manufacturing program continues executing.

system	number	instruction
System 1	C301	[S1] Disable Waiting M Code
System 2	C801	[S2] Disable Waiting M Code

- **Notice**

  In a manufacturing program, waiting M CODEs hav eto be written in independant blocks.

  If two systems execute different waiting M CODEs, warning MOT4010 will be triggered, and the two systems both stop.

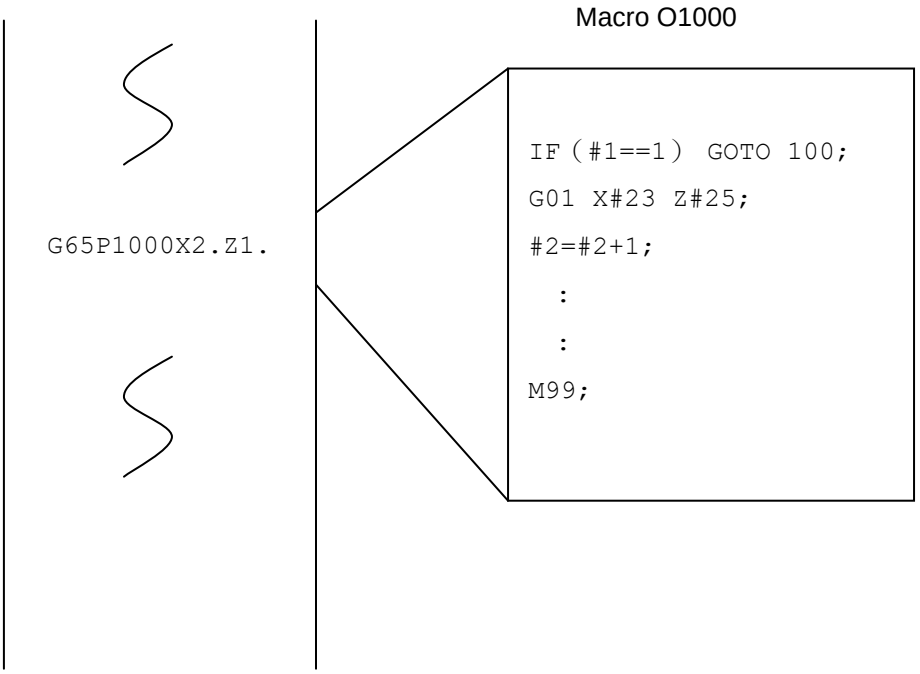
  When execute waiting M CODEs, MLC can not get the M CODEs.

5 MACRO

5.1 Macro program introduction

General numeric program language (NC PROGRAM) has its own restraints. E.g. unable to proceed calculation, unable to do condition distinguishment. MACRO commands provides usages of syntax at higher levels, e.g. IF, GOTO, function, variables, etc, and bring users greater flexibility.

When applying to system, if users want to call the execution of a sequence of frequently used actions with a single and simple command, they can use this MACRO function to develop the actions they want to execute as the following figure depicts:



5.2 User macro call

5.2.1 User macro simple call

Format:

G65 P__ L__ <Argument…>;

Argument:

Please refer to G65.

5.2.2 User macro modal call

Format:

G66 P__ L__ < Argument…>;

Argument:

Please refer to G66.

5.2.3 G code call macro

In a manufacturing program, when NC executes G Codes of the specified parameters in the following chart, it calls and executes corresponding macro programs respectively; if parameter's value is 0, it means that the function is not to be used. In the macro programs called by G CODEs, the G CODEs specified by parameters are deemed as general G CODE, and they can not call macro programs.

(For soft version VER: 01.xx.xxx and VER: 03.xx.xxx)

No. of parameter	File name of macro
P0166	O9010
P0167	O9011
P0168	O9012

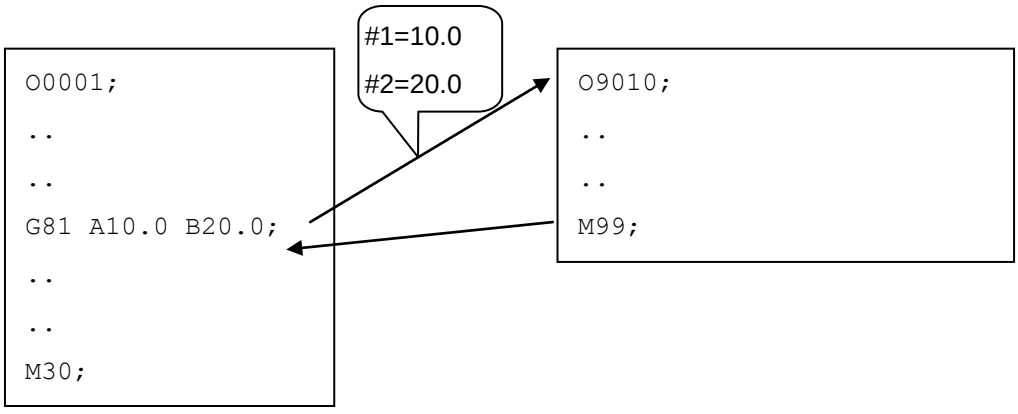
(For soft version VER: 04.xx.xxx)

No. of parameter	File name of macro
P0166	O9010
P0167	O9011
P0168	O9012
P0400	O9013
P0401	O9014

(For T800)

System 1		System 2	
No. of parameter	File name of macro	No. of parameter	File name of macro
P0166	O9010	P5166	O9015
P0167	O9011	P5167	O9016
P0168	O9012	P5168	O9017
P0400	O9013	P5400	O9018
P0401	O9014	P5401	O9019

When use the aforementioned method to call macro program, it can also transit values by arguments.



Parameter P0166=81

5.2.4 M code call macro

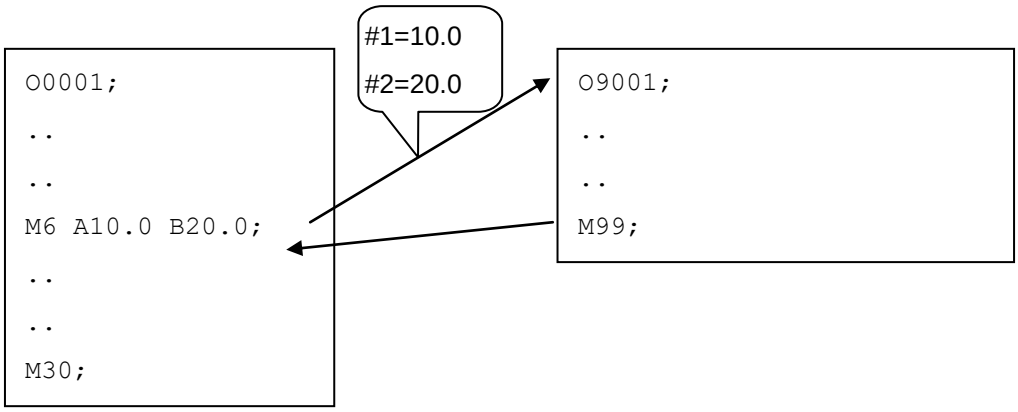
In a manufacturing program, when NC executes M Codes of the specified parameters in the following chart, it calls and executes corresponding macro programs respectively; if parameter's value is 0, it means that the function is not to be used. Besides, M1, M2, M30, M98 and M99 are stored function codes in NC, they can not be used to call macro program. In the macro programs called by M CODEs, the M CODEs specified by parameters are deemed as general M CODE, and they can not call macro programs.

No. of parameter	File name of macro
P0146	O9001
P0147	O9002
P0148	O9003
P0161	O9004
P0162	O9005
P0163	O9006
P0164	O9007
P0165	O9008

(For T800)

System 1		System 2	
No. of parameter	File name of macro	No. of parameter	File name of macro
P0146	O9001	P5146	O9001
P0147	O9002	P5147	O9002
P0148	O9003	P548	O9003
P0161	O9004	P5161	O9004
P0162	O9005	P5162	O9005
P0163	O9006	P5163	O9006
P0164	O9007	P5164	O9007
P0165	O9008	P5165	O9008

When use the aforementioned method to call macro program, it can also transit values by arguments.



Parameter P0146=6

5.2.5 T code call macro

When parameter169 is set as 1, T CODE in the manufacturing program calls macro, and the number after the T CODE will be input to system variables $xxx. In the macro programs called by T CODEs, the T CODEs specified by parameters are deemed as general M CODE, and they can not call macro programs.

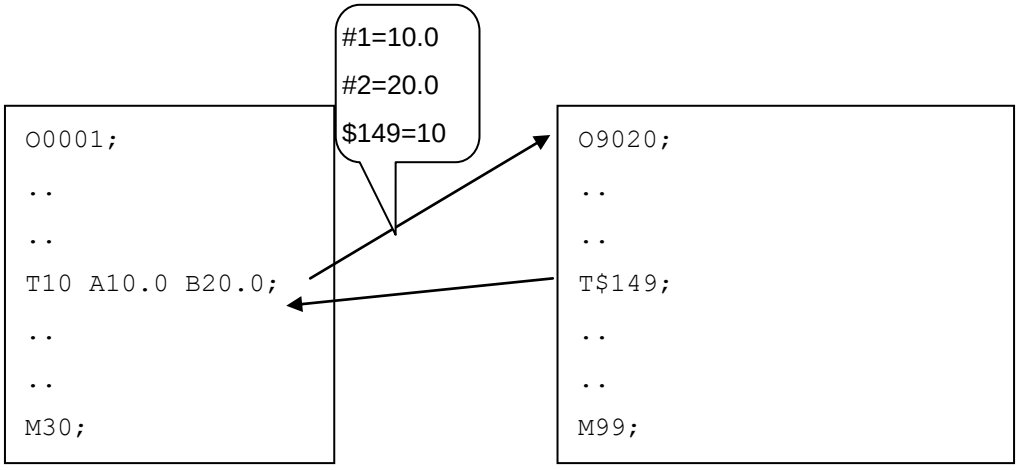
Soft version	System variable ($xxx)
VER: 01.xx.xxx VER: 03.xx.xxx	$149
VER: 04.xx.xxx	$139

No. of parameter	File name of macro
P0169	O9020

(For T800)

System 1		System 2	
No. of parameter	File name of macro	No. of parameter	File name of macro
P0169	O9020	P5169	O9021

When use the aforementioned method to call macro program, it can also transit values by arguments. (VER: 01.xx.xxx).

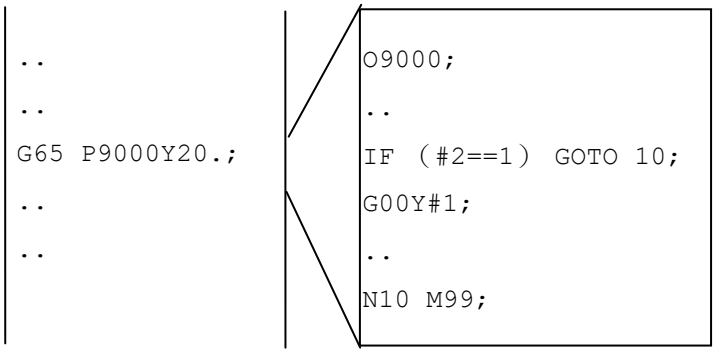


Parameter P0169=1

5.3 Difference between macro program call (G65) and general subprogram call (M98)

1. M98 can not specify arguments; G65 command can specify arguments.
2. Level of M98's local variables is fixed; G65's local variables change along with a nested depth (e.g. #1 has the same meaning before and after M98, but not the same with G65).
3. Maximum combination of M98's call level and G65, G66 is 8 levels; maximum level number of G65 and G66 is 4.

5.4 MACRO function list

Application	Function
	➢ VARIABLES LOCAL #,COMMON @,SYSTEM $ ➢ .ARITHMETIC COMMAND = + - * / () ➢ .LOGIC COMMAND AND &&,OR ||,NOT ! ➢ .RELATION COMMAND > < >= <= != == ➢ CONTROL COMMAND IF … GOTO ➢ .FUNCTON COMMAND SIN (),COS (),TAN (), ATAN (), SORT (),ABS (), ROUND (),FIX () ➢ .GENERAL NC COMMAND G,M,X,Y,Z,… ➢ .COMMENT /* …. */

5.5 Variable

5.5.1 Type of variable

1. Local Variables:
   #1 ~ #33 : Read / Write
   Programs of each level all have 33 local variables. Once the program of the level ends, the variables will be cleared. However, clicking RESET button will result in returning to main program level, and local variables of the level can be specified whether to be cleared, when clicking RESET, by parameter P141. But after a reboot, local variables of any level will be cleared. #0 has always an vacant value.

2. Global Variables:

   (For soft version VER: 01.xx.xxx and VER: 03.xx.xxx)

   @1 ~ @100 : Read / Write

   Programs of all levels all use these variables. Values of @1~@100 are decided by parameter P0140 whether to be cleared after clicking RESET button, and @0 has always an vacant value.

   (For soft version VER: 04.xx.xxx)

   @1 ~ @1799 : Read / Write

   Programs of all levels all use these variables. @1~@1499 are decided by parameter P0140 whether to be cleared after clicking RESET button; @401~@999 and @1400~@1499 can be kept when turning off the system. @0 has always an vacant value.

   @1 ~ @999 : open for users to set up, independant 2 systems.
   @1000 ~ @1499 : open for users to set up, shared by 2 systems.
   @1500 ~ @1699 : internal system use is not allowed, independant 2 systems.
   @1700 ~ @1799 : internal system use is not allowed, shared by 2 systems.

3. System Variables

   (For soft version VER: 01.xx.xxx and VER: 03.xx.xxx)

   $1 ~ $99 : Read only
   $100 ~ $199 : Read only / Hold
   $200 ~ $299 : Read / Write
   $300 ~ $399 : Read / Write / Hold

   (For soft version VER: 04.xx.xxx)

   $1 ~ $199 : Read only
   $200 ~ $399 : Read only / Hold
   $400 ~ $599 : Read / Write
   $600 ~ $799 : Read / Write / Hold

「Keep」 means that after clicking〈RESET〉button, and program restarts or mode is switched, the values of the system variables will not be cleared until being overwritten by new values, but they will be cleared after a reboot.

5.5.2 List of variables

(For soft version VER: 01.xx.xxx and VER: 03.xx.xxx)

Number	Type	Note
#00	NULL	
#01 ~ #33	Local Variables	Local variables can only be used with in a marco program.
@000	NULL	
@001 ~ @100	Global Variables (Common)	Global variables can be shared among different marco programs. Variables hold data even when the power is turned off.

(For soft version VER: 01.xx.xxx and VER: 03.xx.xxx)

Number	Description
$001	G code of group 01
$002	G code of group 02
$003	G code of group 03
$004	G code of group 04
$005	G code of group 05
$006	G code of group 06
$007	G code of group 07
$008	G code of group 08
$009	G code of group 09
$010	G code of group 10
$011	G code of group 11
$012	G code of group 12
$013	G code of group 13
$014	G code of group 14
$015	G code of group 15
$016	G code of group 16
$017	G code of group 17
$018	G code of group 18
$020	Programming coordinate (X axis)
$021	Programming coordinate (Y axis)
$022	Programming coordinate (Z axis)
$023	Data(I) of the previous block
$024	Data(J) of the previous block
$025	Data(K) of the previous block t
$100	Machine coordinate (X axis)
$101	Machine coordinate (Y axis)

Number	Description
$102	Machine coordinate (Z axis)
$103	Absolute coordinate (X axis)
$104	Absolute coordinate (Y axis)
$105	Absolute coordinate (Z axis)
$106	Relative coordinate (X axis)
$107	Relative coordinate (Y axis)
$108	Relative coordinate (Z axis)
$120	MLC→OP C100
$121	MLC→OP C101
$122	MLC→OP C102
$123	MLC→OP C103
$124	MLC→OP C104
$125	MLC→OP C105
$126	MLC→OP C106
$127	MLC→OP C107
$128	MLC→OP C108
$129	MLC→OP C109
$130	MLC→OP C110
$131	MLC→OP C111
$132	MLC→OP C112
$133	MLC→OP C113
$134	MLC→OP C114
$135	MLC→OP C115
$136	MLC→OP C136
$137	MLC→OP C137
$153	External workpiece zero point offset value (X axis)
$154~$159	G54~G59 workpiece zero point offset value (X axis)
$163	External workpiece zero point offset value (Z axis)
$164~$169	G54~G59 workpiece zero point offset value (Z axis)
$180	Plunge chamfering length of G78.
$181	Plunge chamfering angle of G78.
$182	Retract chamfering length of G78.
$183	Retract chamfering angle of G78
$184	Measurement of X axis. (Diameter / Radius)
$299	Macro error
$320	OP→MLC S100
$321	OP→MLC S101
$322	OP→MLC S102
$323	OP→MLC S103
$324	OP→MLC S104
$325	OP→MLC S105
$326	OP→MLC S106
$327	OP→MLC S107
$328	OP→MLC S108
$329	OP→MLC S109
$330	OP→MLC S110

Number	Description
$331	OP→MLC S111
$332	OP→MLC S112
$333	OP→MLC S113
$334	OP→MLC S114
$335	OP→MLC S115

(For soft version VER: 04.xx.xxx)

Number	Type	Note
#0	NULL	
#1~ #33	Local Variables	Local variables can only be used with in a marco program.
@0	NULL	

(For soft version VER: 04.xx.xxx)

	Level	Type	Hold data
@0		VACANT	None
@1 ~ @400	User	Each system	No
@401 ~ @999		Each system	Yes
@1000 ~ @1399		Common	No
@1400 ~ @1499		Common	Yes
@1500 ~ @1699	System	Each system	Yes
@1700~ @1799		Common	Yes
PS: @1~@1499 can be specified by parameter 0140 whether to be cleared by RESET signal.			

(For soft version VER: 04.xx.xxx)

Number	Description
$1	G code of group 01
$2	G code of group 02
$3	G code of group 03
$4	G code of group 04
$5	G code of group 05
$6	G code of group 06
$7	G code of group 07
$8	G code of group 08
$9	G code of group 09
$10	G code of group 10
$11	G code of group 11
$12	G code of group 12
$13	G code of group 13

Number	Description
$14	G code of group 14
$15	G code of group 15
$16	G code of group 16
$17	G code of group 17
$18	G code of group 18
$20	H code
$21	S code
$22	T code
$23	F code
$24	D code
$25	Sequence number of the current program
$26	File name of the current program. (Oxxxx)
$27	Enable using H code
$28	Enable using T code
$29	Enable using D code
$30	Programming coordinate (X axis)
$31	Programming coordinate (Y axis)
$32	Programming coordinate (Z axis)
$33	Programming coordinate (C axis)
$40	Programming coordinate of the previous block (X axis)
$41	Programming coordinate of the previous block (Y axis)
$42	Programming coordinate of the previous block (Z axis)
$43	Programming coordinate of the previous block (C axis)
$46	Data(I) of the previous block
$47	Data(J) of the previous block
$48	Data(K) of the previous block
$50	Line Number of line 1 of the previous block
$51	Moving G code of line 1 of the previous block
$52	End point coordinate of line 1 of the previous block (X axis) (Diameter)
$53	End point coordinate of line 1 of the previous block (Y axis) (Diameter)
$54	End point coordinate of line 1 of the previous block (Z axis)
$55	Arc center coordinate of line 1 of the previous block (X axis)
$56	Arc center coordinate of line 1 of the previous block (Y axis)
$57	Arc center coordinate of line 1 of the previous block (Z axis)
$58	Line Number of line 2 of the previous block
$59	Moving G code of line 2 of the previous block
$60	End point coordinate of line 2 of the previous block (X axis) (Diameter)
$61	End point coordinate of line 2 of the previous block (Y axis) (Diameter)
$62	End point coordinate of line 2 of the previous block (Z axis)
$63	Arc center coordinate of line 2 of the previous block (X axis)
$64	Arc center coordinate of line 2 of the previous block (Y axis)
$136	MLC→OP C136
$137	MLC→OP C137
$138	MLC→OP C138
$140	Absolute coordinate of G31 skip point (X axis / P1)
$141	Absolute coordinate of G31 skip point (Y axis / P1)

Number	Description
$142	Absolute coordinate of G31 skip point (Z axis / P1)
$143	Absolute coordinate of G31 skip point (C axis / P1)
$144	Machine coordinate of G31 skip point (X axis / P1)
$145	Machine coordinate of G31 skip point (Y axis / P1)
$146	Machine coordinate of G31 skip point (Z axis / P1)
$147	Machine coordinate of G31 skip point (C axis / P1)
$148	Absolute coordinate of G31 skip point (X axis / P2)
$149	Absolute coordinate of G31 skip point (Y axis / P2)
$150	Absolute coordinate of G31 skip point (Z axis / P2)
$151	Absolute coordinate of G31 skip point (C axis / P2)
$152	Machine coordinate of G31 skip point (X axis / P2)
$153	Machine coordinate of G31 skip point (Y axis / P2)
$154	Machine coordinate of G31 skip point (Z axis / P2)
$155	Machine coordinate of G31 skip point (C axis / P2)
$156	Absolute coordinate of G31 skip point (X axis / P3)
$157	Absolute coordinate of G31 skip point (Y axis / P3)
$158	Absolute coordinate of G31 skip point (Z axis / P3)
$159	Absolute coordinate of G31 skip point (C axis / P3)
$160	Machine coordinate of G31 skip point (X axis / P3)
$161	Machine coordinate of G31 skip point (Y axis / P3)
$162	Machine coordinate of G31 skip point (Z axis / P3)
$163	Machine coordinate of G31 skip point (C axis / P3)
$164	Absolute coordinate of G31 skip point (X axis / P4)
$165	Absolute coordinate of G31 skip point (Y axis / P4)
$166	Absolute coordinate of G31 skip point (Z axis / P4)
$167	Absolute coordinate of G31 skip point (C axis / P4)
$168	Machine coordinate of G31 skip point (X axis / P4)
$169	Machine coordinate of G31 skip point (Y axis / P4)
$170	Machine coordinate of G31 skip point (Z axis / P4)
$171	Machine coordinate of G31 skip point (C axis / P4)
$180	Plunge rate ratio of G78
$181	Retract rate ratio of G78
$190	ISR count
$191	System date (year): yyyy
$192	System date (month): mm
$193	System date (day): dd
$194	System date (hour): hh
$195	System date (minute): mm
$196	System date (second): ss
$200	MLC→OP C100
$201	MLC→OP C101
$202	MLC→OP C102
$203	MLC→OP C103
$204	MLC→OP C104
$205	MLC→OP C105
$206	MLC→OP C106

Number	Description
$207	MLC→OP C107
$208	MLC→OP C108
$209	MLC→OP C109
$210	MLC→OP C110
$211	MLC→OP C111
$212	MLC→OP C112
$213	MLC→OP C113
$214	MLC→OP C114
$215	MLC→OP C115
$220	Amount of parts
$230	Machine coordinate (X axis)
$231	Machine coordinate (Y axis)
$232	Machine coordinate (Z axis)
$233	Machine coordinate (C axis)
$240	Absolute coordinate (X axis)
$241	Absolute coordinate (Y axis)
$242	Absolute coordinate (Z axis)
$243	Absolute coordinate (C axis)
$250	Relative coordinate (X axis)
$251	Relative coordinate (Y axis)
$252	Relative coordinate (Z axis)
$253	Relative coordinate (C axis)
$303	External workpiece zero point offset value (X axis)
$304~$309	G54~G59 workpiece zero point offset value (X axis)
$313	External workpiece zero point offset value (Y axis)
$314~$319	G54~G59 workpiece zero point offset value (Y axis)
$323	External workpiece zero point offset value (Z axis)
$324~$329	G54~G59 workpiece zero point offset value (Z axis)
$333	External workpiece zero point offset value (C axis)
$334~$339	G54~G59 workpiece zero point offset value (C axis)
$599	Number of macro alarm
$600	OP→MLC S100
$601	OP→MLC S101
$602	OP→MLC S102
$603	OP→MLC S103
$604	OP→MLC S104
$605	OP→MLC S105
$606	OP→MLC S106
$607	OP→MLC S107
$608	OP→MLC S108
$609	OP→MLC S109
$610	OP→MLC S110
$611	OP→MLC S111
$612	OP→MLC S112
$613	OP→MLC S113
$614	OP→MLC S114

Number	Description
$615	OP→MLC S115
$620	Mirror image (X axis / Write)
$621	Mirror image (Y axis / Write)
$622	Mirror image (Z axis / Write)
$623	Mirror image (C axis / Write)
$630	Enable Tapping (Write)
$640	Programming coordinate (X axis / Write)
$641	Programming coordinate (Y axis / Write)
$642	Programming coordinate (Z axis / Write)
$643	Programming coordinate (C axis / Write)
$650	Element of Matrix for Proportion axis (a_{11} / Write)
$651	Element of Matrix for Proportion axis (a_{12} / Write)
$652	Element of Matrix for Proportion axis (a_{13} / Write)
$653	Element of Matrix for Proportion axis (a_{14} / Write)
$654	Element of Matrix for Proportion axis (a_{21} / Write)
$655	Element of Matrix for Proportion axis (a_{22} / Write)
$656	Element of Matrix for Proportion axis (a_{23} / Write)
$657	Element of Matrix for Proportion axis (a_{24} / Write)
$658	Element of Matrix for Proportion axis (a_{31} / Write)
$659	Element of Matrix for Proportion axis (a_{32} / Write)
$660	Element of Matrix for Proportion axis (a_{33} / Write)
$661	Element of Matrix for Proportion axis (a_{34} / Write)
$662	Element of Matrix for Proportion axis (a_{41} / Write)
$663	Element of Matrix for Proportion axis (a_{42} / Write)
$664	Element of Matrix for Proportion axis (a_{43} / Write)
$665	Element of Matrix for Proportion axis (a_{44} / Write)

Tool offset

Use G10 to specify tool compensation. Use GET function to read tool compensation values within a MACRO. Refer to instructions in 5.10 for the usage of GET function.

5.5.3 Variable usage

1.

#i	ith local variable
@i	ith common variable
$i	ith system variable

2.

(<formula>)
@ (<formula>)
$ (<formula>)

Example:

(#10) ---------- Correct
(#10-1) ---------- Correct
(#6/2) ---------- Correct
(#3-FIX (#2)) ---------- Correct
##2---------- Wrong

5.5.4 Variables citation

<address> #i	or	<address> -#i
<address> @i	or	<address> -@i
<address> $i	or	<address> -$i

Example:

X#33;---------- if #33 is 1.2, equals to X1.2;)
Z-#33;---------- if #3 is 2.1, equals to Z-2.1;)
G#33; ---------- if #33 is 3, equals to G3;)

5.5.5 Instruction for VACANT value

1. When doing MACRO CALL, all local variables of the level will be cleared and become VACANT by default.
2. When decoding CNC command, VACANT values will be ignored.
   E.g.:
   when #1=10, #2=VACANT
   G00X#1Y#2; equals to G00X10;
   This is especially useful for MACRO content compilation.
3. When executing, except for assign (=), same as 0
   E.g.:
   If #1=VACANT
   When #2=#1 #2=VACANT
   When #2=#1+#1 #2=0
4. When with conditional expression, except for == and !=, sames as 0
   E.g.:
   If #1=VACANT
   #1==#0 is true (true)
   #1==0 is false (false)
   #1 >= 0 is true (true)

5.6 Mathematics command

（1） Substitution, =

#i = #j

（2） Addition, +

#i = #j + #k

（3） Subtration, -

#i = #j - #k

（4） Multiplication, *

#i = #j * #k

（5） Quotient, /

#i = #j / #k

（6） Parenrhesis, ( )

#i = #j *　(#k + #l)

5.7 Logic command

（1） AND, &&

#i = #j && #k

0: False, 1: True.

（2） OR, | |

#i = #j | | #k

（3） NOT, !

#i = ! #j

5.8 Compare command

（1） Greater than (GT), >

#i = #j > #k

If #j is greater than #k, #i is equal to 1.

（2） Less than (LT), <

#i = #j < #k

If #j is less than #k, #i is equal to 1.

（3） Greater than or Equal or (GE), >=

#i = #j >= #k

If #j is equal or greater than #k, #i is equal to 1.

（4） Less than or Equal or (LE), <=

#i = #j <= #k

If #j is equal or less than #k, #i is equal to 1.

（5） Equal, ==

#i = #j == #k

If #j is equal to #k, #i is equal to 1.

（6） Not equal, !=

#i = #j != #k

If #j is not equal to #k, #i is equal to 1.

5.9 Procedures control command

Use IF ~ GOTO to control program procedures.

（1） Conditioned skip

IF (<Conditional express>) GOTO n

instruction:

if <Conditional express> is true, skip to #n block and execute; if not, continue executing next block.

E.g.:

```
IF  (  (#1+#2) > 3 ) GOTO 10;
        :
        :
N10 G01 X#3;
```

（2） Non-conditioned skip

GOTO n

instruction:

Directly skip to #n block and execute.

5.10 Function

（1） SIN

#i=SIN (#J)　　(Uniti: degree)

（2） COS Function

#i=COS (#j)　　(Uniti: degree)

（3） TAN

#i=TAN (#j)　　(Uniti: degree)

（4） ATAN

#i=ATAN (#j)　　(Uniti: degree)

（5） SQRT

#i=SQRT (#j)　　(Square root)

（6） ABS

#i=ABS (#j)　　(Absolute value)

（7） ROUND

#i=ROUND (#j)　　(Nearest)

（8） FIX

#i=FIX (#j)　　(Round down)

（9） GET

#i=GET (#k,#j)

#j: The number of tool compensation. (1 ~ 30)

#k: 1: The value of tool wear compensation. (X axis)

2: The value of tool wear compensation. (Z axis)

3: The Radius value of tool wear compensation.

4: The value of tool length compensation. (X axis)

5: The value of tool length compensation. (Z axis)

6: The Radius value of tool length compensation.

7: Type of tool compensation.

8: The value of tool wear compensation. (Y axis)

9: The value of tool length compensation. (Y axis)

5.11 Note

Inputs between "/*" and "*/" symbols all be abandoned.

Example:

```
/* test1 */;
G00 X10. /* test 2 */;
/* test3 */ G01 Y20.;
G01 X10. Y20.; /* test
```