SYNTEC

Instruction Guide of
Lathe Programming

By: SYNTEC

Date: 2015/11/13

Version: 7.13

The record of version update

item	The content	Date	Author	The latest version
01	the first craft	2005/10/01	Jerry	V7.1
02	1.add G68, G69 2.Modify G10	2006/01/25	Jerry	V7.2
03	modify the specification of G01, G04, C modify the feedrate F in the examples and define its unit mm/rev	2006/06/06	Jerry	V7.3
04	1.add the specification of G65 G66 G67	2006/07/18	Jerry	V7.4
05	1. add the specification of G12.1 G13.1	2006/07/20	Jerry	V7.5
06	1. add the specification of G07.1	2006/10/05	Jerry	V7.6
07	1. mdify M99 descriptions	2006/11/10	Jerry	V7.7
08	1.add G73 H variable 2.add G74 Type II cutting mode 3.add the specification of G34 4.fix the error in M99 example 5.add the specification of G51.2 6.add spindle synchronization 7.add peck tapping descriptions 8.fix G40 G41 G42 figure 9.fix G73 figure 10.add Postscript 3, Description of Lathe graph assist G code 11.fix G78 figure and action description	2009/12/30	James Lin	V7.16
09	Sync with the Chinese version programming manual	2012/01/02	Adam Hsu	V7.8
10	1. Add the notice of G21/G78	2013/11/18	Bryan Ho	V7.9
11	1. Modify G01 description	2013/11/26	Andy Ngo	V7.10
12	1. Modify G77 description	2015/05/25	Mars Kao	V7.11
13	Modify G51.2 description	2015/08/23	Otis Siah	V7.12

14	Add Chinese topic, and increase front size	2015/11/13	Linda Chen	V7.13

Contents

1 G Code Instruction Description

1.1 G Code List

Function Name	G code			Index
	Type A	Type B	Type C	
Positioning(Rapid traverse)	G00	G00	G00	7
Linear interpolation(cutting feed)	G01	G01	G01	9
Circular interpolation(CW)	G02	G02	G02	11
Circular interpolation(CCW)	G03	G03	G03	11
Dwell	G04	G04	G04	16
Cylinder interpolation	G07.1	G07.1	G07.1	17
Exact stop check	G09	G09	G09	19
Programmable data input	G10	G10	G10	20
Start polar coordinates interpolation	G12.1	G12.1	G12.1	22
polar coordinates interpolation Cancel	G13.1	G13.1	G13.1	22
X-Y plane selection	G17	G17	G17	25
Z-X plane selection	G18	G18	G18	25
Y-Z plane selection	G19	G19	G19	25
Outer/internal diameter cutting cycle	G90	G77	G20	26
Threadingcycle	G92	G78	G21	31
End-face cutting cycle	G94	G79	G24	36
Return to reference position	G28	G28	G28	41
Return from any reference position	G30	G30	G30	43
Skip function	G31	G31	G31	44
Thread cutting	G32	G33	G33	46
Cancel tool nose radius compensation	G40	G40	G40	56
Tool nose radius compensation(left)	G41	G41	G41	56
Tool nose radius compensation(right)	G42	G42	G42	56
Polygon Cutting	G51.2	G51.2	G51.2	66
Local coordinate system setting	G52	G52	G52	69
Machine coordinate system setting	G53	G53	G53	70
Workpiece coordinate system selection	G54 ~G59.9	G54 ~G59.9	G54 ~G59.9	72
Single Marco calling	G65	G65	G65	74
Custom marco modal call	G66	G66	G66	74

Custom marco modal call cancel	G67	G67	G67	74
Input in imperial system	G20	G20	G70	75
Input in metric system	G21	G21	G71	75
Fine cutting cycle	G70	G70	G72	76
Stock removal in turning	G71	G71	G73	81
Stock removal in facing	G72	G72	G74	88
Pattern repeating	G73	G73	G75	95
End face peck drilling	G74	G74	G76	99
Outer diameter/internal diameter drilling	G75	G75	G77	102
Multiple threading cycle	G76	G76	G78	105
Canned cycle for drilling cancel	G80	G80	G80	
Cycle for face drilling	G83	G83	G83	112
Cycle for face tapping	G84	G84	G84	115
Cycle for face boring	G85	G85	G85	119
Cycle for side drilling	G87	G87	G87	112
Cycle for side tapping	G88	G88	G88	115
Cycle for side boring	G89	G89	G89	119
Coordinate system setting/max. spindle speed setting	G50	G92	G92	121
Feedrate per minute(mm/min.)	G98	G94	G94	122
Feedrate per revolution(mm/rev.)	G99	G95	G95	122
Constant surface speed control	-	G96	G96	123
Constant surface speed control cancel	-	G97	G97	123
Return to initial point	-	G98	G98	-
Return to R point	-	G99	G99	-

1.2 Positioning (G00)

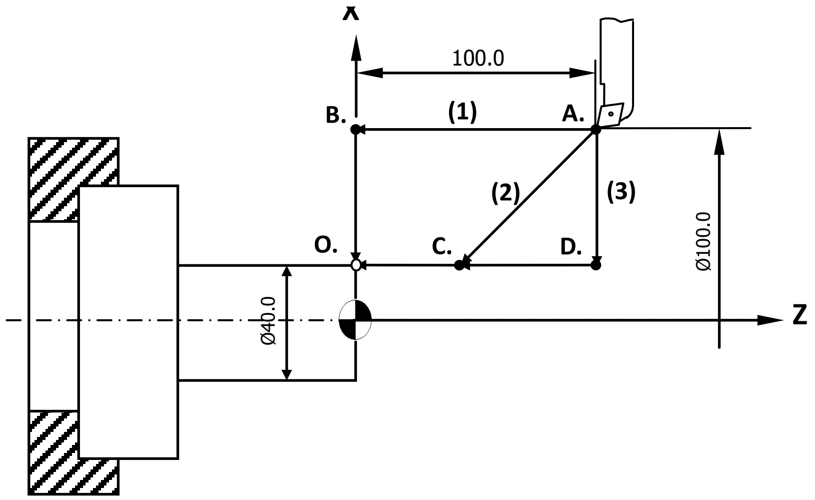
The G00 command moves a tool to the position, in the workpiece system, specified with an absolute or an incremental command at a rapid traverse rate. There is no any cutting action in this command. The main aim is to save the movement time when action includes no cutting. In the lathe program, it is usually used in the tool from machine zero point to start cutting point, or from end point to machine zero point. In absolute mode (G90), tool moves to specified position in coordinate system in increment mode(G91), tool moves to specified position by specified distance.

1.2.1 Format

G00 X(U)_Z(W)_

X ,Z: specified position(absolute mode)
U ,W: specified position(increment mode)

1.2.2 Example



There are several ways to make tool move from point A to point O. Three of them are as below:

1.2.2.1 Absolute mode

G00 Z0.0 // A.→B.
X40.0 // B.→O.

G00 X40.0 Z0.0 //A.→C.→O.

G00 X40.0 //A.→D.
Z0.0 //D.→C.→O.

1.2.2.2 Incremant mode

G00 W-100.0 // A.→B.
U-60.0 // B.→O.

G00 U-60.0 W-100.0 //A.→C.→O.

G00 U-60.0 //A.→D.
W-100.0 // D.→C.→O.

1.2.2.3 Combination of absolute mode and increment mode

G00 Z0.0 or G00 W-100.0
U-60.0 X40.0

G00 X40.0 or G00 U-60.0
W-100.0 Z0.0

G00 X40.0 W-100.0 or G00 U-60.0 Z0.0

1.3 Linear Interpolation (G01)

G01 executes linear interpolation, moves to specified position with feed rate defined by F value. It can process: outer (inner) diameter, end face, outer (inner) taper, outer (inner) tank, chamfer…etc.

1.3.1 Format

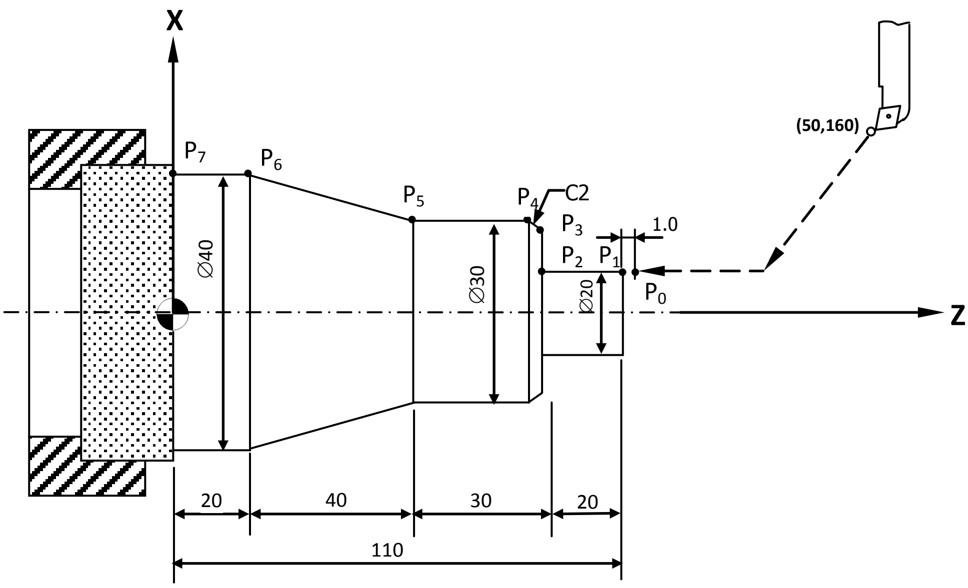
G01 X(U)_Z(W)_F_

X ,Z: specified position(absolute mode)
U ,W: specified position(increment mode)
F: Feed rate, Unit: mm/min (inch/min) for G94 mode,
mm/rev (inch/rev) for G95 mode← default mode

1.3.2 Note

- The max. feed rate of G01 is defined by PR405-maximum cutting feed rate or (PR621~PR636)-each axis maximum cutting feed rate
- Default value F: 1000mm/min(inch/min) for G94 mode and 1.mm/rev(inch/rev) for G95 mode
- Default mode G94/G95 can be changed by parameter Pr3836 (reboot controller to activate setting).

1.3.3 Example



G92 X50.0 Z160.0 S10000 //set the program zero point, max.
// speed 10000 rpm

```
T01                    //use tool NO. 1
G96 S130 M03           //constant surface speed, surface speed
                       //=130m/min, spindle rotate CW
M08                    //cutting liquid ON
G00 X20.0 Z111.0       //positioning to specified point P0
G01 Z90.0 F600              //linear interpolation P0→P2
  X26.0                     //P2→P3
  X30.0 Z88.0          //P3→P4
  Z60.0                //P4→P5
  X40.0 Z20.0          //P5→P6
  Z0.0                 //P6→P7
G00 X50.0              //return the tool
  Z160.0                    //return to zero point
M05 M09                //spindle stops, setting liquid OFF
M30                    //program end
```

1.4 Circular Interpolation (G02/G03)

The G02 ,G03 command will move a specified tool along a circular arc on XZ plane, the parameter settings are as below:

Data setting		Command	Definition
1	Tool direction	G02	CW
		G03	CCW
2	End position	X ,Z	The end position of specified arc
		U ,W	Vector value from starting point to end point
3	Distance from starting point to centered	Two axes among I ,J ,K axis	Vector value from arc starting point to centered
	Radius of arc	R	Radius of arc
4	Feedrate	F	Feedrate along the arc

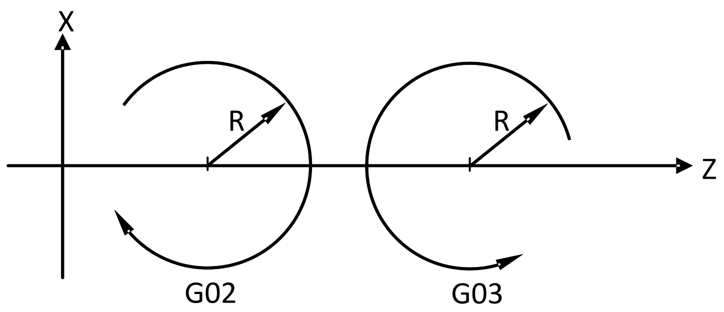
1.4.1 Format

$$\begin{Bmatrix} \mathrm{G02} \\ \mathrm{G03} \end{Bmatrix} \mathrm{X(U)}\_\ \mathrm{Z(W)}\_ \begin{Bmatrix} \mathrm{R}\_ \\ \mathrm{I}\_\ \mathrm{K}\_ \end{Bmatrix} \mathrm{F}\_\ ;$$

G02: Circular Interpolation (CW)
G03: Circular Interpolation (CCW)
X(U) ,Z(W): end point of the arc
R: radius of arc(under 180°)
I, K: X(Z) axis distance from starting point of arc to the center of circle. Positive or negative is determined by the direction.
F: Feedrate of cutting

1.4.2 PIC

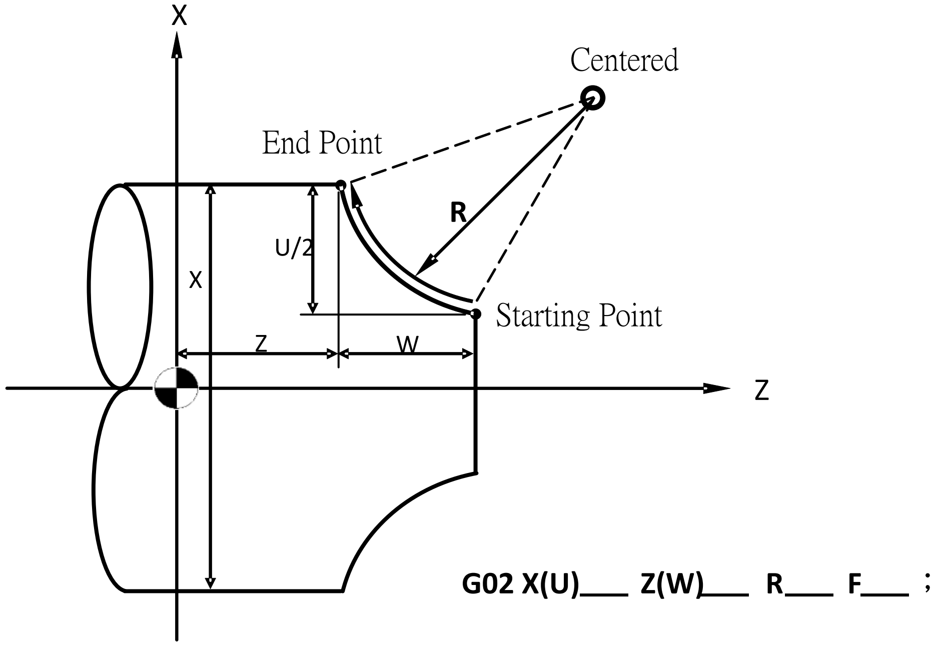
1.4.2.1 G02/G03 Direction Decision



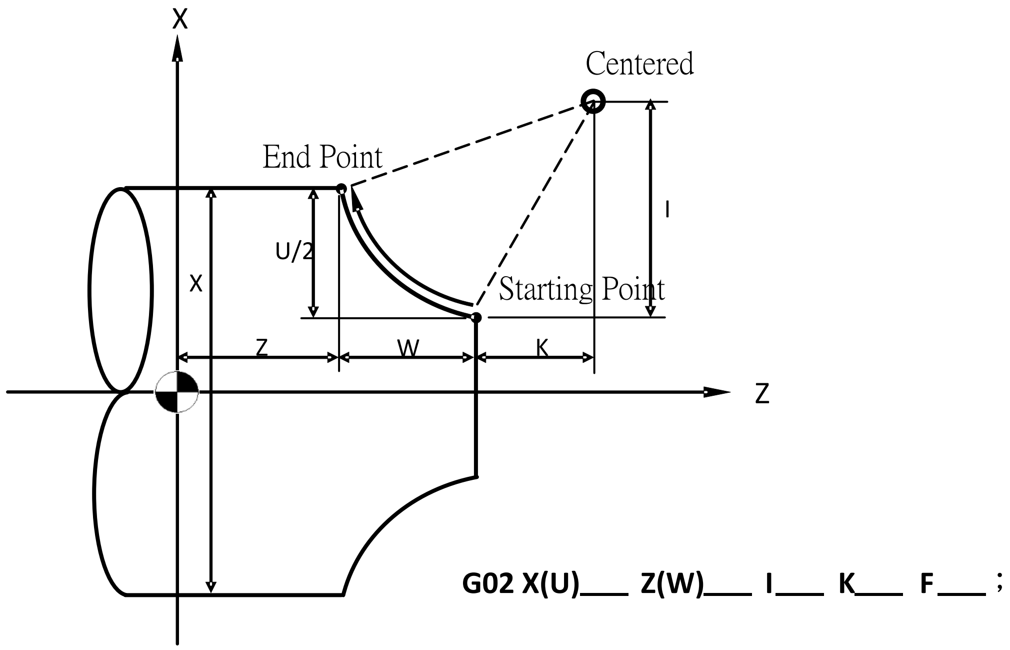
1.4.2.2 Parameter setting in process

(1). G02 circular interpolation

a. Use R value

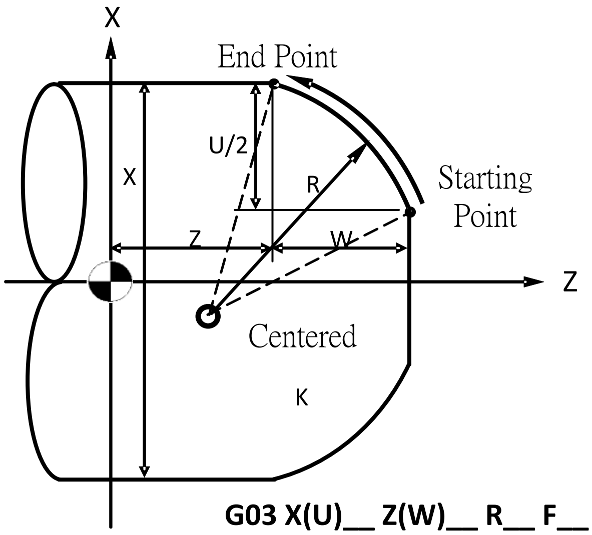


b. Use I ,K



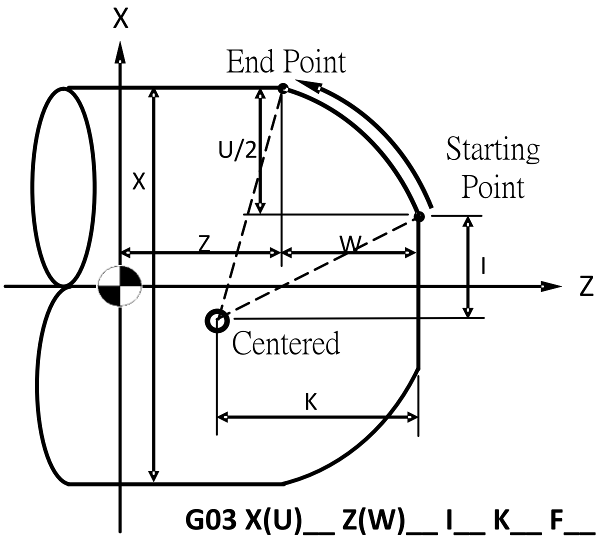
(2). G03 circular interpolation

a. Use R value



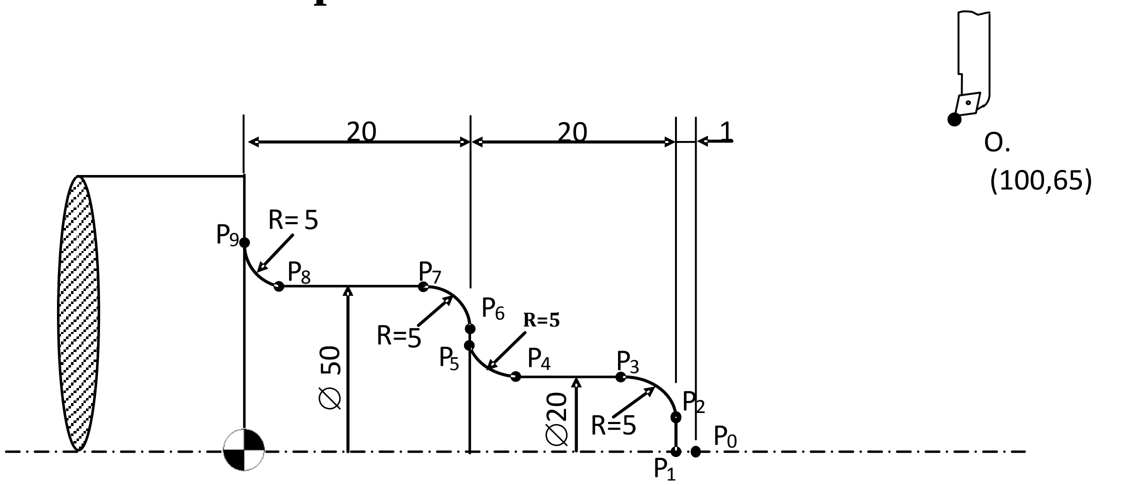
G03 X(U)__ Z(W)__ R__ F__ ;

b. Use I ,K



G03 X(U)__ Z(W)__ I__ K__ F__ ;

1.4.3 Example 1



T01 //use tool NO.1

G92 S10000 // max spindle speed 10000 rpm

G96 S130 M03 //constant surface speed, surface speed =130 //mm/min, spindle rotate CW

M08 //cutting liquid ON

G00 X0.0 Z41.0 //positioning **O.**→P_0

G01 Z40.0 F600 //linear interpolation, feedrate 600 mm/rev, //P_0→P_1

X10.0 //P_1→P_2

G03 X20. Z35.0 R5.0 //circular interpolation CCW P_2→P_3, //radius 5mm

G01 Z25.0 //P_3→P_4

G02 X30.0 Z20. R5.0 //circular interpolation CW P_4→P_5, //radius 5mm

G01 X40.0 //P_5→P_6

G03 X50.0 Z15.0 R5.0 //circular interpolation CCW P_6→P_7, //radius 5mm

G01 Z5.0 //P_7→P_8

G02 X60.0 Z0.0 R5.0 //circular interpolation CW P_8→P_9, //radius 5mm

G00 X100.0 //tool escape, escape from workpiece

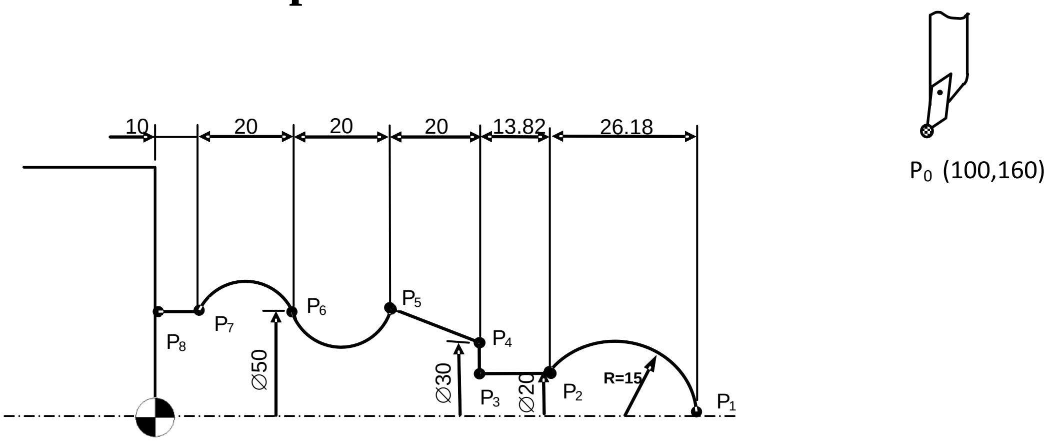
G00 Z65.0 //return to initial point

M09 //cutting liquid OFF

M05 //spindle stops

M30 //program end

1.4.4 Example 2




```
T01                          //use tool NO.1
G92 S10000                   // max spindle speed 10000 rpm
G96 S130 M03                 //constant surface speed, surface speed=
                             //130 mm/min, spindle rotate CW
M08                          //cutting liquid NO
G00 X0.0 Z110.5              //positioning, close to the starting point
G01 Z110.0 F500              //linear interpolation, feedrate
                             //500 mm/min
G03 X20.0 Z83.82 K-15.0      //circular interpolation CCW,
                             //P1→P2, radius 15 mm
G01 Z70.0                    //linear interpolation, P2→P3
X30.0                        //P3→P4
X50.0 Z50.0                  //P4→P5
G02 X50.0 Z30.0 R10.0        //circular interpolation CW,
                             //P5→P6, radius 10 mm
G03 X50.0 Z10.0 R10.0        //circular interpolation CCW,
                             //P6→P7, radius 10 mm
G01 Z0.0                     //linear interpolation, P7→P8
M09                          //cutting liquid OFF
G00 X100.0                   //tool escape, escape from workpiece
Z160.0                       //return to initial point
M05                          //spindle stops
M30                          //program end
```

1.5 Dwell (G04)

We can use G04 command to let the tool dwell a specified time when we process to an appropriate position. It can help cutting off the scouring of iron, improving the precision of cutting depth, and better the surface finish to achieve roundness (as below). When G04 command coordinates with G94 or G95, the time unit is in second.

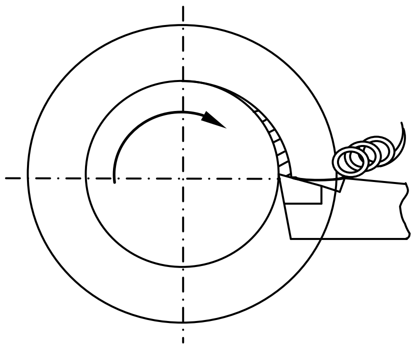
※G04 command is only effective in single block.

1.5.1 Format

$$G04 \begin{Bmatrix} X(U)\_ \\ P\_ \end{Bmatrix};$$

X(U) ,P: dwell time

1.5.2 Example



G04 X0.5 //dwell 0.5s
G04 U0.5 //dwell 0.5s
G04 P500 //dwell 0.5s
//※Notice: P_is not allowed to be decimal point

1.5.3 Referenced formula

$$T = \frac{Z \times 60}{N}$$

T: dwell time (s)
Z: dwell numbers of revolution
N: rev/min

Notice: Syntec controller didn't offer direct Command to input dwell coils. Operator must input the dwell time calculated from the given formula.

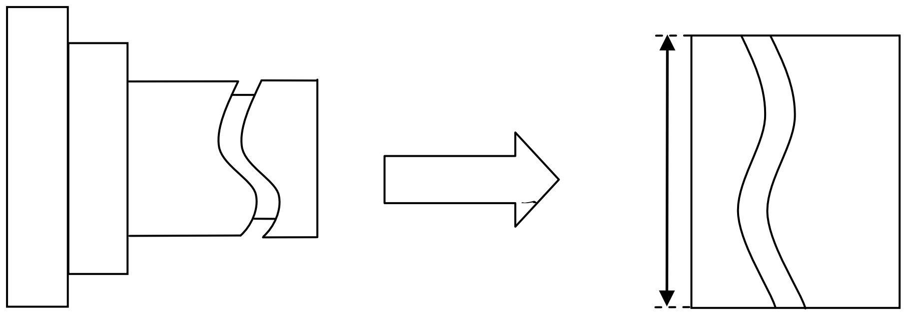
1.6 Cylinder Interpolation (G07.1)

G07.1 starts the cylinder difference, G02/G03-> circular interpolation function, G40/G41/G42-> compensation function for tool nose radius. Because of the difficulty of the calculation of the vector in the center of a cirle, we use the way of R_radius address. Feedrate F_ is linear velocity in the surface of the cylinder. About the way of feed we must swith it into G94 in the lathe system in the first, for the C-axis is the main shaft probably.

1.6.1 Format

```
G19 Z0 C0   //select the working platform
G07.1 C__   //start the cylinder difference, C_the cylinder radius
...
…           //the description of the route
…
G07.1C0     //end the cylinder difference
```

1.6.2 Example



```
G28 U0 W0
T0202
G97 S1000          // set up the rotational speed of the main shaft
G00 X50.0 Z0.
G94 G01 X40.0 F100.
G19 C0 Z0          // choose CZ the working platform
G07.1 C20.0        // start G07.1, the radius is 20.0
G41                // start process
G01 Z-10.0 C80.0 F150.0
G01 Z-25.0 C90.0
G01 Z-80.0 C225.0
G03 Z-75.0 C270.0 R55.0
G01 Z-25.0
```

```
G02 Z-20.0 C280.0 R80.0
G01 C360.0
G40                  // end process
G07.1 C0             // cancel G07.1
G01 X50.0
G00 X100.0 Z100.0
M30
```

1.7 Exact Stop (G09)

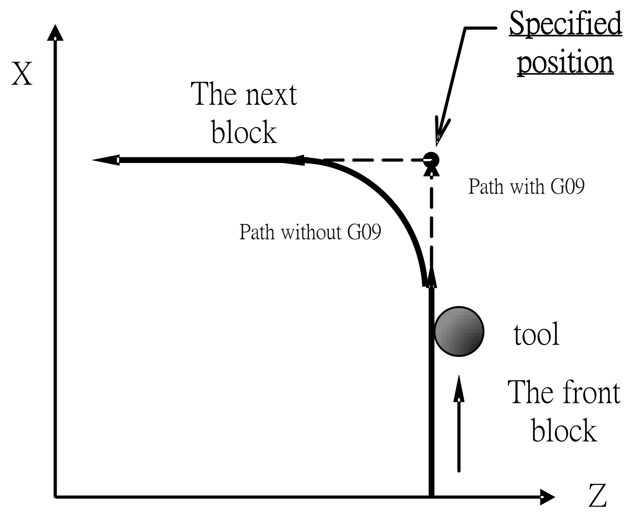
When processing the corner, we cannot cut the exactly corner and the rounding off occurs because the tool moves too fast or servo system delays. In the situation that we need a right-angled, we can use G09 to prevent the CNC from rounding off sharp corners. It controls the tool to decelerate when approaching to the corner and come to a complete stop at the end of block. When the tool reach to specified position, then the next block will be executed.

1.7.1 Format

G09 X__ Z__

X ,Z: specified corner position

1.7.2 PIC



1.8 Programmable Data Input (G10)

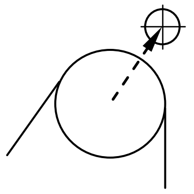
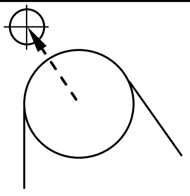
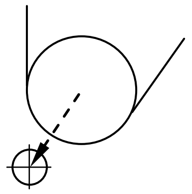
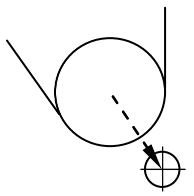
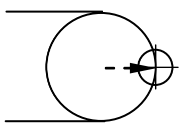
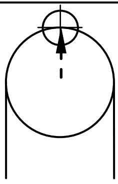
G10 command is programmable data input command. We can use this command to change the tool offset value when programming.

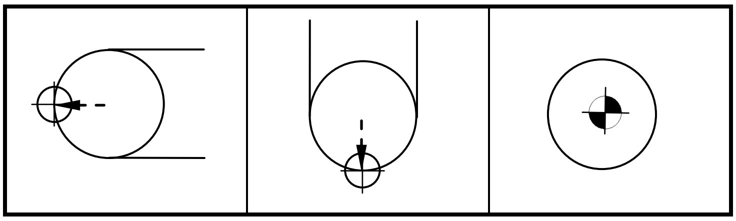
1.8.1 Format

G10 P_X_Z_R_Q_
or
G10 P_U_W_C_Q_

P: offset number
Tool wear offset value: P = number of tool wear offset
Tool geometry offset value: P = 10000 + number of tool geometry offset value
X: offset value on X axis(absolute)
Y: offset value on Y axis(absolute)
Z: offset value on Z axis(absolute)
U: offset value on X axis(incremental)
V: offset value on Y axis (incremental)
W: offset value on Z axis (incremental)
R: tool nose radius offset value(absolute)
C: tool nose radius offset value (incremental)
Q: imaginary tool nose number(setting method is in next page)

1.8.2 Imaginary tool nose setting

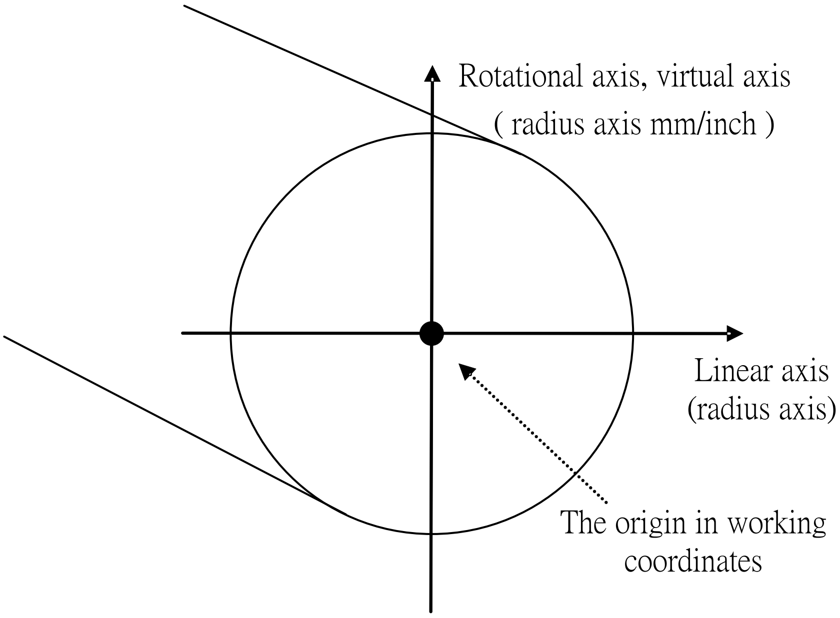
Imaginary tool nose NO.1	Imaginary tool nose NO.2	Imaginary tool nose NO.3
		
Imaginary tool nose NO.4	Imaginary tool nose NO.5	Imaginary tool nose NO.6
		
Imaginary tool nose NO.7	Imaginary tool nose NO.8	Imaginary tool nose NO.9



1.9 Polar coordinates interpolation (G12.1/G13.1)

The function of the polar coordinates interpolation transfers the program instructions of the patterns in the rectangular coordinate to linear motion (knife motion) and rotational motion (workpiece motion). This way is usually used in cutting end face and milling cam shaft in lathe.

The plane of the polar coordinates interpolation: G12.1 starts polar coordinates interpolation and selects the plane of the polar coordinates interpolation (below). The polar coordinates interpolation is completed in the plane.



When system power on or reset, polar coordinates interpolation will be cancelled (G13.1).

With G12.1, the planes (chose by G17 ,G18 or G19)used before are cancelled but with G13.1(polar coordinate interpolation cancel) they are restored. When we reset the system, polar coordinates interpolation is cancelled and restore ,the plane G17 ,G18 or G19 assigned before.

We can use G code with polar coordinates interpolation

G01	linear interpolation
G02, G03	circular interpolation
G04	pause
G40, G41, G42	tool nose radius compensation
G65, G66, G67	sub-program call

Circular interpolation in polar coordinate plane: In the polar coordinate plane, the arguments of the arc's radius with Circular interpolation (G02 or G03) are I and J.

The motion along the axis of the plane of the Cartesian coordinate interpolation in polar coordinates interpolation: The tool moves along these axes and has no relationship with polar coordinates interpolation.
The display of the coordinates in polar coordinates interpolation: Linear axes(X) and rotational axes(C) show their real location by radius axes and others show theirs by original parameters.

1.9.1 Format

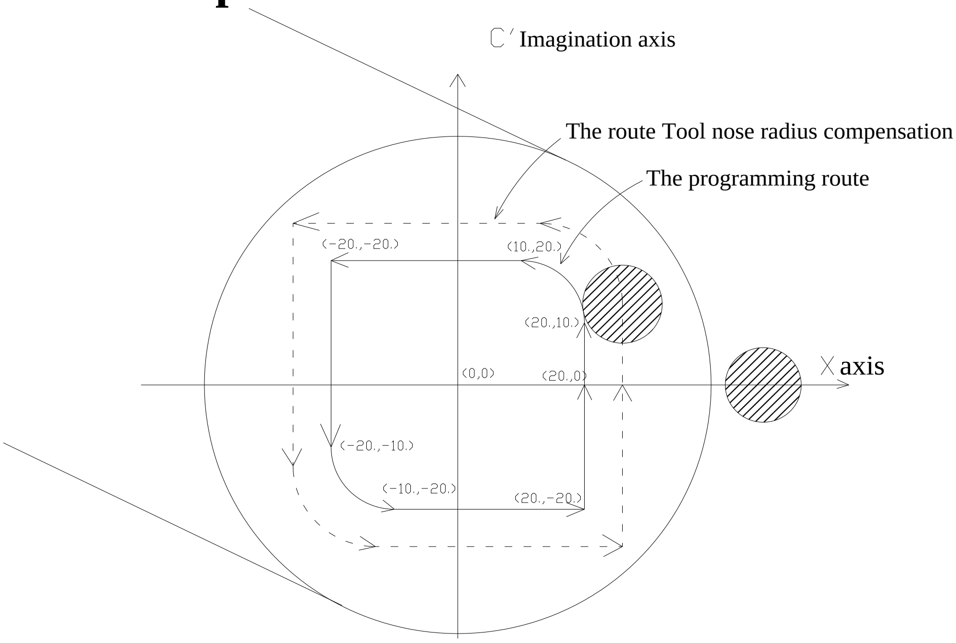
```
G12.1    //Start polar coordinates interpolation
…
…        // (Start linear or circular interpolation in rectangular
…        //   coordinate and rectangular coordinate is composed of
…        //   linear and rotational axes)
…
G13.1    //Cancel polar coordinates interpolation
```

1.9.2 Restriction

The coordinates in polar coordinates interpolation: We should set new working coordinates before G12.1 and the center of the rotational axis is the origin in the coordinate. With G12.1 we must not change coordinates (G92 ,G52 ,G53 ,G54~G59 and so on.).
The instructions of the tool nose radius compensation, With G41 or G42 mode active. We cannot start or cancel G12.1 or G13.1. Only after G40(tool nose radius compensation cancel) can we start or cancel G12.1 or G13.1.
Program restart: We cannot execute “restart” in the programs with G12.1.

The programming of diameter and radius:
We program both linear axes(X) and rotational(C) with the programming of radius.

1.9.3 Example



T0101
G00 X110. C0 Z_ //positioning
G40 G94
G12.1 //start polar coordinates interpolation use Cartesian //coordinate X－C plane edit program
G42 G01 X20. F_
C10.
G03 X10. C20. R10.
G01 X-20.
C-10.
G03 X-10. C-20. R10.
G01 X20.
C0
G40 X110.
G13.1 //cancel polar coordinates interpolation
M30

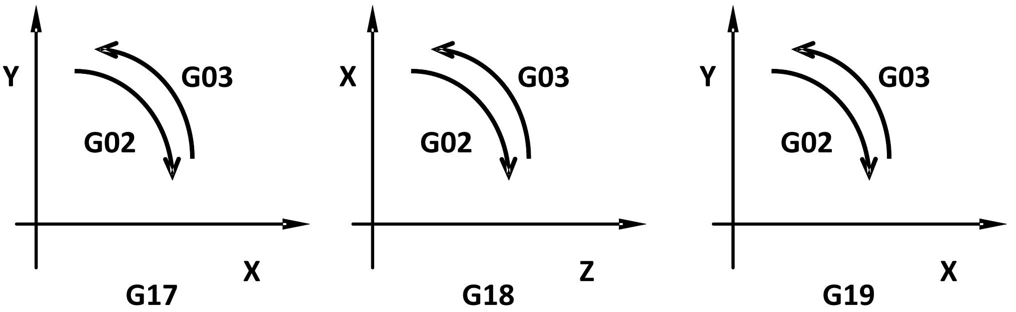
1.10Plane Selection (G17/G18/G19)

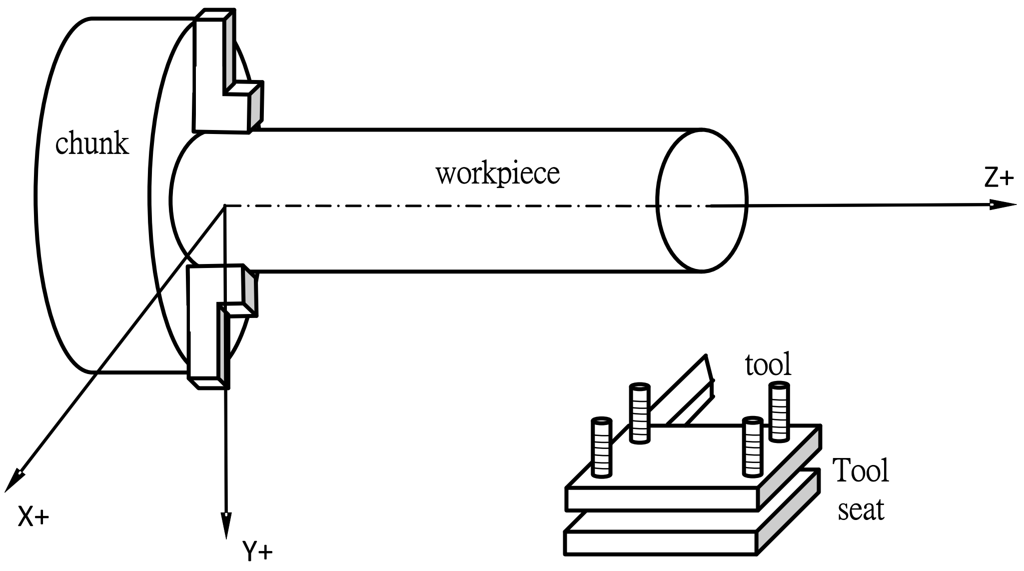
We must make prior change when using circular interpolation command tool and radius compensation command. G17 ,G18 ,G19 activates different planes. The controller will only process on the selected cutting plane.

1.10.1 Format

G17 XY plane selection
G18 ZX plane selection ←controller default planeG19
YZ plane selection

1.10.2 PIC





1.11 Outer(Internal) Diameter Cutting Cycle (G20)

G20 can be used in outer (internal)diameter cutting and taper cutting cycle. By cycle function, we can use only one block to repeat the program, thus simplify the process.

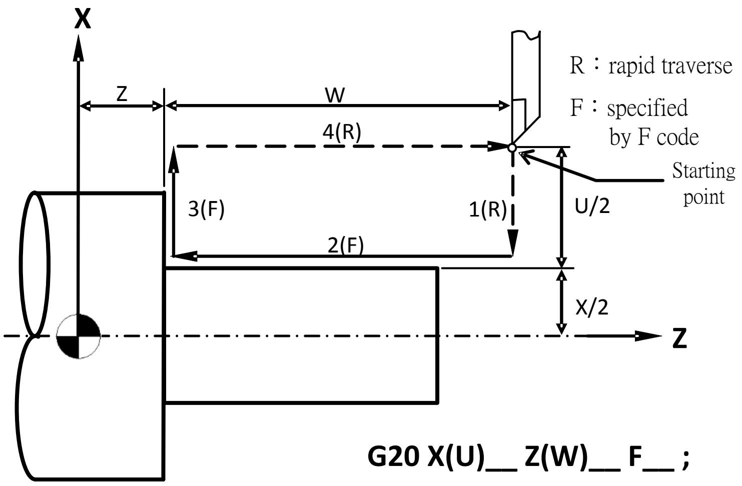
1.11.1 Format

1. Linear cutting cycle: G20 X(U)_Z(W)_F_
2. taper cutting cycle: G20 X(U)_Z(W)_R_F_

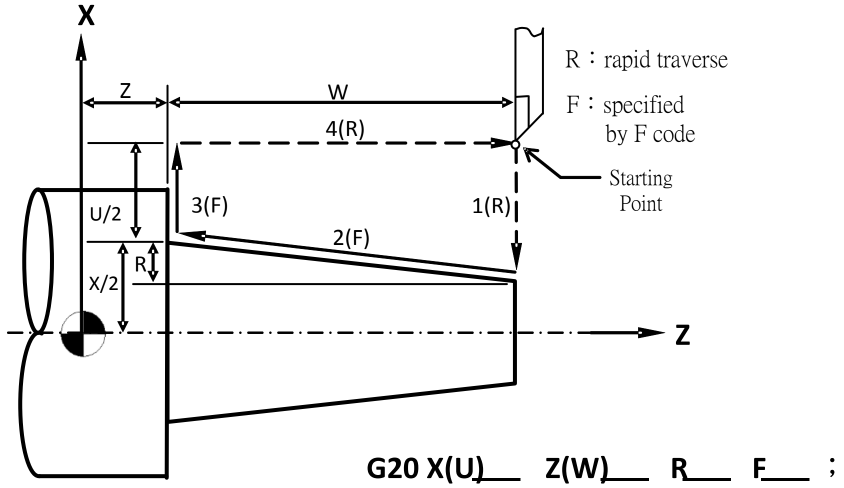
X ,Z: end position of cutting(absolute)
U ,W: end position of cutting(incremental)
R: difference radius value between starting point and end point
F: feedrate

1.11.2 PIC

1.11.2.1 Linear cutting cycle



1.11.2.2 Taper cutting cycle

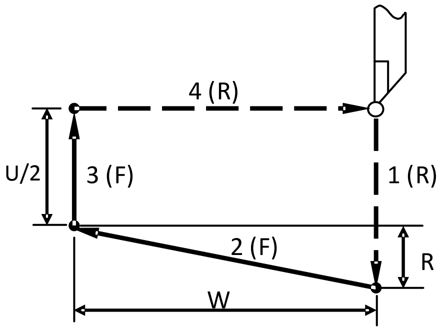


1.11.3 Action description

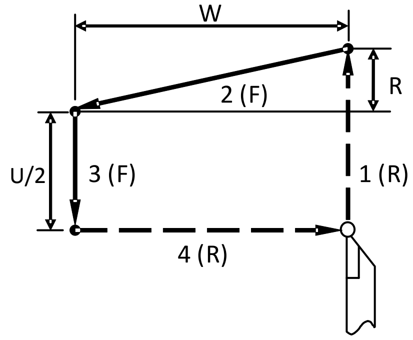
Position the tool to start point before cycle starts.
After executing G20 command, tool move to specified X(U) position in X direction.
The tool starts cutting to the specified X(U) ,Z(W) position in specified federate.
Tool rapidly return to start point once finish each cutting cycle.
After reaching the start point, tool will repeat cutting in the path by changed X(U) value
When cutting to specified size, the tool will stop at start point.

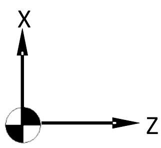
※ When using increment mode, the relationship of U ,W ,R(plus or minus) and the tool path as below:

(a). U < 0，W < 0，R < 0

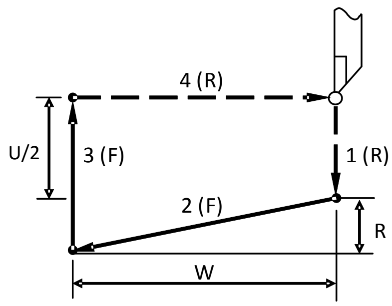


(b). U > 0，W < 0，R > 0

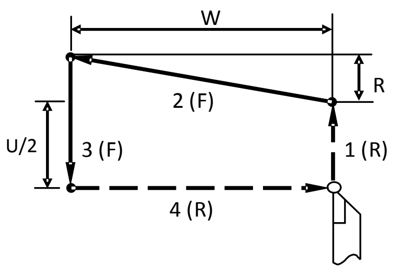




(c). U < 0，W < 0，R > 0，at | R | ≦ | U/2 |

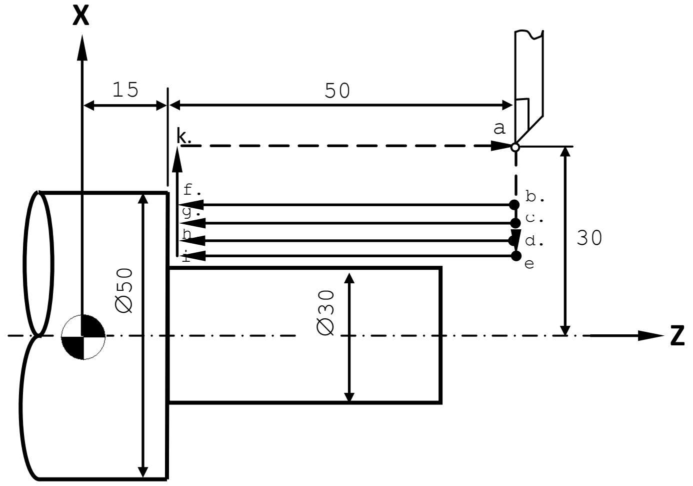


(d). U > 0，W < 0，R > 0，at | R | ≦ | U/2 |



1.11.4 Example 1

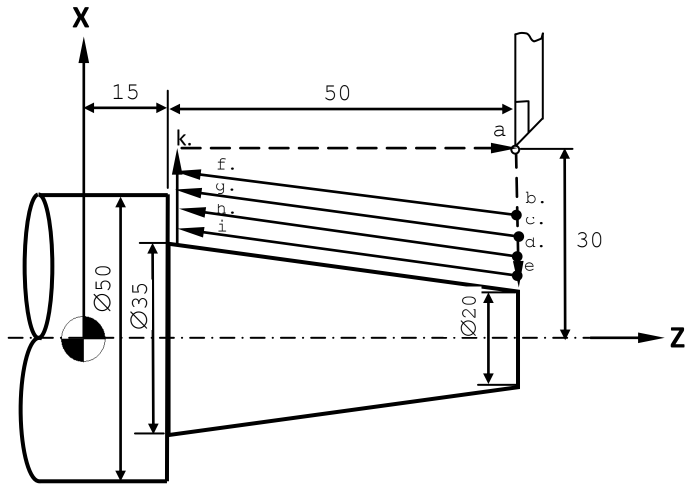
Straight cutting cycle



G92 S5000	//max. speed 5000 rpm
T01	//use tool NO. 1
G96 S130 M03	//constant surface speed, surface
speed	
	//130 m/min, spindle rotate CW
M08	//cutting liquid ON
G00 X60.0 Z65.0	//positioning to **a.**(starting
point)	
G20 X45.0 Z15.0 F600	//execute straight cutting cycle,
feedrate	
	//600 μm/rev, a.→b.→f.→k.→a.
X40.0	//a.→c.→g.→k.→a.
X35.0	//a.→d.→h.→k.→a.
X30.0	//a.→e.→i.→k.→a.
N007 G28 X60.0 Z70.0	//positioning to specified mid-point
then	
	//return to machine zero point
M09	//cutting liquid OFF
M05	//spindle stops
M30	//program ends

1.11.5 Example 2

Taper cutting cycle




```
G92 S5000                      //max. speed 5000 rpm
T01                            //use tool NO.1
G96 S130 M03                   //constant surface speed,
surface
                               //speed 130m/min, spindle
rotate
                               //CW
M08                            //cutting liquid ON
G00 X60.0 Z65.0                   //positioning to a.(starting
point)
G20 X53.0 Z15.0 R-7.5 F600        //Taper cutting cycle,
feedrate 600
                               //μm/rev, a.→b.→f.→k.→a.
X48.0                          //a.→c.→g.→k.→a.
X42.0                          //a.→d.→h.→k.→a.
X35.0                          //a.→e.→i.→k.→a.
G28 X60.0 Z70.0                   // positioning to specified
mid-point
                               //then returns to machine zero
point
M09                            //cutting liquid OFF
M05                            //spindle stops
M30                            //program ends
```

1.12Thread Cutting Cycle (G21)

G21 command is thread cutting cycle. It simplifies many repeating thread cutting blocks into one single block.

1.12.1 Format

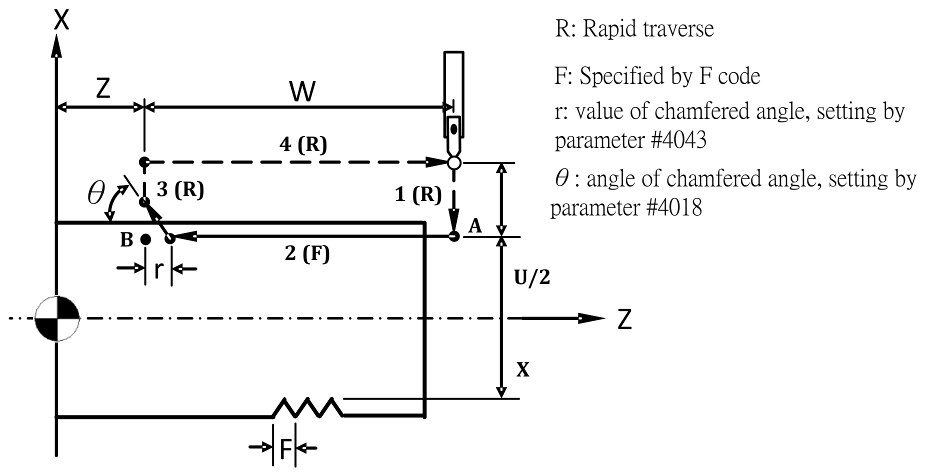
Straight thread cutting cycle:
G21 X(U)_Z(W)_H___ (F___ or E___)

Taper thread cutting cycle:
G21 X(U)_Z(W)_R_H___ (F___ or E___)

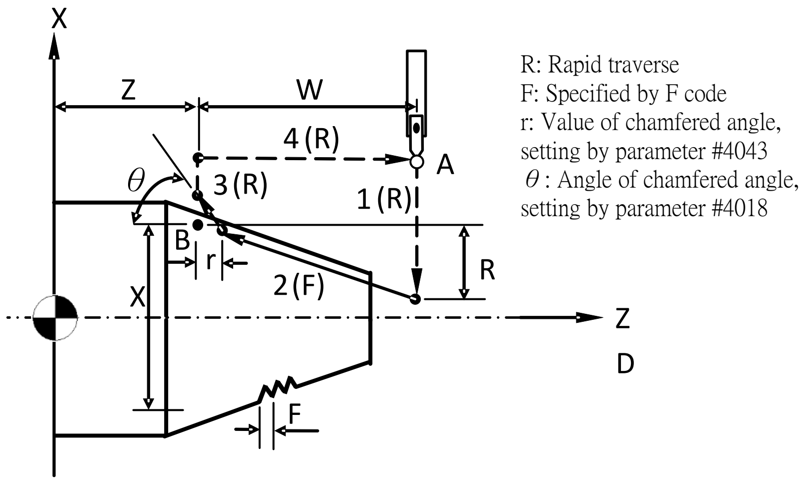
X, Z: end point of cutting (absolute)
U, W: end point of cutting (incremental)
R: difference radius value between starting point and end point
F: screw lead of Metric system (unit: mm/tooth)
E: screw lead of English system(unit: tooth/mm)
H: number of teeth (ex: H3 for cutting the screw of 3 teeth type. In case of H command, F: pitch of teeth)

1.12.2 PIC

1. Straight thread cutting cycle: G21 X(U)_Z(W)_F_



2. Taper thread cutting cycle: G21 X(U)_Z(W)_R_F_

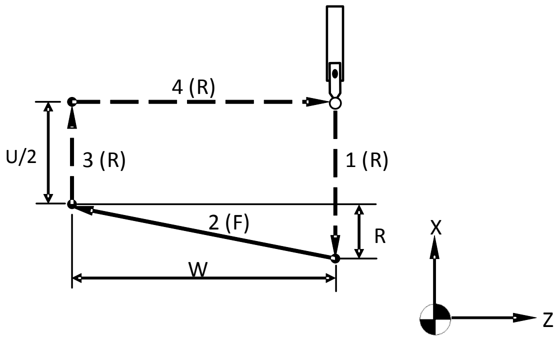

X
Z
W
4 (R)
A
θ
3 (R)
1 (R)
B
r
R
2 (F)
X
Z
D
F
R: Rapid traverse
F: Specified by F code
r: Value of chamfered angle, setting by parameter #4043
θ : Angle of chamfered angle, setting by parameter #4018

1.12.3 Action description

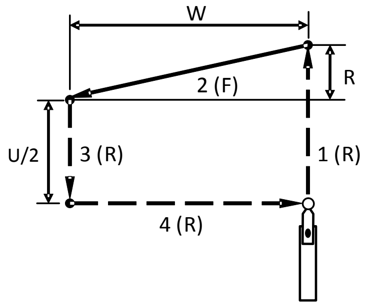
Positioning the tool to start point before cycle starts.
After executing G24 command, tool moves along the X axis direction and reaches to the specified X(U) position.
Then tool start cutting to the specified X(U) ,Z(W) by specified F code
4. The tool returns to start point after cutting.
5. After reaching to the starting point, tool will repeat cutting in the path by changed X(U) value(the changed value is the value that we cutting each time, it can reference tool feed value table in G33).
6. When cutting to specified size, the tool will stop at starting point.

※ When we use increment mode, the relationship of U ,W ,R(plus or minus) and the tool path are as below:

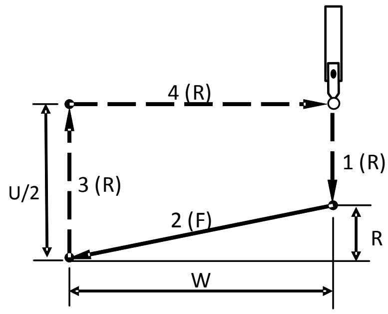
(a). U < 0，W < 0，R < 0



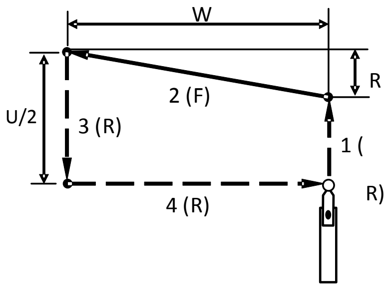
(b). U > 0，W < 0，R > 0



(c). U < 0，W < 0，R > 0，at | R | ≦ | U/2 |



(d). U > 0，W < 0，R > 0，at | R | ≦ | U/2 |



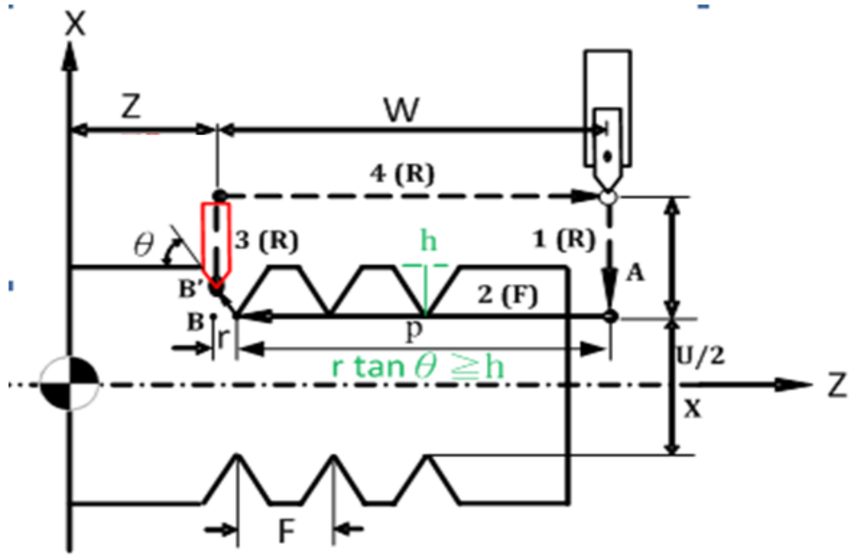
1.12.4 Notice

1. From version 10.114.56E/10.116.0E/10.116.5 (included), the spindle override of entire thread-cutting cycle is locked at the value of the start of cycle, i.e., the spindle override button is in vain during thread-cutting cycle.
2. Before version 10.114.56E/10.116.0E/10.116.5, during thread-cutting cycle, the spindle override is locked at 100% when cutting and resume to setting of control panel while retracting. Therefore, one apply

thread-cutting cycle with a spindle override that is not equal to 100% will find the spindle is under a frequent acceleration and deceleration situation.

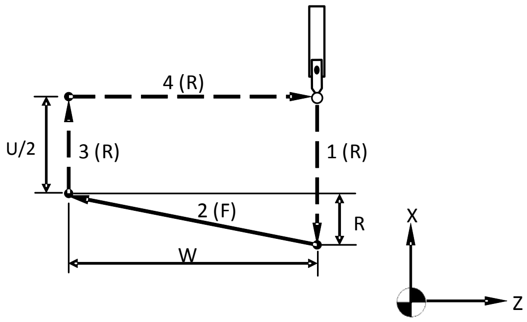
3. The parameter Pr4018 - Chamfer angle (θ) of thread cutting G21, must be set as the actual cutting tool's degree, e.g. the actual cutting tool is 60 degrees, so Pr4018 is set as 60.

4. The setting of parameter Pr4043 - Chamfer amount for threading must meet the condition r*tan $\theta \geqq$ h (where h is the depth). If r is set too large, it will shorten the total thread length (p), as W = r+p In contrast , if r is set too small, it will cause the retraction end point B' appear on the last tooth, hence the last tooth will be relatively lower than the others (see the graph below).

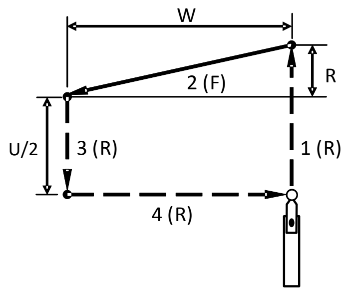


5. Under the incremental mode, the definition of the signs of values in U, W, and R as well as the relation between the cutting tools and the cutting path is as follows:

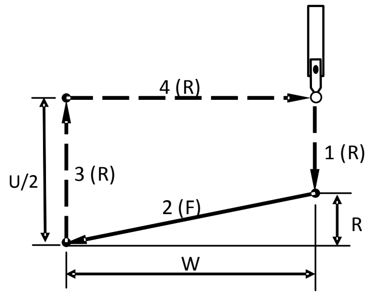
(a). U < 0，W < 0，R < 0



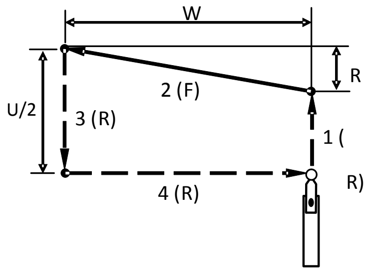
(b). U > 0，W < 0，R > 0



(c). U < 0，W < 0，R > 0，at | R | ≦ | U/2 |

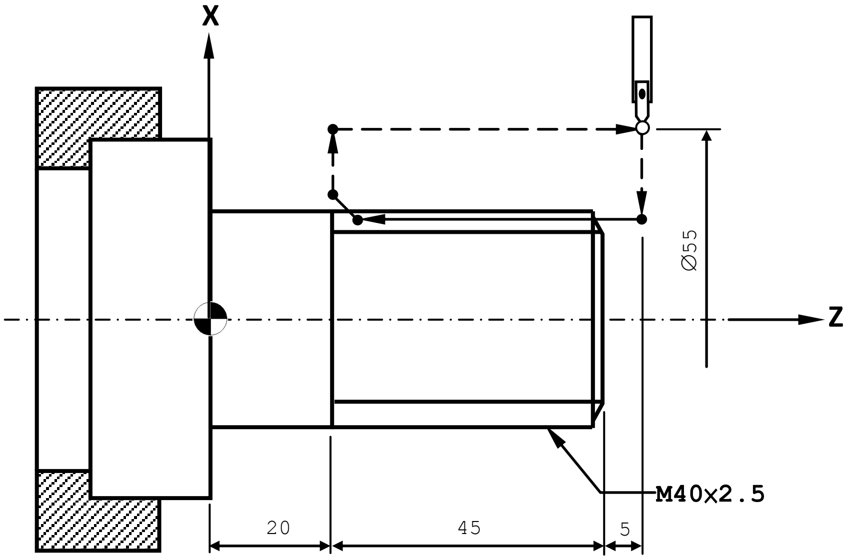


(d). U > 0，W < 0，R < 0，at | R | ≦ | U/2 |



1.12.5 Example 1

Straight thread cutting cycle, 3 teeth type

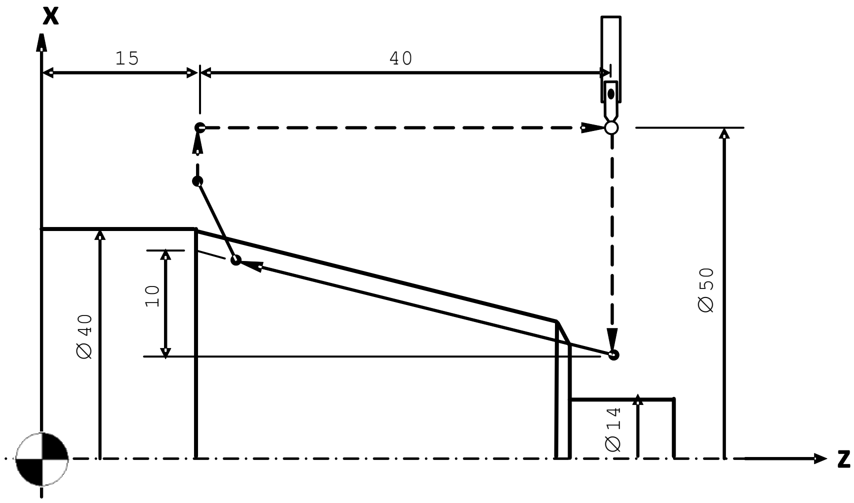


T03 //use tool NO.3
G97 S600 M03 //constant rotate speed, 600 rpm CW
G00 X50.0 Z70.0 //positioning to the starting point of cycle
M08 //cutting liquid ON
G21 X39.0 Z20.0 H3 F2.5 //execute thread cutting, 3 teeth type,
//first cycle

X38.3 **//second cycle**
X37.7 **//third cycle**
X37.3 **//fourth cycle**
X36.9 **//fifth cycle**
X36.75 **//sixth cycle**
G28 X60.0 Z75.0 //positioning to specified mid-point and
//return to machine zero point
M09 //cutting liquid OFF
M05 //spindle stops
M30 //program ends

1.12.6 Example 2

Taper thread cutting cycle, single tooth type



T03 //use tool NO.3
G97 S600 M03 //constant rotate speed, 600 rpm CW
G00 X50.0 Z55.0 //positioning to the starting point of cycle
M08 //cutting liquid ON
G21 X39.0 Z15.0 R-10.0 F2.5 //execute taper thread cutting cycle,
//first cycle
X38.3 **//second cycle**
X37.7 **//third cycle**
X37.3 **//fourth cycle**
X36.9 **//fifth cycle**
X36.75 //sixth cycle
G28 X60.0 Z70.0 // positioning to specified mid-point and

```
                              //return to machine zero point
M09                           //cutting liquid OFF
M05                           //spindle stops
M30                           //program ends
```

1.13End Face Turning Cycle (G24)

G24 command is end face cutting cycle. It simplifies several repeating end face cutting blocks into one single block.

1.13.1 Format

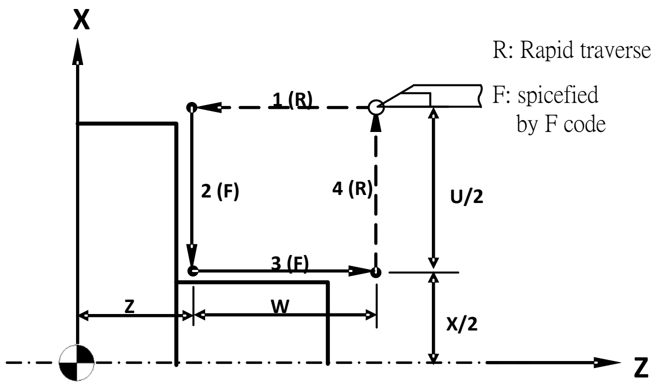
1. Straight end face cutting cycle: G24 X(U)_Z(W)_F_
2. Taper end face cutting cycle: G24 X(U)_Z(W)_R_F_

X ,Z: end position of cutting(absolute)
U ,W: end position of cutting(incremental)
R: difference length from starting point to end point
F: feedrate

1.13.2 PIC

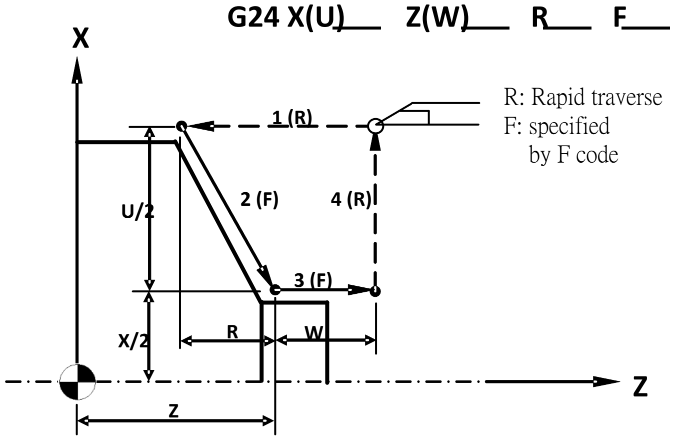
1.13.2.1 Straight end face cutting cycle

G24 X(U)___ Z(W)___ F___ ;



1.13.2.2 Taper end face cutting cycle

G24 X(U)___ Z(W)___ R___ F___ ;



1.13.3 Action description

Positioning the tool to start point before cycle starts.

After executing G24 command, the tool will move along Z direction and reach the specified Z(W) position.

Then the tool will cut to specified X(U) ,Z(W) by specified feedrate.

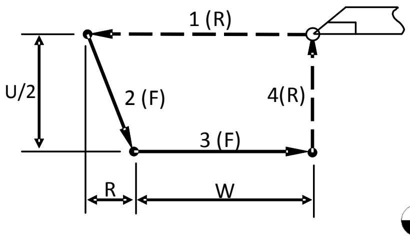
After finishing cutting, the tool returns to start point.

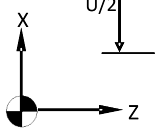
After reaching to the start point, tool will repeat cutting in the path by changed Z(W) value.

When reaching to the specified size, the tool will stop at start point.

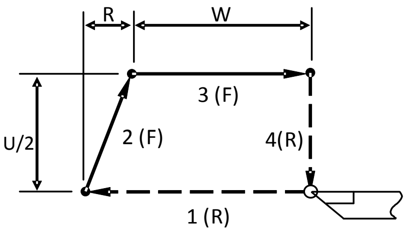
※ when we use increment mode, the relationship of U ,W ,R(plus or minus) and the tool pathare as below:

(a). $U < 0$，$W < 0$，$R < 0$

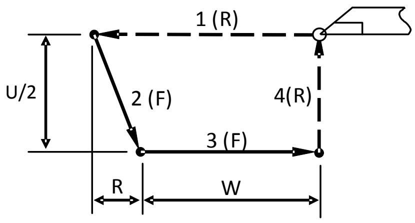




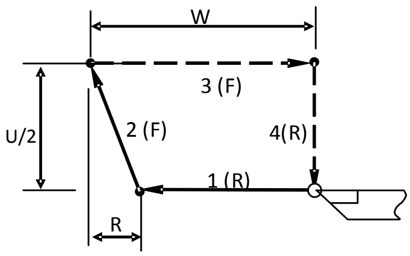
(b). $U > 0$，$W < 0$，$R > 0$



(c). $U < 0$，$W < 0$，$R > 0$，at $|R| \leqq |W|$

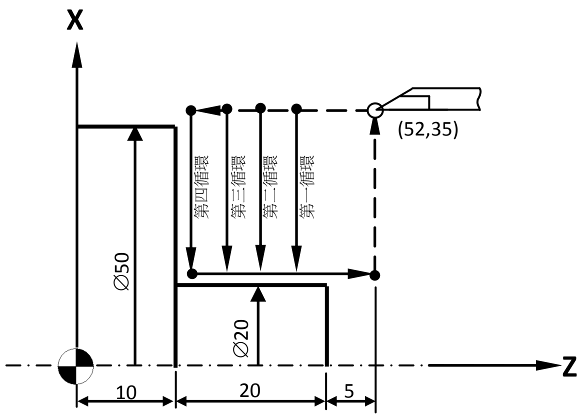


(d). $U > 0$，$W < 0$，$R > 0$，at $|R| \leqq |W|$



1.13.4 Example 1

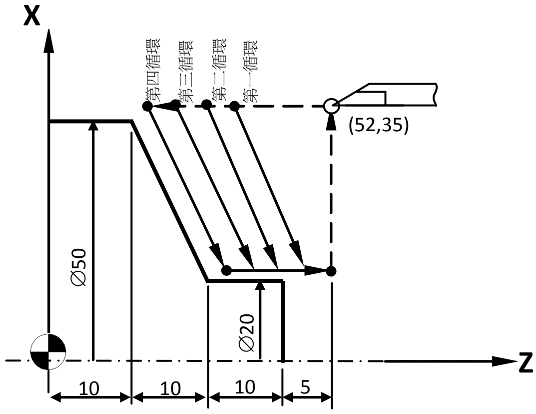
Straight end face cutting cycle.



G92 S3000 //max. rotate speed 3000 rpm
T01 //use tool NO. 1
G96 S130 M03 //constant surface speed, surface speed
//130 m/min
M08 //cutting liquid ON
G00 X52.0 Z35.0 //positioning to starting point of cycle
G24 X20.0 Z25.0 F600 **//execute straight end face cutting,**
//feedrate 600 µm/rev, first cycle
Z20.0 **//second cycle**
Z15.0 **//third cycle**
Z10.0 **//fourth cycle**
G28 X70.0 Z40.0 //positioning to specified mid-point, then
//return to the machine zero point
M09 //cutting liquid OFF
M05 //spindle stops
M30 //program ends

1.13.5 Example 2

Taper end face cutting cycle



G92 S3000 //max. rotate speed 3000 rpm
T01 //use tool NO.1
G96 S130 M03 //constant surface speed, surface speed 130 m/min
M08 //cutting liquid ON
G00 X52.0 Z35.0 //positioning to starting point of cycle
G24 X20.0 Z32.0 R-10.0 F600
//execute taper end face cutting cycle, feedrate 600 μm/rev, first cycle
Z28.0 //second cycle
Z24.0 //third cycle
Z20.0 //fourth cycle
G28 X70.0 Z35.0
//positioning to specified mid-point, then return to machine zero point
M09 //cutting liquid OFF
M05 //spindle stops
M30 //program ends

1.14Reference point return (G28)

When G28 command is executed, tool will move to specified intermediary point and then return to reference point(machine zero point) by the speed of G00 To prevent interference between tool and workpiece, G28 keep tool clear of the workpiece when returning. In absolute mode, it is the absolute value to the intermediary-point in increment mode, it is the increased value from start point to intermediary point.

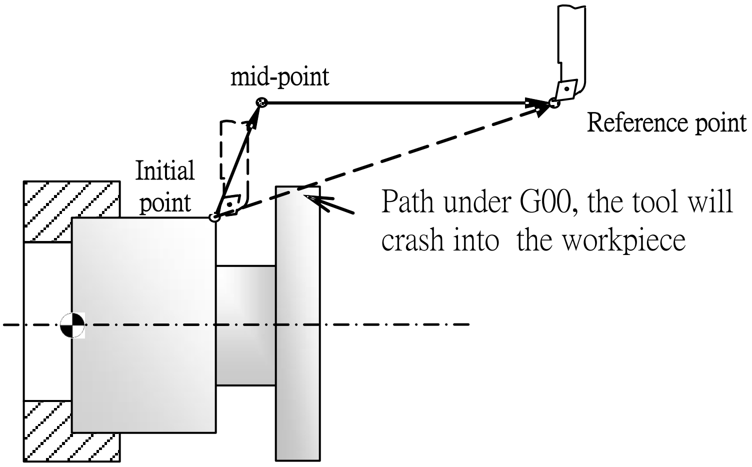
※ Notice: We must cancel tool compensation function prior G28 command to insure the returning action correct.

1.14.1 Format

G28 X(U)_Z(W)_

X ,Z: specified mid-point(absolute)
U ,W: specified mid-point(incremental)

1.14.2 PIC



1.14.3 Additional remark

If the axis (parameter221~236) is set to be rotated axis, see the attachment “parameter manual” for your reference.

1.15Return from reference point (G29)

G29 command is used in conjunction with Reference point return (G28). Noted that G29 is not allowed to be executed alone. G29 does not specify its own intermediary point, it uses the intermediary point that G28 specifies, so G29 can only be executed after executing G28 command.

G29 returns the CNC to the intermediary point programmed in G28, then to the coordinates programmed in the G29 block. In short, it can move to specified position through intermediary point from reference point.

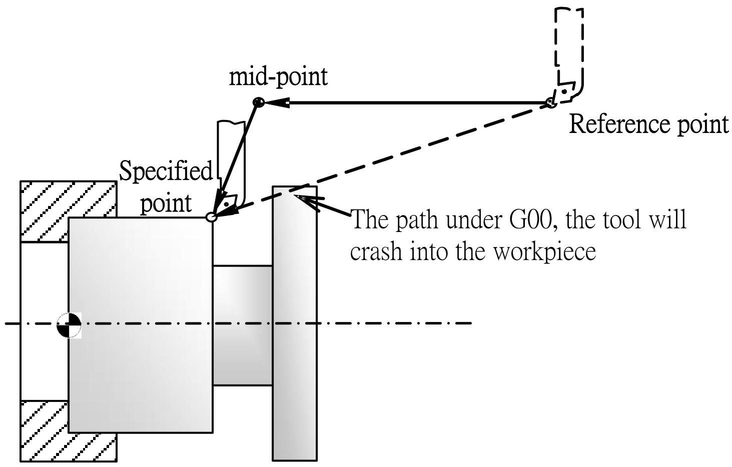
In absolute mode, it is the absolute value to the intermediary point in increment mode, it is the increased value from starting point to intermediary point.

1.15.1 Format

G29 X(U)_Z(W)_

X ,Z: specified point(absolute)
U ,W: specified point(incremental)

1.15.2 PIC



1.16Any reference point return (G30)

In order to be convenient in tool change and inspection, we specify another reference point from the machine zero point by parameter. The machine needs not to return to machine zero point when changing tool, thus increases the changing efficiency. Usage of this command is same as G28 command except the different tool return point. G30 command is usually used in the situation which tool changing position differs from origin. The moving mode is G00 (positioning) mode.

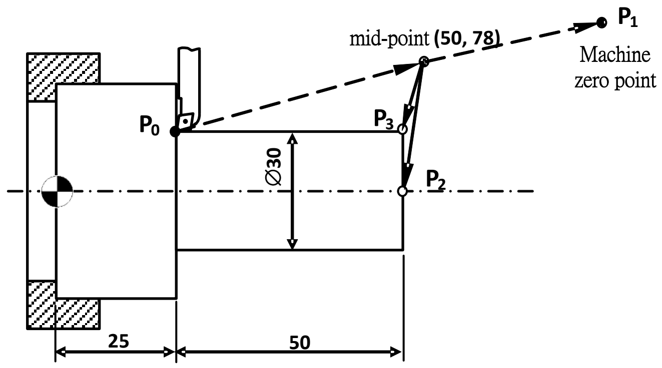
<Notice>this command usually use in auto tool change. We should cancel tool compensation function before executing G30 in safety.

1.16.1 Format

G30 Pn X(U)_Z(W)_

X ,Y ,Z: coordinate value of mid-point
Pn: specify the reference point (setting parameter #2801 ~ #2860)
P1: machine zero point
P2: second reference point
P_default is P2
Description:

1.16.2 Example



Path 1
G30 P01 X50.0 Z78.0 // P_0→mid-point→P_1

Path 2
G30 P02 X50.0 Z78.0 // P_0→mid-point→P_2
Or
G30 X50.0 Z78.0 //default P_2

Path 3

G30 P03 X50.0 Z78.0 // $P_0 \rightarrow$ mid-point $\rightarrow P_3$

1.17 Skip Function (G31)

G31 is to be issued with an associated axis move. When the G31 is executed, it moves at current feedrate selected from G1 until the touch probe selected is deflected. At this point, the move is stopped, and the position where the probe touched the part is read and passed to system variables. Machine will receive the signal then ladder C-bit turns on. Skip function is used in unknown end program. G31 command will then record the present position of machine whenever be interrupted and terminate the current action, continuing to execute the next block.

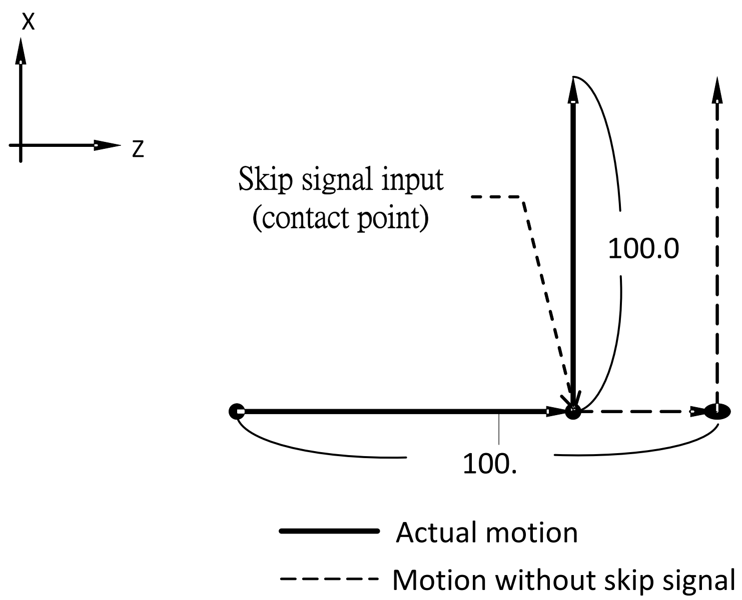
1.17.1 Format

G31 X(U)__ Z(W)__ F__

X ,Z: specified position(absolute)
U ,W: specified position(incremental)
F: feedrate

1.17.2 Example 1

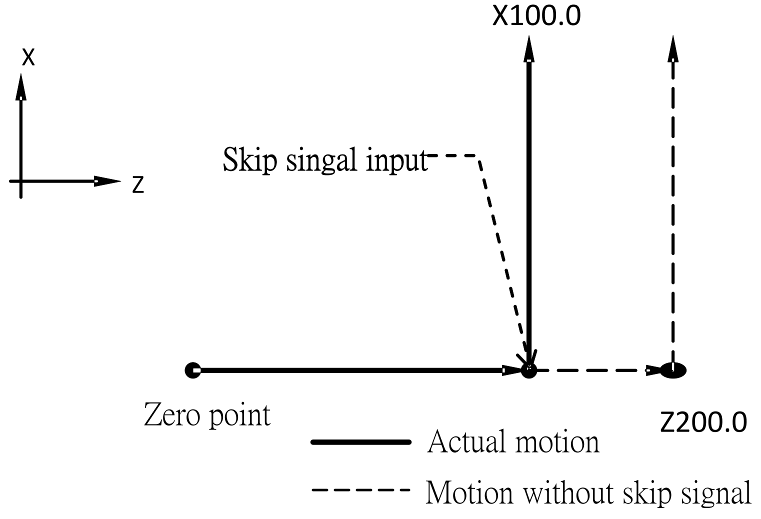
Increment mode



G31 W100.0 F100 //origin path until run into contact point
U100.0 //use contact point to be the relative //coordinate and change the path to specified //position

1.17.3 Example 2

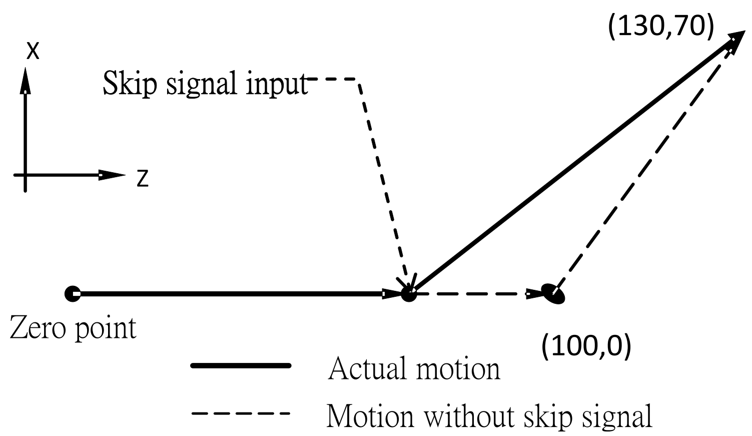
Absolute mode for one axis



G31 Z200.0 F100 // origin path until run into contact point
X100.0 // use contact point to be the relative coordinate
//and change the path to specified position

1.17.4 Example 3

Absolute mode for two axes



G31 Z100.0 F1000 // origin path until run into contact point
Z130.0 X70.0 // use contact point to be the relative coordinate
//and change the path to specified position

1.17.5 Additional Remark

To avoid several G31 command be skipped simultaneously, in conjunction with PLC , the C-bit should be positive edge triggered.

1.18Thread cutting (G33)

G33 command executes endface thread cutting, taper threading, and straight thread cutting. It based on spindle rotates and tool feed executing synchronously.

1.18.1 Format

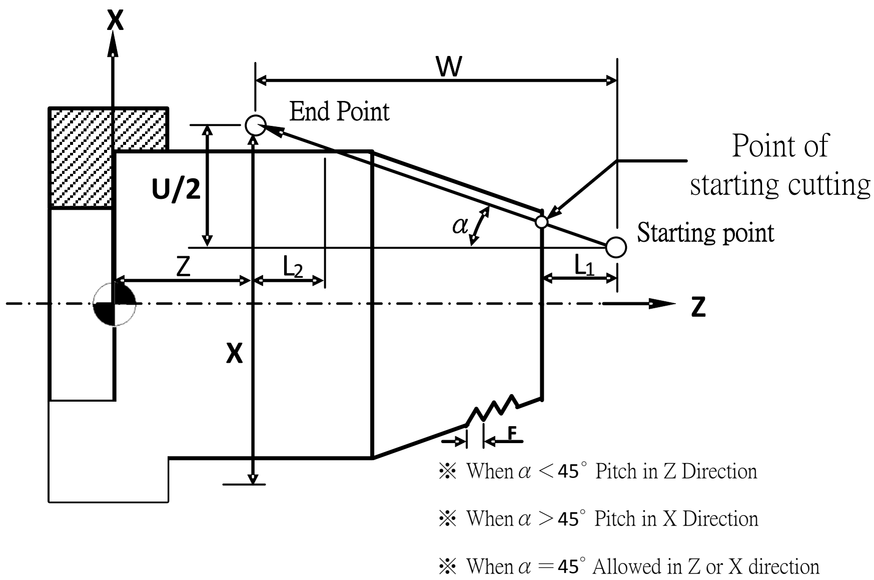
Straight thread cutting:
G33 Z(W)_Q___ (F___ or E___)

Taper threading:
G33 X(U)_Z(W)_Q___ (F___ or E___)

Endface thread cutting:
G33 X(U)_Q___ (F___ or E___)

X ,Z: specified position(absolute)
U ,W: specified position(incremental)
F: lead in longitudinal direction←common thread ,Metric system
E: lead in longitudinal direction←pricise thread ,English system
Q: the shift of the threading start angle, this function can let all the tool start point be the same when cutting rotating workpiece and usually use in multiple-thread cutting. (range: 0.001~360.000°) When single-thread cutting, we can ignore the Q argument and apply default value Q= 0°
(range：0.001~360.000°)

1.18.2 PIC



1.18.3 Notice

Input unit and modal of E ,F value as below table: table 1. Metric system ,table 2. English system

Input unit	A(0.01mm)			B(0.001mm)			C(0.0001mm)		
Command position	F(mm/rev)	E(mm/rev)	E(pc/inch)	F(mm/rev)	E(mm/rev)	E(pc/inch)	F(mm/rev)	E(mm/rev)	E(pc/inch)
Min. command unit	1(-0.001) (1, -1.000)	1(-0.0001) (1, -1.0000)	1(-1) (1.-1.0)	1(-0.0001) (1.-1.0000)	1(-0.00001) (1.-1.00000)	1(-1) (1.-1.00)	1(-0.00001) (1.-1.00000)	1(-0.000001) (1.-1.000000)	1(-1) (1.-1.000)
Command range	0.001～9999.999	0.0001～9999.9999	0.1～9999999.9	0.001～999.9999	0.00001～999.99999	0.01～999999.9	0.00001～99.99999	0.000001～99.999999	0.001～99999.999

Table 1. input by Metric system

Input unit	A(0.00inch)			B(0.0001inch)			C(0.00001inch)		
Command position	F(inch/rev)	E(inch/rev)	E(pc/inch)	F(inch/rev)	E(inch/rev)	E(pc/inch)	F(inch/rev)	E(inch/rev)	E(pc/inch)
Min. command unit	1(-0.00001) (1, -1.00000)	1(-0.000001) (1, -1.000000)	1(-1) (1.-1.000)	1(-0.000001) (1.-1.000000)	1(-0.0000001) (1.-1.0000000)	1(-1) (1.-1.0000)	1(-0.0000001) (1.-1.0000000)	1(-0.00000001) (1.-1.00000000)	1(-1) (1.-1.00000)
Command range	0.00001～999.99999	0.000001～99.999999	0.001～99999.999	0.000001～99.999999	0.0000001～9.9999999	0.0001～9999.9999	0.0000001～9.9999999	0.00000001～0.99999999	0.00001～999.99999

Table 2. input by English system

【Note 1】If the converted feedrate is greater than Max. cutting feedrate, the pitch will vary. Thus the pitch is not the originally specified one. Tilt thread cutting command and spiral thread cutting command are unavailable in constant surface speed mode.

The spindle speed should be fixed from coarse cutting to fine cutting.

If we use dwell in thread cutting, the thread will be damaged. So we cannot use dwell when thread cutting. If we press the dwell button, the thread cutting will be terminated (not in G33 mode) and will stop at the next block.

In the beginning of thread cutting, the varying cutting feed rate will be compared with the limitation of cutting speed. The alarm of error operation will occur if the speed limitation is exceeded. 【Note 1】

In the thread cutting, it is possible that the varying cutting speed exceeds the limitation of cutting speed for keeping the constant pitch.

The limitation of spindle speed is as below:

$$1 \leqq \text{Revolution}(\mathbf{R}) \leqq \frac{\text{Max feedrate}}{\text{Lead of thread}}$$

R: spindle rotate speed (rpm)

Lead of thread (F): mm or inch

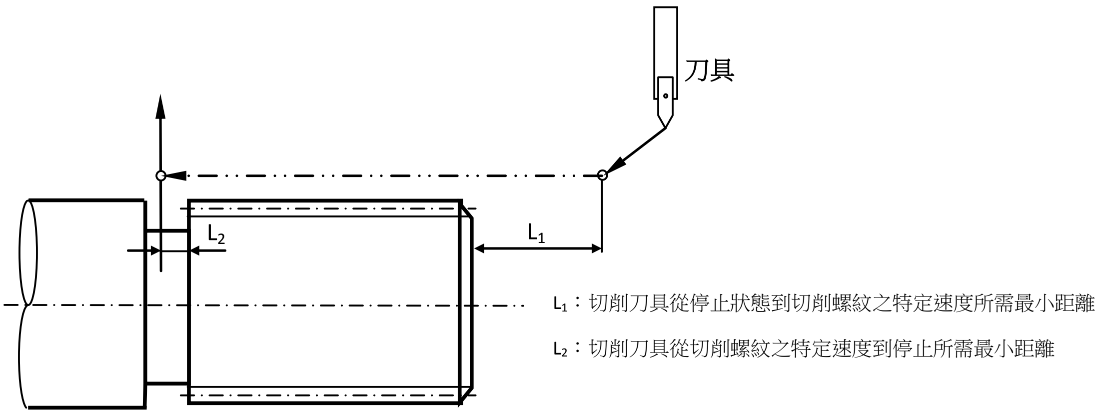
Feedrate: mm/min or inch/min

Around the start and end thread cutting point, incorrect pitch length will occur due to the servo lag. Therefore the thread length we want should be the specified thread length (L_1 ,L_2) plus the thread length.

※L_1 ,L_2 formula are as below:

$$L_1 \approx \frac{S \times P}{400}$$

$$L_2 \approx \frac{S \times P}{1800}$$



The external speed control is effective during the thread cutting, but the feedrate of external speed control cannot synchronize with spindle revolution cannot be in synchrony.

In non-synchronous feed(G94) command, the thread cutting command will become synchronous feed type.

During the thread cutting, manual adjustment of spindle speed is also effective. If manually adjust the speed during thread cutting, an incorrect thread cutting due to delay of servo system may also occur.

When thread cutting command executed during tool nose radius compensation, tool nose radius compensation will be temporarily canceled.

During the G33 command, thread cutting will be canceled if changing to other automatic modes. Automatic spinning will be terminated after executing a block.

During the G33 command, thread cutting will be canceled if change to manual mode, Automatic spinning will be terminated after a block.

During the spinning of single block, thread cutting will also be canceled. Automatic spinning will be terminated after executing a block.

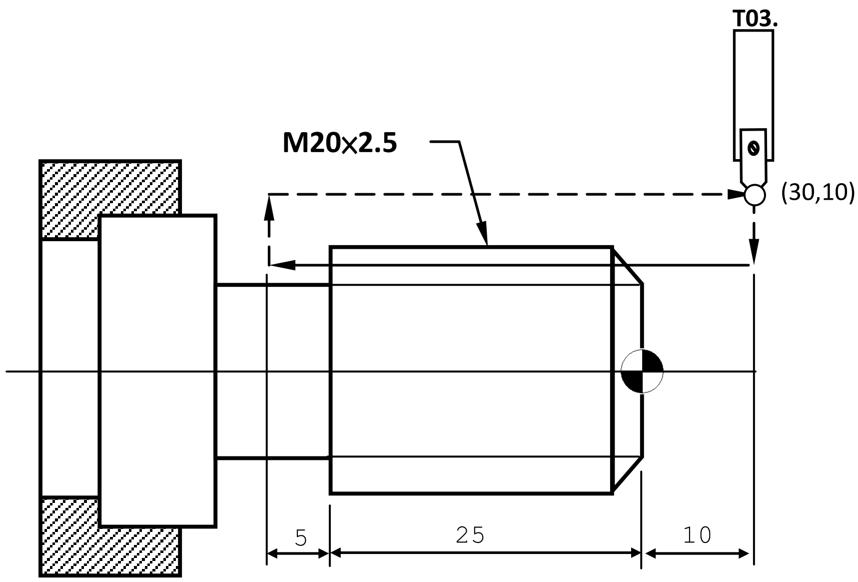
During the thread cutting, it begins to move till the appearance of synchronous signal per one revolution from spinning encoder. But in case of a thread cutting in a system which there is another thread cutting command, it will start to move instead for waiting the appearance of synchronous signal per revolution from backward encoder. Therefore, do not execute duplicated system of thread cutting command.

Tool feed value of thread cutting reference table:

English system depth of tooth h = 0.6403P P = Pitch								
Thread number per inch		8	10	12	14	16	18	24
Pitch of thread(in)		0.1250	0.1000	0.0833	0.0714	0.0625	0.0556	0.0417
Height of thread 0.6403P(in)		0.0800	0.0640	0.0533	0.0457	0.0400	0.0356	0.0267
Numbers of cutting and the value of cutting(diameter)	1	0.0472	0.0394	0.0354	0.0315	0.0315	0.0315	0.0315
	2	0.0276	0.0276	0.0236	0.0236	0.0236	0.0236	0.0157
	3	0.0236	0.0236	0.0236	0.0197	0.0197	0.0118	0.0062
	4	0.0200	0.0157	0.0157	0.0118	0.0052	0.0043	
	5	0.0200	0.0157	0.0083	0.0048			
	6	0.0158	0.0060					
	7	0.0058						

Metric system depth of tooth = 0.06495P P = Pitch								
Pitch of thread(mm)		4.0	3.5	3.0	2.5	2.0	1.5	1.0
Height of thread 0.6495P(mm)		2.598	2.273	1.949	1.624	1.299	0.974	0.650
Numbers of cutting and the value of cutting(diameter)	1	1.5	1.5	1.2	1.0	0.9	0.8	0.7
	2	0.8	0.7	0.7	0.7	0.6	0.6	0.4
	3	0.6	0.6	0.6	0.6	0.6	0.4	0.2
	4	0.6	0.6	0.4	0.4	0.4	0.16	
	5	0.4	0.4	0.4	0.4	0.1		
	6	0.4	0.4	0.4	0.15			
	7	0.4	0.2	0.2				
	8	0.3	0.15					
	9	0.2						

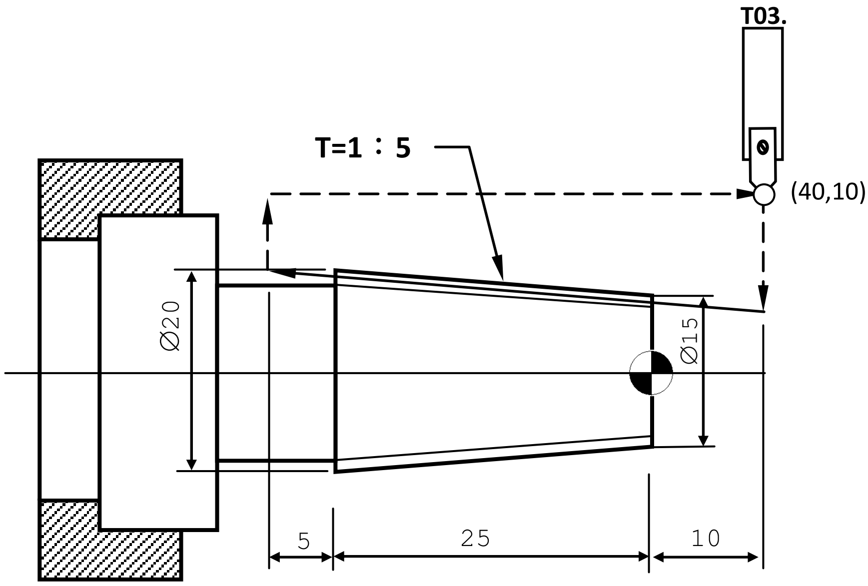
1.18.4 Example 1



```
T03                 //use tool NO.3
G97 S1000 M03       //spindle rotate CW 1000 rpm, constant rotate
                    //speed
M08                 //cutting liquid ON
G00 X30.0 Z10.0     //positioning to starting point of cutting
  X19.0             // First cutting 1.0 mm
G33 Z-30.0 F2.5
 G00 X30.0
  Z10.0
   X18.3            // Second cutting 0.7 mm
G33 Z-30.0 F2.5
G00 X30.0
  Z10.0
  X17.7             // Third Sixth cutting 0.6 mm
G33 Z-30.0 F2.5
G00 X30.0
  Z10.0
  X17.3             // Fourth Sixth cutting 0.4 mm
G33 Z-30.0
G00 X30.0
  Z10.0
  X16.9             // Fifth Sixth cutting 0.4 mm
G33 Z-30.0 F2.5
G00 X30.0
  Z10.0
    X16.75          // Sixth cutting 0.15 mm
G33 Z-30.0 F2.5
G00 X30.0
  Z10.0
G28 X50.0 Z30.0     //positioning to specified mid-point, then return to
                    //machine zero point
M09                 //cutting liquid OFF
M05                 //spindle stops
M30                 //program ends
```

1.18.5 Example 2

Pitch = 2.5



T03 //use tool NO.3
G97 S1000 M03 //spindle rotate CW 1000 rpm, constant rotate
//speed
M08 //cutting liquid ON
G00 X40.0 Z10.0 //positioning to starting point of cutting
X12.0 // First cutting 1.0 mm
G33 X20.0 Z-30.0 F2.5
G00 X40.0
Z10.0
X11.3 // Second cutting 0.7 mm
G33 X19.3 Z-30.0 F2.5
G00 X40.0
Z10.0
X10.7 // **Third cutting 0.6 mm**
G33 X18.7 Z-30.0 F2.5 //
G00 X40.0
Z10.0
X10.3 // **Fourth cutting 0.4 mm**
G33 X18.3 Z-30.0 F2.5
G00 X40.0
Z10.0
X9.9 // **Fifth cutting 0.4 mm**
G33 X17.9 Z-30.0 F2.5
G00 X40.0

```
  Z10.0
  X9.75                // Sixth cutting 0.15 mm
G33 X17.75 Z-30.0 F2.5
G00 X40.0
  Z10.0
G28 X50.0 Z30.0      //positioning to specified mid-point, and return to
                     //machine zero point
M09                  //cutting liquid OFF
M05                  //spindle stops
M30                  //program ends
```

1.19Variable lead threading cutting (G34)

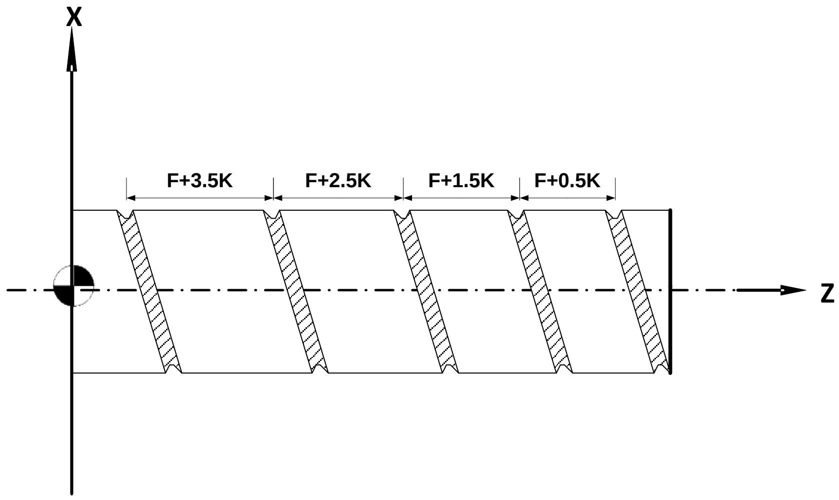
G34 command executes straight thread, taper thread, and endface thread cutting that have variable pitch, based on spindle rotation and tool feed synchronously. (Note. G34 is available in version 10.112.0 or later 9.0 version is unavailable)

1.19.1 Format

(1) Straight thread cutting: G33 Z(W)_Q_F___K___
(2) Taper threading: G33 X(U)_Z(W)_Q_F___K___
(3) Endface thread cutting: G33 X(U)_Q_F___K___

X ,Z: specified position(absolute)
U ,W: specified position(incremental)
F: lead in longitudinal axis direction(at the start point) ←common thread ,Metric system
E: lead in longitudinal direction←pricise thread ,Imperial system
Q: the shift of the threading start angle, this function can let the tool start point the same when cutting rotating workpiece and usually use in multiple-thread cutting. (range: 0.001~360.000°) When single-thread cutting, we can use ignore the Q argument and will apply default value Q= 0°
K: Increment and decrement of lead per spindle revolution.

1.19.2 PIC



1.19.3 Notice

If specified K value cause latter pitch to be negative, an alarm 「Invalid Threading Lead」 will be issued If the feedrate is greater than maximum allowable feedrate, the pitch will decrease and an alarm 「threading block feedrate exceed」 will be issued.

Total move distance in one block: [F+(F+Rev*K)]* Rev/2

Other notices are the same with G33.

1.19.4 Example 1

```
T03                    // use tool no. 3
G97 S1000 M03          //Spindle rotate CW 1000rpm, and the speed
                       //is constant
M08                    //cutting liquid On
G00 X0.0 Z0.0          // G00 move to cutting original point
G34 Z-50.0 F1.0 K0.2   // pitch increase 0.2 per rev to cut
M09                    //cutting liquid Off
M05                    //Spindle stop
M30                    //finish
```

1.19.5 Example 2

```
T03                    //use tool no. 3
G97 S1000 M03          //Spindle rotate CW 1000rpm, and the speed
                       //is constant.
M08                    //cutting liquid On
G00 X0.0 Z0.0          //G00 move to cutting original point
G33Z16F4               //Threading with fix pitch that is 4mm
G34W19F4K5.5           //Pitch increase 5.5mm per rev. Pitch is 4mm
                       //become 15mm.
G33W4F15               //Threading with fix pitch that is 15mm
G34W18F15K-4           //Pitch decrease 4mm per rev. Pitch is 15mm
                       //become 4mm .
G33W12F9               //Threading with fix pitch that is 9mm
M09                    //cutting liquid Off
M05                    //Spindle stop
M30                    //finish
```

1.20Tool Nose Radius Compensation (G41/G42/G40)

A rounded tiny nose on the tool tip increases its strength, the tool life, decreases the stress, help heat radiation and improve the smoothness of surface. It is called tool nose, and its radius is called tool nose radius. But when we use tool nose to cut corner, ,slanting line or an arc, errors will occur because of the arc of tool tip, we cannot perform the exactly shape of workpiece. We can use G41 ,G42 to compute the error of tool nose radius accurately and make adjustment to compensate it.

G code	Function	Position of tool
G40	Tool nose compensation cancel	Tool moves along the path of program
G41	Tool nose compensation (left)	Tool offsets right a specified value to the path of program
G42	Tool nose compensation (right)	Tool offsets left a specified value to the path of program

1.20.1 Format

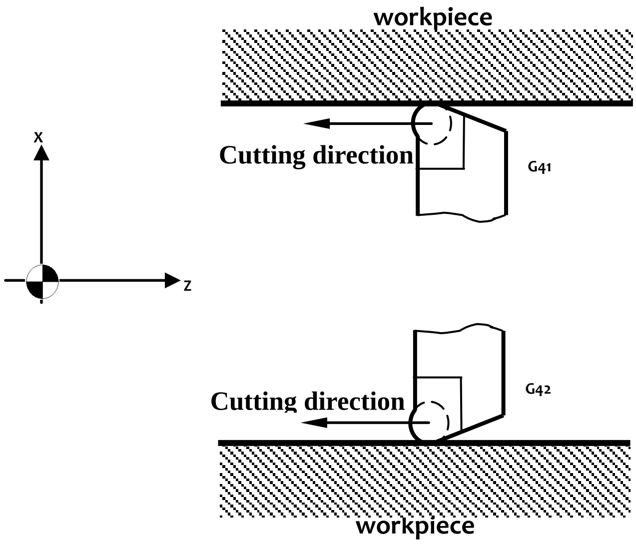
G41 X(U)___ Z(W)___
G42 X(U)___ Z(W)___
G42

G40 compensation cancel

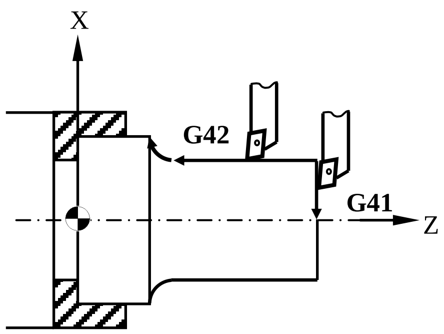
X ,Z: specified position(absolute)
U ,W: specified position(incremental)

1.20.2 PIC

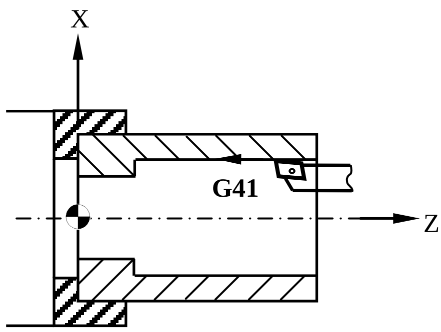
1.20.2.1 Relationship between tool feed direction and workpiece, setting of compensation:



1.20.2.2 Compensation setting of actually perform

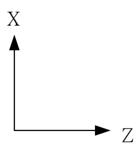


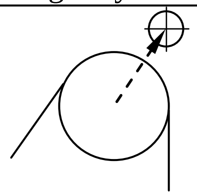
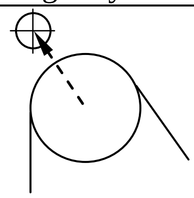
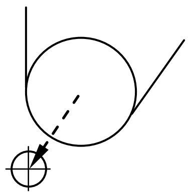
a. cutting outer radius and end surface

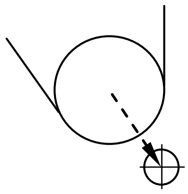
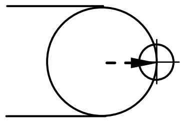
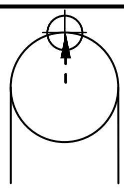
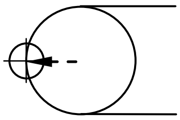
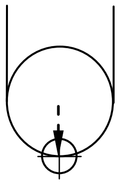
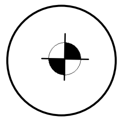


b. cutting internal radius

1.20.2.3 Imaginary tool nose number setting:

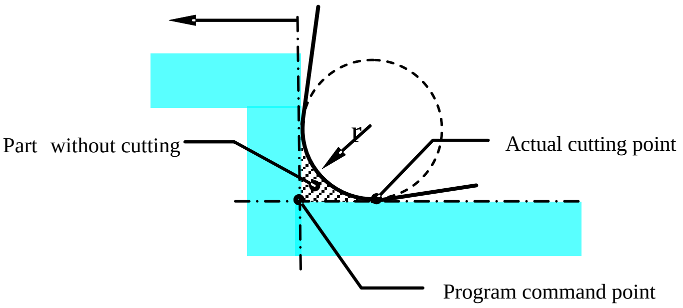


Imaginary tool nose NO.1	Imaginary tool nose NO.2	Imaginary tool nose NO.3
		
Imaginary tool nose NO.4	Imaginary tool nose NO.5	Imaginary tool nose NO.6

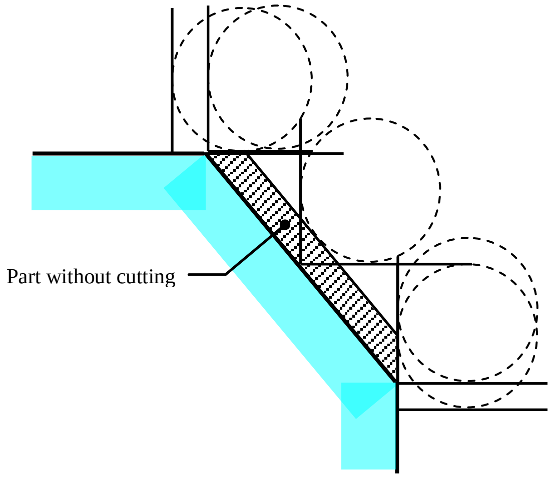
		
Imaginary tool nose NO.7	Imaginary tool nose NO.8	Imaginary tool nose NO.9
		

1.20.2.4 Compensation without tool nose:

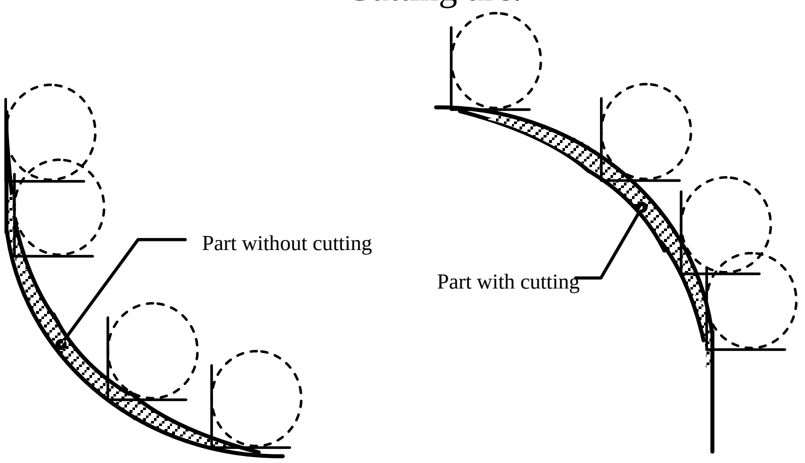
Surface cutting:



Corner or slant surface:



Cutting arc:



a.cutting internal of a cycle **b.cutting outer of a cycle**

1.20.3 Tool Radius (R) compensation

1.20.3.1 Compensation Starts

When a block which satisfies all the following conditions is executed, the system enters the offset mode. Control during this operation is called compensation start-up.

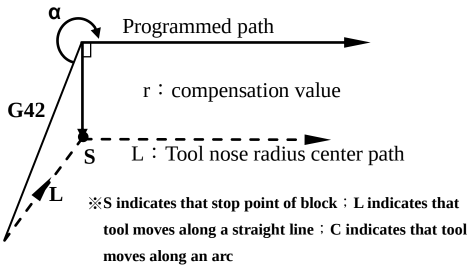
G41 or G42 is contained in the block, or has been specified to set the system to enter the offset mode.

The offset number of tool nose compensation is not “ 00 ”.

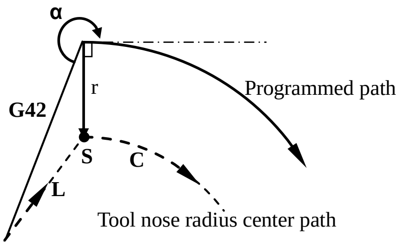
X or Z moving are specified in the block and the move distance is not zero.

(a). Inner Side (180°≧ α)

(i). Linear→Linear

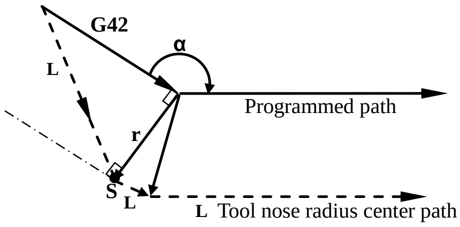


(ii). Linear→Circular

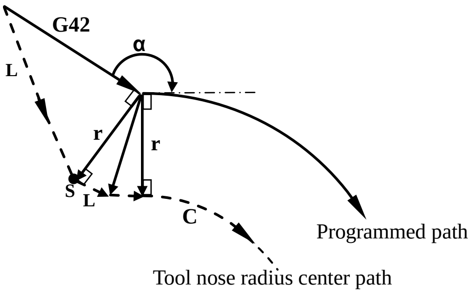


(b). Outer Side (90°≧α＜180°)

(i). Linear→Linear

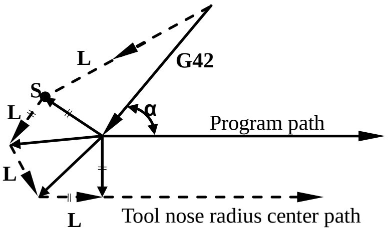


(ii). Linear→Circular

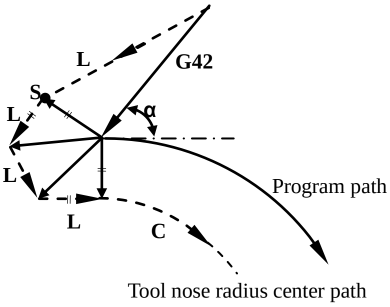


(c). Outer Side (α＜90°)

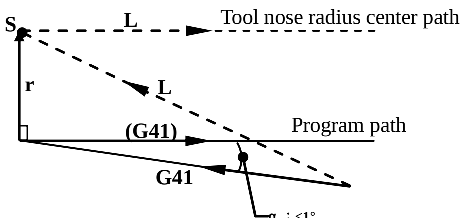
(i). Linear→Linear



(ii). Linear→Circular

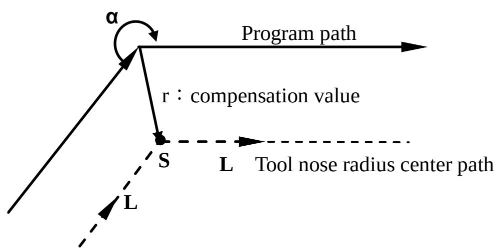
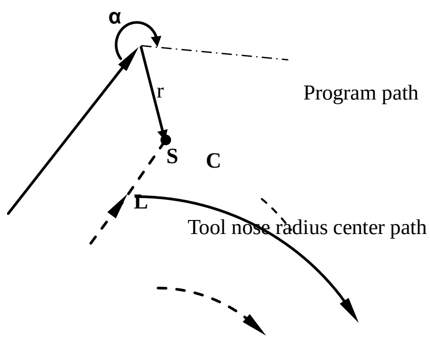
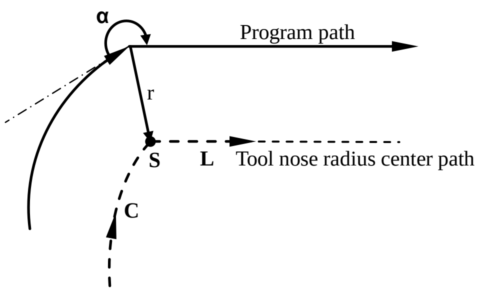
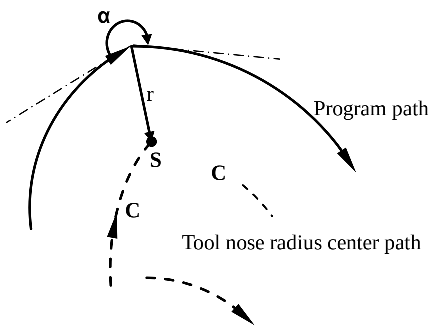
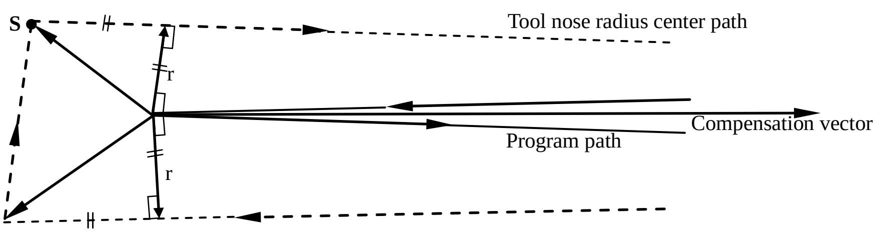


(d). At corner(<1°) outer Linear→Linear Perform(α<1°)



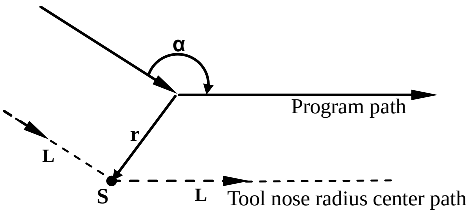
1.20.3.2 2. Compensation mode

In compensation mode, it is the same as straight and circular interpolation. It uses compensation even during positioning. In compensation mode, it does not specify tool movement block (M Function or dwell .etc.) it cannot be specified continuously. If it is specified continuously, overcut or undercut will occur.

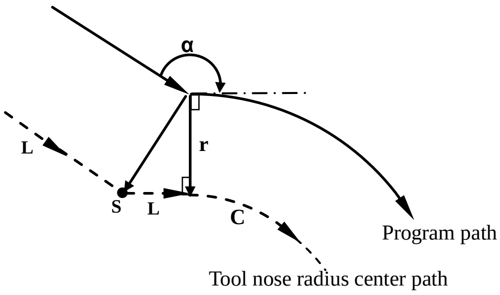
<table>
<tr><th colspan="2">(a). Inner Side (180°≦α)</th></tr>
<tr><td>(i). Linear→Linear

</td><td>(ii). Linear→Circular

</td></tr>
<tr><td>(iii). Circular→Linear

</td><td>(iv). Circular→Circular

</td></tr>
<tr><td colspan="2">(v) Perform inner side at little angle(<1°) and enlarge compensation vector

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

(b). Outer Side(90°≦α＜180°)

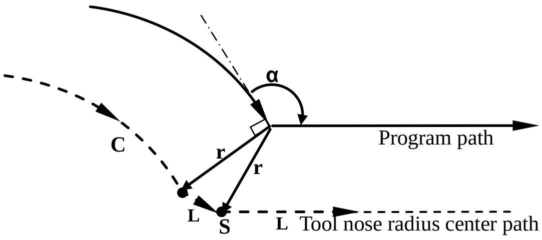
(i). Linear→Linear



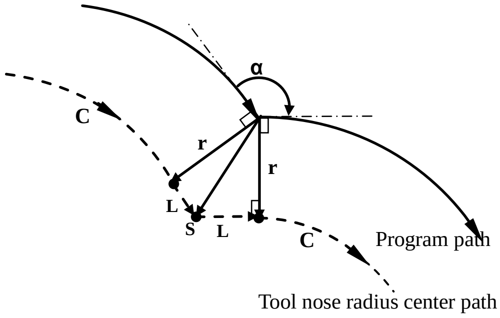
(ii). Linear→Circular



(iii). Circular→Linear

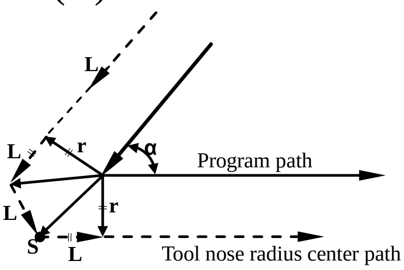


(iv). Circular→Circular

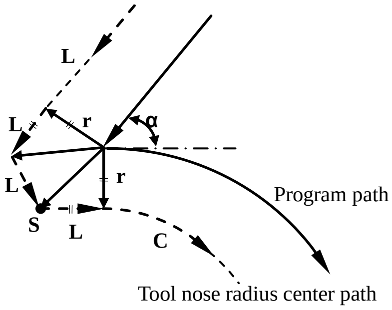


(c). Outer Side (α＜90°)

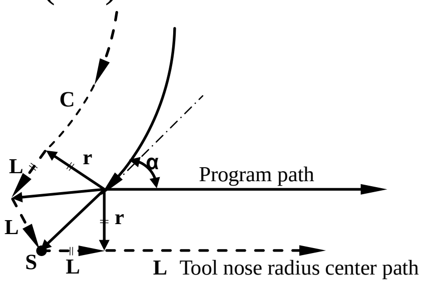
(i). Linear→Linear



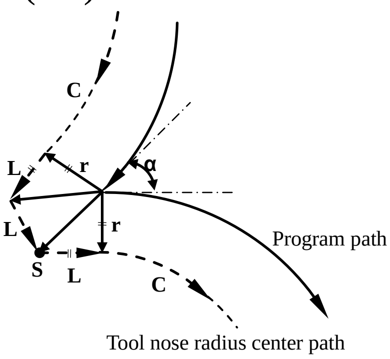
(ii). Linear→Circular



(iii). Circular→Linear



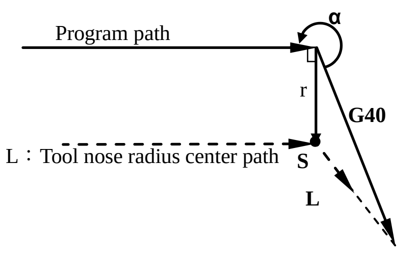
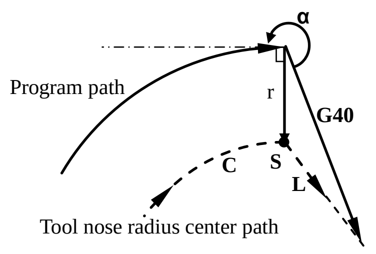
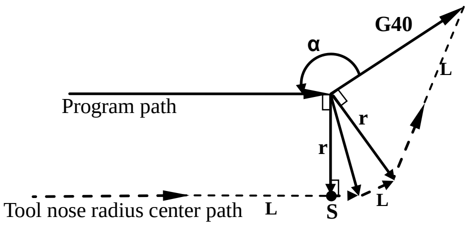
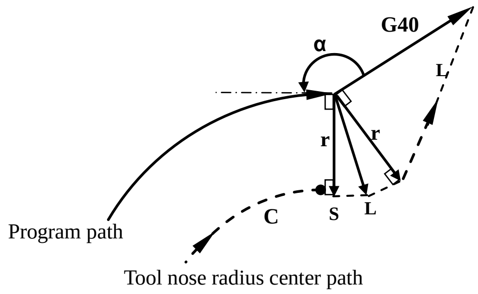
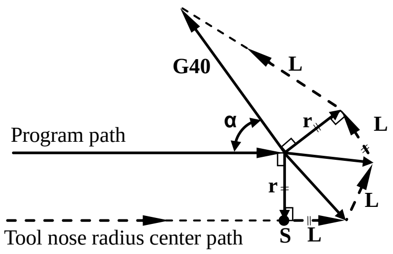
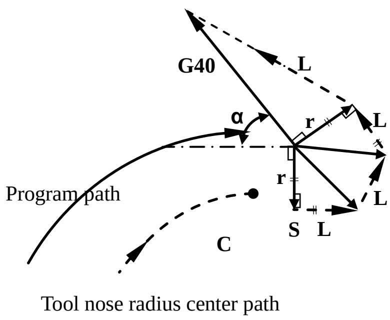
(iv). Circular→Circular

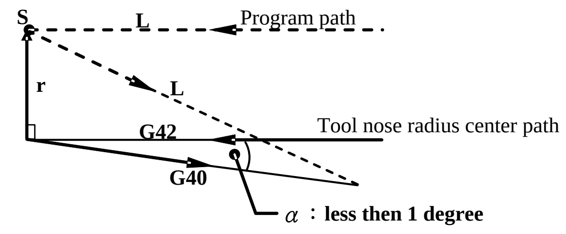


1.20.4 3. Compensation Cancel

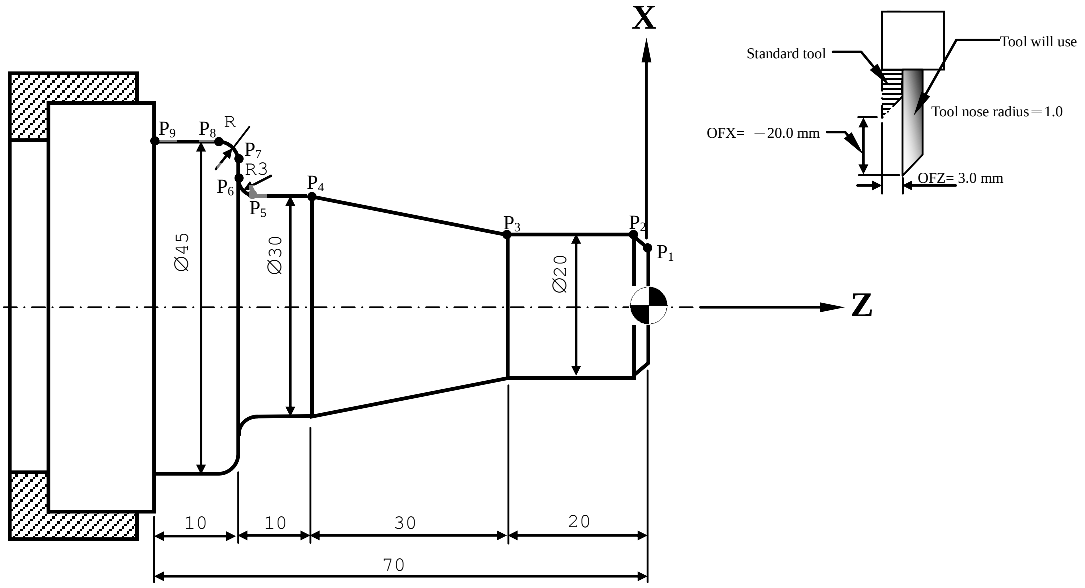
In compensation mode, when block satisfies following conditions, system will enter cancel mode:

1. Specify G40
2. The number of tool nose radius compensation is specified to “0”

<table>
<tr><td colspan="2">(a). Inner Side (180°≦ α)</td></tr>
<tr><td>(i). Linear→Linear

</td><td>(ii). Linear→Circular

</td></tr>
<tr><td colspan="2">(b). Outer Side (90°≦α＜180°)</td></tr>
<tr><td>(i). Linear→Linear

</td><td>(ii). Linear→Circular

</td></tr>
<tr><td colspan="2">(c). Outer Side (α＜90°)</td></tr>
<tr><td>(i). Linear→Linear

</td><td>(ii). Linear→Circular

</td></tr>
<tr><td colspan="2">(d). Tool movement around the outside linear→linear at an acute angle less than 1 degree(α<1°)</td></tr>
</table>

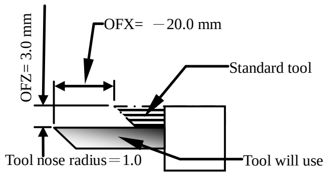

S
L
Program path
r
L
G42
Tool nose radius center path
G40
α ：less then 1 degree

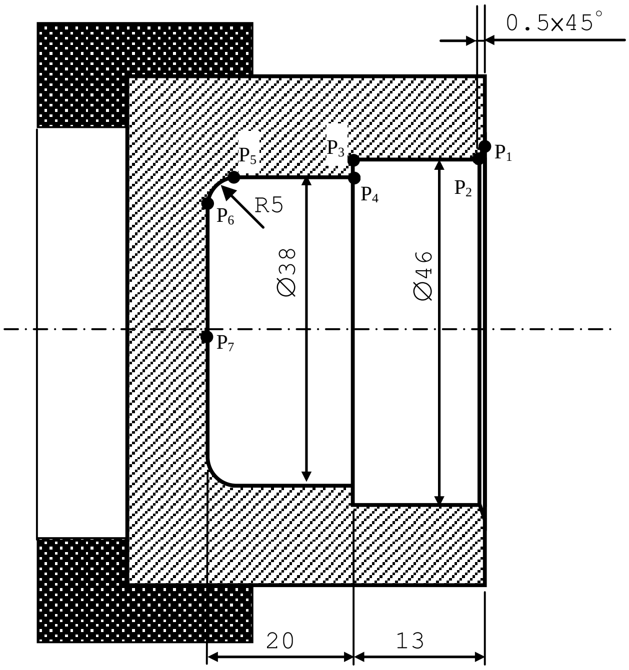
1.20.5 Example 1




```
T02                         //use tool NO.2
G92 S10000                  //max. rotate speed, 10000rpm
G96 S130 M03                //constant surface speed, spindle rotate
                            //130 m/min CW
M08                         //cutting liquid ON
G42 X21.0 Z0.0              //tool compensation start-up, move to P1
G01 X25.0 Z-2.0 F600        //linear interpolation, feedrate 600
                            //µm/rev, P1→P2
  Z-20.0                    // P2→P3
  X30.0 Z-50.0              // P3→P4
  Z-57.0                    //P4→P5
G02 X36.0 Z-60.0 R3.0       // P5→P6
G01 X39.0                   // P6→P7
G03 X45.0 Z-63.0 R3.0       // P7→P8
G01 Z-70.0                  // P8→P9
  X60.0                     //return the tool
G28 X70.0 Z-60.0            //positioning to specified mid-point, then
                            //return to machine zero point
M09                         //cutting liquid OFF
M05                         //spindle stops
M30                         //program ends
```

1.20.6 Example 2





T02 //use tool NO.2
G92 S1000 //max. rotate speed, 10000rpm
G96 S130 M03 //constant surface speed, spindle rotate 130
//m/min CW
M08 //cutting liquid ON
G41 X47.0 Z0.0 //start tool compensation, move to P1
G01 X46.0 Z-0.5 F600 // linear interpolation, feedrate 600µm/rev,
//$P_1 \rightarrow P_2$
Z-13.0 //$P_2 \rightarrow P_3$
X38.0 //$P_3 \rightarrow P_4$
Z-28.0 //$P_4 \rightarrow P_5$
G03 X28.0 Z-33.0 R5.0 //circular interpolation CCW, radius 5
//mm, $P_5 \rightarrow P_6$
G01 X-1.0 //linear interpolation
M09 //cutting liquid OFF
G28 Z20.0 //positioning to specified mid-point, then
//return to machine zero point
M05 //spindle stops

```
M30                 //program ends
```

1.21Polygon cutting (G51.2)

G51.2 is polygon cutting by workpiece axis and tool axis that is synchronous rotates and have fix phase difference and rotation rate.
Synchronous spindle rotate speed: basic spindle speed * Q / P
Synchronous phase difference: the clockwise angle difference between synchronous spindle and basic spindle. If user doesn't use R statement, it would not synchronize the phases.
G50.2 cancels the polygon cutting.
G51.2 is available in version 10.113.0 or later unavailable in earlier versions (9.0 and 10.0).

1.21.1 Format

G51.2 P_Q_R_

P: Basic spindle (workpiece axis) rotation speed rate. Default is P=1 (range from integer 1to 999)
Q: Synchronous spindle (tool axis) rotation speed rate. Default is Q=1(range from integer 1to 999).
R: Synchronous phase difference (range from 0°to 359.999°).

1.21.2 Note

1. Two synchronous spindles must be servo motors. Spindle type is only available in Type3 (Pr1791~1796). If set the wrong type, alarm (Cor093) will occur. If basic and synchronous spindle (Pr4021, 4022) does not exist, alarm (Cor091, Cor092) will occur as well.
2. If set two servo motor in different motion parameter. EX: acceleration/deceleration time (Pr1831~1836) and acceleration time of spindle motor acceleration up to 1000RPM/Sec (Pr1851~1856). They will use their own parameter to arrive the synchronized feedrate before reaching synchrony. After synchrony, two motor will use the slower parameter to control motion to synchronize the spindle feedrate.
3. If the position loop gain (Kp, Pr 181~196) of two servo motor are not the same, Kp of controller is used to compensate. User have to check if Kp of controller and Kp of driver are the same, or the motion will not be controlled as expected.
4. G51.2 is model G-Code. When the signal of spindle synchronization is on and both of spindles have rotate command (M03, M04), spindle synchronization will start and output spindle synchronization success signal.
5. After spindle synchronization, rotational direction of basic spindle and synchronous spindle are assigned by Pr1861~1866 (Spindle Sync. basic spindle direction). M03 and M04 enable to control the direction.
6. During spindle synchronization, commanding to synchronous spindle is invalid. If the speed of the basic spindle is greater than maximum allowed synchronous spindle speed, the speed of the basic spindle will descend to be P/Q times the speed of maximum allowed synchronous spindle speed.(Ratio between basic and synchronous spindle speed is P:Q)
7. After spindle synchronization, synchronous spindle doesn't act on M03, M04, M05 and S code but only record the mode until synchronization disabled.
8. Pressing emergency stop will terminate spindle rotation and spindle synchronization.
9. After spindle synchronization finish, user can't orientate for spindle.

10. After finishing synchronization (S62, ON) and pressing reset G51.2 synchronization will be disabled until two spindles stop.
11. When reading feedback from the encoder, 8-μs delay time exists between the port and its adjoining ports. The further two ports are, the longer the delay time results. Spindle synchronization has to take care about phase. If using spindle synchronization to implement polygon cutting, user have to put two spindles on the port that is next to each other. EX: Since P1 and P2 are on the same servo card, connecting feedback ports to them decreases time delay as well as the phase error.
12. If G50.2 is commanded during spindle synchronization, only until the speed reaching the specified speed, will the system disable spindle synchronization.

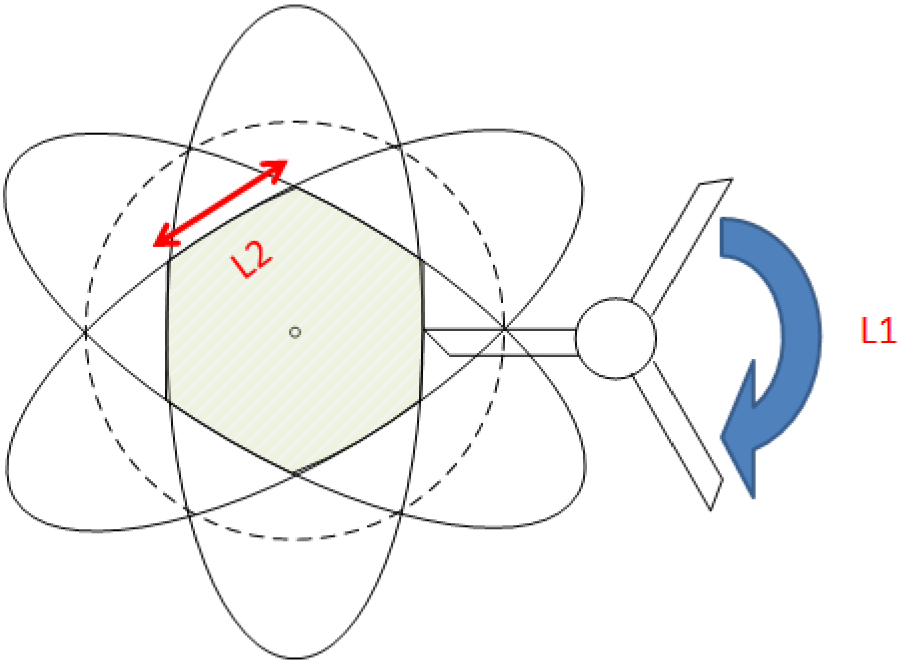
13. To synchronize again, G50.2 must be performed to cancel synchronization first, otherwise an alarm will occur.

14. When synchronize phase difference is used, the value of R shall equal to the amount of angle different between the tool and the workpiece, times Q, and divided by P. (Please refer to the example)

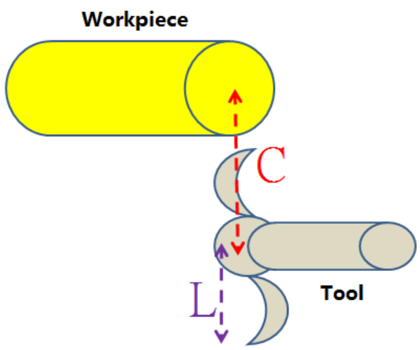
15. P and Q value can be only integer. In case of the ratio is not integer, such as 1:2.5, user shall use equivalence integer ratio, e.g. 2.:5.

16. In order to assure the absolute position of the workpiece, tool's home position teaching needed to be set. (Please refer to the first section of Example)

17. The length of arc from a tip of the tool to another, is require to be larger than the target side length of the workpiece, as referred to the following diagram, L1 needs to be larger than L2.



18.G51.2 performs a polygon cutting through the speed difference of the tool and the workpiece (spindle). The result of surface (convex, flat, or concave) depends on the cutting conditions. The chart below provides some information of the cutting conditions and the results of surface for references.

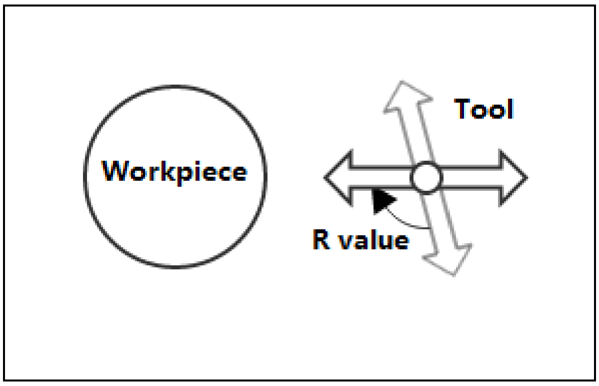
<table>
<tr><th>Speed Ratio (i)</th><th colspan="4">Result of Surface</th></tr>
<tr><td rowspan="3">>2</td><td>K>L</td><td>K=L</td><td>K<L</td><td rowspan="5">
L: Tool Rotational Radius
i : Speed Ratio (Q/P) 、
C: Distance from center of tool to center of workpiece</td></tr>
<tr><td>Convex</td><td>Flat</td><td>Concave</td></tr>
<tr><td colspan="3">Note: K = C/(i-1)^2</td></tr>
<tr><td>=2</td><td colspan="3">Convex</td></tr>
<tr><td><2</td><td colspan="3">Convex</td></tr>
</table>

1.21.3 Example

- **Setting Up**

  In order to assure the absolute position of the workpiece, tool's home position needed to be setup. There are three different ways to setup the tool's home position:

  1. Input the phase difference via R input

  2. Reset the zero position of tool to where the tip of the tool is perpendicular to the workpiece.
  3. Turn the tool to where the tip of the tool is perpendicular to the workpiece, and then proceed the phase difference teaching (F4>F4>F3), the angle will be input to the Registry Table automatically.

- **Sample Command**

  Ex1. Hexagon with 3 flute tool : G51.2 P3 Q6 (or G51.2 P1Q2)

  Ex2. Pentagon with 2 flute tool : G51.2 P2 Q5

- **Sample Program**

```
S1 = 1000          //Workpiece axis (basic spindle) rotate
speed 1000 RPM
M03             //Workpiece axis (basic spindle) spindle rotate
CW
S2 = 500        //Tool axis (Synchronous spindle) rotate speed
500
                //RPM
M204               //Tool axis (Synchronous spindle) spindle
rotate CCW
G51.2 P1 Q2 R30  // Tool axis (Synchronous spindle)
                //synchronoziation arrive to 2000RPM and the
phase
                //difference is 30 degree. Cut for quadrangle
```

```
M81            // reading S62. Check the synchronization
success.
G01 X50        // start cutting
G04 X5
G01 X0              //return
G51.2 P1 Q3 R60  //Tool axis (Synchronous spindle)
               //synchronoziation arrive to 3000RPM and the
               //phase difference is 60 degree. Cut for
hexagon.
G01 X50        // start cutting
G04 X5
G01 X0              // return
G50.2               // cancel polygon cutting
M05            // Workpiece axis (basic spindle) stop
M205                // Tool axis (Synchronous spindle) stop
M30            // program finish
```

- **Synchronizing Error**

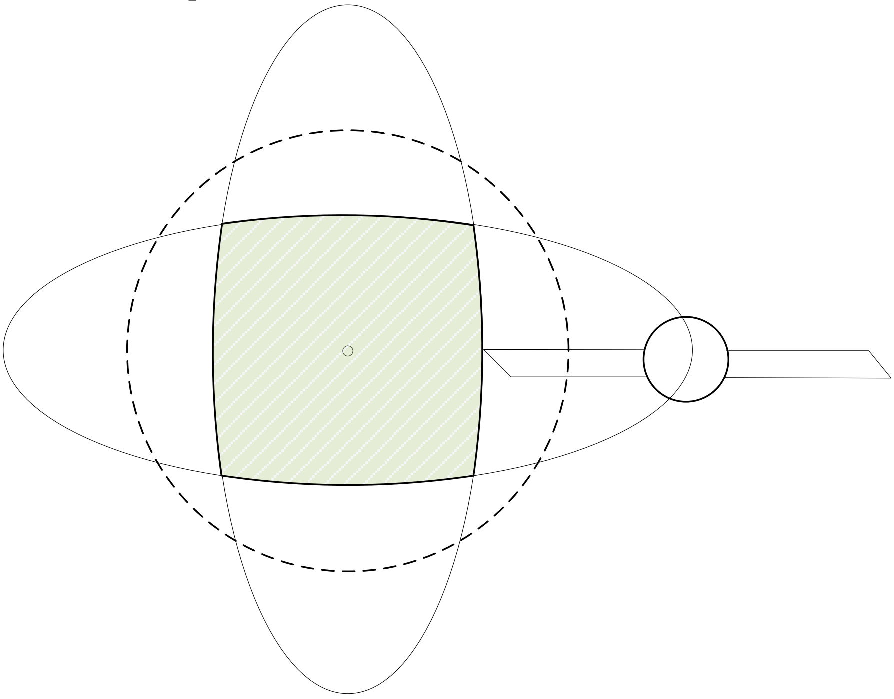
  The speed of main and sub spindle can be different during synchronization, therefore, the synchronizing error is calculated through the formula as below:

  Synchronizing Error = (Actual Position of sub-spindle – Datum Angle of sub-spindle) - Speed Ratio*(Actual Position of main spindle – Datum Angle of main spindle) – Phase Difference

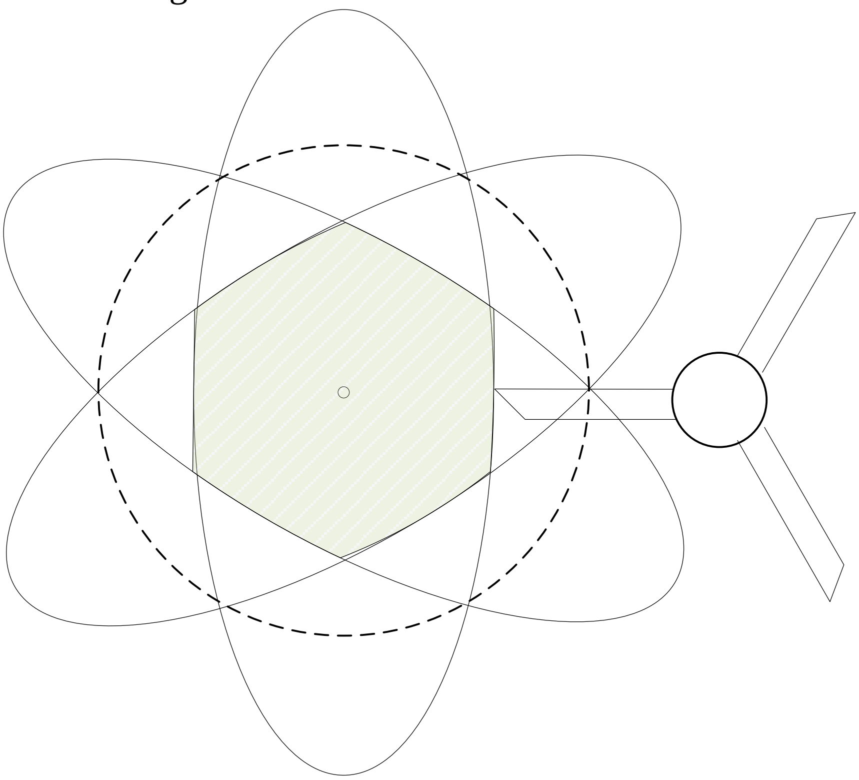
  Note 1 : Datum Angle is referred to the Registry Table
  Note 2 : Speed Ratio is Q/P
  Note 3 : Actual position is referred to the feedback from the motor encoder

1.21.4 Polygon machining PIC

A square can be machined as shown below



A hexagon can be machined as shown below



1.21.5 Reference

Device Type	Device	Description
R	R761~R776	Corresponding machine coordinate. Unit is 0.001 degree.
S	S62	signal of spindle synchronization success
Registry	L10031	signal of spindle synchronization. Basic spindle datum angle θ_1
	L10032	signal of spindle synchronization. Synchronous spindle datum angle θ_2
Paramter	181~196	Position loop gain(Kp)(1/sec) of servo
	881~896	Home offset
	1791~1796	Spindle type
	1831~1836	spindle motor acceleration time(ms)
	1851~1856	spindle motor speed up to 1000RPM/Sec acceleration time(ms)
	1861~1866	spindle direction, 0: CW, 1: CCW
	4021	Basic spindle number(1~6)
	4022	Sync spindle number (1~6)
Alarm	Cor091	Invalid number of basic spindle
	Cor092	Invalid number of synchronous spindle
	Cor093	Invalid type of sync. spindle
	Cor095	Invalid ratation speed rate of basic spindle
	Cor096	Invalid ratation speed rate of synchronous spindle

1.22 Local Coordinate System Setting (G52)

When a program is created in a workpiece coordinate system (G54~G59.9), another sub-coordinate can be established for easier programming, this sub-coordinate system is called local coordinate system.

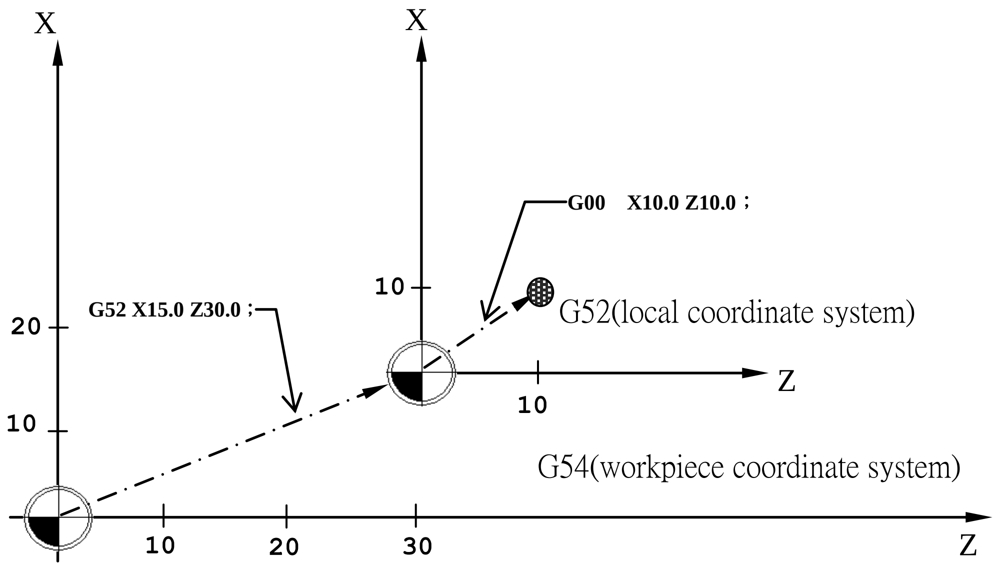
G52 X0.0 Z0.0: cancel local coordinate.

1.22.1 Format

G52 X__ Y__ Z__

X, Y, Z: Set the local coordinate system

1.22.2 Coordinate System



G54	//specify workpiece coordinate system G54
G52 X15.0 Z30.0	**//specify the zero point of local coordinate**
	//system to X15.0 Z30.0 of workpiece
	//coordinate system
G00 X10.0 Z10.0	//positioning to X10.0 Z10.0 of local
	//coordinate
G52 X0.0 Z0.0	**//local coordinate system cancel**
…	

1.23 Machine Coordinate System (G53)

The point that is specific to a machine and serves as the reference of the machine is referred to as the machine zero point. A machine coordinate system, once set, remains unchanged until the power is turned off. A machine tool builder sets a machine zero point for each machine. When a position has been specified as a set of machine coordinates, the tool moves to that position by means of rapid traverse.

1.23.1 Format

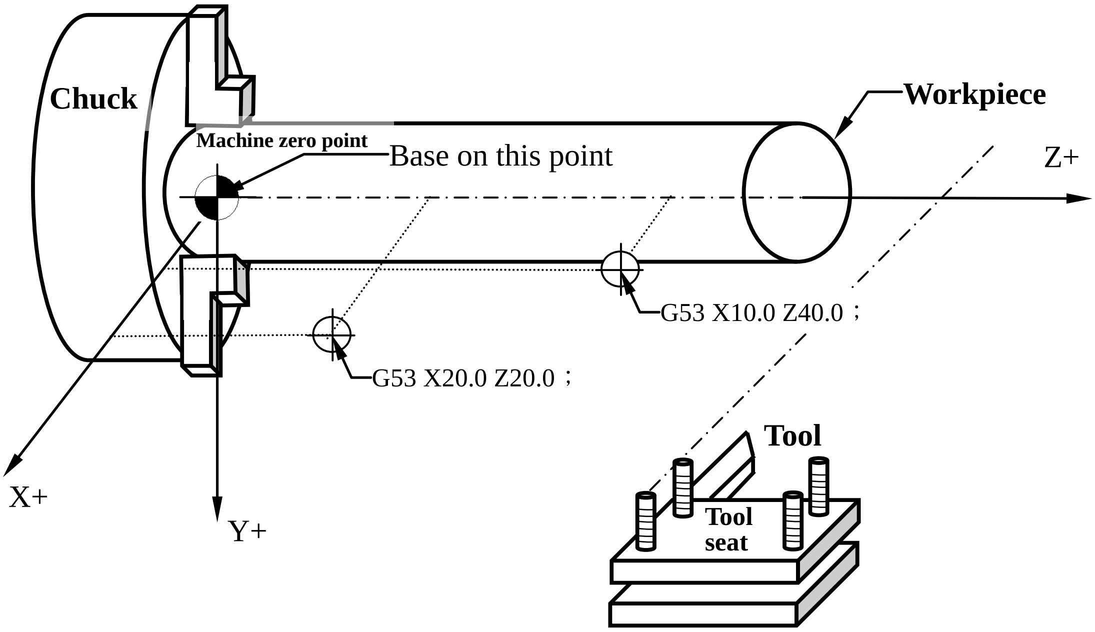
G53 X___ Y___ Z___

X: move to specified X in machine coordinate.
Y: move to specified Y in machine coordinate.
Z: move to specified Z in machine coordinate.

1.23.2 Notice

1. G53 command is valid only in the block in which it is specified on a machine coordinate system .
2. G53 is valid only in absolute mode. When an incremental command is specified, the G53 command will be ignored.
3. Prior to specifying G53, cancel related tool radius ,length or position compensation .
4. Prior to G53 command is specified, manual reference position return must be performed.

1.23.3 Example



G53 X20.0 Z20.0 //move to specified position in machine coordinate

G53 X10.0 Z40.0 //move to specified position in machine coordinate

1.24 Workpiece Coordinate System (G54…G59.9)

When operating the lathe, we may repeat performing the same process in different positions which are in one workpiece. By specifying G code from G54 to G59 and G59.1 to G59.9, one of the 15 workpiece coordinate systems can be selected for easier repeating processes. It can be set by parameter #3229「disable workpiece coordinate system」(0: enable, 1: disable).

1.24.1 Format

```
G54       X___ Y___ Z___
G55       X___ Y___ Z___
G56       X___ Y___ Z___
G57       X___ Y___ Z___
G58       X___ Y___ Z___
G59       X___ Y___ Z___
G59.1     X___ Y___ Z___
G59.2     X___ Y___ Z___
…
G59.9     X___ Y___ Z___
```

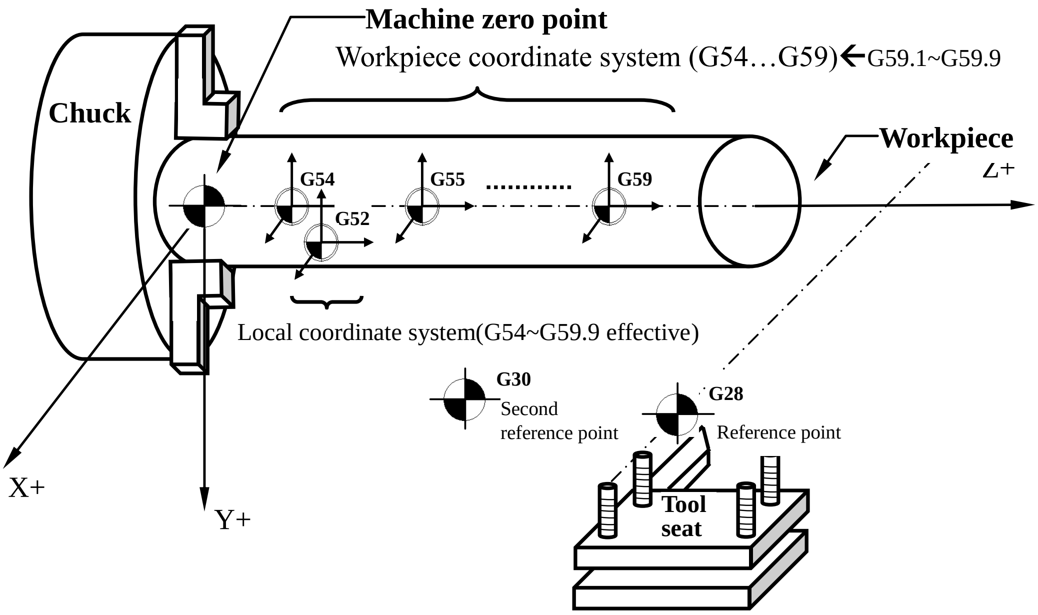
G54: First workpiece coordinate system
…
G59: Sixth workpiece coordinate system
G59.1: Seventh workpiece coordinate system
…
G59.9: 15th workpiece coordinate system

X,Y,Z: Move to specified position in workpiece coordinate system which has been set

1.24.2 How to set G54……G59.9

By selecting “Set workpiece coordinate system” in controller operation interface, workpiece coordinate system G54 …G59.9 can be set one by one.

1.24.3 Example



1.25Simple Marco Call (G65)

After G65, specify at address P the program number of the custom Marco to call. When repetition is required, specify the repetition count after address L. The execution is valid only in the block in which G65 performed. Refer to SYNTEC『OPEN CNC Macro Develop Tool Guide』 for more instructions.

1.25.1 Format

G65 P_L_

P: number of the program to call
L: Repetition count (1 by default)

1.25.2 Example

```
G65 P10 L20 X10.0 Y10.0    //call the marco program O0010.
                           //Execute the program repeatedly for 20
                           //times with value X10.0 Y10.0 being
                           //operated.
```

1.26Modal Marco Mode (G66/G67)

After G66, specify the number of the program to call P_. When repetition is required, specify the repetition count after address L_. After finishing the execution in one block, it will automatically execute the contents of G66 in the next “moving” block. (If next command is not a moving block, G66 does not act until moving command appears.)

G66 movement performs repeatedly until G67 is issued. (if the subprogram called contains calculation of variables, notice the problems of pre-calculated variable values.)

1.26.1 Format

G66 P_L_ Modal Marco call
G67 Modal Marco cancel

P: number of the program to call
L: Repetition count (1 by default)

1.26.2 Example

```
G91
G66 P10 L2 X10.0 Y10.0      //call O0010 two times and input the
                            //value X10.0 Y10.0 into the program for
                            //calculation
X20.0                       //move X axis to 20.0, then execute
                            //G66 P10 L2 X10.0 Y10.0
Y20.0                       //move Y axis to 20.0, then execute
                            //G66 P10 L2 X10.0 Y10.0
G67                         //cancel the model marco mode
```

1.27English/Metric Unit Setting (G70/G71)

G70: Imperial unit system

G71: Metric unit system

After changing Imperial/Metric, workpiece coordinate offset ,tool data ,system parameter ,and reference position are still correct. System will convert the unit automatically. After unit converting, the function units list below will change as well:

Coordinate display ,unit of speed

Incremental JOG unit

MPG JOG unit

1.27.1 Format

G70

G71

1.28 Finishing Cycle (G72)

G72 command is finishing cycle (contour cutting cycle), this command must be in conjunction with stock removal cycle in the previous block. In general, finishing cycle is executed after stock removal cycle in the program, The execution range is from "P(ns)" to "Q(nf)".

While rough cutting is performed by G73 / G74 / G75, G72 command must be implemented to reach the final specified size.

1.28.1 Format

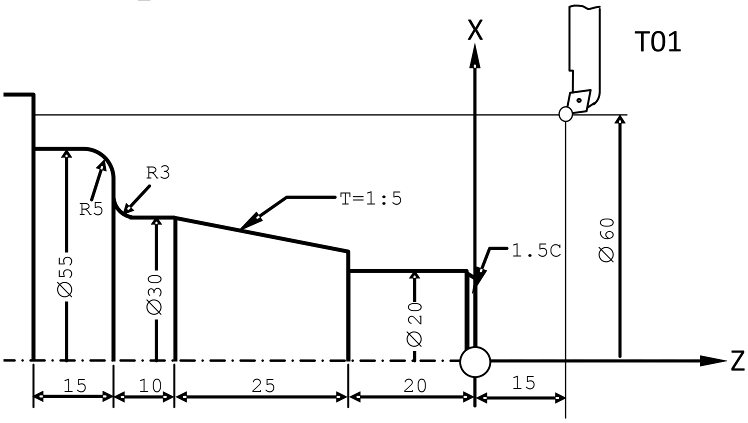
G72 P(ns) Q(nf)

ns: Sequence number of starting block in finishing cycle
nf: Sequence number of ending block in finishing cycle

1.28.2 Notice

1. F,S and T functions specified in the block G73,G74 and G75 are not effective while those specified between the blocks determined by addresses P and Q("**ns**"→"**nf**") are effective in G72.
2. When the cycle machining through G72 is terminated, the tool is returned to the start point and the next block is read.
3. In blocks between "ns" and "nf" referred in G72 through G75, the subprogram cannot be called.

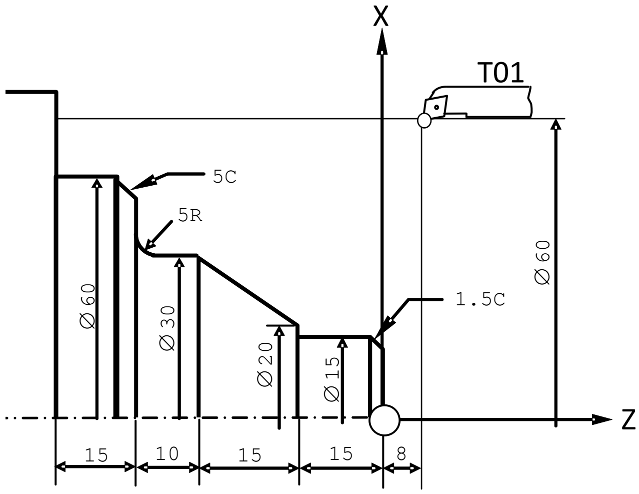
1.28.3 Example 1




```
T01                //use tool NO. 1
G92 S5000          //Max. rotate speed 5000 rpm
G96 S130 M03       //constant surface speed, surface speed 130 m/min,
                   //spindle rotate CW
G00 X60.0 Z15.0    //positioning to start point
M08                //cutting liquid ON
G73 U2.0 R1.0      //cut 3.0 mm in X axis direction, tool returned
                   //value 1.0 mm
G73 P01 Q02 U0.8 W0.1 F300   //execute stock removal in turning,
                   //sequence number N01→N02, leave 0.8mm for
                   // finishing allowance in X axis direction, leave
                   //0.1mm for finishing allowance in Z axis direction
                   //feedrate 300 µm/rev
N01 G00 X17.0
    G01 Z0.0
        X20.0 Z-1.5
        Z-20.0
        X25.0
        X30.0 Z-45.0 //contour for cutting
        Z-52.0
    G02 X36.0 Z-55.0 R3.0
    G01 X45.0
    G03 X55.0 Z-60.0 R5.0
N02 G01 Z-70.0
```

G72 P01 Q02 //execute fine cutting cycle, sequence number
//N01→N02
M09 //cutting liquid OFF
M28 X60.0 Z20.0 //tool positioning to specified mid-point, then
//return to machine zero point
M05 //spindle stops
M30 //program ends

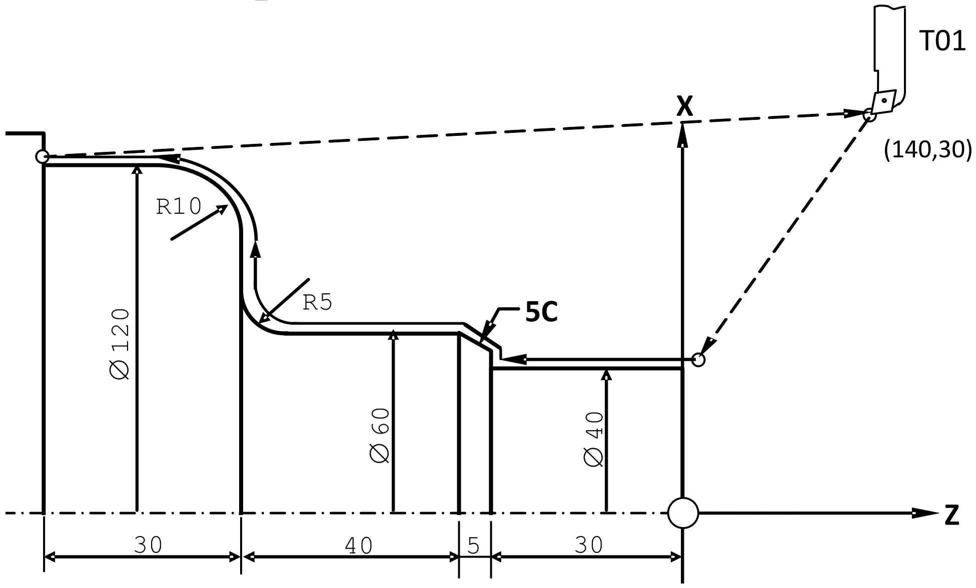
1.28.4 Example 2




```
T01               //use tool NO. 1
G92 S5000         //Max. rotate speed 5000 rpm
G96 S130 M03      //constant surface speed, surface speed 130 m/min,
                  //spindle rotate CW
G00 X60.0 Z8.0        //positioning to start point
M08               //cutting liquid ON
G74 W3.0 R1.0     //cut 3.0mm in Z axis direction, tool returned value
                  //1.0 mm
G74 P01 Q02 U0.8 W0.2 F600    //execute stock removal in facing,
                  //the sequence number N01→N02,leave 0.8mm for
                  //finishing allowance in X axis direction, leave
                  //0.2mm for finishing allowance in Z axis direction
                  //, feedrate 600 μm/rev
N01 G00 Z-55.0
    G01 X60.0
        Z-45.0
        X50.0 Z-40.0
        X40.0
    G03 X30.0 Z-35.0 R5.0
    G01 Z-30.0
        X20.0 Z-15.0
        X15.0
        Z-1.5
N02 X12.0 Z0.0
```

```
G72 P01 Q02     //execute fine cutting cycle, the sequence number
                //N01→N02
M09             //cutting liquid OFF
G28 X60.0 Z10.0 //positioning to specified mid-point, then return to
                //machine zero point
M05             //spindle stops
M32             //program ends
```

1.28.5 Example 3




```
T01                //use tool NO.1
G92 S5000          //max. rotate speed 5000 rpm
G96 S130 M03       //constant surface speed, surface speed 130 m/min,
                   //spindle rotate CW
G00 X140.0 Z30.0         //positioning to start point
M08                      //cutting liquid ON
G75 U15.0 W15.0 R3.0     //cut 15.0mm in X axis direction, cut
                   //3.0mm in Z axis direction, repeat 3 times
G75 P01 Q02 U0.8 W0.2 F300    //execute pattern repeating cutting,
                   //the sequence number N01→N02,leave 0.8mm for
                   //finishing allowance in X axis direction, leave
                   //0.2mm for finishing allowance in Z axis direction
                   //feedrate 300 µm/rev
N01 G00 X40.0 Z5.0
    G01 Z-30.0
        X50.0
        X60.0 Z-35.0
        Z-70.0
    G02 X70.0 Z-75.0 R5.0
    G01 X100.0
    G03 X120.0 Z-85.0 R10.0
N02 G01 Z-105.0
```

```
G72 P01 Q02      //execute fine cutting cycle, the sequence number
                 //N01→N02
M09              //cutting liquid OFF
G28 X140.0 Z30.0 //positioning to specified mid-point, then return to
                 //machine zero point
M05              //spindle stops
M30              //program ends
```

1.29Stock Removal in Turning (G73)

G73 command (stock removal in turning) processes the workpiece to specified shape, leaving a specified value of distance for finishing allowance. This cutting cycle needs to define the block path range of workpiece, the depth of each cut and both the distance and direction of finishing allowance.

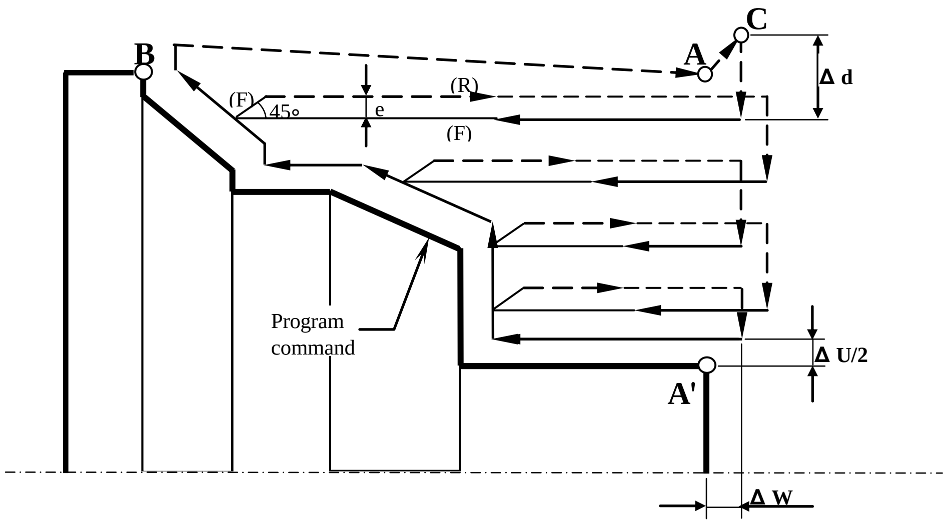
1.29.1 Format

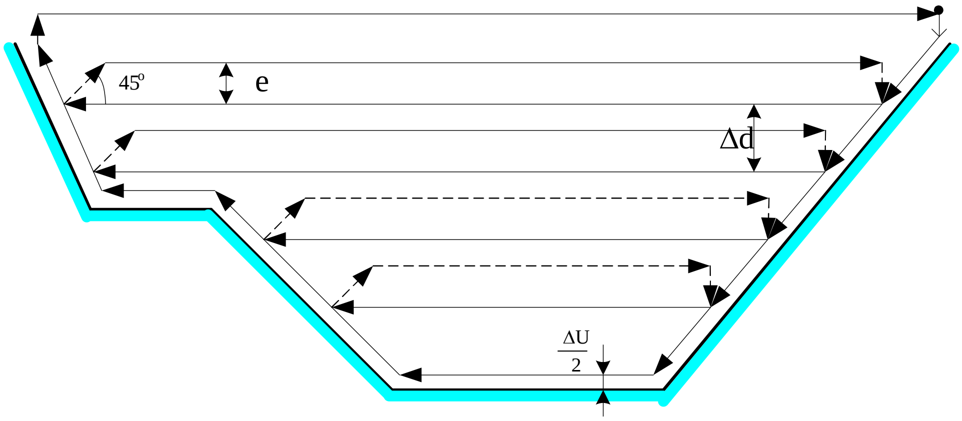
G73 U(Δd) R (e) H__
G73 P (ns) Q (nf) U(Δu) W(Δw) F S T

Δd: depth of each cut in X axis direction, it can be specified by the parameter#4013 -when this statement is not applied.
e: escaping amount, it can be specified by the parameter#4012 when this statement is not applied.
ns: sequence number of the first block for the program of stock removal in turning.
nf: sequence number of the last block for the program of stock removal in turning.
Δu: distance and direction of finishing allowance in X direction (diameter/radius designation)
Δw: distance and direction of finishing allowance in Z direction.
F: feedrate
T: number of the tools
S: spindle rotate speed
H: cutting type. Tpye I set 0. Type II set 1.
If user dosen't set value in H, system will check the type automatically.

1.29.2 PIC

TYPE I: The figure must show monotone increase or decrease along both X and Z axes. Each block must satisfy that the amount of cut is always increase or decrease. Usually start cutting from end face.





1.29.3 Description:

Tool should be positioned to **point A(start point)** before cycle starts.

Tool offsets to point C by specified finishing allowance **(ΔU/2 for X axis, ΔW for Z axis)**.

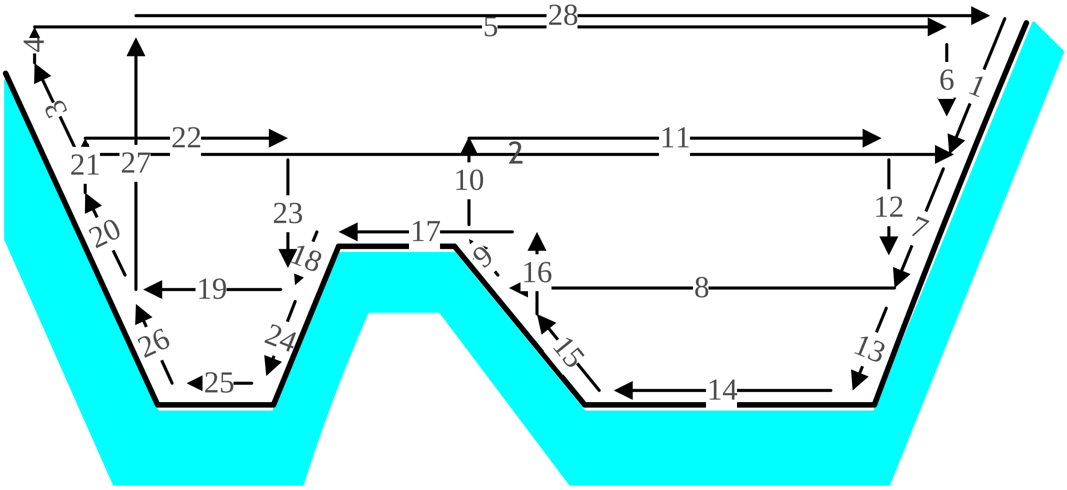
Tool moves Δd amount of distance in X axis direction. Tool begins to move to the endface of contour.

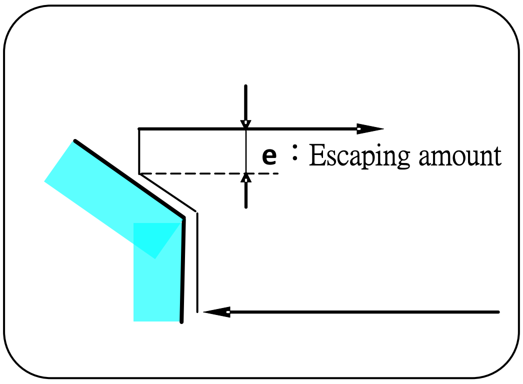
Tool escapes(retracts) **e** amount of distance in X axis direction, but moves by the direction of 45°.Tool then retracts in reversed Z axis feed direction to the point that parallels in X direction to the start point.

Move Δd amount of distance in X direction, continuing next cycle

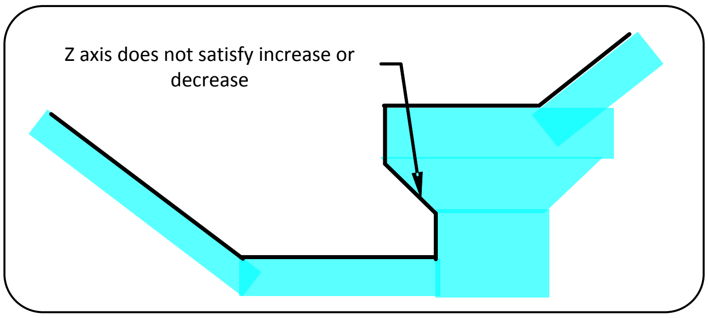
1. In last cycle, tool cuts along contour **A'→ B** once
2. After finishing last cycle, tool positions to point A.

TYPE II: Usually be performs in the middle part of the workpiece. The figure need not show monotone increase or decrease in the direction of X axis. Only Z axis needs to satisfy the condition that cutting amount is always increasing or decreasing.





← Escaping mode



← Can not perform

1.29.4 Notice

When **ns and nf** are not specified, specified **U** in **G73** block is depth of cut **Δd**. Otherwise, **U** is finishing allowance in X direction.

Contour path is described by the blocks **ns and nf**, passing through point A → A' → B. If Z coordinate of contour path is not monotone,

System will send out [MAR-002 the profile must be monotone along X, Z axis] alarm. If starting point (defined by the block before G73 command)

is lower than contour path, System will send out [MAR-005 the position higher than the cycle start point] alarm.

F, S or T function issued within block range of **ns→nf** will be ignored. The relevant F,S and T functions specified in G73 block are effective instead.

G00/G01 command in the G73 blocks will be used to perform linear cut to the workpiece.

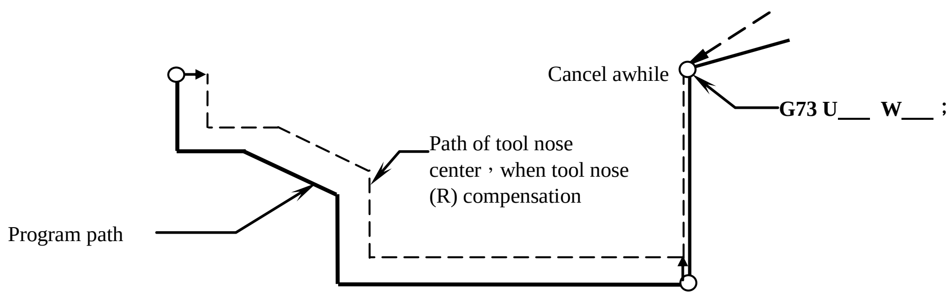
Using G73 command, If H is equal to 0, system will interpret as TYPE I. If H is equal to 1, system will interpret as TYPE II. If H is not specified,

system will diagnose automatically. If H variable is specified wrong, System will send out [MAR-018 ERROR INPUT OF G73/G74 H VALUE] alarm

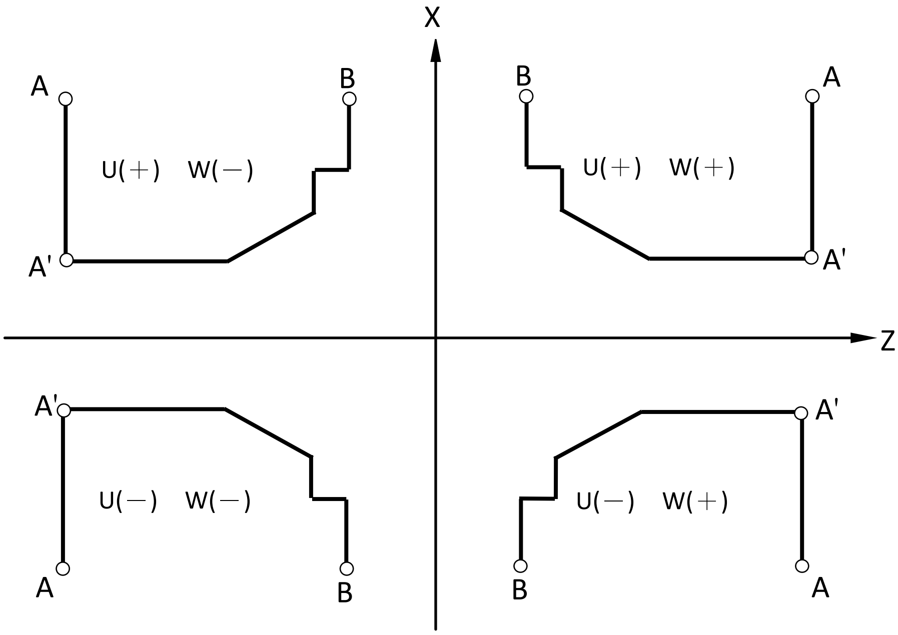
Sub-program cannot be called during blocks **ns→nf**.

All tool nose compensation commands will be disabled when G73 is in the block. However, the compensation value will be added to the finishing allowance.

When H value is not specified in G73, and the first block contains only movements along X axis, system will take TYPE-I as default.

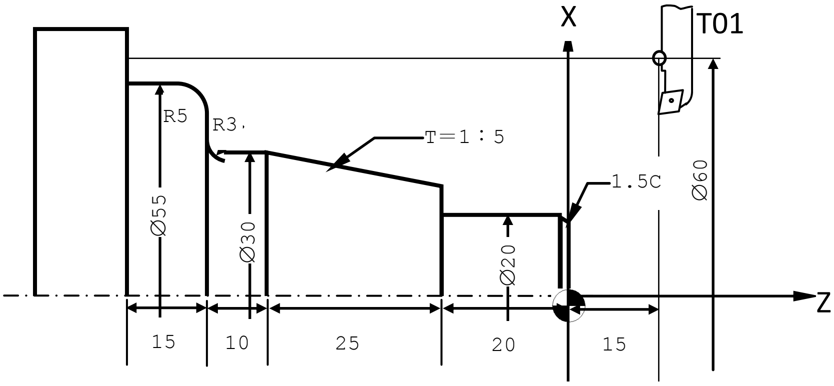


Direction of finishing allowance: the direction depends as figures shown below, passing through point A→A'→B.


X
A
B
U(+) W(−)
A'
B
A
U(+) W(+)
A'
Z
A'
U(−) W(−)
A
B
A'
U(−) W(+)
B
A

1.29.5 Example one:

TYPE I

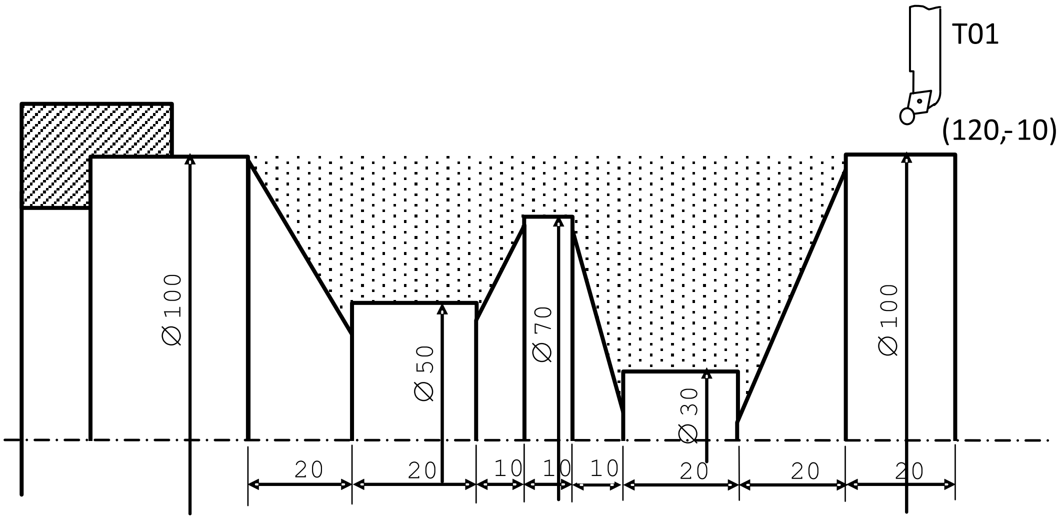


T01 //use tool NO. 1
G92 S5000 //max. rotate speed 5000 rpm
G96 S130 M03 //constant surface speed, surface speed 130 m/min,
//spindle rotate CW
G00 X60.0 Z15.0 //position to start point
M08 //cutting liquid ON
G73 U2.0 R1.0 H0 //depth of cutting in X direction is 2.0 mm,
//escaping amount 1.0 mm
//(equivalent to G73 U2.0 R1.0)
G73 P01 Q02 U0.8 W0.1 F0.3 //execute stock removal in turning,
//the sequence of block N01→N02, finishing
//allowance in X direction is 0.8 mm, finishing
//allowance in Z direction is 0.1mm, feedrate
//0.3mm/rev
N01 G00 X17.0 //← TYPE I
G01 Z0.0
X20.0 Z-1.5
Z-20.0
X25.0
X30.0 Z-45.0
Z-52.0
G02 X36.0 Z-55.0 R3.0
G01 X45.0
G03 X55.0 Z-60.0 R5.0
N02 G01 Z-70.0

```
M09               //cutting liquid OFF
M28 X60.0 Z20.0   //positioning to specified mid-point, then return to
                  //machine zero point
M05               //spindle stops
M30               //program ends
```

1.29.6 Example 2

TYPE II




```
T01                    //use tool NO. 1
G92 S5000              //max. rotate speed 5000rpm
G96 S130 M03           //constant surface speed, surface speed 130
                       //m/min
M08                    //cutting liquid ON
G00 X120.0 Z-10.0      //positioning to start point
```

G73 U2.0 R1.0 H1 //depth of cutting in X direction is 2.0 mm,
//escaping amount is 1.0 mm

```
G73 P01 Q02 U0.8 W0.1 F300   //execute stock removal in turning,
                       //the sequence of block N01→N02, finishing
                       //allowance in X direction is 0.8 mm,
                       //finishing
                       //allowance in Z direction is 0.1mm, feedrate
                       //0.3mm/rev
```

N01 G00 X101.0 Z-20.0 //← **TYPE II**

```
    G01 X100.0
        X30.0 Z-40.0
        Z-60.0
        X70.0 Z-70.0
        Z-80.0
        X50.0 Z-90.0
        Z-110.0
N02 X100.0 Z-130.0
```

```
G28 X150.0 Z40.0   //positioning to specified mid-point, then
                   //return to machine zero point
M09                //cutting liquid ON
M05                //spindle stops
M30                //program ends
```

1.30Stock Removal in Facing (G74)

G74 command is stock removal in facing, generally used when the diameter of workpiece is relatively greater than its length. That is, G74 is used when cutting amount in diameter direction is larger than axle direction, .

1.30.1 Format

G74 W(d) R(e) H__

G74 P(ns)_ Q(nf)_ U(Δu)_ W(Δw)_ F___ S___ T___

Δd: depth of each cut in Z axis direction, it can be specified by the parameter#4013 -when this statement is not applied

e: escaping amount, it can be specified by the parameter#4012 when this statement is not applied.

ns: sequence number of the first block for the program of stock removal in facing.

nf: sequence number of the last block for the program of stock removal in facing.

Δu: distance and direction of finishing allowance in X direction (diameter/radius designation)

Δw: distance and direction of finishing allowance in Z direction

F: feedrate

T: number of the tools

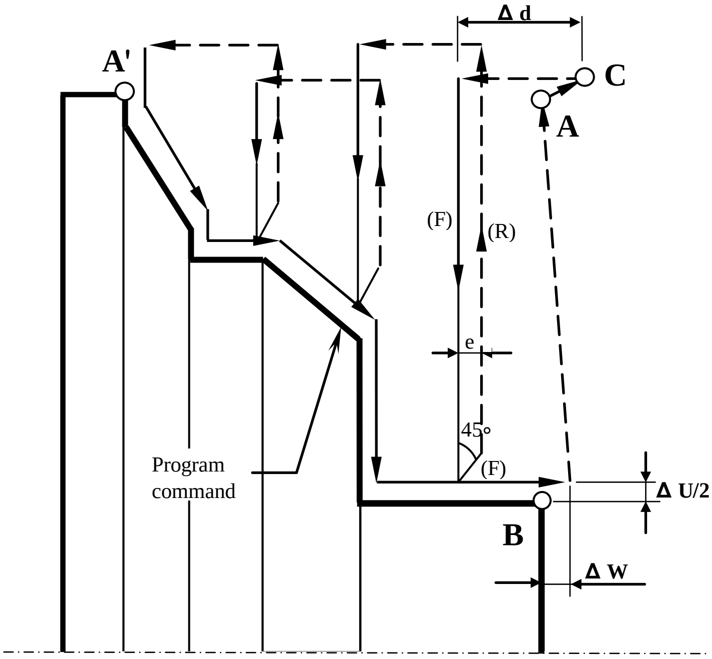
S: spindle rotate speed

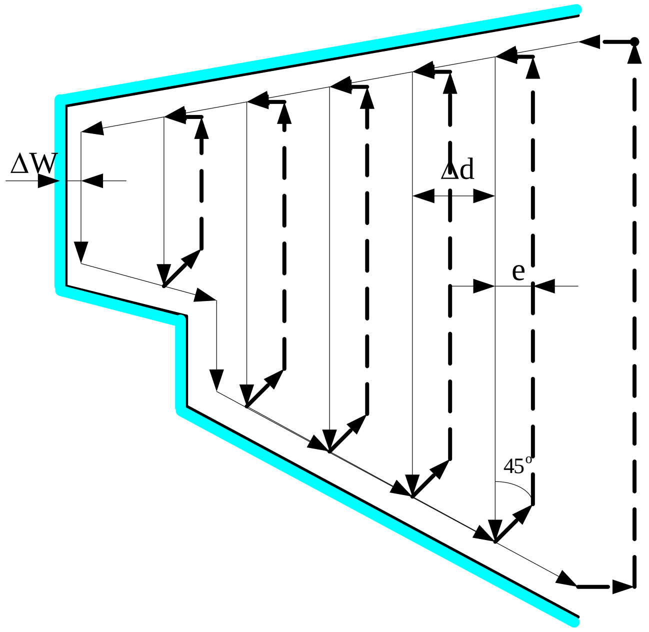
H: cutting type. Tpye I set 0. Type II set 1.

If user dosen't set value in H, system will check determine the type automatically.

1.30.2 PIC

Type I:

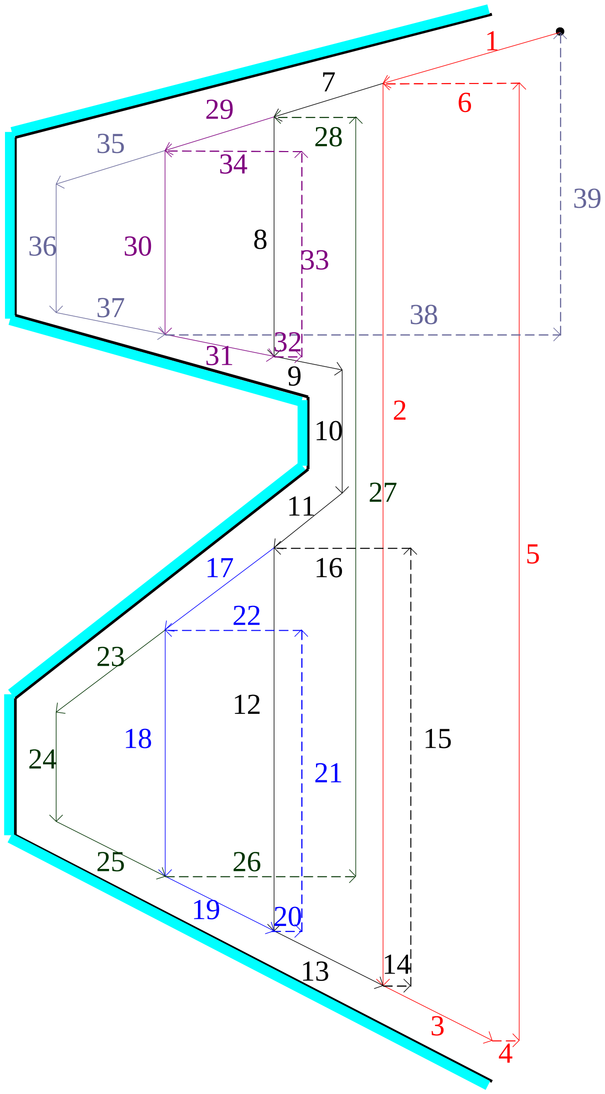


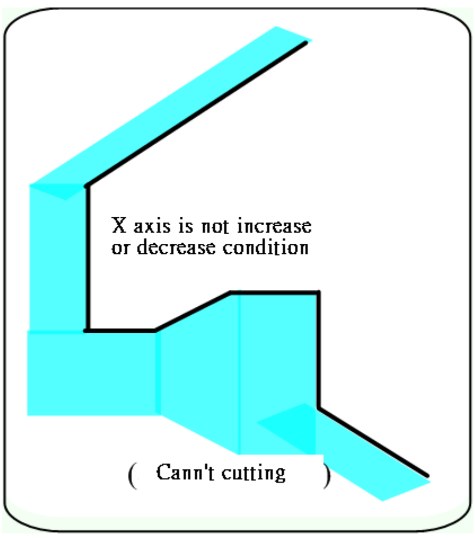


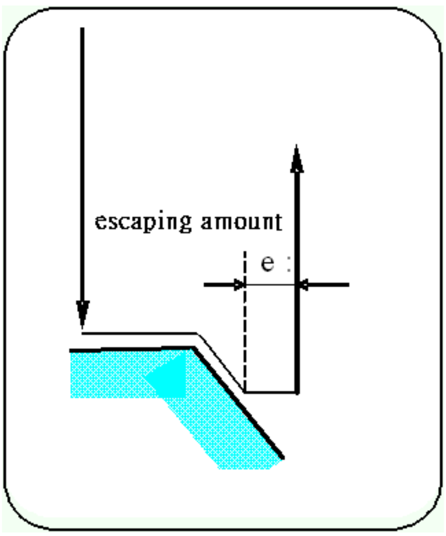
1.30.3 Action description:

1. Positioning to point A (start point) through rapid traverse (G00) before cycle starts.
2. Tool offsets to C point according to specified finishing allowance. (ΔU/2 in X direction, ΔW in Z direction)
3. Tool moves Δd distance in Z axis direction, feed to the contour endface.
4. Tool escapes(retracts) **e** amount of distance in Z axis direction, but moves by the direction of 45°.Tool then retracts in reversed X axis feed direction to the point that parallels in Z direction to the start point.
5. Move Δd amount of distance in Z direction, continuing next cycle.
6. In last cycle, tool cuts once along contour **A'→ B.**
7. Positioning to point A through rapid traverse.

TYPE II: Usually be performs in the middle part of the workpiece. The figure need not show monotone increase or decrease in the direction of Z axis. Only X axis needs to satisfy the condition that cutting amount is always increase or decrease.

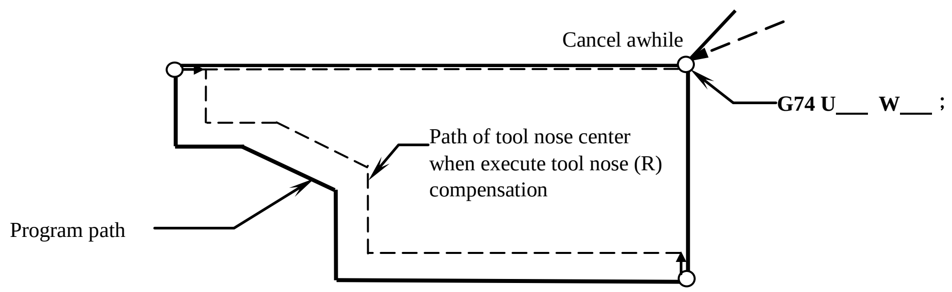




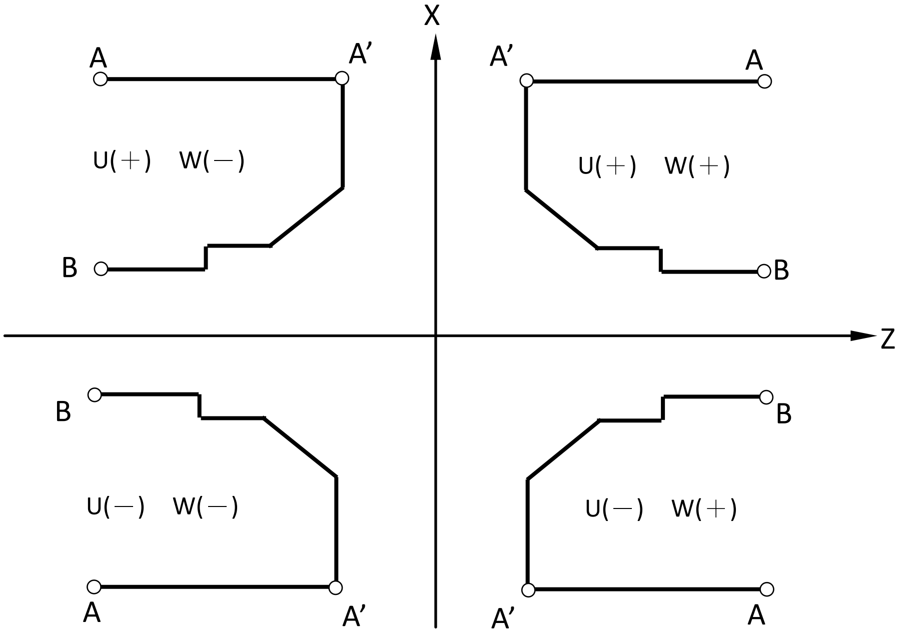


1.30.4 Notice

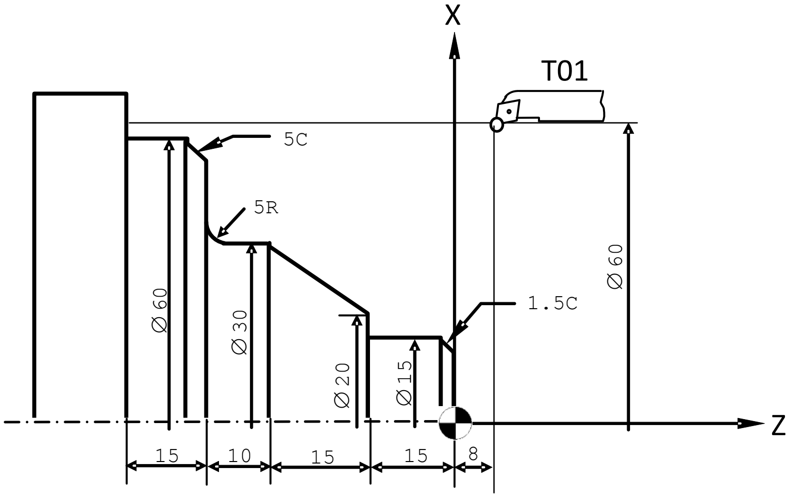
1. When **ns and nf** are not specified, specified **W** in **G74** block is depth of cut **Δd**. Otherwise, **W** is finishing allowance in Z direction.
2. Contour path is described by the blocks **ns and nf**, pathing through point A → A' → B. If X coordinate of contour path is not monotone,
   System will send out [MAR-002 the profile must be monotone along X, Z axis] alarm. If starting point (defined by the block before G74 command)
   is lower than contour path, System will send out [MAR-005 the position higher than the cycle start point] alarm.
3. F, S or T function issued within block range of **ns→nf** will be ignored. The relevant F,S and T functions specified in G74 block are effective instead.
4. G00/G01 command in the G74 blocks will be used to perform linear cut to the workpiece.
5. Using G74 command, If H is equal to 0, system will interpret as TYPE I. If H is equal to 1, system will interpret as TYPE II. If H is not specified,
   system will diagnose automatically. If H variable is specified wrong, System will send out [MAR-018 ERROR INPUT OF G73/G74 H VALUE] alarm
6. Sub-program can not be called during blocks **ns→nf**.
7. All tool nose compensation commands will be disabled when G74 is in the block. However, the compensation value will be added to the finishing allowance if exists.



8. Direction of finishing allowance: the direction dependeds on figures shown below. Path is A→A'→B.


X
Z
A
A'
U(+) W(−)
B
A'
A
U(+) W(+)
B
B
U(−) W(−)
A
A'
B
U(−) W(+)
A'
A

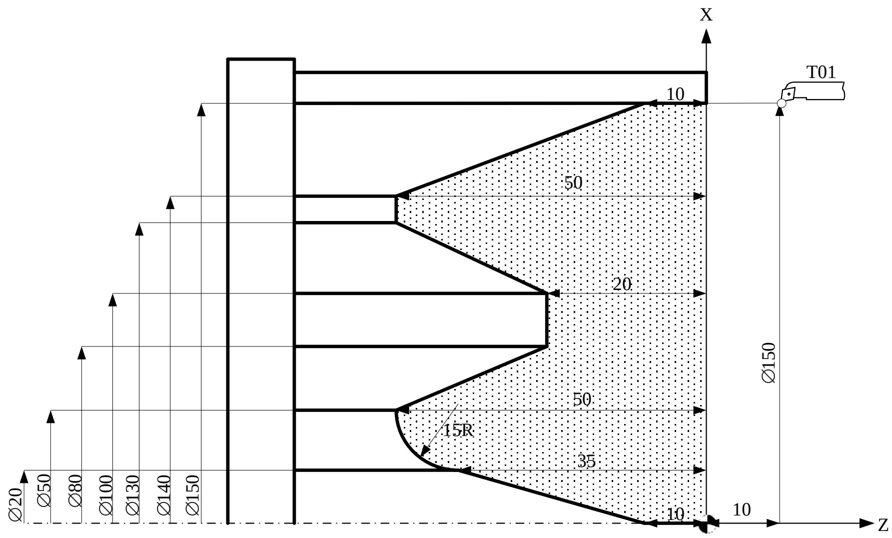
1.30.5 Example 1



T01 //use tool NO. 1
G92 S5000 //max. rotate speed 5000 rpm
G96 S130 M03 //constant surface speed, surface speed 130 m/min,
//spindle rotate CW
G00 X60.0 Z8.0 //positioning to start point
M08 //cutting liquid ON
G74 W3.0 R1.0 H0//depth of cutting in Z direction is 3.0 mm,
//escaping amount is 1.0 mm
G74 P01 Q02 U0.8 W0.2 F0.6 //execute stock removal in turning,
//the sequence of block N01→N02, finishing
//allowance in X direction is 0.8 mm, finishing
//allowance in Z direction is 0.2mm, feedrate 0.6
//mm/rev
N01 G00 Z-55.0
G01 X60.0
Z-45.0
X50.0 Z-40.0
X40.0
G03 X30.0 Z-35.0 R5.0
G01 Z-30.0
X20.0 Z-15.0
X15.0
Z-1.5

N02 X12.0 Z0.0
M09 //cutting liquid OFF
G28 X60.0 Z10.0 //positioning to specified mid-point, then return to
//machine zero point
M05 //spindle stops
M32 //program ends

1.30.6 Example 2



T01 //use tool NO. 1
G92 S5000 //max. rotate speed 5000 rpm
G96 S130 M03 //constant surface speed, surface speed 130 m/min,
//spindle rotate CW
M08 //cutting liquid ON
G00 X150.0 Z10.0 //positioning to start point
G74 W2.0 R1.0 H1 **//depth of cutting in Z direction is 2.0 mm,**
//**escaping amount is 1.0 mm**
G74 P01 Q02 U0.8 W0.1 F0.6 // execute stock removal in turning,
//the sequence of block N01→N02, finishing
//allowance in X direction is 0.8 mm, finishing
//allowance in Z direction is 0.1mm, feedrate 0.6
//mm/rev
N01 G00 X150.0 Z0.0

```
          G01 Z-10.0
                X140.0 Z-50.0
                X130.0
                X100.0 Z-20.0
                X80.0
                X50.0 Z-50.0
          G03 X20.0 Z-35.0 R15.0
          G01 X20.0
                X0.0 Z-10.0
N02             X0.0 Z0.0
M05                         //spindle stops
M32                         //program ends
M30
```

1.31Pattern Repeating Cycle (G75)

G75 command is a pattern repeating cycle. By this cutting cycle, it is possible to efficiently cut work whose rough shape has already been made by a rough machining, and only slightly larger than finishing shape, such as forged or cast workpieces, etc. Using G73, G74 wastes time on unnecessary routes. G75 command, instead, repeats cutting along the contour of workpiece for specified times. Each cutting cycle the tool moves toward the fringe for an appropriate amount of distance (depth).

1.31.1 Format

G75 UΔi WΔk R d

G75 P (ns) Q (nf) UΔu WΔw F___ S___ T___

Δi: distance in the X axis direction, this value can be specified by the parameter #4015 when this statement is not applied.

ΔK: distance in the Z axis direction, this value can be specified by the parameter #4016 when this statement is not applied.

d: number of cuts parallel to the contour, it can be specified by parameter #4017

ns: sequence number of the first block for the program of finishing shape

nf: sequence number of the last block for the program of finishing shape

Δu: distance and direction of finishing allowance in X direction

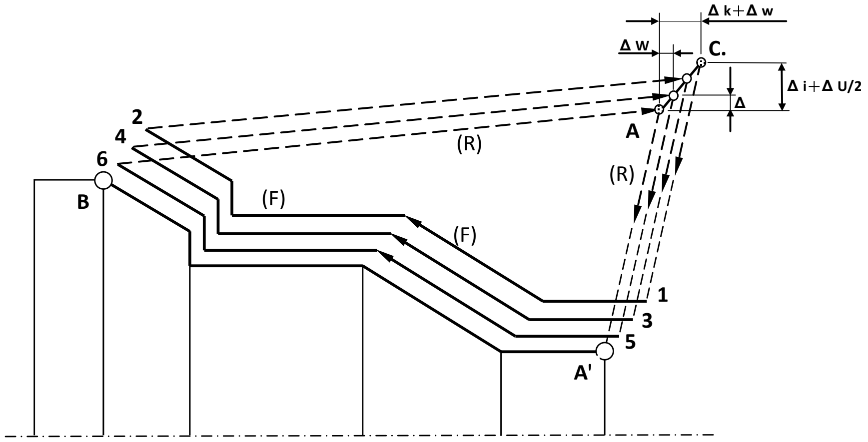
Δw: distance and direction of finishing allowance in Z direction

F: feedrate

T: **tool** number of the tool in use (tool selection)

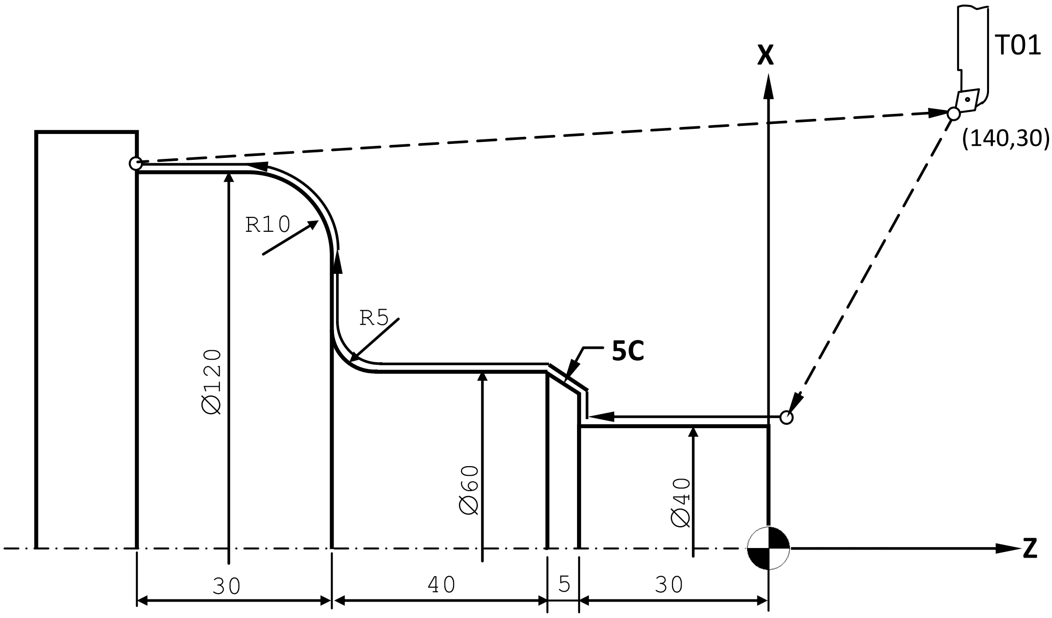
S: spindle speed

1.31.2 Action description



1. Positioning to **point A** (start point) before cycle starts
2. Tool offsets to point C by the sum of specified finishing allowance (ΔU/2 for X axis, ΔW for Z axis) and cutting value. (Δi for X axis, ΔW for Z axis)
3. Tool cuts through path A→A'→B, according given feed value and times of cutting to finish the cyclic processes.
4. After finishing last cutting cycle, tool will position back to point automatically.

1.31.3 Example




```
        T01                        //use tool NO. 1
G92 S5000               //max. rotate speed 5000 rpm
G96 S130 M03            //constant surface speed, surface speed
130                              //m/min,
                        //spindle rotate CW
G00 X140.0 Z30.0        //positioning to start point
M08                     //cutting liquid ON
```

G75 U15.0 W3.0 R3.0 //cutting value of X axis 15.0 mm, cutting

//value of Z axis 3.0 mm, cut 3 times

G75 P01 Q02 U0.8 W0.2 F300//execute Pattern Repeating,

```
                        //sequence of the block N01→N02,
finishing
                        //allowance of X axis 0.8 mm, finishing
                             //allowance
                        //of Z axis 0.2 mm, feedrate 300
µm/rev
N01 G00 X40.0 Z5.0        //shape of cutting
      G01 Z-30.0
            X50.0
            X60.0 Z-35.0
            Z-70.0
      G02 X70.0 Z-75.0 R5.0
      G01 X100.0
      G03 X120.0 Z-85.0 R10.0
N02 G01 Z-105.0
M09                     //cutting liquid OFF
```

```
G28 X140.0 Z30.0   //positioning to specified mid-point, then return to
                   //machine zero point
M05                //spindle stops
M30                //program ends
```

1.32End Face (Z axis) Peck Drilling Cycle (G76)

G76 command is end face peck (Z axis) drilling cycle, generally used for grooving on the end face and peck drilling in Z direction A cycle of cutting by Δk and return by e(in Z axis direction) is repeated. Therefore G76 can be used not only in grooving of workpiece endface and grooving during outer diameter cutting/cutting off, but also deep drilling of workpiece.

1.32.1 Format

G76 R e_

G76 X(U)_Z(W)_P(Δi)_Q(Δk)_R_(d)_F_

e: return amount(return amount in Z direction when cut Δk distance) ← the value can be setted by parameter #4011 when this statement is not applied.

X: X coordinate of point B（diameter）

Z: Z coordinate of point C

U: Incremental amount from A to B（diameter）

W: Incremental amount from A to C

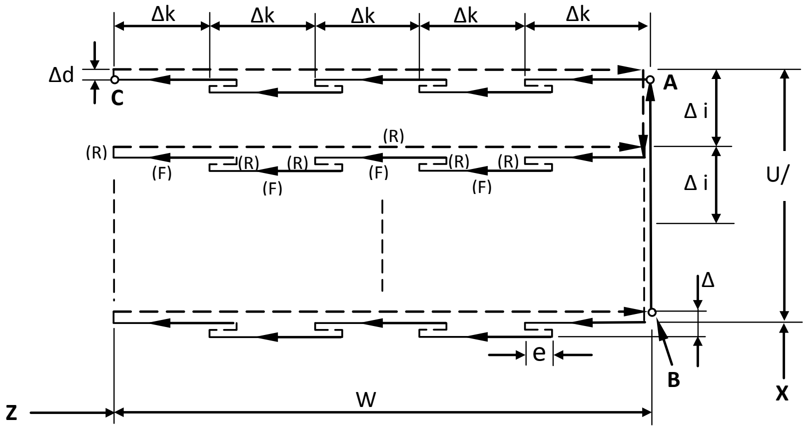
Δi: Travel distance in X direction (display by radius, positive)

Δk: Depth of cutting Z direction (positive)

Δd: Relief amount of the tool at the cutting bottom. (If set to be 0 , tool returns in original path)

F: Feedrate

1.32.2 Action description

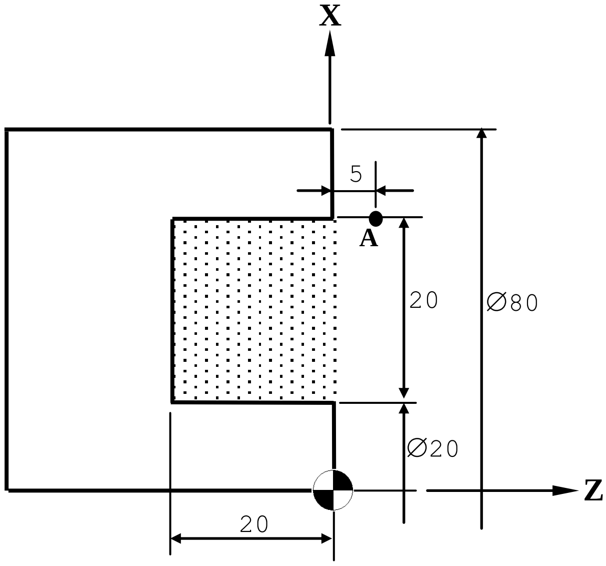


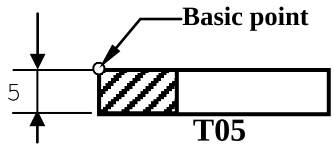
1. Positioning to **point A (start point)** before cycle starts.
2. When execute G76, tool starts peck drilling from point A to point C. Peck drilling is performed in the way that tool returns e amount each time tool cuts Δk distance(in Z direction). Tool immediately escapes Δd distance (in X direction then rapid traverse to the position parallel with start point.
3. Afterwards, tool moves Δi distance in X direction, and continues the cycle (the steps above). When machine performs G76 to point B, tool will return back to point A automatically.

1.32.3 Notice

1. **e and Δd is** specified by parameter **R**, when **X_** or **Z_** are specified. **R_** is escaping amount in X axis direction.
2. When there is only parameter **R** after G76 command, it is escaping amount in **Z** axis direction. This is called modal **G** code, in which G code is always effective until changing to new program.
3. If **Q(Δk)** is not specified, peck drilling will be canceled. Tool cuts once directly to the end point of Z axis.

1.32.4 Example





T05 //use tool NO. 5
G92 S1000 //max. rotate speed 1000 rpm
G96 S100 M03 //constant surface speed, surface speed 100 m/min,
//spindle rotate CW
M08 //cutting liquid ON
G00 X60.0 Z5.0 //positioning to point A
G76 R1.0
G76 X30.0 Z-20.0 P4.0 Q8.0 F100 //execute end face peck
//drilling cycle, after cutting 8.0 mm, tool escape
//1.0 mm distance, X axis moves 4.0 mm after
//cycle starts, feed rate 100 μm/rev
M09 //cutting liquid OFF

```
G28 X100.0 Z30.0 //positioning to specified mid-point, then return to
                 //machine zero point
M05              //spindle stops
M30              //program ends
```

1.33 Outer Diameter/Internal Diameter Drilling Cycle (G77)

G77 is outer diameter/internal diameter drilling cycle, generally used for grooving and peck drilling in X axis direction. To avert the variable pith at the ends of thread, and to make tool retraction easier, grooving on the outer diameter is performed. Furthermore, G77 is often called when cutting off the workpiece is needed in processing.

1.33.1 Format

G77 R e_

G77 X(U)___ Z(W)___ P(Δi)Q(Δk) R(Δd) F_

e: return amount(after cutting Δi distance in X axis direction) ←it can be setted by parameter #4011

X: X coordinate of point C (diameter)

Z: Z coordinate of point C

U: increment amount from B to C(diameter)

W: increment amount from A to B

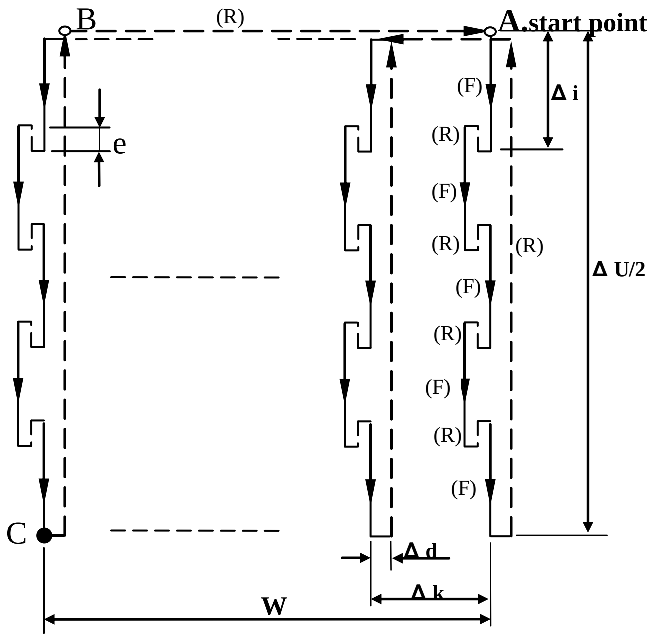
Δi: depth of cut in Z direction (positive)

Δk: travel distance in X direction (display by radius, positive)

Δd: Relief amount of the tool at the cutting g bottom. (this value is 0 when it returns in origin path)

F: feedrate

1.33.2 Action description

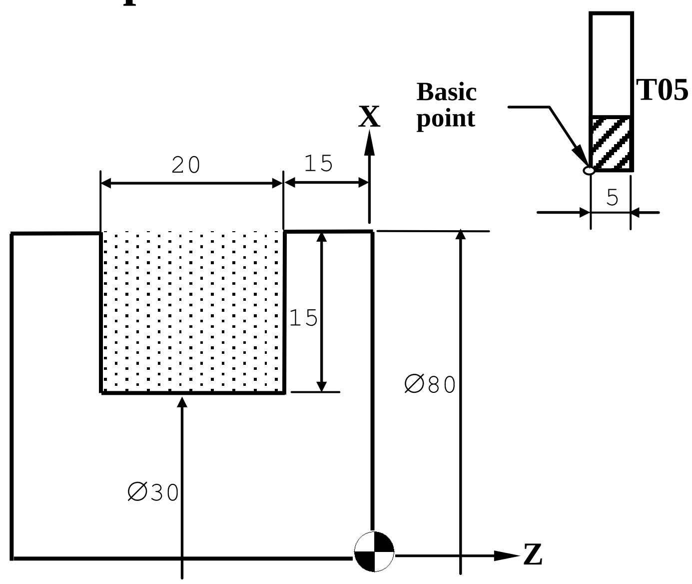


1. Positioning to **point A (start point)** before cycle starts.
2. When execute G77, tool starts peck drilling from point A to the specified point X. Peck drilling is performed in the way that tool returns e amount each time tool cuts Δk distance(in Z direction). Tool immediately escapes Δd distance (in X direction then rapid traverse to the position parallel with start point.
3. Afterwards, tool moves Δi distance in Z direction, and continues the cycle (same steps above in operation 2.). When machine performs G77 to the terminal point B, tool will return back to point A automatically.

1.33.3 Notice

1. **e and Δd is** specified by parameter **R**, when **X_**or **Z_**are specified. **R** is escaping amount in Z axis direction.
2. When there is only parameter **R** after G77 command, it is escaping amount in **Z** axis direction. This is called modal **G** code, in which G code is always effective until changing to new program.
3. If **P(<u>Δi</u>)** is not specified, peck drilling will be canceled. Tool cuts once directly to the end point of X axis.

1.33.4 Example



T05 //use tool NO. 5
G92 S1000 //max. rotate speed 1000 rpm
G96 S100 M03 //constant surface speed, surface speed 100 m/min,
//spindle rotate CW
M08 //cutting liquid ON
G00 X70.0 Z20.0 //approaching to workpiece
Z-20.0 //positioning to cutting start point
G77 R1.0
G77 X30.0 Z-35.0 P8.0 Q4.0 D0.0 F150 //execute Outer
//Diameter/Internal Diameter Drilling Cycle, after
//cut 8.0 mm, then tool escapes 1.0 mm, Z axis
//moves 4.0mm after first cycle, feed rate 100
//µm/rev
M09 //cutting liquid OFF

```
G28 X80.0 Z50.0    //positioning to specified mid-point, then return to
                   //machine zero point
M05                //spindle stops
M30                //program ends
```

1.34Multiple Thread Cutting Cycle (G78)

By G78 command (multiple thread cutting cycle),system automatically program the repeated paths which is needed to accomplish the thread cutting process. The controller computes counts of thread cutting needed , depth of cutting and start points of each cutting cycle according to the specified parameter assigned by user.

1.34.1 Format

G78 P m r a Q Δdmin R d ;

G78 X(U)___ Z(W)___ R Δi P(Δk) Q(Δd) H___ (F__ or E__) ;

P:

m: repetition count in finishing, specified by system parameter #4044.

r: chamfering amount, specified by system parameter #4043.

a: angle of tool tip, angle from 80^0, 60^0, 55^0, 30^0, 29^0 and 0^0 can be specified or specified by system parameter #4042.

Q(Δdmin): minimum cutting depth$(\Delta d\sqrt{n}-\Delta d\sqrt{n-1})<Q$, specified by system parameter #4045

d: finishing allowance, specified by system parameter #4041

X(U): X coordinate in end point(bottom of tooth)

Z(W): Z coordinate in end point(bottom of tooth)

Δi: difference of thread radius

Δk: height of thread

Δd: depth of cut in first cycle

F: lead of thread in metric system (unit: mm/tooth)

E: lead of thread in imperial system (unit: tooth/inch)

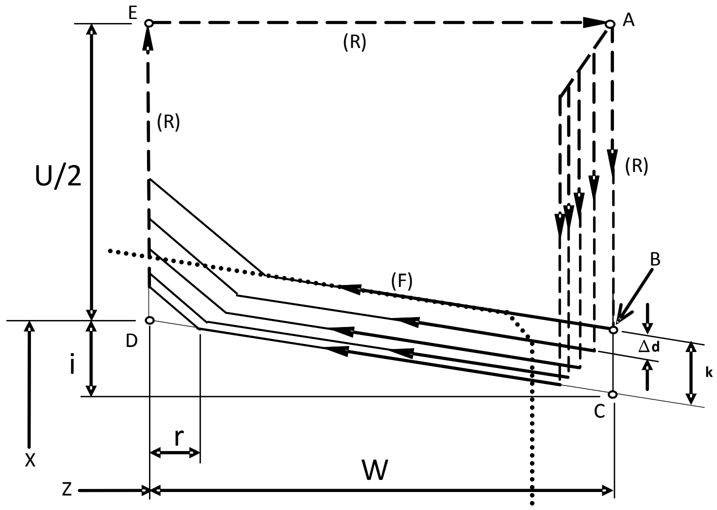
H: numbers of thread (ex: H3 is three thread type cutting. Multiple thread F function is the distance neighbor thread)

1.34.2 Ways of thread cutting

1. G33(thread cutting): A 4-block sequence of commands is needed to finish one thread cutting, thus the programming of thread cutting in G33 is inconvenient and time-consuming.
2. G21(thread cutting cycle): A “single” cycle command of thread cutting. we can use one block of command to finish thread cutting, but it also need to repeating thread cutting many times so the program is also too long.
3. G78(multiple thread cutting cycle): Byusing only one commandG78finishes all needed cycle in thread cutting.Therefore G78 much simplifies shortens the procedure of programming.

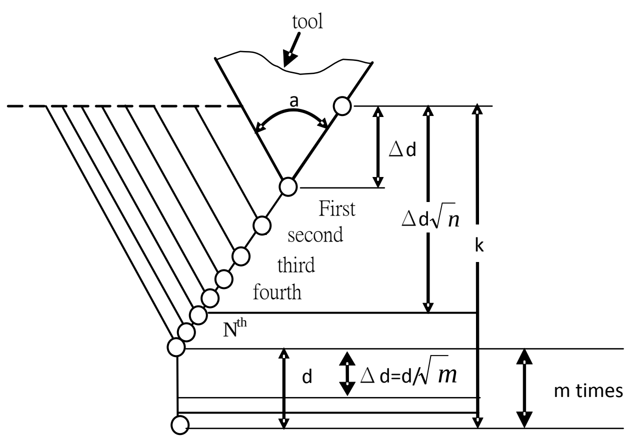
1.34.3 Action description

cutting path:



1. Positioning to **point A(start point)** by rapid traverse before cycle starts.
2. The tool cuts along path A→B→E→A, depending on the cutting feed to finish first time of threading.
3. After rough thread cutting, machine performs equivolume cutting (in sequence) accordong to **finishing allowance(d)** and **repetition count in finishing(m)** to accomplish finishing cutting.
4. After final cutting (along A→C→D→E→A), tool stops at point A.

The way of feed in thread cutting and the depths of each cut:



1.34.4 Notic

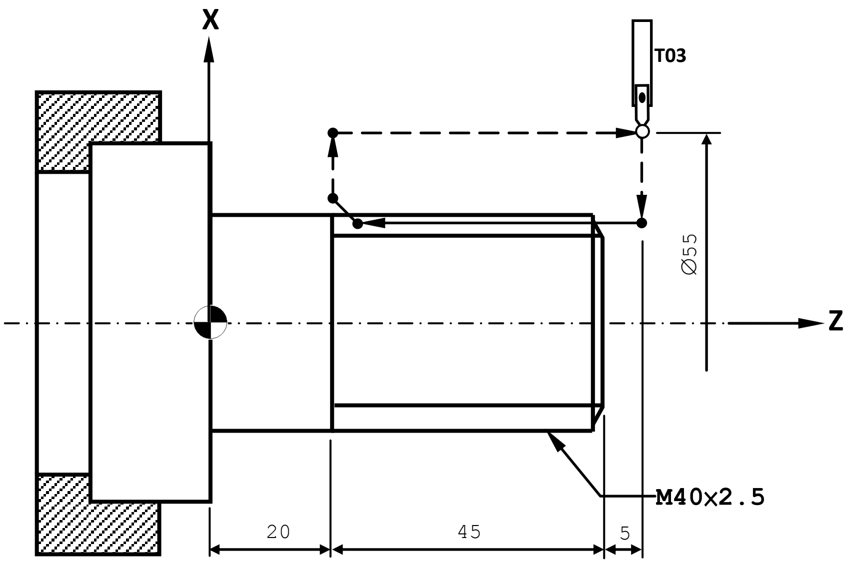
1. From version 10.114.56E/10.116.0E/10.116.5 (included), the spindle override of entire thread-cutting cycle is

locked at the value of the start of cycle, i.e., the spindle override button is in vain during thread-cutting cycle.

2. Before version 10.114.56E/10.116.0E/10.116.5, during thread-cutting cycle, the spindle override is locked at 100% when cutting and resume to setting of control panel while retracting. Therefore, one apply thread-cutting cycle with a spindle override that is not equal to 100% will find the spindle is under a frequent acceleration and deceleration situation.

1.34.5 Example 1

Compare with example one of G21



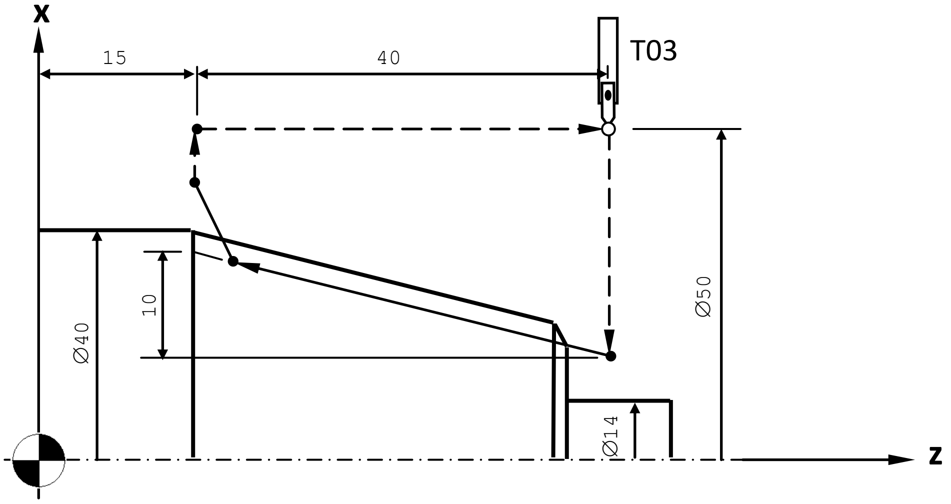
T03 //use tool NO. 3
G97 S600 M03 //constant rotate speed, 600 rpm CW
G00 X50.0 Z70.0 //positioning to the start point of cycle
M08 //cutting liquid ON
G78 P011060 Q0.15 R0.02//**execute multiple repetitive cycle,**
//**finishing cutting once, escaping amount=Lead,**
//angle of tooth 60°, Min. depth of cutting 0.15
//mm, finishing allowance 0.02 mm
G78 X36.75 Z20.0 R0.0 P1.624 Q1.0 H3 F2.5//**difference radius**
//**of multiple thread cutting cycle is 0 mm, depth**

//of thread 1.624 mm, first cutting value is 1.0
//mm, lead of thread 2.5 mm, three tooth thread
//cutting

G28 X60.0 Z75.0 //positioning to specified mid-point and return to
//machine zero point
M09 //cutting liquid OFF
M05 //spindle stops
M30 //program ends

1.34.6 Example 2

compare with example two of G21, single tooth type, Pitch＝2.5 mm




```
T03                 //use tool NO. 3
G97 S600 M03        //constant rotate speed, 600 rpm CW
G00 X50.0 Z55.0     //positioning to start point of cycle
M08                 //cutting liquid ON
G78 P011060 Q0.15 R0.02//execute multiple repetitive cycle,
                    //finishing cutting once, escaping amount=Lead,
                    //angle of tooth 60°, Min. depth of cutting 0.15
                    //mm, finishing allowance 0.02 mm
G78 X36.75 Z15.0 R-10.0 P1.624 Q1.0 F2.5//difference radius of
                    //multiple thread cutting cycle is 10.0 mm, depth of
                    //thread 1.624 mm, first cutting value is 1.0 mm,
                    //lead of thread 2.5 mm, single tooth thread cutting
G28 X60.0 Z70.0     //positioning to specified mid-point and then return
                    //to machine zero point
M09                 //cutting liquid OFF
M05                 //spindle stops
M30                 //program ends
```

1.35Canned Cycle For Drilling (G80～G89)

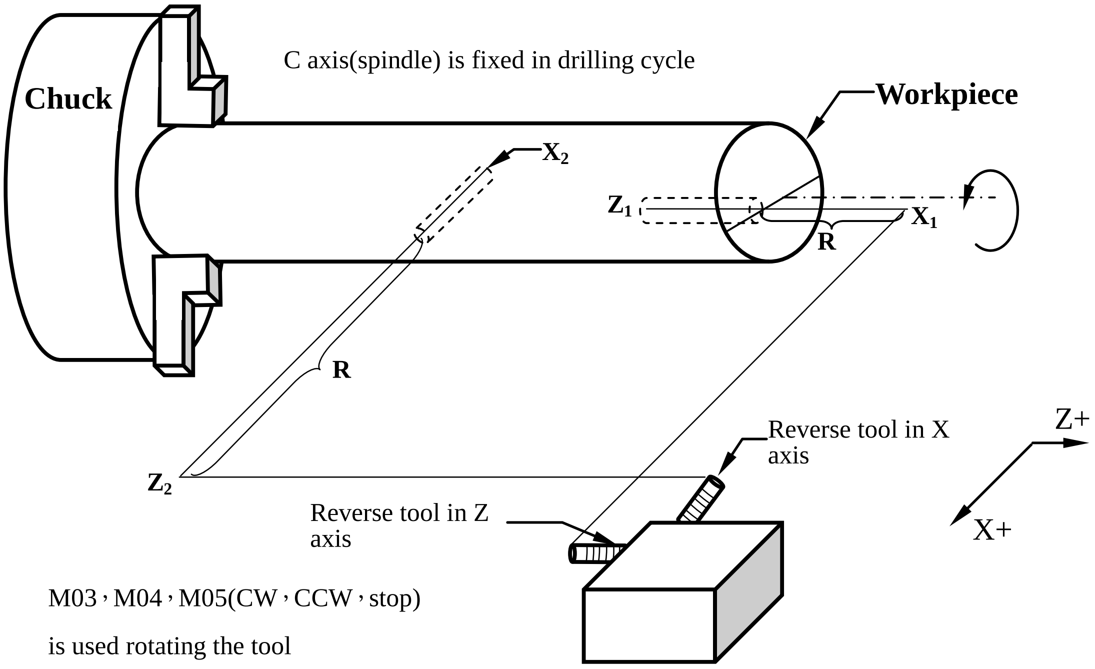
The canned cycle for drilling simplifies the program by instruct the CNC to perform necessary moves in only one block containing G functions. A customized canned cycle is a preset sequence of events initiated by a single block of data. The objective of a canned cycle is to simplify the process normally be performed through several blocks.

Table of Canned Cycle

G code	Drilling axis	Operation in the bottom hole position	Retraction operation	Applications
G80	----	----	----	Cancel
G83	Z	Dwell	Rapid traverse	Front drilling cycle
G84	Z	Spindle CCW	Cutting feed	Front tapping cycle
G85	Z	Dwell	Cutting feed	Front boring cycle
G87	X	Dwell	Rapid traverse	Front drilling cycle
G88	X	Spindle CCW	Cutting feed	Front tapping cycle
G89	X	Dwell	Cutting feed	Front boring cycle

Note 1: use M04 command to reverse the spindle.
Note 2: Whether G83 ,and G87 is cutting feed or intermittent feed is decided by Q_command.

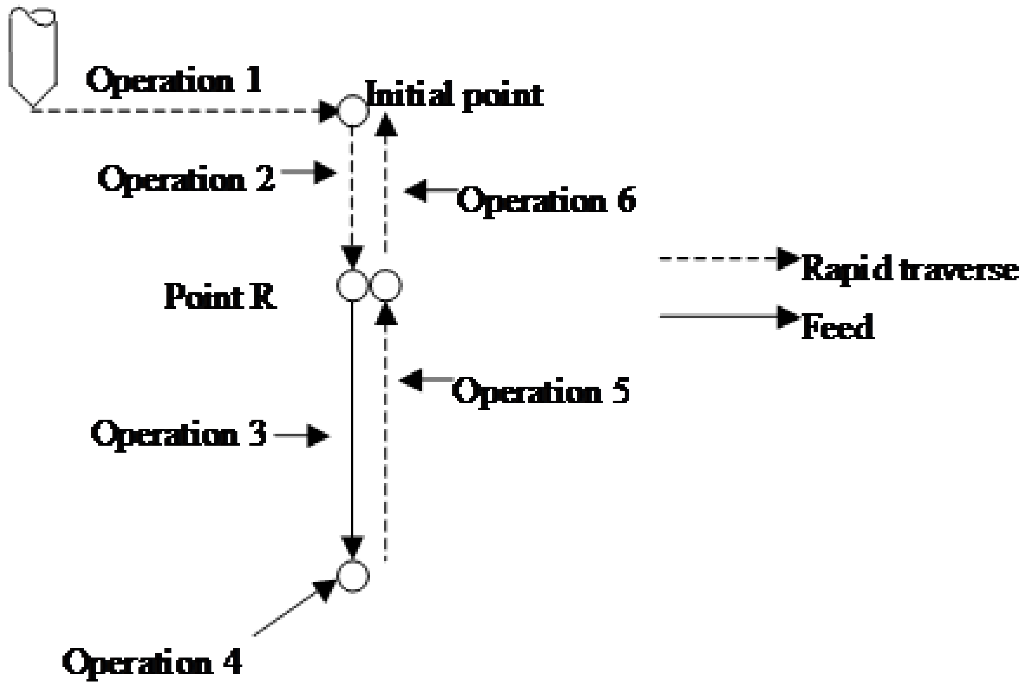
1.35.1 Drilling cycle figure



※The difference between G83/G87, G84/G88, G85/G89 is the direction of drilling-axis. G83, G84, G85 are for Z axis and G87, G88, G89 are for X axis.

In general, the drilling cycle consists of the following six operation sequences:

Operation 1 positioning of X(Z) and C axis
Operation 2 Rapid traverse up to point R level
Operation 3 Hole machining
Operation 4 Operation at the bottom of a hole
Operation 5 Retraction to point R level
Operation 6 Rapid traverse up to the initial point



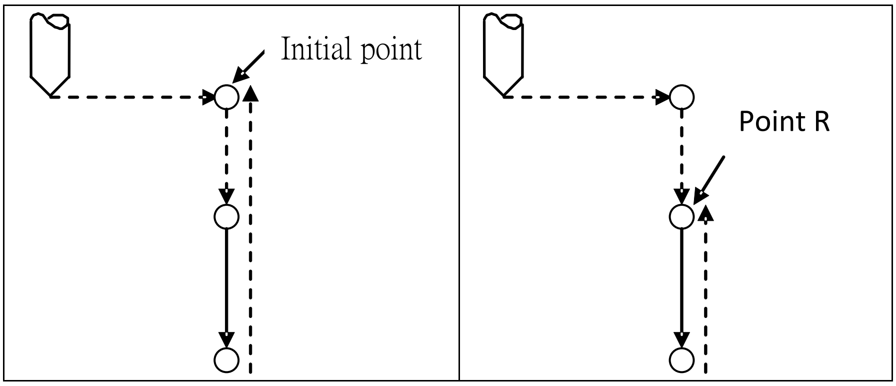
※Two modal G code specify the ways of drilling cycle.

In returning, G98/G99 specifies whether the tool retract to point-R level or initial level.(an illustration is shown below)

If the regression position is initial level/ point-R level, the start point of next cutting is initial level/ point-R level.

The initial level doesn't change even when drilling is performed in G99 mode.

G98	G99


Initial point
Point R

1.36Front/Side Drilling Cycle (G83/G87)

G83/G87 command is front/side drilling cycle, generally used in drilling of the lathe, it uses rotating tool to do front/side drilling cycle to clamped workpiece(fixed).

1.36.1 Format

G83 X(U)_C(H)_Z(W)_R_Q_P_F_K_M_

or

G87 Z(W)_C(H)_X(U)_R_Q_P_F_K_M_

X(U)_C_or Z(W)_C_: Hole position data

Z(W)_C_or X(U)_C_: The distance from point R to the bottom of the hole

R_: The distance from the initial level to point R level

Q_: Depth of cut for each cutting feed

P_: Dwell time at the bottom of the hole (sec)

F_: cutting federate

K_: Number of repetitions

M_: M code for clamping C axis. C axis is unclamped when Clamp Code adds 1 (Unclamp Code)

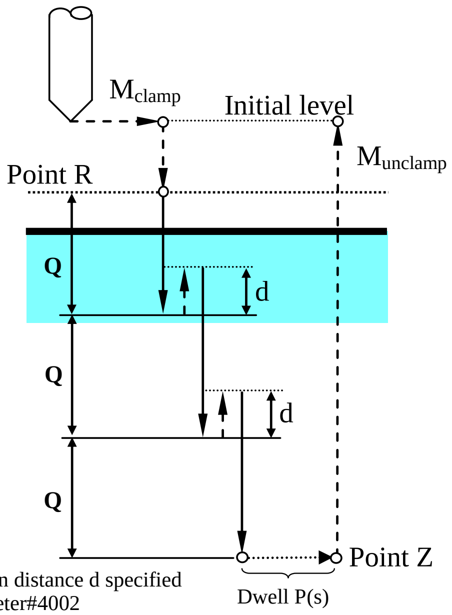
1.36.2 Note

Using G83 or G87, if absolute value of argument R is larger than the relative Z or X coordinate of 【G83 or G87 command】 and the block before, System will send out [MAR-011 the R level is lower the bottom level of hole] alarm. If Z or X coordinate is not specified, System will send out [MAR-012 absent bottom level, Z, in canned cycle] alarm.

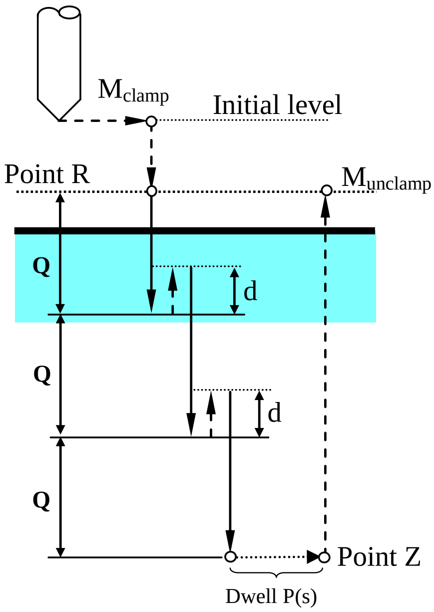
1.36.3 PIC

TYPE I: High speed deep hole drilling cycle (Custom Parameter No.4001= 1)

G83/G87(G98)



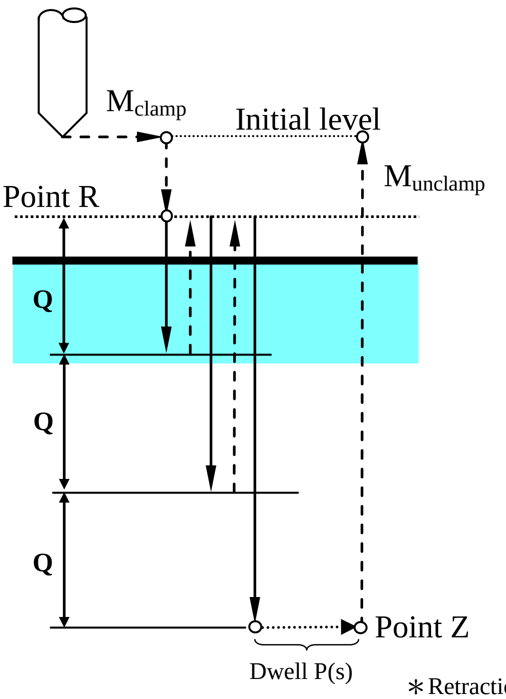
G83/G87(G99)



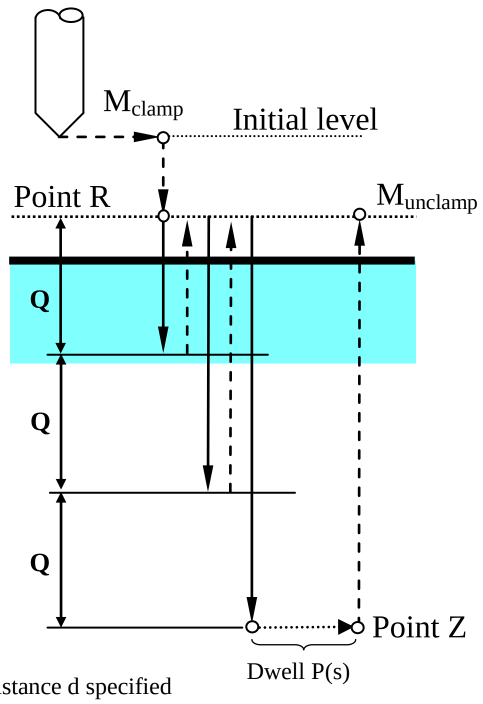
* Retraction distance d specified in parameter#4002

TYPE II: deep hole drilling cycle 1(Custom Parameter No.4001=0

G83/G87(G98)



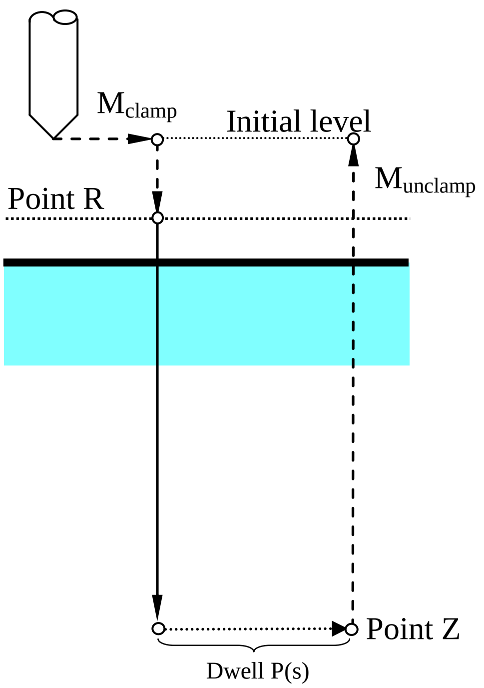
G83/G87(G99)



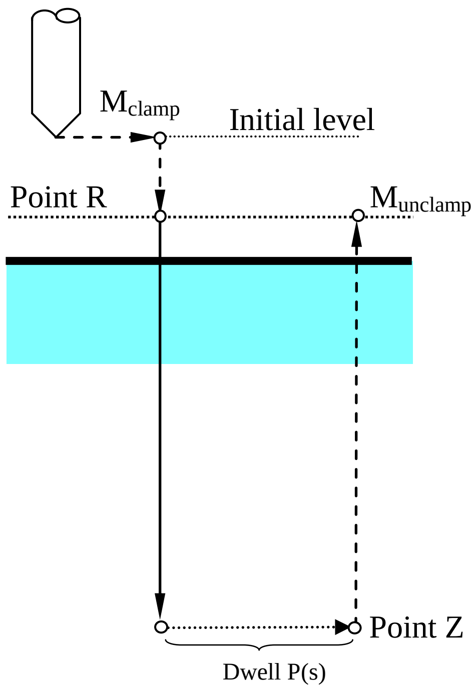
＊Retraction distance d specified in parameter#4002

TYPE III: Drilling without specified Q

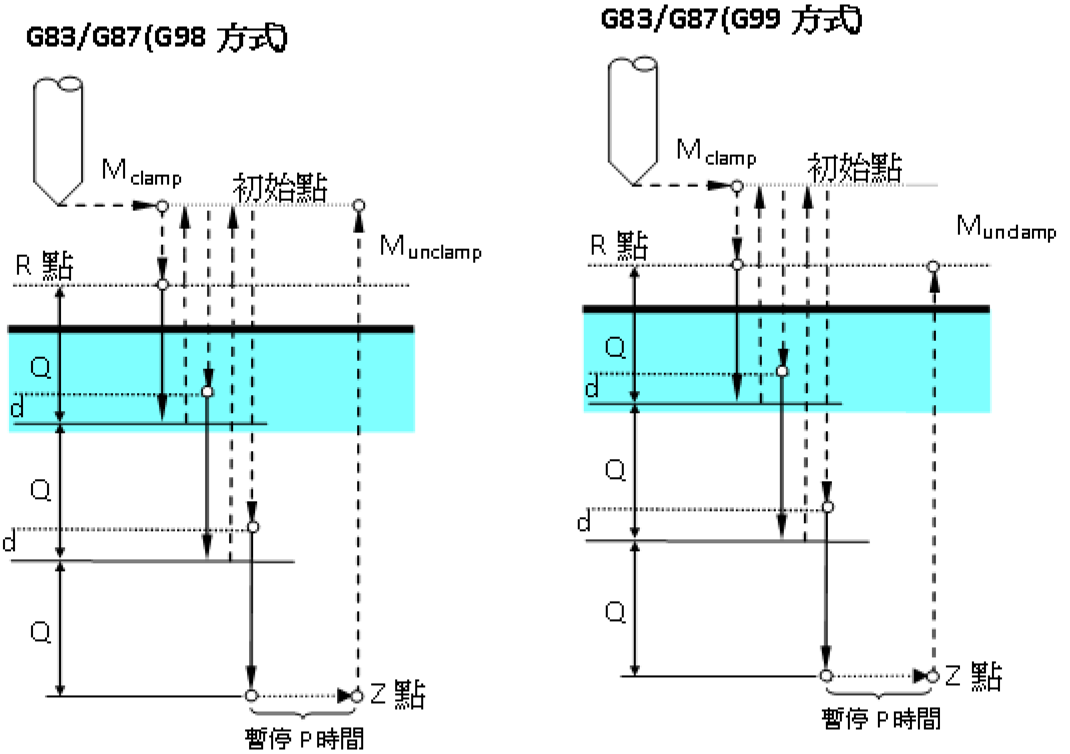
G83/G87(G98)



G83/G87(G99)



Type IV：Deep hole drilling cycle 2 (Custom Parameter No. 4001=2)



1.36.4 Example

Suppose M31 is the command of Clamp for C axis, M32 is the command of Unclamp for C axis.

```
S1000                  //spindle speed 1000 rpm
G00 X50.0              //rapid traverse to start point
G98 G83 Z-40.0 C0.0 R-5.0 P10.0 Q500 F500 M31
                       // first hole drilling of C axis at 0°
C90.0 M31              // second hole drilling of C axis at 90°
C180.0 M31             // third hole drilling of C axis at 180°
G80                    //cycle cancels
M30                    //program ends
```

1.37Front/Side Tapping Cycle (G84/G88)

G84 / G88 command is Front(Z)/Side(X) Tapping cycle, generally used in tapping of the lathe. Rotating tool performs front/side tapping cycle on clamped workpiece(fixed).

1.37.1 Format

G84 X(U)_C(H)_Z(W)_R_P_F_K_M_

or

G88 Z(W)_C(H)_X(U)_R_P_F_K_M_

X(U)_C_or Z(W)_C_: coordinate of the hole

Z(W)_C_or X(U)_C_: position(absolute mode) of the bottom of the hole (The distance from point R to the bottom of the hole)

R_: The distance from the initial level to point R level(always positive)

P_: Dwell time at the bottom of the hole (sec)

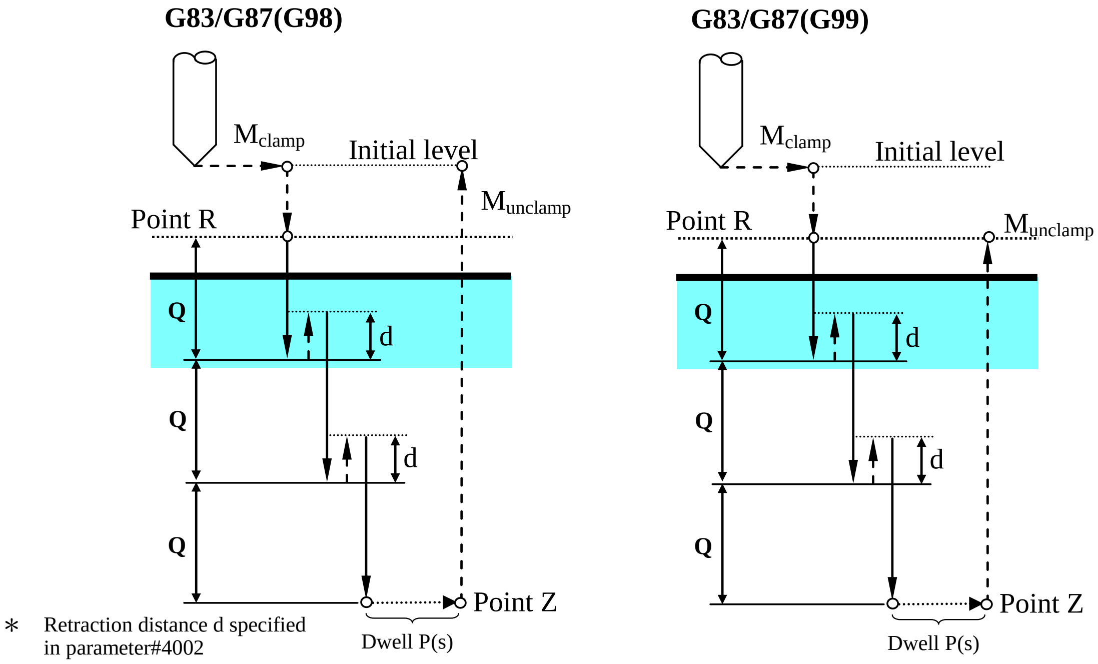
F_: cutting federate(mm/rev), equivalent to the pitch of metric system

K_: Number of repetitions

M_: M code for clamping C axis. C axis is unclamped when Clamp Code adds 1 (Unclamp Code)

Q_: Depth of cut in peck tapping, incremental and positive. (System is set to be normal tapping without specified value)

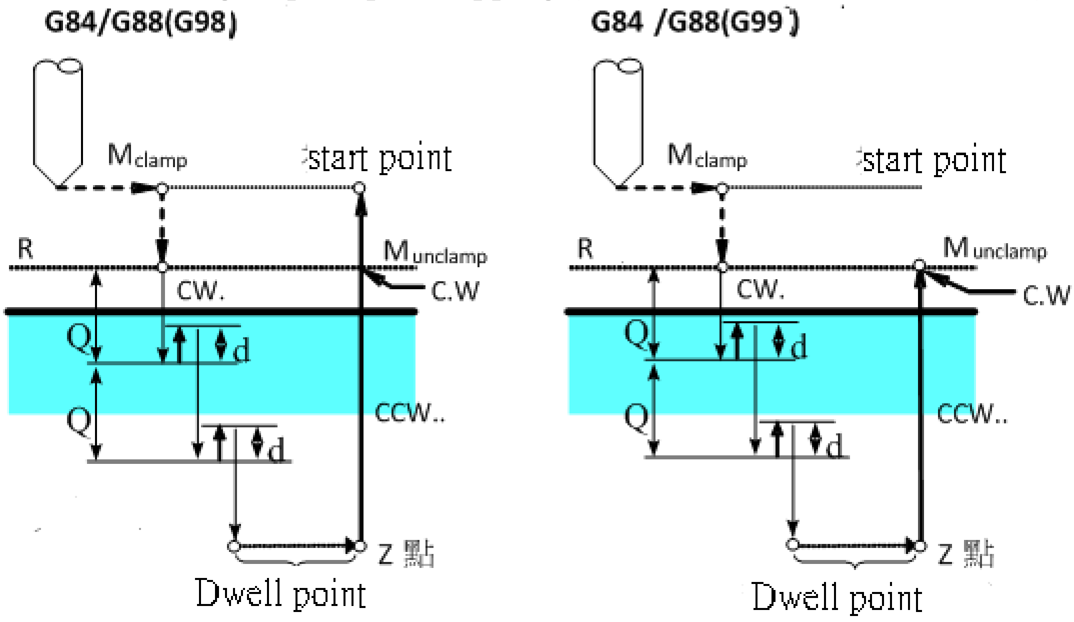
TPYE I



1. Action starts, Z axis rapid traverse to point R by G00 (R mudt be in incremental value)
2. Start tapping, pitch is the specified F value.

3. When Z axis reachs the specified Z depth of G84(Z absolute / W incremental)
4. Spindle stops.
5. Dwell P(sec) (with floating point, unit : 1 s, without floating point, unit : 0.001 s)
6. Spindle rotates CCW (use M04 in CNC)
7. Escape to point R by the feedrate of tapping.
8. Dwell several second (Dwell time set at Pr4003. Defaul value is 0 second)
9. Spindle rotates CW (M03)
10. Return to initial point(G98) or stop at point R(G99)

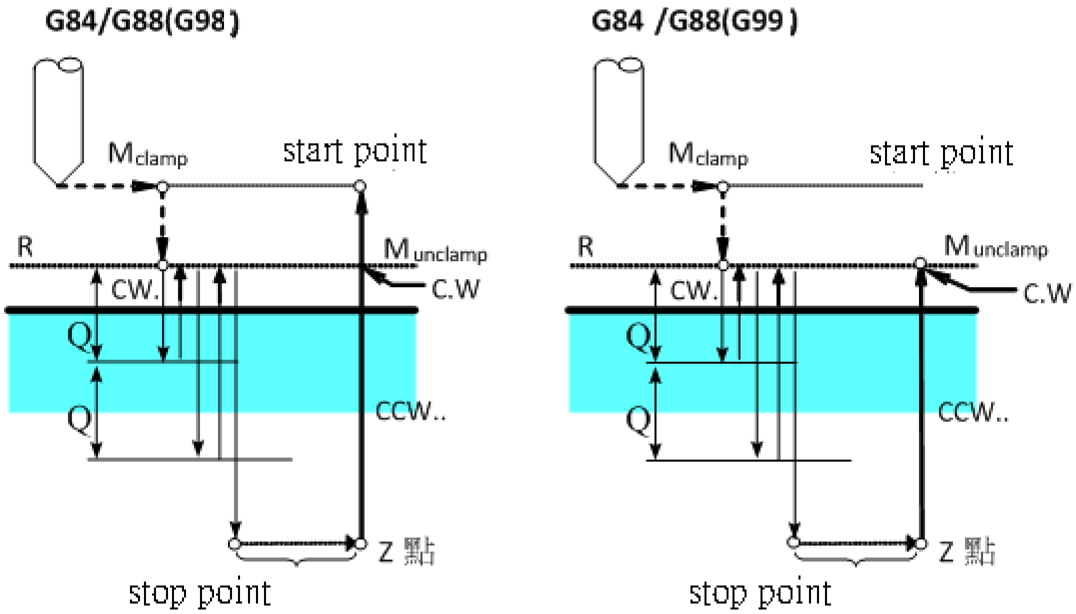
TYPE II: High-Speed peck tapping (Custom Parameter No.4004=1)



1. Rapid traverse (G00) to specified point (X, C) when process starts.
2. Rapid traverse (G00) to specified point R.
3. Interpolate (G01) depth of cut (Q) from the present position.
4. Spindle stops and rotates CCW. Use G01 to retract height d (set by Pr4005).
5. Spindle stops and rotate CW. Interpolate (G01) to the position which is a depth of cut (Q) below the last tapping position.
6. Spindle stops and rotates CCW. Use G01 to retract height d (set by Pr4005).
7. Repeat the above steps until tapping to the end of hole.
8. Dwell P seconds then rotate CCW.
9. By G01 feedrate, return to specified point R (G99).

10. Dwell a few seconds (set by Pr 4003, default value is 0 second) then rotate CCW.
11. Rapid traverse(G00) to the start point (G98).

TYPE II: Peck tapping (Custom Parameter No.4004=0)



1. Rapid traverse (G00) to specified point (X, C) when process starts.
2. Rapid traverse (G00) to specified point R.
3. Interpolate (G01) depth of cut (Q) from the present position.
4. Spindle stops and rotates CCW. Use G01 to retract to the point-R level.
5. Spindle stops and rotate CW. Interpolate (G01) to the position which is a depth of cut (Q) below the last tapping position.
6. Spindle stops and rotates CCW. Use G01 to retract to the point-R level.
7. Repeat the above steps until tapping to the end of hole.
8. Dwell P seconds then rotate CCW.
9. By G01 feedrate, return to specified point R (G99).
10. Dwell a few seconds (set by Pr 4003, default value is 0 second) then rotate CCW.
11. Rapid traverse (G00) to the initial level (G98).

1.37.2 Notice

1. when first time tapping, spindle has to be initiated to rotate CW.
2. If initial point is the same as point R, we do not need to specify R.
3. If there is no power tool seat on lathe, the parameter X ,C ,K ,M of G84 need not to be specified.
4. When G84/G88 ends, spindle returns to rotate CW.

5. G84/G88 is canceled by G80. When G00/G01/G02/G03 in the program be executed, G84/G88 will also be canceled automatically.
6. If M4 is specified behind G84/G88, it is set to be the left hand tapping.
7. Please avoid to use M code behind G84/G88.
8. Using G84 or G88, if absolute value of argument R is larger than the relative Z or X coordinate of 【G84 or G88 command】 and the block before, System will send out [MAR-011 the R level is lower the bottom level of hole] alarm. If Z or X coordinate is not specified, System will send out [MAR-012 absent bottom level, Z, in canned cycle] alarm.

1.37.3 Example

Suppose M31 is Clamp command of C axis M32 is Unclamp command of C axis

```
M03 S500          //spindle is initiated to rotate CW 500rpm
G00 X50.0         //positioning to start point by rapid traverse
G98 G84 Z-40.0 C0.0 R-5.0 P10.0 F500 M31    // first hole drilling
                                            //of C axis at 0°
C90.0 M31         // second hole drilling of C axis at 90°
C180.0 M31        // third hole drilling of C axis at 180°
G80 M05           //cancel tapping mode, spindle stops
M30               //program ends
```

1.38 Front/Side Boring Cycle (G85/G89)

G85/G89 command is Front/Side Boring cycle, used in boring of the CNC lathe. The rotating tool performs front/side tapping cycle to process the clamped workpiece(fixed).

1.38.1 Format

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

or

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

X(U)_C_or Z(W)_C_: Hole position/coordinate of the hole

Z(W)_C_or X(U)_C_: The distance from point R to the bottom of the hole/ position of the bottom of the hole (absolute mode)

R_: The distance from the initial level to the point R level

P_: Dwell time at the bottom of hole (s)

F_: Feedrate

K_: Number of repetitions

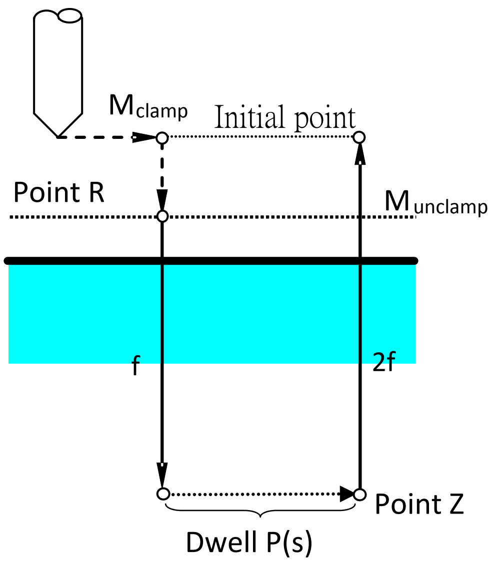
M_: M code for C axis clamping, C axis is unclamped when Clamp Code adds 1 (Unclamp Code)

1.38.2 Note

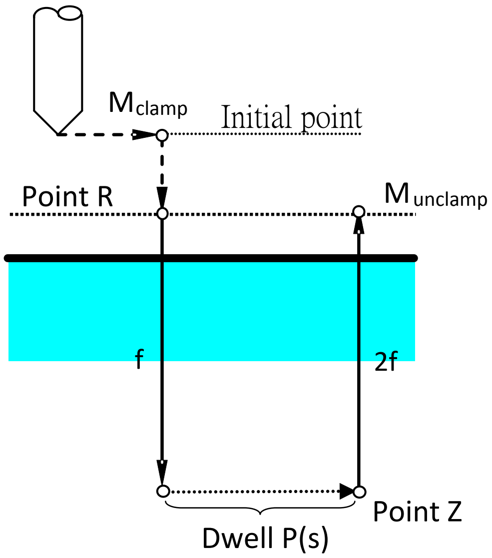
Using G85 or G89, if absolute value of argument R is larger than the relative Z or X coordinate of 【G85 or G89 command】 and the block before, System will send out [MAR-011 the R level is lower the bottom level of hole] alarm. If Z or X coordinate is not specified, System will send out [MAR-012 absent bottom level, Z, in canned cycle] alarm.

1.38.3 PIC

G85/G89(G98)



G85/G89(G99)



1.38.4 Example

Suppose M31 is Clamp command of C axis M32 is Unclamp command of C axis.

S1000 M03 //spindle rotates CW, rotate speed 1000 rpm
G00 X50.0 //positioning to start point
G98 G85 Z-40.0 C0.0 R-5.0 P100 F500 M31
// first hole drilling of C axis at 0
C90.0 M31 // second hole drilling of C axis at 90°
C180.0 M31 // third hole drilling of C axis at 180°
G80 //cycle cancels
M30 //program ends

1.39Coordinate System Setting/Max. Spindle Speed Setting (G92)

G92 command has two functions. One is coordinate system setting and another is Max. speed of spindle setting G92 can define any appropriate position to be zero point of workpiece coordinate. The distance from the position of tool to the machine zero point is used to set a zero point of new coordinate. After setting, tool starts machining from this point and the absolute command is calculated according to the new reference coordinate. This command can also be used in the offset of coordinate system. If the old coordinate is (X, Z), the new coordinate will be (X＋ΔU, Z＋ΔW). When using G96 (constant surface speed control) command, in order to avoid the excessively high speed of spindle speed due to too small effective diameter of workpiece , G92 is also used to limit the max. speed of spindle.

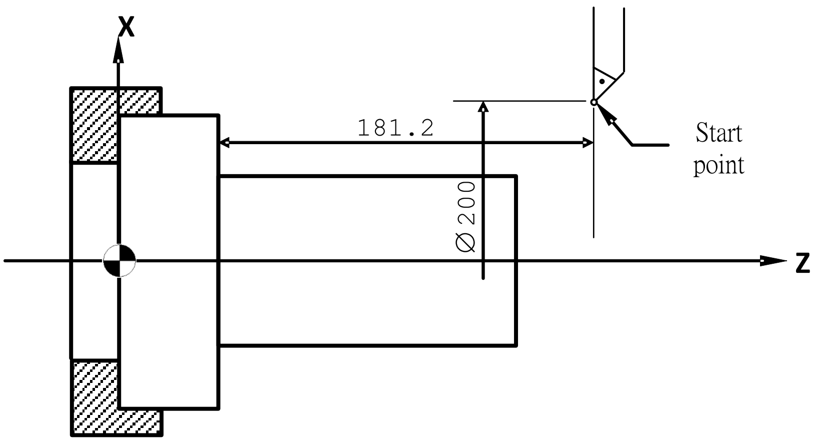
1.39.1 Format

G92 X_Z_
or
G92 S_

X ,Z: basic coordinate system position setting (G92) in program coordinate system
S: spindle speed

1.39.2 Example 1

Coordinate system setting



Example：G92 X200.0 Z181.2;
//tool is starting from specified point

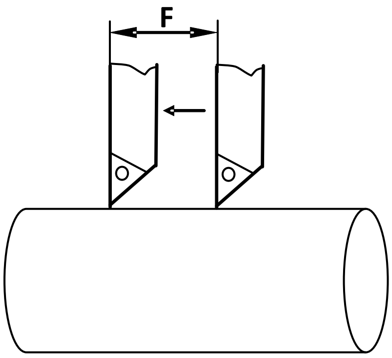
Unit Setting of Feed Amount (G94/G95)

This command set feed amount unit of F_function (tool movement per minute or per revolution). G94 is for feed per minute (mm/min inch/min), G95 is for feed per revolution (mm/rev, inch/rev).

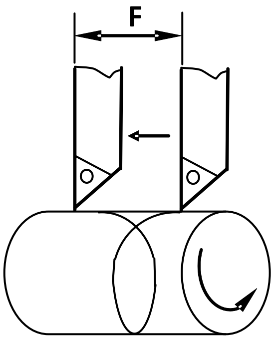
1.39.3 Format

G94 F_
G95 F_

1.39.4 PIC



G94. feed per minute (mm/min or inch/min)



G95. feed per revolution (mm/rev or inch/rev)

1.40 Constant Surface Speed Control (G96/G97)

G96 command specifies the surface speed of the contact point between tool and workpiece. G97 is constant surface speed cancel command, and also functions to set spindle speed. To control the surface speed while the diameter of the workpiece varies, a lathe operator uses G96 to specify the constant surface speed. If a constant spindle rotate speed is to be set, regardless the value of the diameter workpiece has, G97 can be performed. The surface speed follows the formula shown below:

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

V: surface speed, (use G96 to specify a value, unit M/MIN or FEET/MIN).
D: valid diameter of workpiece, unit mm or inch
N: spindle rotate speed, specified by G97, unit RPM.

1.40.1 Format

G96 S_ constant surface speed control ON
G97 S_ constant surface speed control OFF

1.40.2 Example

1.40.2.1 Constant surface speed:

```
G92 S2000        //limit max. rotate speed of spindle by G92
G96 S130 M03     //cutting speed maintains to be130m/min
```

Notice: G92 often be used with G96. G92 can limit max. rotate speed of spindle. I Following above example, spindle rotate speed of the workpiece with 10mm diameter is

$$N = \frac{1000 \times 130}{\pi \times 10} = 4140\text{rpm}$$

By G92, the max. rotate speed of spindle is limited to be no more than 2000rpmtherefore preventing accidental unclamping due to the excessively large centrifugal force and insufficient clamping force. G92 is sometimes working in conjunction with G96.

$$N = \frac{1000 \times 130}{\pi \times 10} = 4140\text{rpm}$$

1.40.2.2 Constant rotate speed

G97 S1300 M03 //spindle rotate speed maintains to be is 1300 rev/min

1.41Chamfer, Corner Round, Angle Command (,C ,R ,A)

In the mechanical drawing, we can input the angle of straight lines, chamfering, corner rounding, and other specification values directly by using the following functions. The system will insert the rounding and chamfering values in the straight lines under enough space. This program would be effective in automatic operation mode only.

1.41.1 Chamfer (C), Corner Round (R) function

In the continuous single blocks, straight and arc commands formed the corner of arbitrary angles user assigned. System could execute the cutting of chamfering and R rounding by adding", C_"or", R_", in the end of the first single block. Chamfering C and rounding R commands are available in both absolute value and incremental value command.

The feedrate of "**,C_**"及"**,R_**"can be specified by E_. When user leaves E_ unspecified, the feedrate of "**,C_**"及"**,R_**" in that block is set to be the same as the feedrate in next block.

1.41.2 Chamfering (,C_)

In the first single block of two continuous blocks (including no arc), specify ",C_" command could execute corner chamfering. In the case including arc, it will base on the length of arc.

1.41.3 Format

Standing for the length that supposed corner to the start or end of chamfering

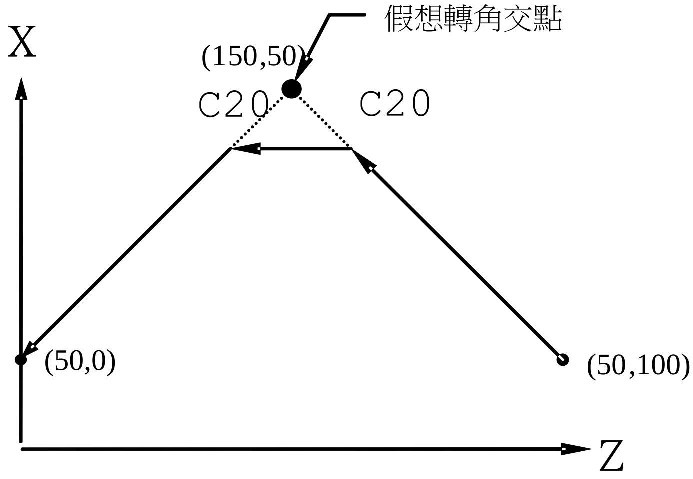
N100 G03X _ Z _ I _ K _ , C_ ;

Chamfering in the intersection of N100 and N200

N200 G01X _ Z _ ;

1.41.4 Example

(the chamfer of straight line and arc)



1. absolute command:

G28 X0.0 Z0.0 // Chamfering C20.0 between the
G00 X50.0 Z100.0 // movement of these two blocks
G01 X150.0 Z50.0 F100.0 , C20.0
G01 X50. Z0

2. incremental command:

G28 X0.0 Z0.0 // Chamfering C20.0 between the
G00 U50.0 W100.0 // movement of these two blocks
G01 U100.0 W-50.0 F100, C20.0
G01 U-100.0 W-50.0

1.41.5 Corner Round R(,R_)

In the first single block of two continuous blocks (including no arc), specify ",R_" command could execute corner corner round R function.

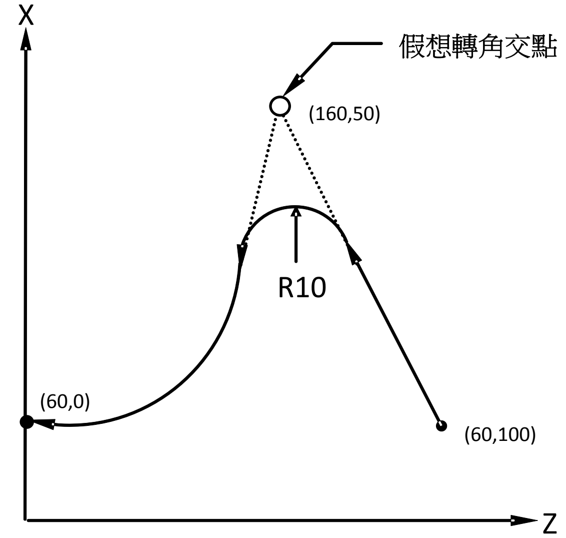
1.41.6 Format

,R_

R: for radius of corner and arc.

1.41.7 Example

(Corner between straight line and arc)



1. Absolute command

G28 X0.0 Z0.0 // Rounding R10.0 between the
G00 X60.0 Z100.0 // movement of these two blocks
G01 X160.0 Z50.0 F100 , R10.0
G02 X60.0 Z0.0 I0.0 K-50.0

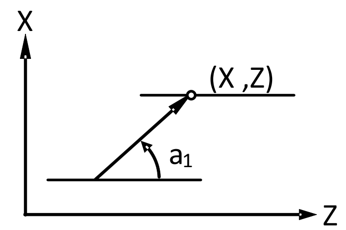
2. Incremental command

G28 X0.0 Z0.0 // Rounding R10.0 between the
G00 U60.0 Z100.0 // movement of these two blocks
G01 U100.0 W-50.0 F100, R10.0
G02 U-100.0 W-50.0 I0.0 K-50.0

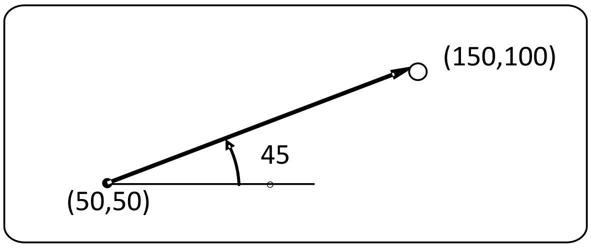
1.41.8 Angle Command (, A_):

1.41.8.1 Format

G01 Z_(X_) , A_ //specify the angle and the coordinate of X or Z.



1.41.8.2 Example



Program description：
N01 G00 X50.0 Z50.0；
//positioning to specified point
N02 G01 Z100.0, A45.0；
//the angle between tool path and horizontal axis is 45° end point absolute coordinate value of Z is 100
＊after executing program□coordinate value of X is 150

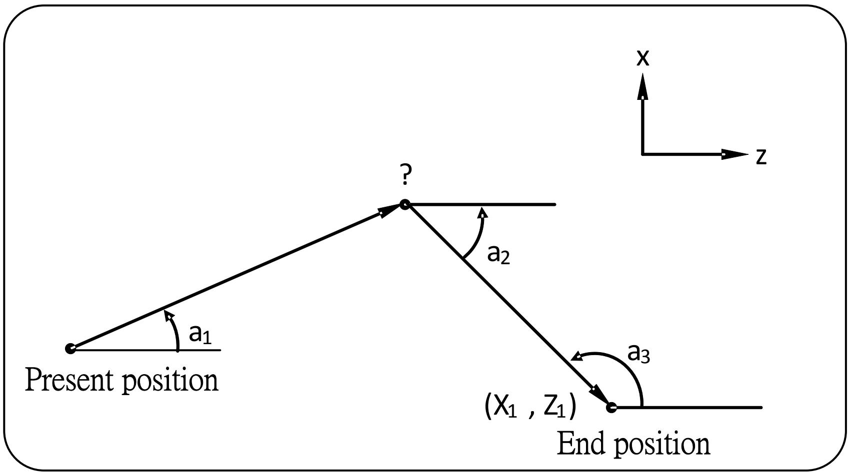
1.41.9 Geometric Function Command:

In continuous linear interpolation command, if it is hard to get the intersection point of two lines. We can use the sloping angle of the first line, absolute coordinate value of second line and the sloping angle of the first line to be the command, and then NC controller will compute the end of the first line. The continuous straight line corner function can be executed.

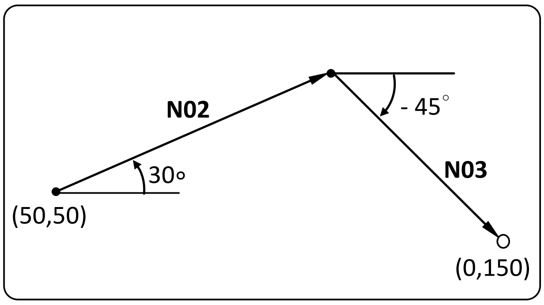
1.41.9.1 Format

G01 , A_F_ //specified angle

X_Z_, A_ //specified the end coordinate value and the angle of the next block



1.41.9.2 Example




```
Program description：
N01 G00 X50.0 Z50.0；
//positioning to specified point
N02 G01 , A30.0 F300；
//angle (30°) between the first path and
horizontal axis
N03 X0.0 Z150.0, A45.0；
// angle (-45°) between the first path and
horizontal axis，end point (0, 150)
＊after executing program□the node of path
(104.904 , 97.548)
```

1.41.9.3 Notice

1. This function is effective only under G01. It is not effective under other interpolation or positioning command.
2. From the + direction of horizontal axis in selected plane, the angle is positive for CCW , negative for CW.
3. The sloping angle can be specified in start point or end point of start side or end side. The sloping angle is specified in start side or end side is determined by NC automatically.
4. If we use the second way to specify, we need to specify the end point of the second block to be absolute coordinate.

1.41.10 Relative usage:

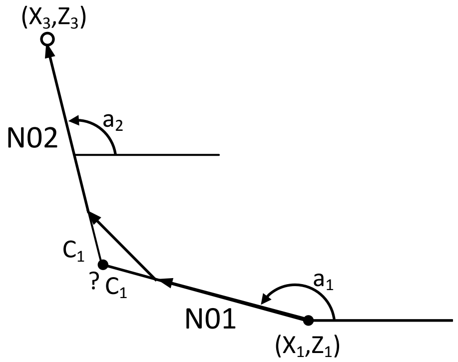
1.41.11 TYPE I

In the first angle command, we can specify Chamfer command or Angle Round command.

1.41.11.1 Format

N01 ,Aa_1 ,Cc_1 ;

N02 Xx_3 Zz_3, Aa_2 ;

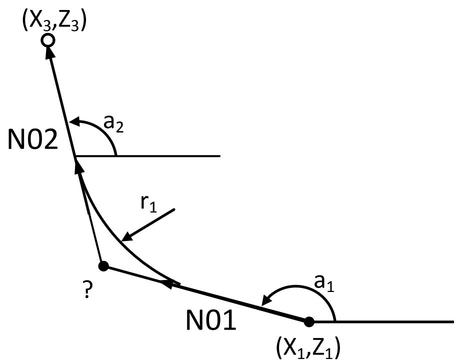


Description：Tool reaches to specified position(X3, Z3) according to the command，and there are specified angle『a1』、『a2』 between the twice movement path and horizontal axis，and there is a chamfer angle『C1』of the corner of two path。Contorller use specified value to computer the unknow intersection “?” of two path，and tool do cutting to specified position(X3, Z3) along the two path。

1.41.11.2 Command Format

N01 ,Aa_1 ,Rr_1 ;

N02 Xx_3 Zz_3 Aa_2 ;



Description：Tool reaches to specified position(X3, Z3) according to the command，and there are specified angle『a1』、『a2』 between the twice movement path and horizontal axis，and there is a round angle『r1』of the corner of two path。Contorller use specified value to computer the unknow intersection “?” of two path，and tool do cutting to specified position(X3, Z3) along the two path。

1.41.12 TYPE Ⅱ

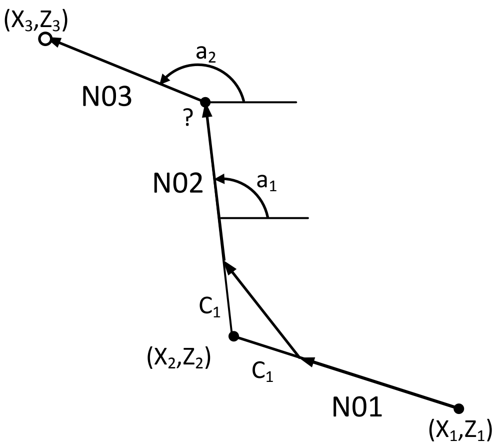
After Chamfering command ,angle round command (R), we can continue to do linear angle command.

1.41.12.1 Format

N01 Xx_2 Zz_2, Cc_1；

N02 ,Aa_1；

N03 Xx_3 Zz_3, Aa_2；



Description：Tool reaches to specified position (X2, Z2), (X3, Z3) according to the command，and there is a chamfering angle『C1』between the corner of the front two path，and there are specified angle『a1』、『a2』between the back two path and horizontal axis。Contorller use specified value to computer the unknow intersection ” ?” of two path，and tool cuts to end point (X3, Z3) along the three pathes

1.41.13 TYPE Ⅲ

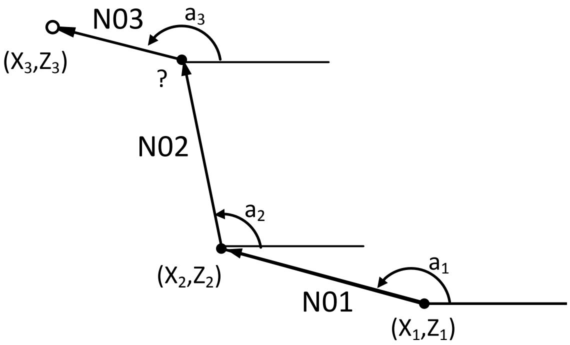
After linear angle command, we can continue to do linear angle command.

1.41.13.1 Format

N01 Xx_2 Aa_1；

N02 ,Aa_2；

N03 Xx_3 Zz_3, Aa_3；



Desctiption：Set the the X axis coordinate value “X2” of the first movement path according to the command，and the angle『a1』to horizontal axis，and the end point value(X3, Z3) of the third movement path，and the angle『a2』、『a3』between the front path，the angle between horizontal axis and the axis of the front path；Contorller use specified value to computer the unknow intersection ” ?” of two path，and tool cuts to end point (X3, Z3) along the three pathes。

1.41.14 Notice

1. Round angle value cannot be inserted in threading area.
2. By directly entering the continuous command in next area according to the drawing size, the end point of front area is already specified. Stop cannot be executed in single area, but dwell can be executed in the front area.
3. Allowance range of angle computing is ±1° .

(1) X_ , A_

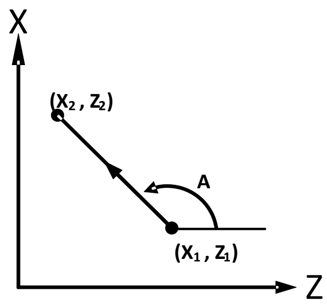
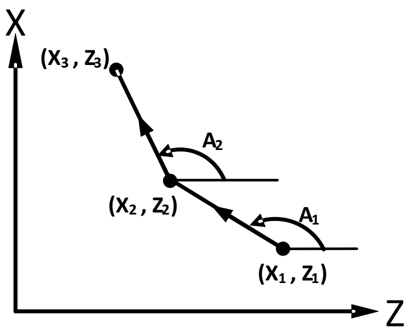
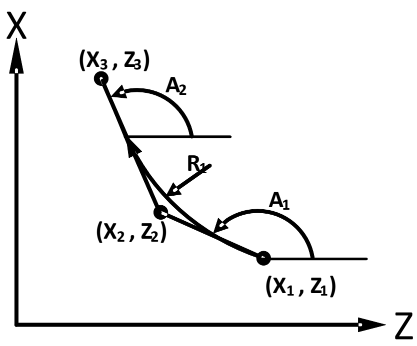
(when the angle is 0° ±1 , 180° ±1 , the alarm will be issued.)

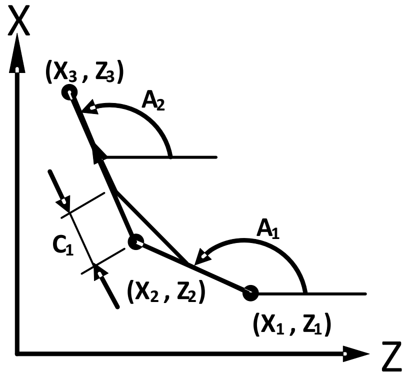
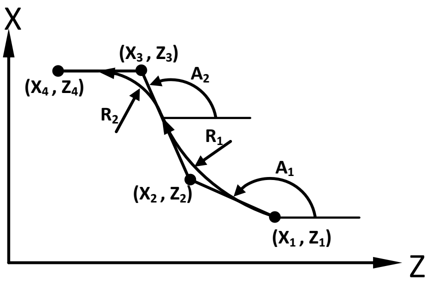
(2) Z_ , A_

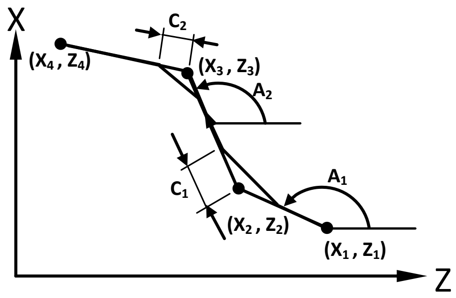
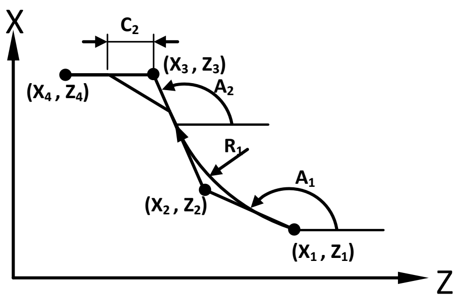
(when the angle is 90° ±1 , 270° ±1 , the alarm will be issued.)

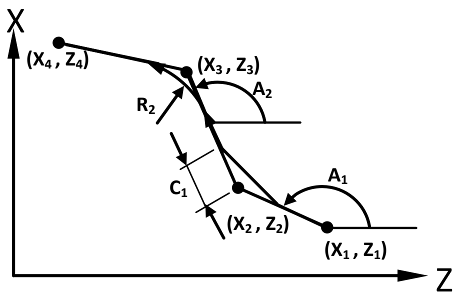
4. If the angle between two lines is in between of ±1° , it will be alarming when we computer the intersection.
5. If the angle between two lines is in between of ±1° , chamfer angle and round angle can be ignored.

1.41.15 Geometric Function Usage Table

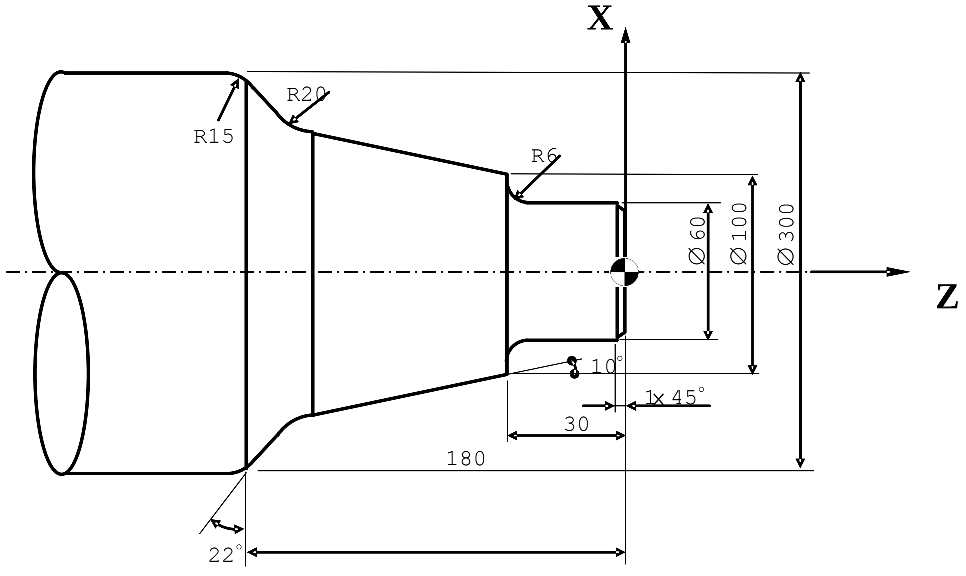
	Command	Movement	Description
1.	X_2_(Z_2)_, A_		According to any X_2(or Z_2) coordinate and the angle 『A』 which is between the path and horizontal axis. Controller computes the other unknown Z2(or X2), and tool can cut to specified position (X2, Z2) along this paths
2.	, A_1_ X_3_Z_3_, A_2_		According to the command setting and the specified angle 『A_1』 , 『A_2』which are between each path and horizontal axis, the controller computes the unknown intersection(X_2 , Z_2) to reach specified point (X_3, Z_3). Tool will cut to specified point(X_3, Z_3) along these two paths.
3.	X_2_Z_2_, R_1_ X_3_Z_3_ Or , A_1_, R_1_ X_3_Z_3_, A_2_		According to the setting command to reach specified point (X_3, Z_3), and the specified angle 『A_1』 ,『A_2』which are between each path and horizontal axis, and the corner is the round angle 『R_1』. Use controller to compute the unknown intersection(X_2 , Z_2), and tool will cut to specified point(X_3, Z_3) along the two paths

	Command	Movement	Description
4.	X_2_Z_2_, C_1_ X_3_Z_3_ Or , A_1_, C_1_ X_3_Z_3_, A_2_		According to the setting command to reach specified point (X_3, Z_3), and the specified angle 『A_1』 ,『A_2』 which are between each path and horizontal axis, and the corner is the chamfer angle 『R_1』. Use controller to compute the unknown intersection(X_2 , Z_2), and tool will cut to specified point(X_3, Z_3) along the two path
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_		According to the command to reach to the specified position (X_2, Z_2)→ (X_3, Z_3) → (X_4, Z_4), the corner of the front two path is a round angle『R_1』, the corner of the back two path is a round angle『R_2』, (or we do not specify (X_2, Z_2) but we add『A_1』 ,『A_2』). Controller will computer『A_1』,『A_2』or unknown intersection(X_2 , Z_2) by the specified value. Tool will cut to end point (X_4, Z_4) along these paths.

	Command	Movement	Description
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_		According to the command to reach to the specified position (X_2, Z_2)→ (X_3, Z_3) → (X_4, Z_4), the corner of the front two paths is a chamfer angle『C_1』, the corner of the back two paths is a chamfer angle『C_2』, (or we do not specify (X_2, Z_2) but we add『A_1』,『A_2』). Controller will computer『A_1』,『A_2』or unknown intersection(X_2 , Z_2) by the specified value. Tool will cut to end point (X_4, Z_4) along these paths.
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_		According to the command to reach to the specified position (X_2, Z_2)→ (X_3, Z_3) → (X_4, Z_4), the corner of the front two paths is a round angle『R_1』, the corner of the back two path is a chamfer angle『C_2』, (or we do not specify (X_2, Z_2) but we add『A_1』,『A_2』). Controller will computer『A_1』,『A_2』or unknown intersection(X_2 , Z_2) by the specified value. Tool will cut to end point (X_4, Z_4) along these paths.

	Command	Movement	Description
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_		According to the command to reach to the specified position (X_2, Z_2)→ (X_3, Z_3) → (X_4, Z_4), the corner of the front two paths is a chamfer angle『C_1』, the corner of the back two path is a round angle『R_2』, (or we do not specify (X_2, Z_2) but we add『A_1』,『A_2』). Controller will computer『A_1』,『A_2』or unknown intersection(X_2 , Z_2) by the specified value. Tool will cut to end point (X_4, Z_4) along these paths

1.41.16 Example



Program description: (input diameter by Metric system)

G01 X60.0 A90.0, C1.0 F80 //linear interpolation, the angle
//between the straight line and horizontal axis is "+90°", and
//chamfering C1.0 angle at the next block, feed rate
//80μm/rev
Z-30.0, A180.0 R6.0 // linear interpolation, the angle
//between the straight line and horizontal axis is "+180°",
//and rounding R6.0 angle at the next block
X100.0, A90.0 // linear interpolation, cutting to
//specified point, the angle between the straight line and
//horizontal axis is "+90°"
,A170.0 ,R20.0 // linear interpolation, the angle
//between the straight line and horizontal axis is "+170°",
//and rounding R20.0 angle at the next block, the end point is
//specified in the next block
X300.0 Z-180.0, A112.0, R15.0 // linear interpolation, the angle

//between the straight line and horizontal axis is "+112°",
//and rounding R15.0 angle at the next block
Z-230.0, A180.0 // linear interpolation, the angle
//between the straight line and horizontal axis is "+180°",
//cutting to specified position

1.42Tool Compensation Function (T Function)

Tool compensation function is mainly for selecting the using tool, also be called as T function, usually used in conjunction with tool exchange command (M06). Therefore tool switch can be automatically executed by tool numbers.

Two code form: the specification is for tool number ,tool length compensation and wear compensation selection.

Four code form: the first two codes are for tool number, the other two codes are for tool length and wear compensation.

When an user executes T__ command, the compensation value is first selected but the compensation action is not yet performed. When a block with movement action in it is performed, compensation action is executed.

1.42.1 Format

T ＊＊ (two code form)

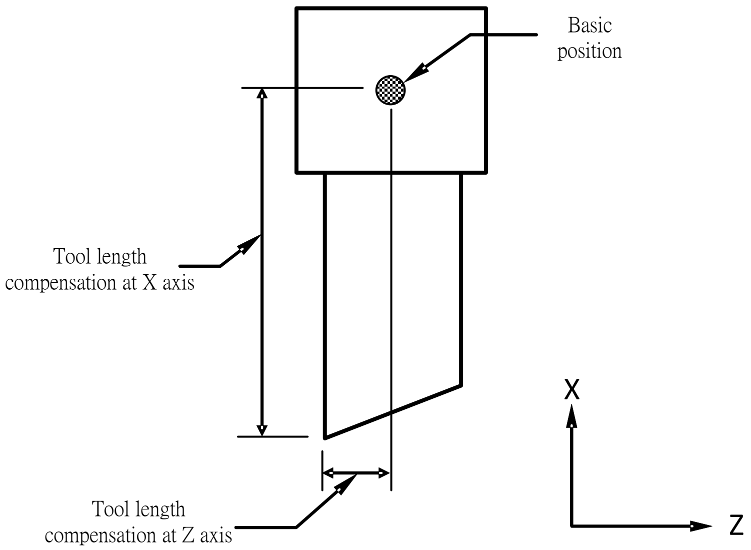
T ＊＊ ＊＊ (four code form)

1.42.2 Modal Of Tool Length Compensation

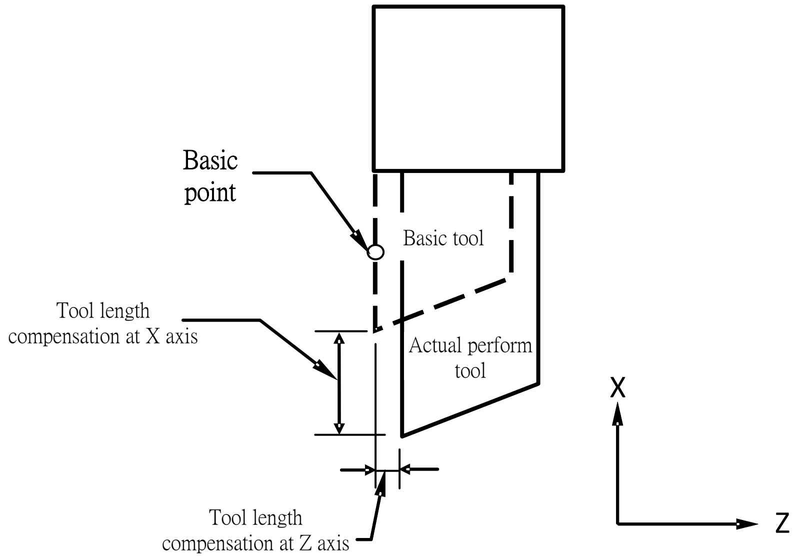
1.42.2.1 Tool length compensation

Execute tool length compensation at the basic position of program.

The basic position of program: center of tool seat or tool nose of basic tool:



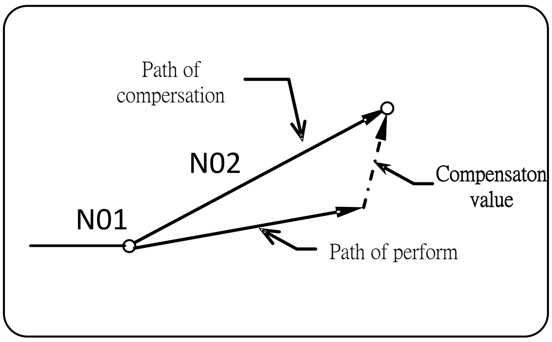
1.42.2.2 Tool nose of basic tool



1.42.3 Principle of Tool Length Compensation

1.42.3.1 Tool compensation starts

Tool compensation action starts after executing T command and executing movement command.

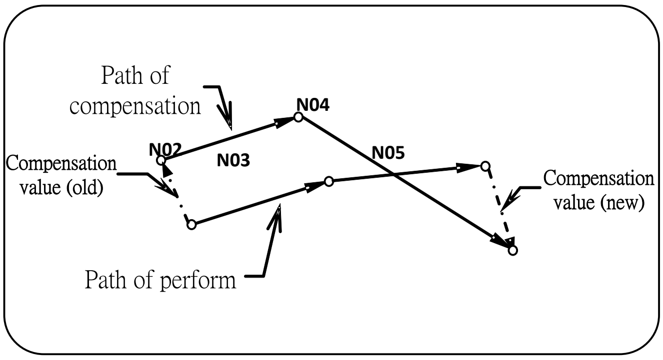


N01 T0101;

N02 X10.0 Z10.0;

1.42.3.2 Number change of tool length compensation

When number of tool changes, the corresponded tool compensation value of the new tool is added into the original offset.



N01 T0100;

N02 G01 X10.0 Z10.0 F0.2;

N03 G01 X13.0 Z15.0 F0.3;

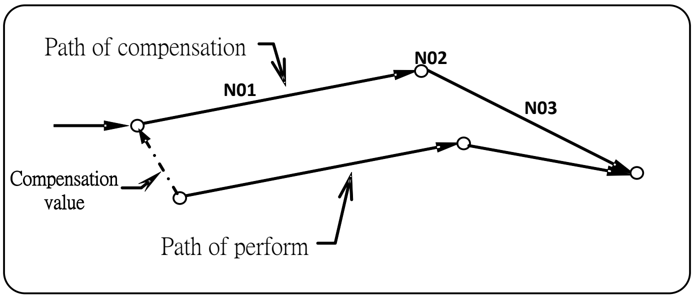
N04 T0200;

N05 G01 X13.0 Z20.0 F0.205;

1.42.3.3 Tool length compensation cancel

Number of compensation is 0.

When number of compensation is “0” in T command, compensation cancels.



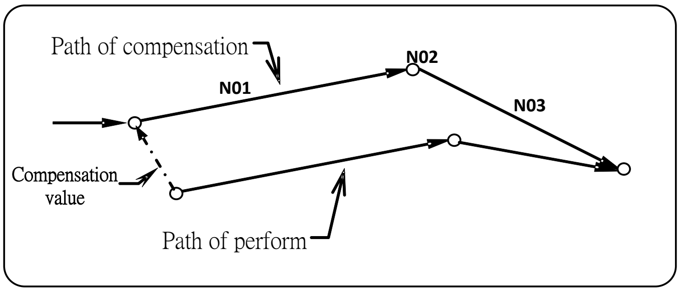
N01 X10.0 Z10.0 F0.1 ;

N02 T0000 ;

N03 G01 X10.0 Z20.0 ;

Compensation value of command is “0”

When compensation value of tool length compensation number is “0” in T function, the compensation cancels.



N01 G01 X10.0 Z10.0 F0.1；

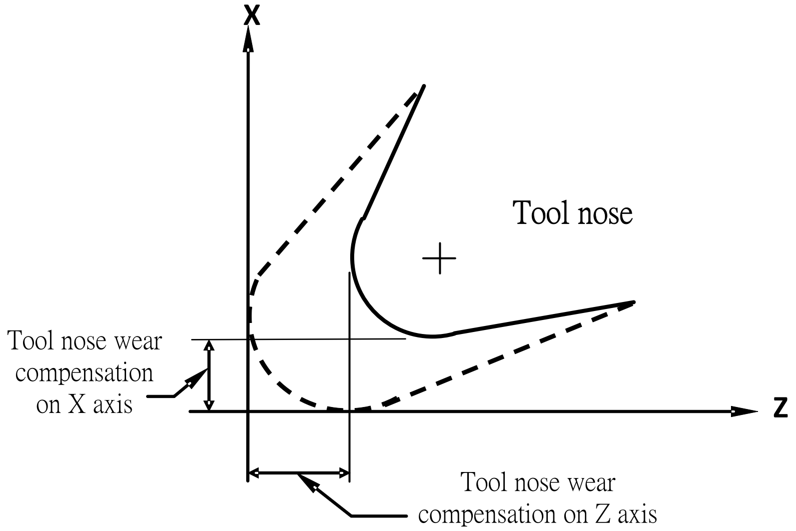
N02 T0100；

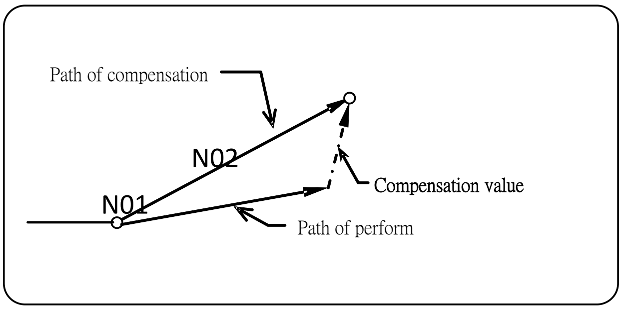
N03 G01 X10.0 Z20.0；

1.42.4 Tool Nose Wear Compensation

1.42.4.1 Tool nose wear compensation value setting

System can performs compensate function when tool nose wears. The compensation value will be added into geometric compensation. Geometric compensation = tool length compensation + wear compensation. When we specify the number of compensation, geometric compensation will be executed.






```
N01 T0102；
//start tool No.1 compensation, the numbe
of compensation is 2
N02 X10.0 Z10.0;
```

1.43 Spindle Rotate Speed Function : S code command

S function is spindle speed command, specifying constant revolution per minute or constant surface speed per minute of spindle by G96/G97.

1.43.1 Format

S

1.43.2 Example

G96 S150 M03 //constant surface speed of spindle, 150 m/min.
G97 S500 M03 //spindle keeps 500 rev/min

1.43.3 Notes

Consider the situation when the tool spindle of processing is shifted among the spindle group. For example, if the current processing spindle is the second spindle and the first spindle is to be selected at speed of 150RPM clockwise, "M03 S1=150" should be specified in order to avoid the situation that the speed is specified to the second spindle due to insufficient time for spindle shifting.

1.44Feed Function: F code command

In cutting mode, the specified movement speed of tool in the program is called feedrate. The axis feed mode to be used is selected by designating the feed function G code (G94 or G95).G94 is the designation of feed per minute(mm/min) mode, while G95 is feed per revolution(mm/rev) mode. For example, command F300 in G94 mode represents for 300 mm/min and command F0.5 in G95 mode represents for 0.5 mm/rev.

1.44.1 Format

F

1.44.2 Example

```
G94 G01 X100.0 Y100.0 F300     //linear interpolation, feed rate 300
                               //mm/min
G95 G01 X100.0 Y100.0 F0.5     //linear interpolation, feed rate 0.5
                               //mm/rev
```

1.45Programmable Mirror Image (G68)

With double turrets in lathe, we can mirror the X coordinate with X0 by G code. It is more convenient with double turrets because it is not necessary to consider the moving direction of the turret while programming.

1. The reversal or compensation direction of the circular interpolation, tool nose radius compensation and the coordinate reversal is opposite.
2. Because this instruction is used in local coordinate, the center of the mirror still moves when the counter is reset or the working coordinate is changed.
3. When execute the instructions (G28, G30) within the G68, programmable mirror image is effective between the paths through the start point to the middle point, ineffective between path through the middle point to the origin.
4. When execute the return from reference point instruction (G29) within G68, programmable mirror image is effective between the paths through the start point to the middle point.

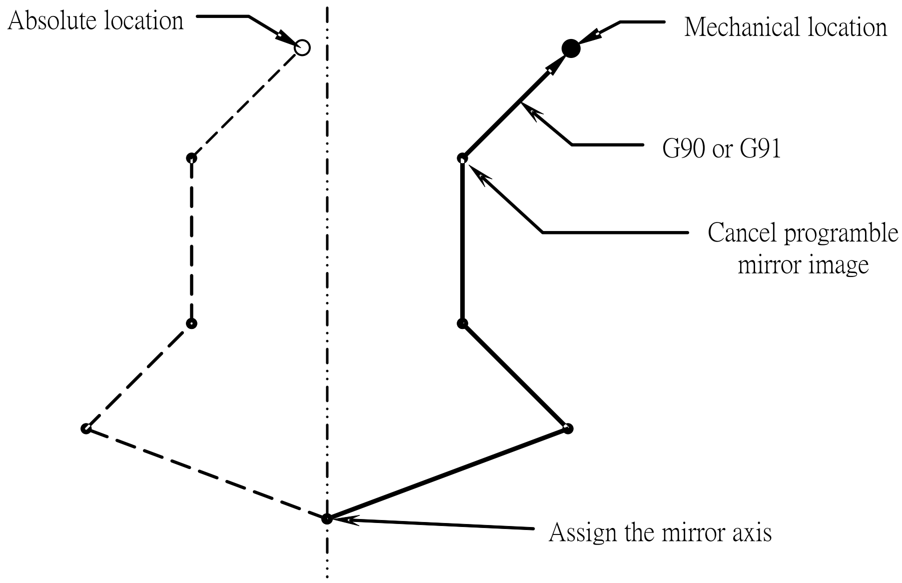
1.45.1 Format

G68 Start X axis programmable mirror image

G69 Cancel programmable mirror image

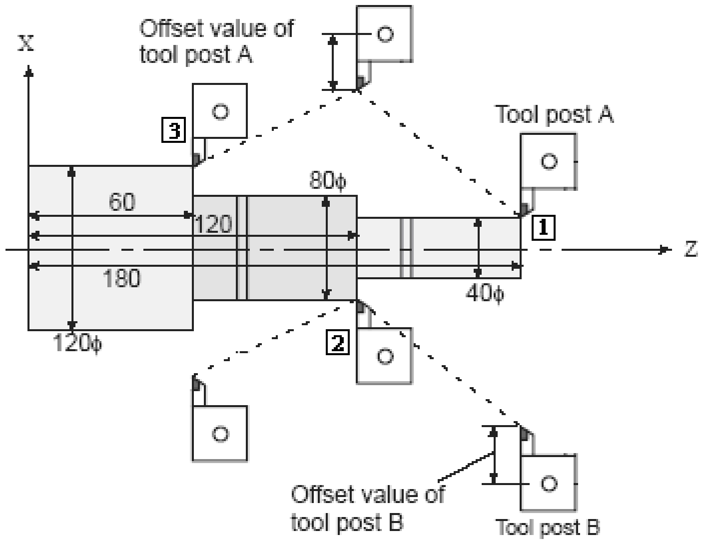
1.45.2 Attention

If cancel programmable mirror image when the tool is out of the center of the mirror , the absolute location cannot match the mechanical location . As figure shown below, this situation lasts until the instruction is set to specify the absolute location [positioning of G90] or machine zero point return G28 and G30 are set. If we re-assign location of the center mirror under motionless mode in absolute location, the specified location will be unexpected. So we should cancel programmable mirror image at the center of the mirror or use G90 after cancelling programmable mirror image.



The center of the mirror axis (Z axis)

1.45.3 Example



Program illustration:

```
T0101            //turret 1
G01 Z180. X40. //position-1
Z120.
T0202            //turret 2
G68                  //enable X-axis mirror image
G01 Z120. X80. //position 2
Z60.
T0101            //turret 1
G69                  //disable X-axis mirror image
G01 Z60. X120. //position 3
M99
```

1.46Decimal Point Input

The input parameter with decimal point is interpreted as prevailing unit (mm, inch, sec...etc). On the other hand, the input parameter without decimal point is interpreted as least input unit (μm, ms…etc) which has been set in a system.

1.46.1 Example

Decimal point: ○○.○○

Whole number: ○○○○

1.47 Spindle Synchronization

When two or more spindles of tool are available, user can make some special application by cooperating with the spindles. For example, in catching workpiece, two spindles have to have the same rotate speed and the phase angles need to be the same or fixed. Two spindles have to be synchronization. This is Spindle Synchronization.

1.47.1 Action description

1.47.1.1 Spindle synchronization position adjust

Spindle has to change the sleeve when cutting different workpieces(cylinder, hexagonal, octagonal pillar…etc.). When the sleeve on the spindle is set up, it can hardly maintain a fixed angle each time. The difference of angle phase has to be known before spindle synchronization, thus calibration of the origin is necessary before synchronization starts. The steps of calibration of the origin are as followed:

1. Set the home offset to be 0 (Pr881~Pr896).
2. Clamps the workpiece by basic spindle and use micrometer gauge to adjust the datum on the workpiece. Then set the location of basic spindle to the home offset. .
3. Calibrate synchronous spindle by the same way in step 2 to adjust datum.
4. Home positions of two spindles are the workpiece synchronization position.

1.47.1.2 Format

Enable spindle synchronization

G114.1 R_

R phase difference (When R isn't set, R stands for the synchronization speed. It is usually used on the cylinder.)

Disable spindle synchronization

G113

1.47.1.3 Synchronization success signal

1. "S62 Bit On" indicates the synchronization success and the basic spindle and synchronous spindle both achieve the same speed/phase.
2. After the synchronization success, if the synchronization error is too high (over 0.5 degrees), system will turn S62 Bit off.

1.47.2 NOTE

1. Two synchronization spindles have to be servo motor. The spindle type only support Type3 (Pr1791~1796). If user set the wrong type, the alarm (Cor093) will be issued.
2. The motion parameters of two servo motors have to be set the same. EX: acceleration time (Pr1831~1836), spindle motor speed up to 1000RPM/Sec acceleration time (Pr1851~1856).
3. If the position loop gain (Kp, Pr 181~196) of two servo motor are not the same, user has to check that Kp of controller and driver are the same. Otherwise, the motion system performed will be unexpected.G114.1 is a model G-Code. Only when the signal of spindle synchronization is on and both of spindles have positive (negative) rotate command (M03, M04), spindle synchronization will be start and output the signal of spindle synchronization success.
4. To start synchronization transfer worlpiece from static condition, set the minimum spindle rotate speed to be 0.
5. After spindle synchronization, synchronous spindle doesn’t act on M03, M04, M05 and S code but only record until synchronization is disabled.
6. After spindle synchronization, rotation command will be sent to basic spindle and synchronous spindle. Synchronous spindle direction of rotation will follow basic spindle and Pr1861~1866 (Spindle Sync. basic spindle direction). M03 or M04 can’t control the direction.
7. When user pushes emergency stop, both spindle rotation and spindle synchronization stop.
8. If signal of spindle synchronization is on but Pr4021, 4022 is not exist, the alarm (Cor91, Cor92) will be issued.
9. After spindle synchronization success, user can’t orientate spindle.
10. When signal of spindle synchronization success (S62) is on, by pressing down “Reset” button, system will disable G114.1 synchronization state only after two spindles have stopped.
11. When reading feedback from the encoder, a 8-usec delay exists between the port and the next one. The more ports in between the longer delay time is. Spindle synchronization has to take care about phase. If using spindle synchronization to catch workpiece, user has to put two spindles on the port that is next to each other. EX: P1 and P2 are on the same servo card to decrease the phase error.

12. When signal of spindle synchronization success (S62) is on and G113 is specified to disable spindle synchronization, system will practically disable synchronization until achieve the speed program specifies.

1.47.3 Example

Take the first spindle to be basic spindle and the second spindle to be synchronous as example. M103 and M104 are command for “spindle rotate in positive direction”. M105 and M205 are command for “spindle stop”. M81 is command for “wait until synchronization success (S62)”. The above M-Code commands are all needed to be specified in PLC.

Dual program example

$1	$2

S1 = 150
M103 // spindle 1 CW on.
G04 X0.4 // wait spindle speed goal.
G114.1 R0. // enable spindle syncronization.
Mxx // wait spindle syncrhonization.
S1 = 200 // change speed.
G04 X0.4
M105 // stop spindle
G113 // diable spindle syncronization.
G04.1 P1 // wait sync. $2
M30 // end.
	S2 = 100
	M203 // spindle 2 CW on.
	G04.1 P1 // wait sync. $1
	M99 // end.

1.47.4 Single Program example

```
G114.1 R0.  // enable spindle syncronization.
S1 = 150
M103        // spindle 1 CW on.
S2 = 100
M203        // spindle 2 CW on.
M81         // wait spindle syncrhonization.
M105        // stop spindle 1.
G113        // diable spindle syncronization.
G04 X1.
M205        // stop spindle2
M30         // end.
```

1.47.5 Reference

Device Type	Device	Description
R	R761~R776	Corresponding machine coordinate. Unit is 0.001 degree.
S	S62	signal of spindle synchronization success
Registry	L10030	Synchronization base difference: base angle difference of spindle synchronization

		Shift Angle = (sign)$\theta_2 - \theta_1$
Display		Synchronization angle difference: angle difference of spindle synchronization (MMI show it) Difference Angle = (sign)$\theta_2 - \theta_1 - \Delta$)
Paramter	181~196	Position loop gain(Kp)(1/sec) of servo
	881~896	Home offset
	1791~1796	Spindle type
	1831~1836	spindle motor acceleration time(ms)
	1851~1856	spindle motor speed up to 1000RPM/Sec acceleration time(ms)
	1861~1866	spindle direction, 0: CW, 1: CCW
	4021	Basic spindle number(1~6)
	4022	Sync spindle number (1~6)
Alarm	Cor091	Invalid number of basic spindle
	Cor092	Invalid number of synchronous spindle
	Cor093	Invalid type of sync. spindle

2 M Code Command Description

Auxiliary function is used to control the On and OFF of the machine function. The miscellaneous function (M code) is specified by a two-digit number following address M.

The specific number and applications of M codes are described below:

M Function Table

M code	Function
M00	Dwell
M01	Optional dwell
M02	End of program
M03	Spindle rotates (CW)
M04	Spindle rotates (CCW)
M05	Spindle stops
M06	Tool exchange
M08	Cutting liquid ON
M09	Cutting liquid OFF
M10	Tight the clamp
M11	Loose the clamp
M19	Spindle location, let spindle stops at a specified position
M30	Program ends, return to start point
M98	Calling of subprogram
M99	End of subprogram

2.1 Dwell (M00)

When M00 is executed by CNC, the spindle stops, the feed dwells, and the cutting liquid is off. The dwell enables an operator to inspect workpiece/tool dimensions, calibrate and make compensation of the workpiec. The “M00 signal button” on the panel is used to control whether a program should be dwelled or not.

2.2 Optional dwell (M01)

The function of M01 is similar to M00. M01 is valid only when "optional stop button" turns ON, and the program is therefore dwelled. On the contrary, M01 is invalid while the button turns OFF.

2.3 End of program (M02)

M02 should be specified at the end of a program (if required). When M02 is executed during operation, system If there is M02 command at the end of program and CNC execute to this command, the machine will stop all action at the same time. If an user wants to restart the program, it is effective only by pressing the "RESET" and the "PROGRAM START" button in sequence.

2.4 Spindle rotates CW (M03)

M03 command spindle to rotate CW. When M03 is used in conjunction with S function, spindle is specified to rotate CW in a given speed. .

2.5 Spindle rotates CCW (M04)

M04 command spindle to rotate CCW.

2.6 Spindle stops (M05)

M05 command spindle to stop. When shifting gears or changing the direction of a rotating axis is required, M05 is specified to stop the spindle first.

2.7 Tool exchange (M06)

M06 is specified to execute tool exchange. Note that M06 include no tool choosing, thus it must be used in conjunction with T_function.

2.8 Cutting liquid ON/OFF (M08/M09)

M08 is specified to turn cutting liquid ON; M09 is specified to turn cutting liquid OFF

2.9 Spindle locates and stops (M19)

This command can locate the spindle at specified corner

2.10Program ends (M30)

M30 command is specified at the end of the program. When program executes M30, all actions will stop, and the memory will be reset and return to the beginning state of the program.

2.11 Subprogram Control (M98/M99)

1. Subprogram is the parameter that includes fixed cutting procedures or repeatedly used parameters. We should prepare it in advance and put it into the memory. We call from the main program when we need to use. Calling subprogram is executed by M98, and it will stop by executing M99.

2. When running M02 and M30 in the subprogram, system regards it as the end of the subprograms and returns to the main program.

2.11.1 Format

(1). M98 P_H_L_ Calling of subprogram

P: the number of subprogram called (when P is unspecified, system specifys the program itself, and it is valid only in memory running or MDI mode)

H: the starting executing sequence number in subprogram called (when H is unspecified, system will execute from the forefront)

L: count of subprogram repeated times.

(2). M99 P_L_ subprogram ends

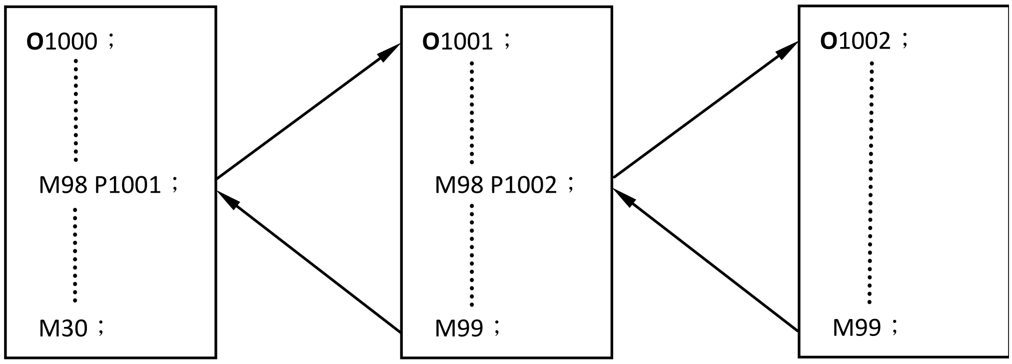
P: the sequence number of caller program for returning back after subprogram ends.

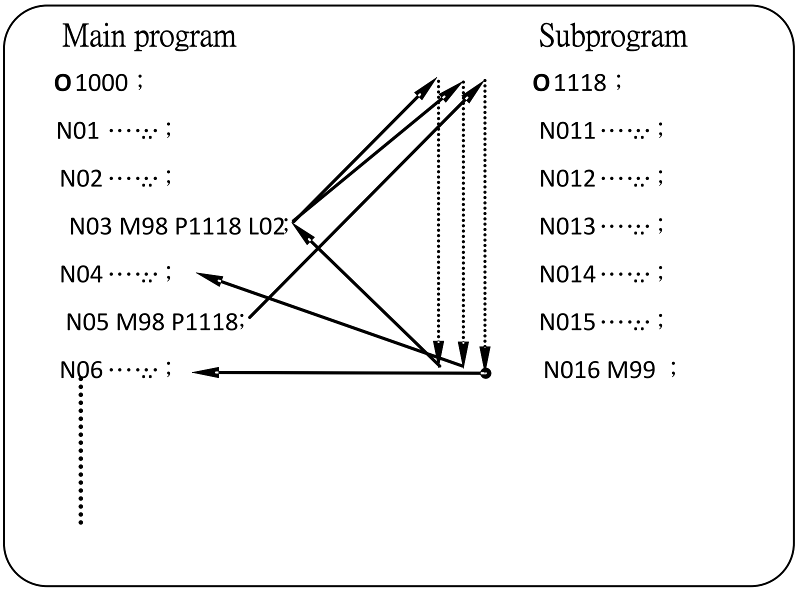
2.12 Making and Executing of Subprogram

(1) The normal format as below

```
O xxxx ;  ------------------------------------  Number of Subprogram
   G01 ·········;
   G02 ·········;        Content of program
   G01 ·········;
     :
M99 ;  ------------------------------------------  Subprogram ends
```

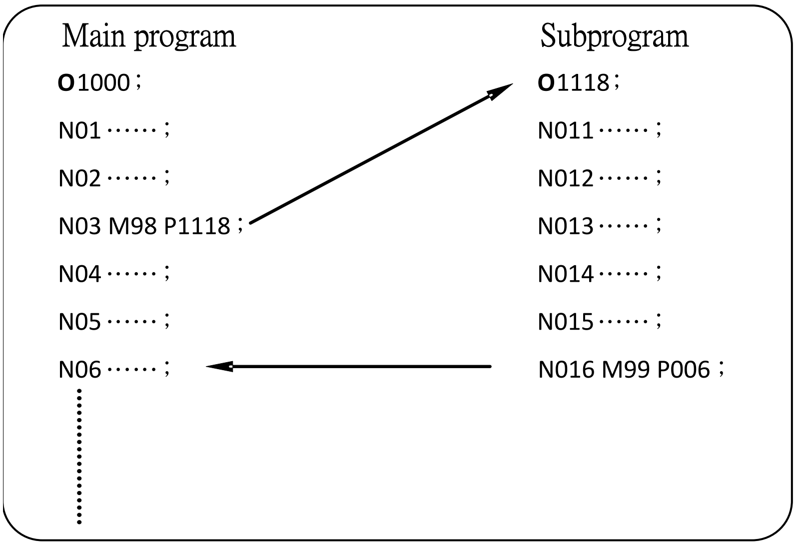
(2) Main program use with calling of subprogram, and sequence of executing



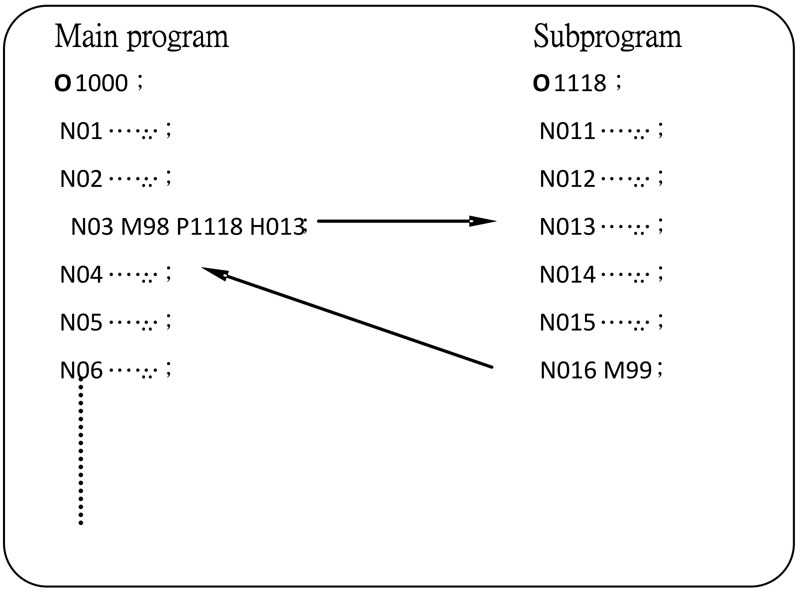


2.12.1 Special usage of subprogram:

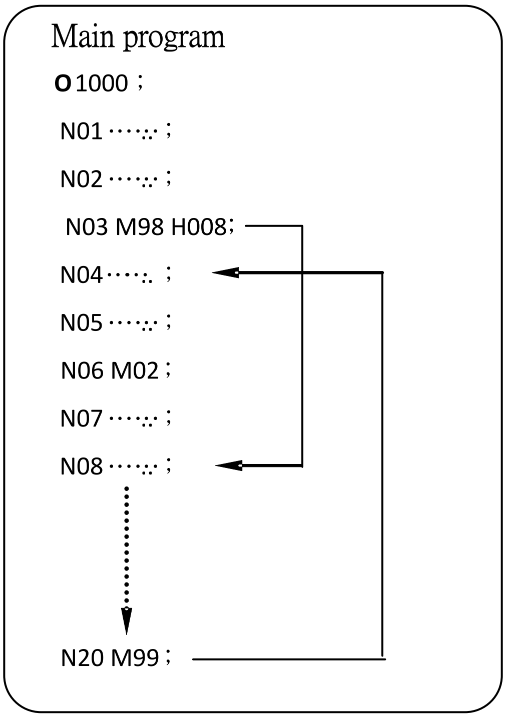
(3) We can execute subprogram by adding **P__** function after M99 in the end of the final block. After finishing this program, it will return to main program, and execute the block which the sequence number specified by P_ function is in.



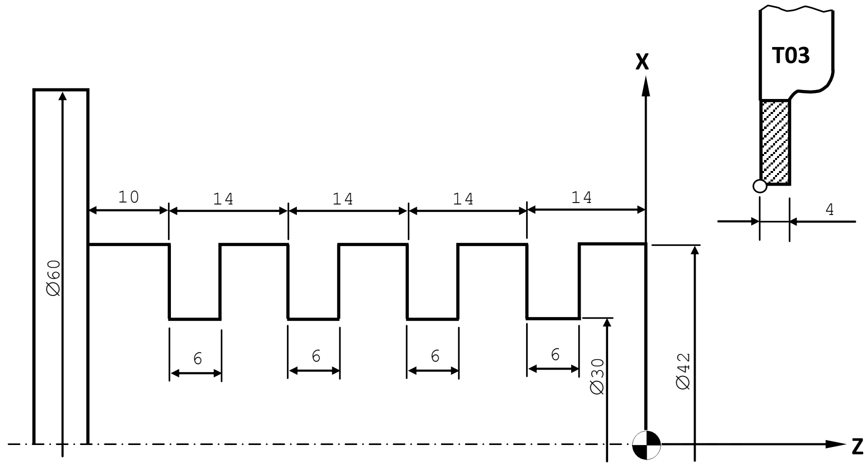
(4) Subprogram also can execute P_ command with H_ command in M98. The system will execute the subprogram (specified by P_) from the sequence number specified by H_. . The subprogram is therefore versatile. With open only one subprogram to execute muti-purpose function, the system can save more memory space.



(5) If user leaves P_ command unspecified and only specify H_ command in M98, the system will execute from the sequence number of main program that specified by H_ command. After executing M99, it will return to the next block of M98 and continue to execute the program.



2.12.2 Example cutting a tank, use "calling of subprogram" to execute repeating machining



(1). First way: P command in block of M98

＊Main program.

T03 //use tool NO.3
G97 S710 M03 //constant rotate speed of spindle, 710 rpm CW
M08 //cutting liquid ON
G00 X45.0 Z-12.0 //positioning to the above of first tank
M98 P1234 H102 L4 //call the subprogram of sequence number
//"O1234", machining from the block of N102
//and repeating 4 times
G28 X80.0 Z80.0 //positioning to specified mid-point and return to
//machine zero point
M09 //cutting liquid OFF
M05 //spindle stops
M30

＊Subprogram.

O1234
G00 X45.0 Z-12.0
G01 X30.0 F200 ←**Start from this block**
//linear interpolation to the bottom of the tank, feedrate
//200µm/rev
G00 X45.0 //escaping to start position

```
    W-2.0     //move 2mm toward negative direction of Z
G01 X30.0     //linear interpolation to the bottom of the tank
G00 X45.0     // escaping to start position
    W-12.0        // move 12mm toward negative direction of Z, and wait
              //for cutting next tank
M99           //return to main program
```

(2). Second way: without executing P_ command in block of M98

＊Main program.

```
T03               //use tool NO.3
G97 S710 M03      //constant rotate speed of spindle, 710 rpm CW
M08               //cutting liquid ON
G00 X45.0 Z-12.0  //positioning above the first tank
M98 H0010 L4      //execute from the block of main program sequence
                  //number N0010, and repeat 4 times
G28 X80.0 Z80.0   //positioning to specified mid-point and return to
                  //machine zero point
M09               //cutting liquid OFF
M05               //spindle stops
M30               //program ends

N0010
G01 X30.0 F200    ←start with this block after executing M98
                  //linear interpolation to the bottom of the tank,
                  feedrate 200μm/rev
G00 X45.0         //escaping to start point
W-2.0             // move 2mm toward negative direction of Z
G01 X30.0         //linear interpolation to the bottom of the tank
G00 X45.0         //escaping to start point
W-12.0            // move 12mm toward negative direction of Z, and
                  //wait for cutting next tank
M99               //return the next block N006 of M98
```

3 Postscript

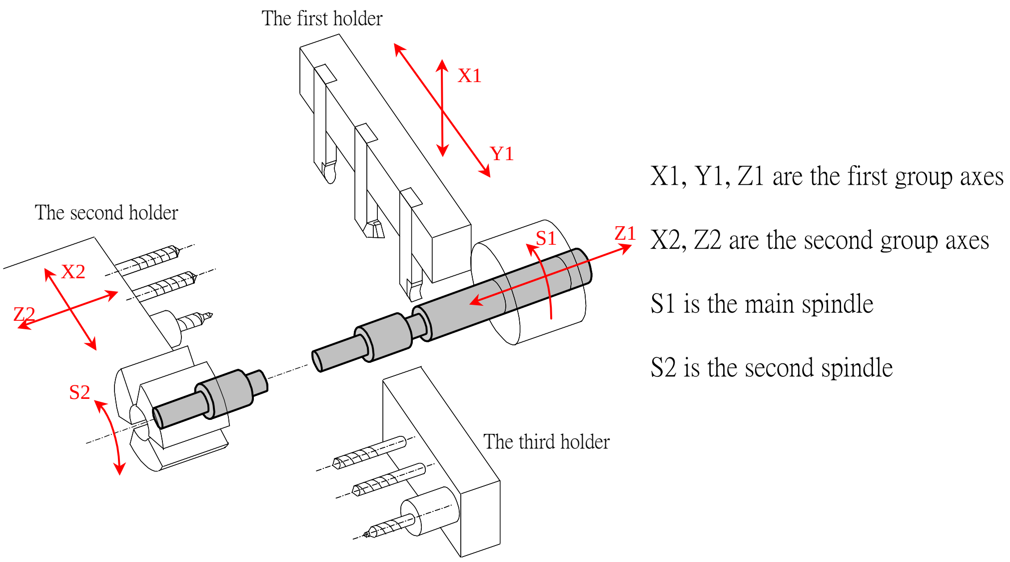
3.1 Description of lathe parameter

NO	Explain	Input range	Unit	Description
4001	Drilling mode	[0, 1]		0: high speed 1: normal
4002	Escaping amount of drilling cycle	[0, 999999999]	LIU	LIU is min. input unit, and it will be affected by Metric or Imperial system in use.
4011	Escaping amount of peck drilling cycle	[0, 999999999]	LIU	LIU is min. input unit, and it will be affected by Metric or Imperial system in use.
4012	Escaping amount of cutting cycle	[0, 999999999]	LIU	LIU is min. input unit, and it will be affected by Metric or Imperial system in use.
4013	Cutting value of cutting cycle	[0, 999999999]	LIU	LIU is min. input unit, and it will be affected by Metric or Imperial system in use.
4015	Cutting value of pattern repeating in X direction	[0, 999999999]	LIU	LIU is min. input unit, and it will be affected by Metric or Imperial system in use.
4016	Cutting value of pattern repeating in Z direction	[0, 999999999]	LIU	LIU is min. input unit, and it will be affected by Metric or Imperial system in use.
4017	Number of repeats of pattern repeating	[1, 999]	Number of times	
4018	Chamfer angle of thread cutting G21	[0, 89]	degree	
4041	Finishing allowance of threading	[0, 999999999]	LIU	LIU is min. input unit, and it will be affected by Metric or Imperial system in use.

NO	Explain	Input range	Unit	Description
4042	Thread angle of threading	{0, 29, 30, 55, 60, 80}	Degree	
4043	Chamfering value of threading	[0, 99]	0.1 pitch	
4044	Times of finishing allowance in threading	[0, 99]	Number of times	
4045	Min. cutting value in threading	[0, 999999999]	LIU	LIU is min. input unit, and it will be affected by Metric or Imperial system in use.
4050	C axis motor is used on spindle or not	[0, 1]		This function is used with Marco command M19 C_, when we use this function, we need entry M18 /M50 /M51 to system parameter 360X M code Marco registry table
4051	Multiple cutting cycle, increasing (decreasing) allowed error range (um)	[0,999999999]		LIU is min. input unit, and it will be affected by Metric or Imperial system in use.

3.2 Description of lathe double program

To save the time of the processing, the SYNTEC lathe's controllers can drive two programs simultaneously. The two program can drive two pairs of turret to execute linear interpolation and circular interpolation at the same time. The system therefore achieves the most effective lathe status while processing workpieces in external diameter and internal diameter simultaneously.



3.2.1 The description of the related instructions with double program:

$1→the contents after the instruction in the program is the first group

$2→the contents after the instruction in the program is the second group

The second group in the program must end with M99.

G04.1 P_→synchronous instruction, G04.1 P1 in the first group and one in the second group would wait for each other until synchronization succeeds and go to next section.

G04.1 P2 waits for each other until synchronization succeeds and go to next section in the same way.

3.2.2 The related M code:

M_code	The specification
M03	The first main axis rotates in positive direction
M04	The first main axis rotates in negative direction
M05	The first main axis stops

M63	The second main axis rotates in positive direction
M64	The second main axis rotates in negative direction
M65	The second main axis stops
M70	Assign the first main axis to be the main axis of first group.
M71	Assign the second main axis to be the main axis of second group.

3.2.3 Matters needing attention when compiling program

1. The first group of the program must start with $1 and the second one must start with $2.
2. The quantities of G04.1 P_ in the first group must be the same as that in second group and the number after P has to be sequentially assigned in increasing order.
3. Put end command M30 or M02 in the first group when program ends and M99 must be specified in the last block of the second group.

4. When repeatedly processing several workpieces automatically is required, specify M99 in the end of the first group program. But notice that in order to enable the synchronization of first and second groups, the same G04.1 P_ code must be specified before M99 of the first and second group.

```
$1
----------
----------
G04.1 P20
M99
                    The Same
$2
-----------
-----------
G04.1 P20
M99
```

5. With the axis set belonged to the second group, G code can only be specified in the second group. With the axis set belonged to the first group, if we specify G code in the second group, commands are ineffective.

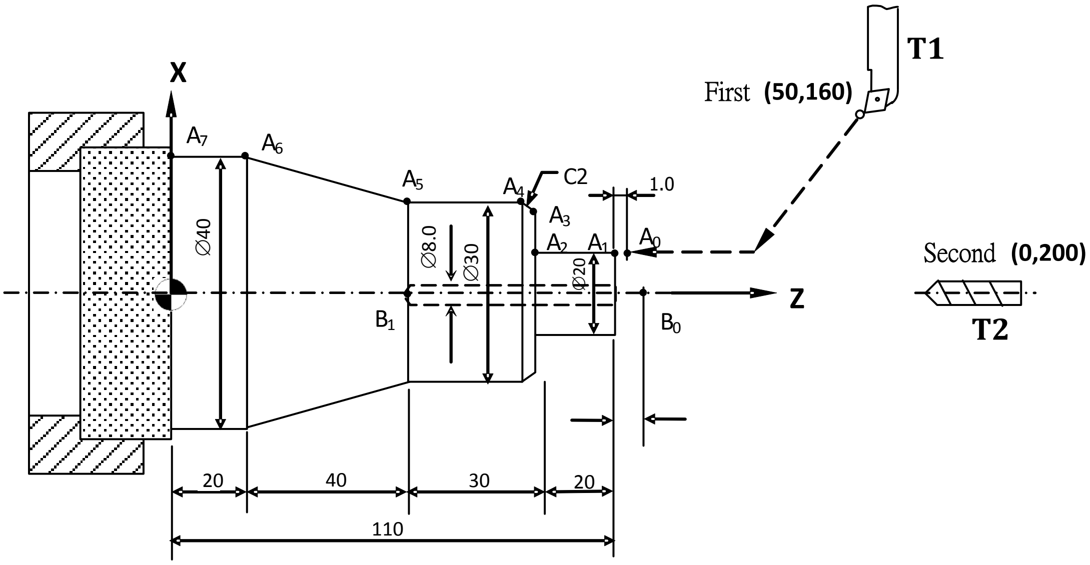
6. M code ,S code and T code are all available in the first and second group. Therefore M code ,S code and T code can be properly executed simultaneously in the first and second group.

3.2.4 Compiling programs:

Start a new file and program the processing file according to the following example.

```
$1
G00 X50.        //Move X axis in the first group
G04.1 P1
Z100.           //Move Z axis in the first group
------------
------------
G04 P30
M30
                                        Simultaneous
$2
G00 X250        //Move X axis in the secon group
Z500.           //Move Z axis in the second group
G04.1 P1
------------                        Simultaneous
------------
------------
------------
G04.1 P30
M99
```

3.2.5 Examples for processing program:




```
$1                              //the first group
G92 X50.0 Z160.0 S10000         //set origin, the highest speed 10000 rpm
T01                             //use knife No.1
```

```
G96 S130 M03          //face speed130m/min, main axis rotates
                      //in positive direction
M08                   //turn on cutting liquid
G04.1 P1
G00 X20.0 Z111.0      //positioning to A0 rapidly
G01 Z90.0 F0.6        //linear cutting A0→A2
    X26.0             //A2→A3
    X30.0 Z88.0       //A3→A4
    Z60.0             //A4→A5
G04.1 P2
    X40.0 Z20.0       //A5→A6
    Z0.0              //A6→A7
G00 X50.0             //back knife rapidly
    Z160.0            //return to origin
G04.1 P3
M05 M09               //stop the main axis, turn off cutting liquid
G04.1 P4
M30                   //end program

$2                    //the second group
G04.1 P1
T02                   // use  knife No.2
G04.1 P2
G00 X0 Z120.          //position to B0 rapidly
G01 Z60. F0.5         //move knife B0→B1
G00 Z120.             //back knife B1→B0
G04.1 P3
G00 Z200.             //back the knife
G04.1 P4
M99
```

3.3 Description of Lathe graph assist G code

Lathe graph assist G code is the special G code specified by inserted cycle in program editing. For example, two lines have to be specified when using G73 command manually. Only a line is to be specified in the special G code which inserted cycle automatically generates. Thus system combines two lines in G73 into special G code G73.1. The following is the instructions of special G code.(Special G code conversational input mode is only available in DOS version)

3.3.1 Assist G code list

- G73.1 Stock Removal in Turning
- G74.1 Stock Removal in Facing
- G75.1 Pattern Repeating
- G76.1 End Face (Z axis) Peck Drilling Cycle
- G77.1 Outer Diameter/Internal Diameter Drilling Cycle
- G78.1 Multiple Thread Cutting Cycle

3.3.2 G73.1 Stock Removal in Turning

G73.1 D(Δd) X(e) P (ns) Q (nf) U(Δu) W(Δw) F___ S___ T___

Δd: depth of cut in X axis direction, default can be specified by the system parameter#4013.

e: escaping amount, specified by the parameter#4012

ns: sequence number of the first block for the program of finishing shape

nf: sequence number of the last block for the program of finishing shape

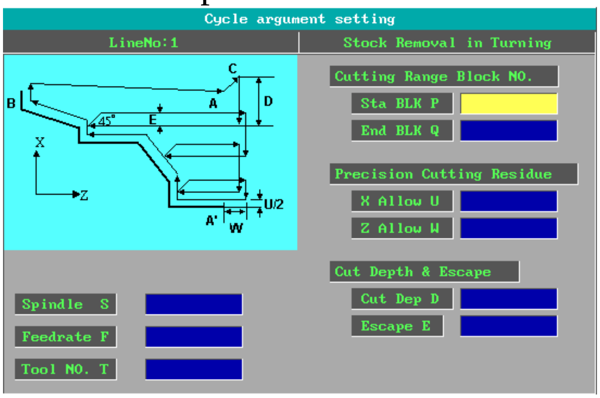
Δu: distance of finishing allowance in X direction

Δw: distance of finishing allowance in Z direction

F: feedrate

T: tool number

S: spindle rotate speed



3.3.3 G74.1 Stock Removal in Facing

G74.1 D_(d)_E_(e) P_(ns)_Q (nf)_U(Δu) W(Δw) F___ S___ T___

d: depth of cut in Z axis direction, it can be specified by the parameter#4013 and the parameter is changed by the program command

e: escaping amount, it can be specified by the parameter#4012

ns: sequence number of the first block for the program of finishing shape

nf: sequence number of the last block for the program of finishing shape

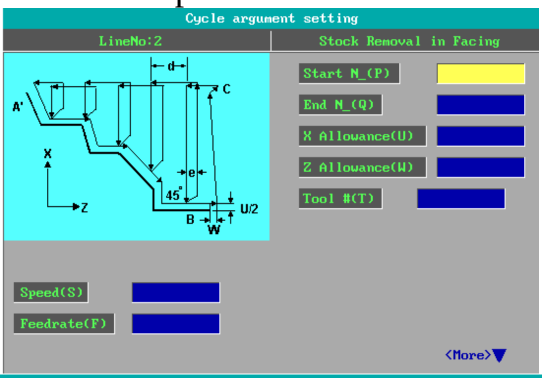
Δu: distance of finishing allowance in X direction

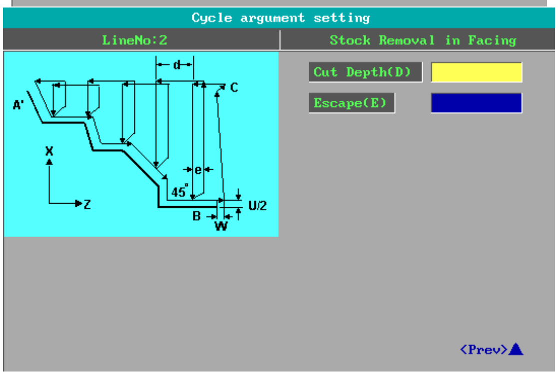
Δw: distance of finishing allowance in Z direction

F: feedrate

T: tool number

S: spindle rotate speed

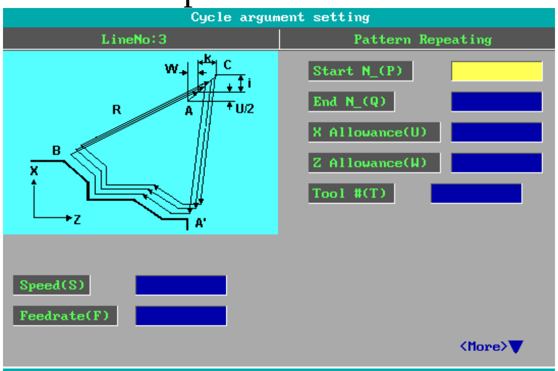


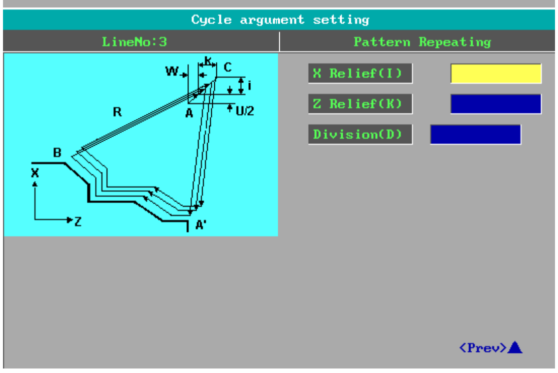


3.3.4 G75.1: Pattern Repeating

G75.1 X(Δi) Z(Δk) D(d)_P (ns) Q (nf) U(Δu) W(Δw) F___ S___ T___

Δi: distance and direction of relief in the X axis direction, this value can be specified by the parameter #4015
ΔK: distance and direction of relief in the Z axis direction, this value can be specified by the parameter #4016
d: the number of division, it can be specified by parameter #4017
ns: sequence number of the first block for the program of finishing shape
nf: sequence number of the last block for the program of finishing shape
Δu: distance and direction of finishing allowance in X direction
Δw: distance and direction of finishing allowance in X direction
F: feedrate
T: number of the tool
S: spindle rotate speed

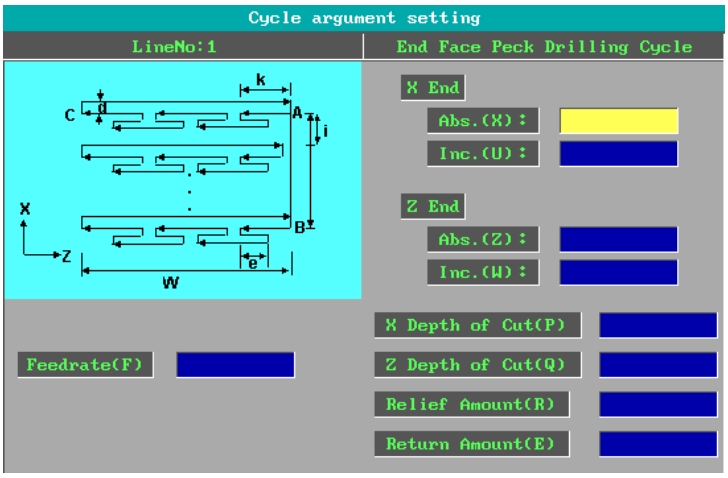




3.3.5 G76.1: End Face (Z axis) Peck Drilling Cycle

G76.1 E_(e)_X(U)_Z(W)_P(Δi)_Q(Δk)_R_(d)_F_

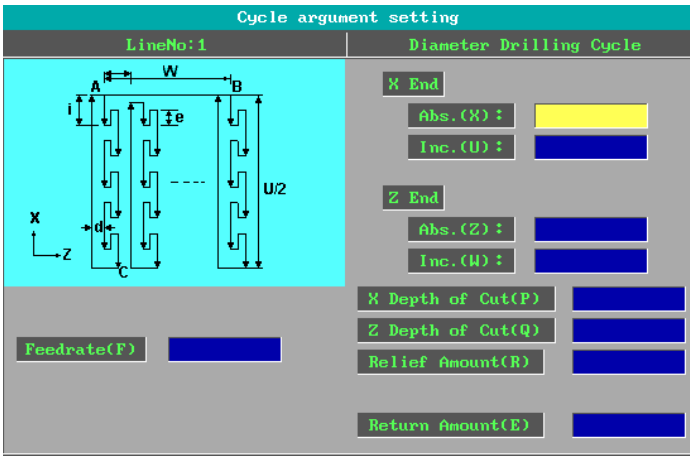
e: escaping amount (escaping amount in Z direction when Δk depth is cut) ← it can be specified by parameter #4011
X: X coordinate of point B（diameter）
Z: Z coordinate of point C
U: Incremental amount from A to B（diameter）
W: Incremental amount from A to C
Δi: Movement amount each cut in X direction(displayed by radius, positive)
Δk: Movement amount each cut in Z direction(positive)
Δd: Relief amount of the tool at the cutting bottom. (The value is 0 when it returns in original path)
F: Feed rate



3.3.6 G77.1: Outer Diameter/Internal Diameter Drilling Cycle

G77.1 E (e) X(U)___ Z(W)___ P(Δi) Q(Δk) R(Δd) F___

e: escaping amount(after cutting Δi distance in X axis direction) ←it can be specified by parameter #4011
X: X coordinate of point C(diameter)
Z: Z coordinate of point C
U: increment amount from B to C(diameter)
W: increment amount from A to B
Δi: movement amount in X direction (display by radius, positive)
Δk: depth of cut in Z direction (positive)
Δd: Relief amount of the tool at the cutting bottom. (The value is 0 when it returns in origin path)
F: feedrate



3.3.7 G78.1: Multiple Thread Cutting Cycle

G78.1 K(m) C(r) A(a) D(Δdmin) B_(d)_ X(U)_ Z(W)_ R (Δi) P (Δk) Q (Δd) (F__ or E__) __

m: repetitive count in finishing, specified by system parameter #4044.
r: chamfering amount, specified by system parameter #4043.
a: angle of tool tip, the angle from 80^0, 60^0, 55^0, 30^0, 29^0 and 0^0 is available. a can also be specified by system parameter #4042.
Q: minimum cutting depth $(\Delta d\sqrt{n} - \Delta d\sqrt{n-1}) < Q$, specified by system parameter #4045d: finishing allowance, specified by system parameter #4041
X(U): X coordinate in end point(bottom of tooth)
Z(W): Z coordinate in end point(bottom of tooth)
Δi: difference of thread radius
Δk: height of thread
Δd: depth of first cut
F: lead of thread in metric system(unit : mm/tooth)
E: lead of thread in imperial system(unit : tooth/inch)
H: numbers of thread (ex: H3 three thread type cutting, multiple thread F function is for neighbor thread)