FINGER CNC

Open Designs Customize The Future

B Series Milling Machine Programming Manual

Version No：F202309MP-CN

Guangzhou Finger Technology Co., Ltd

CONTENTS

Part 1.G-code Summary Table

G-Code	Function	G-Code	Function
* G00	Rapid Positioning/Feed	* G43	Tool Length Compensation (+) Direction
* G01 #	Linear Interpolation	* G44	Tool Length Compensation (-) Direction
* G02/G03	Circular Interpolation	* G49	Cancel Tool Length Compensation
* G02.1/G03.1	Ellipse Interpolation	G50	External Workpiece Coordinate System
* G02.2/G03.2	Parabolic Interpolation	G51/G50	Scale Function
G04	Dwell Command	G51.2/G50.2	Spindle Synchronization Function
G06.2	NURBS Curve Interpolation	G51.4/G50.4	Axial Synchronous Control
G06.3	Spline Curve Function	G52	Set Local Coordinate System
G08	Clearing Mechanical Coordinates	G53	Machine Coordinate Positioning
G09	Single Block Stop Mode	* G54 #	First Work Coordinate
G10		* G55-G59	Second to Sixth Work Coordinates
G11	Output Point Control	* G61	Enable Single Block Stop Mode
G12	Input Point Control	* G62	Disable Single Block Stop Mode
G12.1	Enable Polar Coordinate Interpolation	* G68	Enable Coordinate Rotation
G13.1	Disable Polar Coordinate Interpolation	* G69	Disable Coordinate Rotation
G12.8	Three-Point Circular Command	G73	High-Speed Peck Drilling Cycle
G13.9	Quick Check Variable Bit Status	G74	Reverse Deep Hole Tapping Cycle
G15	Servo Spindle Positioning	G76	Precision Boring Fixed Cycle
G16	Cylinder Interpolation	* G80 $	Cancel Fixed Cycle
* G17、G18、G19	Plane Setting	* G81 $	Drilling Cycle Setting
* G20	Inch Measurement Mode	* G82 $	Drilling Cycle (Pause at Bottom)

* G21	Metric Measurement Mode	* G83 $	Deep Hole Cycle
G27	Tool Positioning	* G84 $	Forward Deep Hole Tapping Cycle
G28	Return to First Reference Point	* G85 $	Boring Cycle
G29	Return to Self-Reference Point	* G86 $	Boring Fixed Cycle
G30	Return to Second Reference Point	G87	Back Boring Fixed Cycle
G31	Jump Function	G88	Semi-Automatic Fine Boring Fixed Cycle
G33	Thread Cutting		
G34 $	Circular Hole Cycle	* G89 $	Boring Fixed Cycle
G35 $	Angular Linear Hole Cycle	* G90	Absolute Value Coordinate Command
G36 $	Circular Hole Cycle	* G91	Incremental Value Coordinate Command
G37.1 $	Checkerboard Hole Cycle	* G94 #	Cutting Speed Specified in mm/min
* G40 #	Cancel Tool Radius Compensation	* G95	Cutting Speed Specified in mm/rev
* G41	Set Tool Radius Compensation (Left)	* G98 #	Return to Initial Point in Drilling Cycle
* G42	Set Tool Radius Compensation (Right)	* G99	Return to Feed Point in Drilling Cycle
. G-codes marked with an asterisk (*) are modal G-codes. . G-codes marked with a pound sign (#) are initial setup functions during power-on. . G-codes marked with a dollar sign ($) are specific functions for milling machines (finger milling machines).			

Table 1-1

1.1 One Shot G-code

This type of G-code (codes without an asterisk (*) in the instruction table) is only valid within the specified block.

- **Example**

  N10 G0 X30.000 Y40.000 ... G0 is a modal G-code.

1. N10 G0 X30.000 Y40.000 ......G0 is a modal G-code.
2. N20 G4 X2.000 ......G4 is a one-shot code, only valid within this block.
3. N30 X20.000 Y50.000 ......No G-code specified, the G0 code from N10 is still valid within this block.

1.2 Modal G-code

This type of G-code (codes marked with an asterisk (*) in the instruction table) remains in effect until replaced by another code within the same group.

- **Within the same group**

1. G00, G01, G02, G03 belong to the same group.
2. G17, G18, G19 belong to the same group.
3. G40, G41, G42 belong to the same group. G43, G44, G49 belong to the same group.
4. G21 to G22 belong to the same group. G54 to G59 belong to the same group.
5. G80 to G89 belong to the same group. G90 to G91 belong to the same group.
6. G94 to G95 belong to the same group. G98 to G99 belong to the same group.

- **Example**

1. N10 G0 X30.000 Y5.000 ... Specifies G0.
2. N20 X50.000 Z10.000 ... No G-code specified, G0 is still valid.
3. N30 Y20.000 ... No G-code specified, G0 is still valid.
4. N30 G1 X30.000 F200. ... G1 is valid (replaces G0).

For finger milling machine controllers, only one G-code from the same group can be specified within a single block. If multiple codes from the same group are specified, only the last one will be valid.

Part 2. G-code Instructions

2.1 G00 Rapid Positioning/Feed

- **Function and Purpose**

This command is accompanied by coordinate names. It moves from the current position to the specified coordinate position in a straight line path.

- **Command Format**

G00 X(U)Y(V) Z(W)_____

X_Y_Z_: Absolute coordinate values of the target position.

U_V_W_: Incremental values of the target position relative to the starting point of the block.

- **Detailed Explanation**

1. Once this command is given, the G00 mode remains active until a G01, G02, G03, or other single-shot G-code is encountered, which changes the mode of G00. Therefore, if the next command is also G00, only the axis addresses need to be specified.
2. The speed of positioning is set by the mechanical parameter items (MCM2320~MCM2359).
3. This command can simultaneously control movement in 1 to 6 axes. If an axis is not specified in the command, it will not move.
4. The G00 command can be set as linear interpolation type or non-linear interpolation type based on parameter MCM42.
5. MCM42 BIT00=0: G00 is linear interpolation type positioning. BIT00=1: G00 is non-linear interpolation type positioning, with each axis moving at its maximum speed independently.
6. The acceleration and deceleration form of G00 is fixed as linear type.

- **Program Example 1**

In the following program, point S is moved rapidly to point P1, and then to point P2.

- **Method 1: G90 command, absolute value program**

```
N1 G90
N2 G00 X25.000 Y20.000 Z10.000 F100.000          ...P1
N3 X60.000 Y50.000 Z40.000                       ...P2
```

- **Method 2: G91 specified, incremental value program**

```
N1 G91
N2 G00 X25.000 Y20.000 Z10.000 F100.000          ...P1
N3 X35.000 Y30.000 Z30.000                       ...P2
```

- **Method 3: G90 specified, incremental value program**

```
N1 G90
N2 G00 U25.000 V20.000 W10.000 F100.000          ...P1
N3 U35.000 V30.000 W30.000                       ...P2
```

- **Method 4: G90 specified, mixed use of incremental and absolute values**

N1 G90

N2 G00 X25.000 V20.000 W10.000 F100.000 ...P1

N3 X60.000 V30.000 W30.000 ...P2

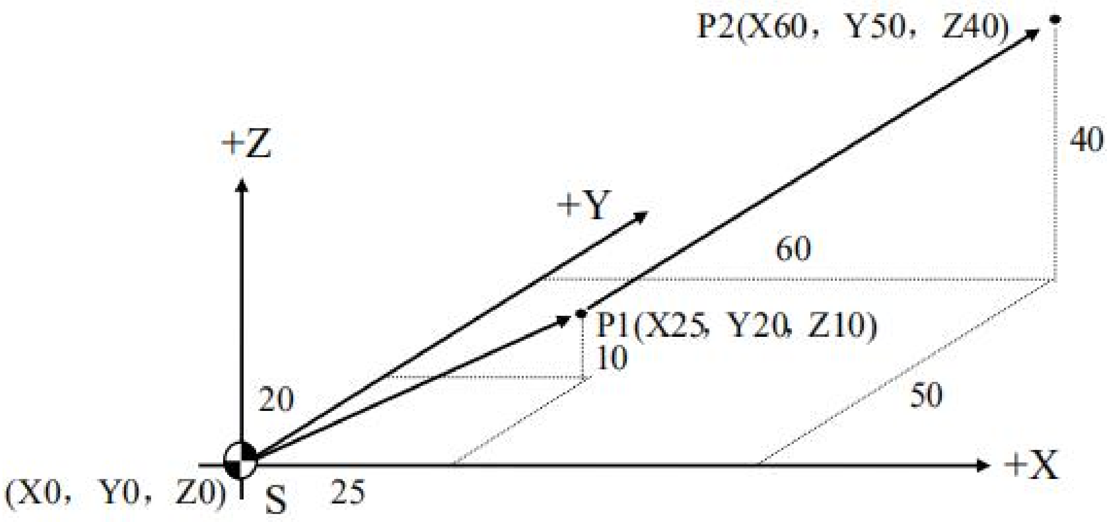


Figure 2.1.2 Program Example

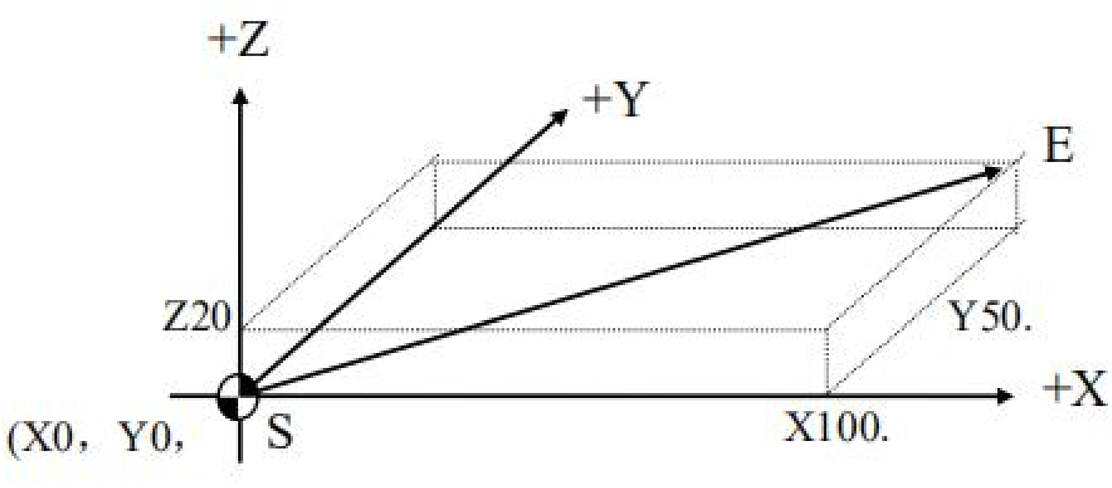
The tool moves rapidly to the position X25.000 Y20.000 Z10.000 in a quick positioning manner. Since all three axes (X, Y, Z) have displacement values, the feed rate is determined by the lower feed rate specified in the MCM2320~MCM2325 "maximum feed rates" parameter.

- **Program Example 2**

Move from point S to point E using rapid positioning.

G90

G00 X100.000 Y50.000 Z20.000



The tool moves to the position X = 100.000, Y = 50.000, Z = 20.000 using rapid positioning. Since the displacement distances are different for each axis, the controller uses the displacement of the longest axis (X-axis) and the feed rate set in the MCM 2320 parameter. If the maximum feed rate for X, Y, and Z is set to 10000 mm/min, the speeds for each axis in the above example are as follows:

X-axis: 10000 mm/min

Y-axis: 5000 mm/min

Z-axis: 2000 mm/min

X-axis: The displacement is 100.000 mm, which is the longest among the displacement distances. Therefore, the controller uses the feed rate set in the MCM2320 parameter, which is 10000 mm/min, as the feed rate for the X-axis.

Y-axis: The displacement is 50.000 mm. To calculate the feed rate for the Y-axis, we divide the displacement by the longest axis distance, which is 100.000 mm, and then multiply it by the maximum feed rate set in the MCM2321 parameter, which is 10000. So the calculation is (50.000/100.000) * 10000 = 5000. The actual feed rate for the Y-axis is 5000 mm/min.

Z-axis: The displacement is 20.000 mm. Following the same calculation method as above, we divide the displacement by the longest axis distance (100.000 mm) and multiply it by the maximum feed rate set in the MCM2322 parameter, which is 10000. So the calculation is (20.000/100.000) * 10000 = 2000. The actual feed rate for the Z-axis is 2000 mm/min.

2.2 G01 Linear Interpolation

- **Function and Purpose**

The G01 command is used for linear cutting. It executes linear interpolation according to the program instructions. The absolute or incremental mode is determined by the G90/G91 command. The movement is performed at the speed specified by the "F" value.

- **Command Format**

G01 X(U)_____ Y(V)_____ Z(W)_____ F_____

X_Y_Z_: Absolute coordinates of the cutting endpoint.
U_V_W_: Incremental values relative to the starting point of the block.
F_: Feed rate for the cutting operation. The F command can be used with any G command. It can also be used within a G00 block without affecting rapid positioning.

- **Detailed Explanation**

1. The G01 command is used for linear cutting and can control 1 to 6 axes simultaneously. The feed rate for the cutting operation is set by the F command and can be set as low as 0.001mm/min.
2. Once the G01 command is given, the G01 mode remains active until a G00, G02, G03, or other single-block G motion command is encountered. Therefore, if these commands are also G01 with the same feed rate, only the coordinate values need to be specified.
3. The cutting start point is the coordinate position of the tool when the command is issued. The feed rate specified by the F command (Modal Code) remains active until redefined and does not need to be specified in every block.

Calculation formula for X, Y, Z-axis cutting feed rates: (In the formula, U, V, W represent the actual incremental values)

X-axis feed rate: $F_x = \dfrac{U}{\sqrt{U^2 + V^2 + W^2}} * F \quad (1)$

Y-axis feed rate: $F_y = \dfrac{V}{\sqrt{U^2 + V^2 + W^2}} * F \quad (2)$

Z-axis feed rate: $F_z = \dfrac{W}{\sqrt{U^2 + V^2 + W^2}} * F \quad (3)$

- **Example Program**

Here is an example of a G01 program with three different variations, but all performing the same linear cutting path:

❖ Absolute Programming with G90 (Figure 3-2)

N1 G90

N2 G01 X25.000 Y20.000 Z10.000 F100.00 ... P1

N3 X60.000 Y50.000 Z40.000 ... P2

❖ **Incremental Programming with G91 (Figure 3-2)**

N1 G91

N2 G01 X25.000 Y20.000 Z10.000 F100.00 ... P1

N3 X35.000 Y30.000 Z30.000 ... P2

❖ **Absolute Programming with G90 (Figure 3-2)**

N1 G90

N2 G01 U25.000 V20.000 W10.000 F100.00 ... P1

N3 U35.000 V30.000 W30.000 ... P2

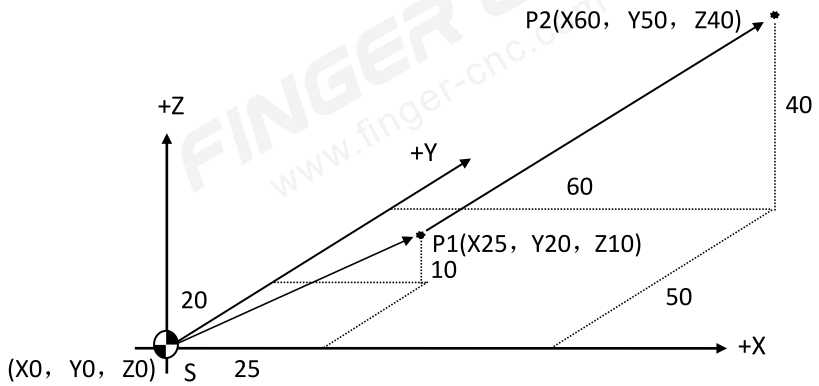


Figure 2-3 G01 Program Example

- **Program Example 2**

G01 X2.0 Z2.01 F0.300 . . . Absolute Instruction

G01 U1.0 W-2.59 F0.300 . . . Incremental Instruction

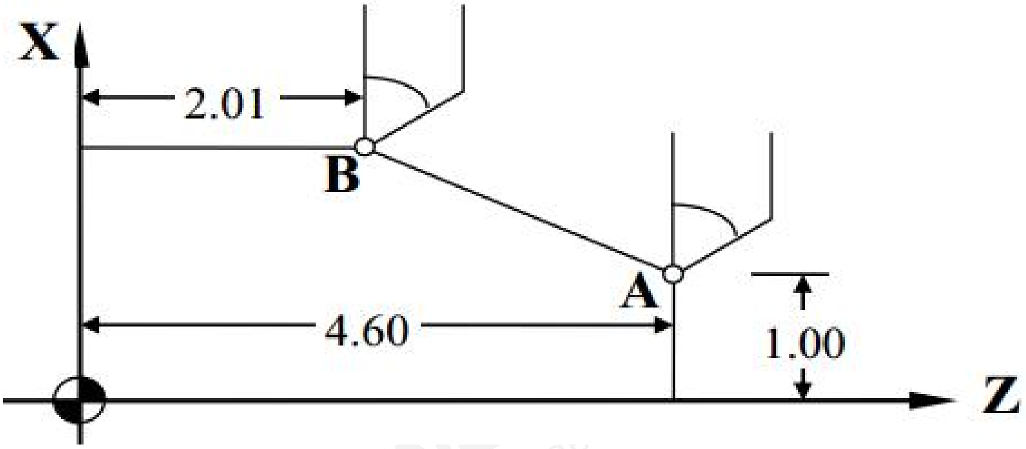


Figure 2.2.1 G01 Program Example

2.3 Circular Interpolation (G02 and G03)

- **Function and Purpose**

The purpose of these commands is to enable the tool to move along an arc in a specified plane.

- **Command Format**

Circular Interpolation (G02 and G03)	
	Arc or Circle
Radius Mode	G17 G02(G03) X(U)_Y(V)R_F: G17 X0 A0 represents the X-Axis and A-Plane.
	G18 G02(G03) X(U)_Z(W)R_F: G18 Z0 A0 represents the Z-Axis and A-Plane.
	G19 G02(G03) Y(V)_Z(W)R_F: G19 Y0 A0 represents the Y-Axis and A-Plane.
Center Mode	G17 G02(G03) X(U)_Y(V)I_J_F: G17 X0 A0 represents the X-Axis and A-Plane.
	G18 G02(G03) X(U)_Z(W)I_K_F: G18 Z0 A0 represents the Z-Axis and A-Plane.
	G19 G02(G03) Y(V)_Z(W)J_K_F: G19 Y0 A0 represents the Y-Axis and A-Plane.

G02: Circular interpolation in a clockwise (CW) direction.

G03: Circular interpolation in a counterclockwise (CCW) direction.

X_: Movement distance along the X-axis or its parallel axis (as set by Mcm1881, Mcm1884, Mcm1887, etc.).

Y_: Movement distance along the Y-axis or its parallel axis (as set by Mcm1882, Mcm1885, Mcm1888, etc.).

Z_: Movement distance along the Z-axis or its parallel axis (as set by Mcm1883, Mcm1886, Mcm1889, etc.).

U_: Increment along the X-axis or its parallel axis.

V_: Increment along the Y-axis or its parallel axis.

W_: Increment along the Z-axis or its parallel axis.

I_: Distance from the starting point on the X-axis to the center of the arc, specified as a radius with a signed value.

J_: Distance from the starting point on the Y-axis to the center of the arc, specified as a radius with a signed value.

K_: Distance from the starting point on the Z-axis to the center of the arc, specified as a radius with a signed value.

R_: Radius of the arc, specified as a signed value (negative value indicates an arc exceeding 180°).

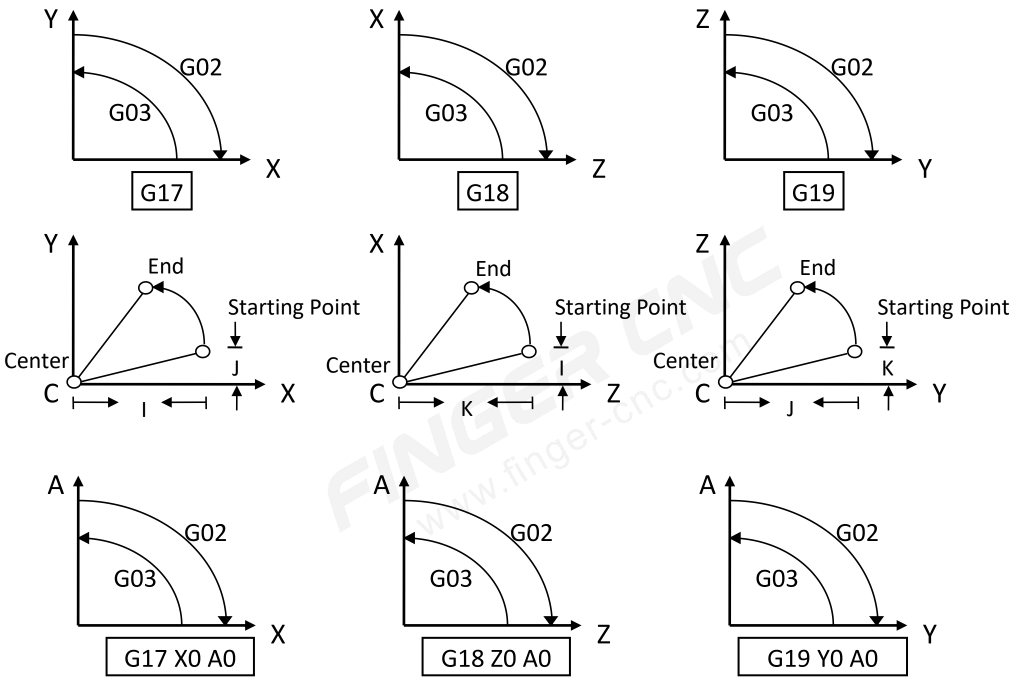
F_: Feedrate for the arc interpolation.

- **Detailed explanation**

1. Clockwise direction (G02) and counterclockwise direction (G03) are referred to as the direction seen from the negative direction of the Z (Y, X) axis in relation to the G17 (G18, G19) plane in the Cartesian coordinate system.

   G02: Clockwise direction (cw)

   G03: Counterclockwise direction (ccw)



2. The movement of the arc refers to the distance from the arc endpoint to the arc starting point. It can be specified using absolute coordinates or incremental coordinates.

3. The arc radius can be specified using I, J, K, or R. The I, J, K values specify the distance from the starting point to the arc center, while the R value specifies the radius of the arc. Here is an example:

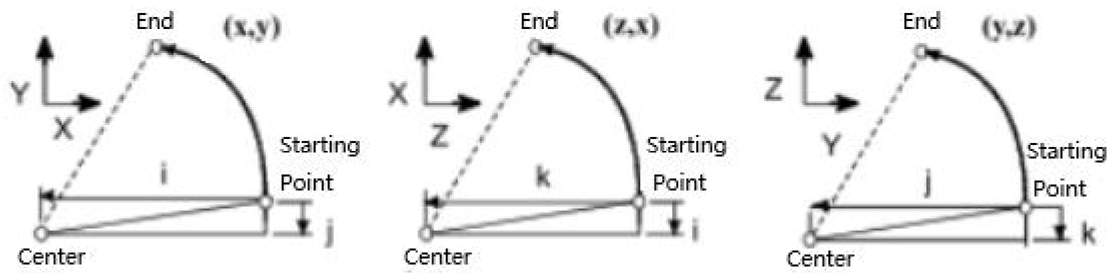


Figure 2.3.1 Circular cutting path diagram.

When the arc radius is specified by R, the center of the arc is determined by the start point, end point, and the direction of machining. When R is specified as a negative value, it indicates machining an arc with a central angle α (180° < α < 360°). For example, G02 Z60.000 X20.000 R-50.000 F0.300.

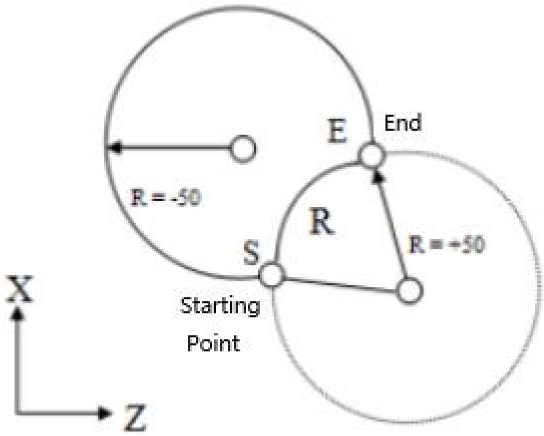


Figure 2.3.2 Arc Cutting Path with Specified Radius.

When both I, J, K and R are specified, the arc is prioritized based on the specified radius (R), and I, J, K values are ignored.

4. Processing a full circle can only be done using I, J, K values, and using R to specify the radius will result in a system alarm.

   Example:

   G90

   G00 X40.000 Y0.000

   G03 X40.000 Y0.000 I50. F100 Full circle command.

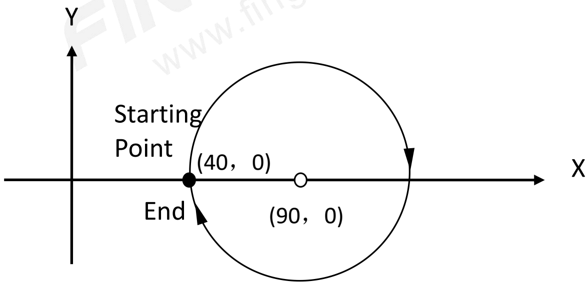


Figure 3-10 Full Circle Cutting

5. In G02 and G03, the specified feed rate value (F) represents the velocity along the tangent of the arc. This tangent velocity is limited by the arc radius and the programmed feed rate. If the calculated maximum tangent velocity of the arc is greater than the programmed feed rate, the programmed feed rate is used as the tangent velocity. Conversely, if the calculated maximum tangent velocity of the arc is smaller than the programmed feed rate, the calculated maximum tangent velocity is used as the tangent velocity.

6. By simultaneously specifying the movement of axes outside the specified plane (axis 1 or axis 2), it is possible to perform helical interpolation, which moves the tool in a spiral motion. The command format is as follows:

   G17: G02/G03 X(U)Y(V)R(I_J)α_(β_)F_

   G18: G02/G03 Z(W)X(U)R(I_K)α_(β_)F_

   G19: G02/G03 Y(V)Z(W)R(K_J)α_(β_)F_

   α(β) represents the axis outside the current plane as shown in the diagram below.

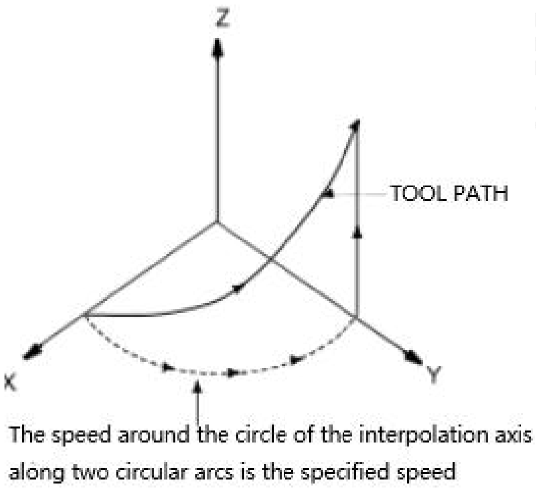


Figure 2.3.3 Full circle cutting

7. The processing of a circular arc is actually done by dividing the arc into n equal linear segments, and the higher the number of segments, the higher the accuracy of the arc. The arc accuracy is controlled by Mcm10. Mcm10 = (radius - chord height) / radius, and the smaller the set value, the higher the accuracy. The minimum arc accuracy can be precise up to 0.000001.

8. For process requirements, it is possible to machine products where the radius values at the arc start and arc end are different. The difference is controlled by Mcm13. When the difference between the radius value at the arc start and the radius value at the arc end is less than Mcm13 * 0.001 * radius at the arc start, the arc is executed normally, creating a spiral machining. When it exceeds the limit, the system will trigger an alarm.

9. When the arc radius is 1mm, the speed is limited by Mcm1781. Mcm1781 = 500 indicates that the maximum speed of the arc cannot exceed 500mm/min when the arc radius is 1mm.

10. When crossing quadrants during the arc, if one axis has a speed that is too fast and

another axis has a speed that is too slow, it can result in sharp corners in the arc. Mcm4040 to Mcm4159 can be used to solve the issue of sharp corners in the arc.

- **Program Example 1**

G17
G0 X200. Y40. Z0.
G03 X140. Y100. R60. F300.
G02 X120. Y60. R50.

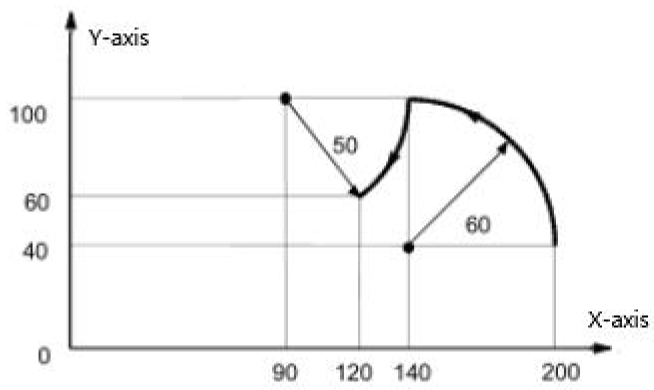
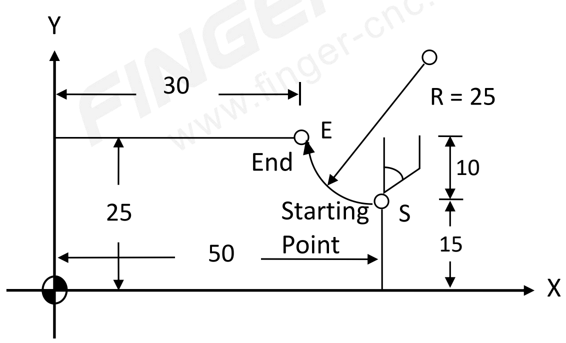


Figure 2.3.4 Program Example

- **Program Example 2**

Below are four different instructions, but they cut the same arc.
The starting coordinates are X=50.000, Y=15.000,
The ending coordinates are X=30.000, Y=25.000,
The radius is R=25.000, or I=0.000, J=25.000.

1. G02 X30.000 Y25.000 J25.000 F200
2. G02 U-20.000 V10.000 J25.000 F200
3. G02 X30.000 Y25.000 R25.000 F200
4. G02 U-20.000 V10.000 R25.000 F200



2.4 G02.1、G03.1Ellipse interpolation

- **Function and Purpose:**

To perform interpolation for an elliptical path.

- **Instruction Format**

G02.1 X/U_ Z/W_ R_ P_ Q_ Ff_ (Clockwise)

G03.1 X/U_ Z/W_ R_ P_ Q_ Ff_ (Counter-clockwise)

X/U_ Z/W_: End point coordinates of the elliptical interpolation.

R: Specifies the value of the major axis of the ellipse (radius specified) (ignores sign) (non-modal).

P: Specifies the value of the minor axis of the ellipse (radius specified) (ignores sign) (non-modal).

Q: Specifies the angle between the major axis of the ellipse and the positive direction of the Z axis (0~180°) (unsigned) (non-modal).

Ff: Specifies the feed rate (modal).

- **Program Example**

G0 X0. Z0.

G01 X72.059 Z69.336 F3000.

G02.1X -95.708Z-28.984R110. P50. Q240.

G01 X72.059 Z69.336 F3000.

G03.1X -95.708Z-28.984R110. P50. Q60.

In the figure, point A is the starting point, point B is the end point. The red line represents the G02.1 curve, and the green line represents the G03.1 curve.

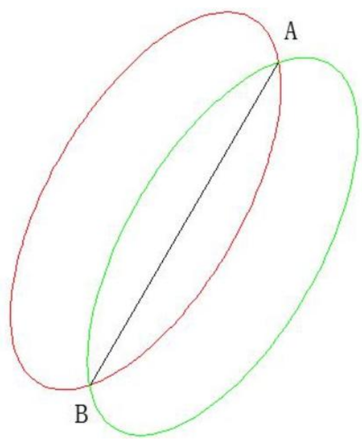


Figure 2.4.1 Elliptical Interpolation

2.5 G02.2, G03.2 Parabolic Instructions

- **Function and Purpose**

To implement parabolic interpolation commands.

- **Command Format**

G02.2 X/U_ Z/W_ P_ Q_ Ff_ (Clockwise direction)

G03.2 X/U_ Z/W_ P_ Q_ Ff_ (Counterclockwise direction)

X/U_ Z/W_: Endpoint coordinates for parabolic interpolation.

P: Specifies the value of P in the parabolic equation X^2=2PZ (radius specified) (ignore sign) (non-modal) Range: 09999999999.

Q: Specifies the angle between the parabolic axis and the positive Z-axis (0180.000°) (unsigned) (non-modal).

Ff: Specifies the feed rate (modal).

2.6 G04 Dwell

- **Command Format**

G04 X_____

G04 P_____

X: Pause time in seconds. (Here, X represents time, not an address).

P: Pause time in milliseconds.

- **Detailed Explanation**

During program design, in certain situations, it may be necessary to temporarily pause the motion of all axes for a specific duration after the execution of a single block, before proceeding to the next block of instructions. This command can be used in such cases.

The minimum pause unit is 0.01 seconds, and the maximum pause time can be set up to 8000.0 seconds.

- **Program Example**

N1 G1 X10.000 Y10.000 F100.

N2 G4 X2.000.　.　.　Pause for 2 seconds

N3 G0 X0.000 Y0.000

2.7 G06.2 NURBS Curve Interpolation

- **Function and Purpose**

This function is used to specify the parameters (degree, weight, knots, control points) of NURBS (non-uniform rational B-spline) curves for surface or curve machining. It does not replace small line segments and allows for NURBS curve machining.

- **Related Parameter**

MCM11

Definition: Special curve (parabolic, spline curve) accuracy setting (output minimum distance between two points).

Range: 0 to 2^31

Default Value: 100

Unit: micrometers

Note: The minimum distance between line segments when decomposing special curves into small segments is set by this parameter.

- **Instruction Format**

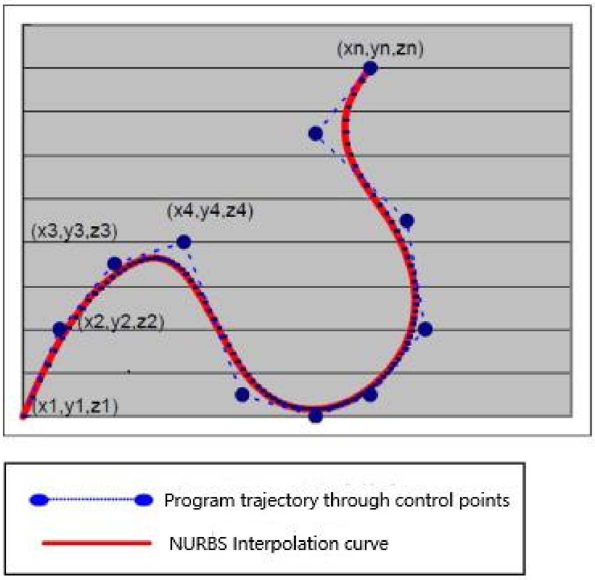
```
G06.2  Pp  Kk1 X1 Yy1 Zz1 Rr1 Ff;  NURBS interpolation ON
           Kk2 Xx2 Yy2 Zz2 Rr2;
           Kk3 Xx3 Yy3 Zz3 Rr3;
           Kk4 Xx4 Yy4 Zz4 Rr4;
           ............................................
           Kkn Xxn Yyn Zzn Rrn;
           Kkn+1;
           Kkn+2;
           Kkn+3;
           Kkn+4;                     NURBS interpolation OFF
```

G06.2	NURBS Curve Interpolation Instruction
P_	NURBS curve degree: The curve is of degree (p-1), and it is equivalent to P4 when omitted.
K_	Knots: The same value is set for the knots from the first program segment to the pth program segment. If there is a program segment with only knots, the NURBS interpolation ends.
X_Y_Z_	Control point coordinates: Coordinates of the control points.

R_	Control point weights: Range from 0.0001 to 99.9999.
F_	Interpolation speed: Speed of the interpolation.

- **Detailed Explanation**

1. In the first program segment of NURBS interpolation, specify the degree P.
2. The control points in the first program segment of NURBS interpolation should be specified at the same coordinates as before NURBS interpolation.
3. In the first program segment of NURBS interpolation, all the axes that will be used in subsequent NURBS interpolation program segments should be set.
4. The set values for the knots from the first program segment to the Pth segment of NURBS interpolation must be consistent.
5. When there is a single program segment with P consecutive specified knots K, and the set values for knot K are the same, the NURBS curve ends.



- **Function Handling**

1. G codes, feed rates, and auxiliary functions:

❖ In NURBS interpolation mode, no G codes, feed rates, or MSTB codes can be set.

❖ From the second program segment of NURBS interpolation onwards, it is not allowed to specify axis addresses, R, K, or any other instructions that are inconsistent with the first program segment.

2. Program Jump/Program Stop can be used in the first program segment of NURBS interpolation, but cannot be set in the second program segment and onwards.
3. Single Block Execution: It is only effective at the final control point and has no effect

in NURBS interpolation.

4. Reset: Reset cancels the NURBS interpolation mode.
5. The following functions are supported: Feed Hold.
   - ❖ The following functions are not supported: Program Stop (M00), Optional Stop (M01), MDI Insert.

- **Note**

1. NURBS interpolation involves 3 basic axes.
2. All axes involved in NURBS interpolation should be specified as control points in the first program segment. Specifying them in the second program segment or later will result in a system alarm (1-14-5) (TBD).
3. The first control point should be consistent with the end point of the previous program segment using G06.2. Otherwise, a system alarm will occur (206-1).
4. In the weight instruction, omitting the decimal point will still be treated as having a decimal point. Specifying 1 is equivalent to specifying 1.0. The valid range for weights is 0.0001 to 99.9999. Setting values beyond this range will trigger a system alarm (206-2).
5. In the node instruction, omitting the decimal point will still be treated as having a decimal point. Specifying 1 is equivalent to specifying 1.0. The node instruction cannot be omitted and must be specified in each program segment. Failure to do so will result in a system alarm (206-3).
6. Nodes should be specified as greater than or equal to the value in the previous program segment. Otherwise, a system alarm will occur (206-4).
7. Reset cancels the NURBS interpolation mode.

- **Program Example**

```
G0 X0.Y0.Z0.
G06.2 P4 X0. Z0. R1. K0 F1000
X1. Z2. R1. K0
X2.5 Z3.5 R1. K0
X4.4 Z4. R1. K0
X6. Z0.5 R1. K1
X8. Z0. R1. K2
X9.5 Z0.5 R1. K3
X11. Z2. R1. K4
X10.5 Z4.5 R1. K5
X8.0 Z6.5 R1. K6
X9.5 Z8. R1. K7
```

K8
K8
K8
K8
M30

2.8 G06.3 Spline Curve Function

- **Command Format**

G06.3 (or G306) Q__ E__ D__ __ __ F__

Q__: Starting address of the spline curve control points (excluding the starting point), not affected by decimal point. (When the Q letter is not specified, the default value is 1000)
E__: Ending address of the spline curve control points (excluding the end point), not affected by decimal point. (When the E letter is not specified, the default value is 1000)
D__: D0 = absolute coordinate mode for control point coordinates, D1 = incremental coordinate mode for control point coordinates. When D is not specified, the default mode is D0 (absolute).
α__β __ : End point coordinates of the spline curve interpolation axis. (α ,β represents any one of the axes X, Y, or Z of the spline interpolation axis)
F: Feed rate (mm/min or mm/rev)

- **Storage of Control Points**

Storage Range: User Variables V1000~V9999
One variable stores one interpolation axis coordinate. Control point coordinates are stored as groups (i.e., horizontal axis interpolation coordinates are stored first, followed by vertical axis interpolation coordinates). A maximum of 4500 groups of control points can be stored. A single G06.3 command can specify up to 4500 control points, including the starting point and ending point coordinates as one control point. The control points are stored in the order of their appearance in the spline curve. The following table illustrates the storage sequence.

- **Table Title**

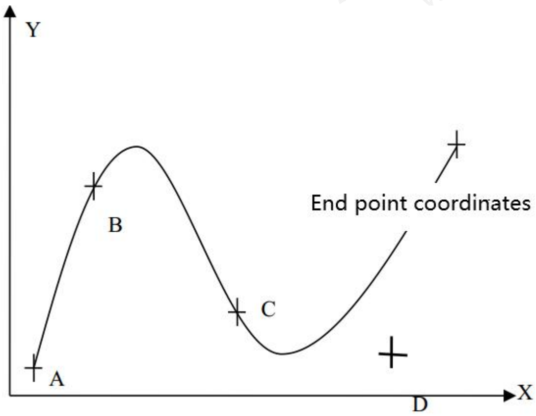
V1000	*x*0 (Vertical Interpolation Axis Coordinates)	The first control point
V1001	*y*0 (Horizontal Interpolation Axis Coordinates)	
V1002	*x*1 (Vertical Interpolation Axis Coordinates)	The second control point

V1003	y1 (Horizontal Interpolation Axis Coordinates)	
V……	………………………	……………….

- **Example Program**

G06.3 Q1004 E1008 Xx4 Yy4

As shown in the diagram below, the curve consists of three control points: B(x1,y1), C(x2,y2), and D(x3,y3). A(x0,y0) is the starting point, B(x1,y1) is the first control point, C(x2,y2) is the second control point, D(x3,y3) is the third control point, and the end point coordinates are (Xx4 Yy4).



V1000		Control Point Starting Address
…	…	…
V1004	x1	B Point
V1005	y1	
V1006	x2	C Point
V1007	y2	
V1008	x3	D Point
V1009	y3	
V1010	x4	
V1011	y4	

Axis determination: It is determined by the G17/G18/G19 plane, similar to the method used for circular interpolation axes.

2.9 G08 is used to clear the machine coordinates of each axis

- **Command Format**

G08 ... Clear all axes, i.e., X, Y, Z machine coordinates.

G08 X__ Y__ ... Clear X, Y axis machine coordinates.

G08 Z__ ... Clear Z axis machine coordinate.

G08 X__ Y__ A__ ... Clear X, Y, A axis machine coordinates.

Or any combination of X, Y, Z, A, B.

- **Detailed Explanation**

In the program format, the numerical values assigned to X, Y, and Z are meaningless but must be provided in order to include the command in the program. They simply indicate which axis's machine coordinate is intended to be cleared. If the G08 command is used as a standalone segment, the machine coordinates for all three axes (X, Y, and Z) will be cleared simultaneously. In other words, the machine coordinates for X, Y, and Z axes will be reset to their respective zero positions.

- **Program Example**

```
N1 G54               ;;; Assume the values of XY in G54 coordinate system are both 0
N2 G01 X100. Y100.   ;;; Move to the machine coordinate position X100. Y100.
N3 G08 X0            ;;; Clear the machine coordinate of the X-axis, setting the current
                         position as the machine zero point for the X-axis
N4 G08 Y0            ;;; Clear the machine coordinate of the Y-axis, setting the current
                         position as the machine zero point for the Y-axis.
```

2.10 G09、G61、G62 Automatic Corner Override Mode

- **Function and Purpose**

1. To mitigate mechanical collisions when there is a sudden change in tool feedrate.
2. To prevent rounding of corners during corner cutting (or control the size of the rounding).
3. To ensure that after mechanical deceleration and stop, the positioning is confirmed or a deceleration check time has elapsed before starting the next program block.
4. To accurately stop at the end of the program block that includes G09 before

continuing with the next program block. This function is used for machining sharp edges. G09 is only effective within the specified program block.

- **Instruction Format**

G09 G01(G02 G03) or G09 X__Y__Z__

Note: When using G09 X_ Y_ Z_, it does not change the modal G code. Precise positioning only takes effect within the G09 instruction.

G61: Enables single-block precision positioning mode (modal command, canceled by G62).

G62: Cancels single-block precision positioning mode (modal command, cancels G61).

The difference between G61 and G09 is that G61 is a modal command, and G61 can be canceled by G64.

- **Program Example**

G01 X50.Z50.

G01 X75.

N001 G09 X100.F150----This is equivalent to G09 G01 X100. (Precise positioning, see the diagram below)

N002 Z100.

G83 X400. Z400.F3.

G09 X500. Z500.----------This is equivalent to G09 G83 X500. Z500.

G00 X600.Z600.

M3 S500

G90 X700.Z700.F2.

X800.

X900.

G09 X1000.---------------- This is equivalent to G09 G92 X1000 Z700. F2.

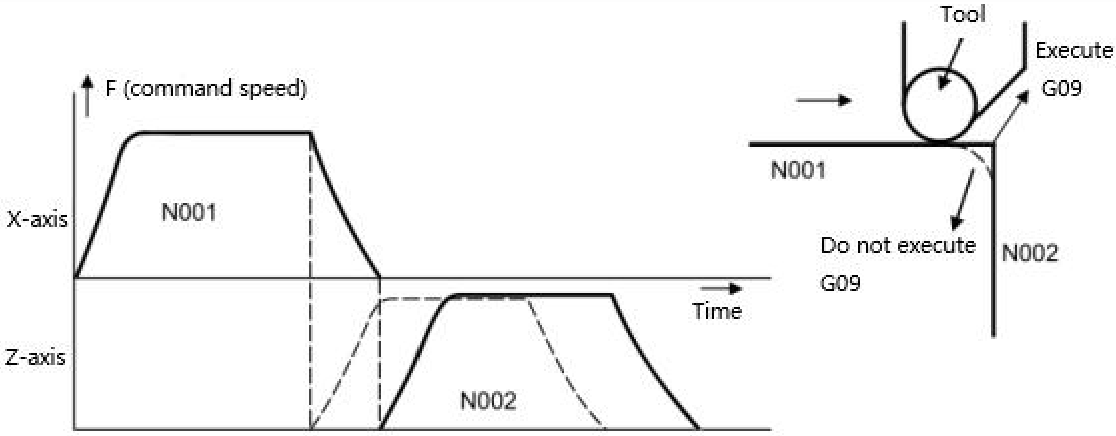
G0 X0.

Z0.

M30

- **Regarding approach width**

N001 G09 G01 X100.000 F150 ;	After the deceleration check time or confirming the positio status after deceleration stops, execute the next section
N002 Z100.000;	



The solid line represents the speed curve with G09, and the dashed line represents the speed curve without G09

Figure 1 Effect of Precise Stop Verification

Figure 2.6.1 when in precise positioning.

1. Continuous Cutting Feed

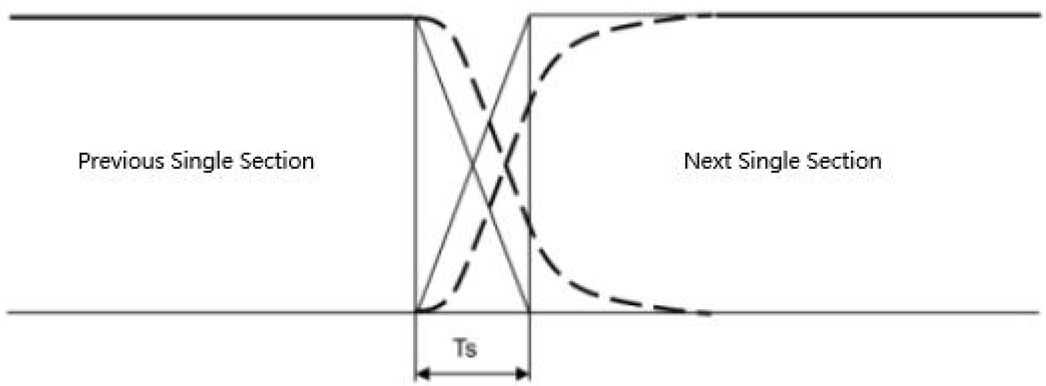


Figure 2 Continuous Cutting Feed Command

Figure 2.6.2Continuous Cutting without G09 (Non-Stop Mode)

2. Cutting Feed Position Check

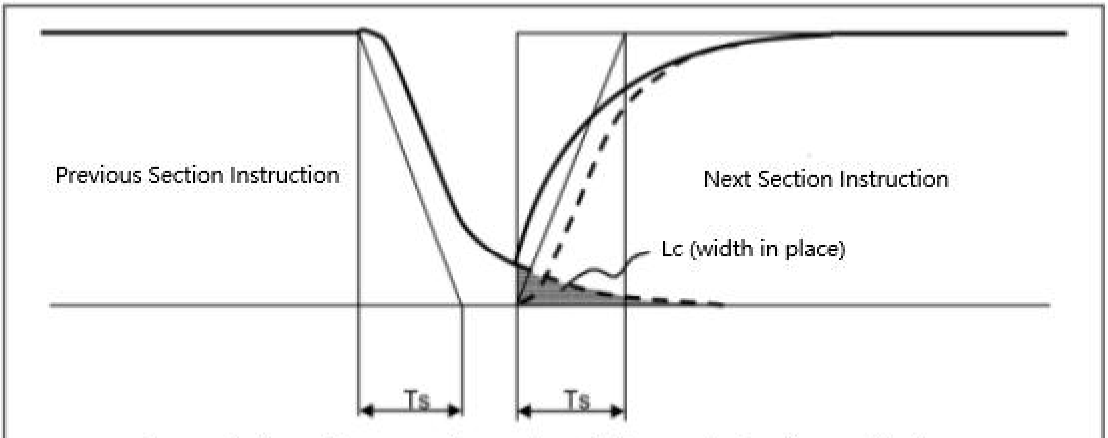


Figure 3 Single Section Connection During Hafnium Cutting in Place Verification

Figure 2.6.2 when in continuous cutting without G09 (non-stop mode).

2.11 G10 Instruction

2.11.1 G10 L2

- **Instruction Format**

G10 L2 P_ IP_ (Setting of Workpiece Coordinate System G54~G59, a maximum of 6 axes can be modified at once)

P: Workpiece coordinate system group number

P = 1~6 corresponds to G54~G59 workpiece coordinate systems.

P = 0, setting of external workpiece coordinate system.

P = Blank, setting of the current work coordinate system.

An alarm will occur when the current work coordinate system is G54.1.

IP_: If it is an absolute command, it specifies the work coordinate offset for each axis. If it is an incremental command, the value is added to the current work coordinate offset for each axis (resulting in the new work coordinate offset). If no axes are specified (i.e., empty), an alarm will be triggered.

- **Program Example 1**

G54	Machine coordinates	Program coordinates
G0 X0 Y0 Z0	0.000;0.000;0.000	0.000;0.000;0.000
G10 L2 P0 X100. Y80. Z50.	0.000;0.000;0.000	100.000;80.000;50.000
G0 X0 Y0 Z0	-100.000;-80.000;-50.000	0.000;0.000;0.000

Explanation:

G10 L2 P0 X100. Y80. Z50. is used to set the external work coordinate system with the parameter SYS2160.

- **Program Example 2**

G54	Machine coordinates	Program coordinates
G0 X0 Y0 Z0	0.000;0.000;0.000	0.000;0.000;0.000
G10 L2 P1 X100. Y80. Z50.	0.000;0.000;0.000	100.000;80.000;50.000
G0 X0 Y0 Z0	-100.000;-80.000;-50.000	0.000;0.000;0.000

Explanation:

G10 L2 P1 X100. Y80. Z50. is used to set the G54 work coordinate system using absolute values. The parameter P=1 indicates that the command is setting the G54 coordinate system.

- **Program Example 3**

G54	Machine coordinates	Program coordinates
G0 X0 Y0 Z0	0.000;0.000;0.000	0.000;0.000;0.000
G10 L2 P1 X100. Y80. Z50.	0.000;0.000;0.000	100.000;80.000;50.000
G0 X0 Y0 Z0	-100.000;-80.000;-50.000	0.000;0.000;0.000
G10 L2 P1 U100. V80.W50.	0.000;0.000;0.000	200.000;160.000;100.000
G0 X0 Y0 Z0	-200.000;-160.000;-100.000	0.000;0.000;0.000

Explanation:

G10 L2 P1 U100.V80. W50.P=1 represents the setting of the G54 coordinate system using incremental instructions.

2.11.2 G10 L20

- **Instruction format**

G10 L20 P_ IP_ (Set G54.1 work coordinate system, can modify up to 6 axes at a time)

P: Work coordinate system group number

P=1-48 corresponds to G54.1 P1G54.1 P48.

P=Blank, set the current work coordinate system.

An alarm will be triggered if the current work coordinate system is G54~G59.

IP_: If it is an absolute command, it represents the work coordinate offset value for each axis.

If it is an incremental command, the value is added to the original work coordinate offset value set for each axis (resulting in the work coordinate offset value).An alarm will be triggered if no axis is specified (i.e., blank).

- **Program Example**

G54.1	Machine coordinates	Program coordinates
G0 X0 Y0 Z0	0.000;0.000;0.000	0.000;0.000;0.000
G10 L20 P1 X100. Y80. Z50.	0.000;0.000;0.000	100.000;80.000;50.000
G0 X0 Y0 Z0	-100.000;-80.000;-50.000	0.000;0.000;0.000

Explanation:

G10 L20 P1 X100. Y80. Z50. P=1 represents the setting of G5.14 coordinate system using absolute commands.

2.11.3 G10 L990

- **Instruction format**

G10 L990 P1 (Enabling real-time computation functionality with MACRO B)

G10 L990 P0 (Disabling real-time computation functionality)

- **Program Example 1**

G10 L990 P1

#1 = 100

#2= #1+1

#3= #1-1

......

G10 L990 P0

So, the MACRO B between G10 L990 P1 and G10 L990 P0 is a real-time computation instruction, meaning that the values of variables are changed only when this line is executed.

Note: If there are non-MACRO B instructions between G10 L990 P1 and G10 L990 P0, an error will occur.

- **Program Example 2**

G10 L990 P1

#1 = 100

#2= #1+1

#3= #1-1

G01 X100. ;This line is not a MACRO B instruction, an error will occur.

G10 L990 P0

2.11.4 G10 L991

- **Instruction format**

G10 L991 P__A__ B__ C__ D__ (Set acceleration/deceleration time)

P: Axis selection for setting acceleration/deceleration time. BIT00=1 for X-axis, BIT01=1 for Y-axis, and so on. When P is blank or 0, set the acceleration/deceleration time for all axes.

A: Set acceleration/deceleration time for G00, in milliseconds.

B: Set acceleration/deceleration time for G01, in milliseconds.

C: Set acceleration/deceleration time for feedrate, in milliseconds.

2.11.5 G10 L994

- **Instruction format**

G10 L994 P__ Q__ H__ D__ I__ J__ K__ IP_ (Add lead-in/lead-out line feature)

Adds a lead-in or lead-out line in the next block. The lead-in line is added at the start point of the next block, while the lead-out line is added at the end point of the next block.

P: Specify as lead-in line or lead-out line.

0 or blank = Lead-in line

1 = Lead-out line

Note: When set as lead-in line, it means that the next block requires a lead-in line. The IP value determines the position where the lead-in line is added (i.e., the starting position of the next block). When set as lead-out line, it means that the lead-out line is added after the completion of the next block.

Q: Lead-in/lead-out line type.

0 or blank = Straight line

1 = Arc

H: Lead-in/lead-out line direction.

0 or blank = Left side of machining direction

1 = Right side of machining direction

D: G-code from the current point to the lead-in line start point.

0 or blank = G00 (rapid move)

1 = G01 (linear interpolation)

I: Length (or radius) of the lead-in/lead-out line.

For a straight line, specify the length of the line.

For an arc, specify the radius of the arc.

J: Angle of the lead-in/lead-out line.

For a straight line, specify the deflection angle.

For an arc, specify the included angle of the arc.

K: Overcut length for the lead-out line.

IP: Specifies the position for adding the lead-in line.

- **Function Introduction**

❖ Lead-in → Lead-out

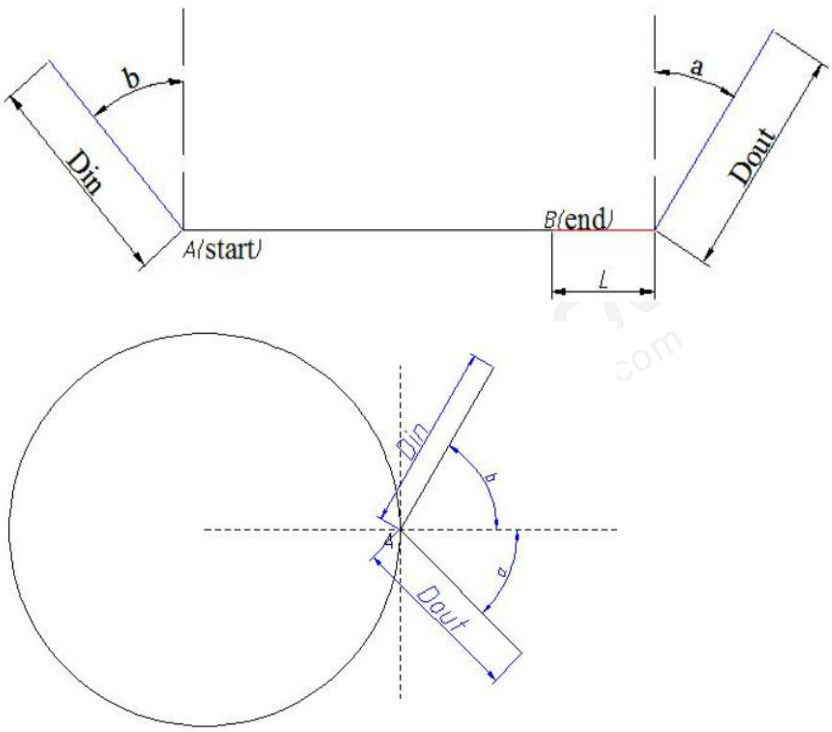


Figure 2.7.1: Schematic Diagram of Linear Entry and Exit

Din: Linear entry line, length specified by I.

Dout: Linear exit line, length specified by I.

b: Entry line angle, angle specified by J.

a: Exit line angle, angle specified by J.

L: Overcut line, length specified by K.

Line type: Straight line, determined by Q.

- **Program Example**

G00 X0. Z0.

G01 Z20. F2000.

G10 Z40. L994 P0 Q0 I10. J30.

G10 L994 P1 I10. K10. J60.

G01 Z80.

M30

Figure 2.7.2: Program Example - Machining Path

BC: Entrance Line with a length of 10.000mm.

EF: Exit Line with a length of 10.000mm.

DE: Overcut Line with a length of 10.000mm.

AB: Straight line segment from the end point before the entrance line to the starting point of the entrance line, the speed is determined by the letter.

(2) Entrance Line - Straight Line - Arc Exit;

Arc Entrance - Straight Line - Exit;

Arc Entrance - Arc Exit (similarly).

2.11.6 G10 L995

- **Instruction format**

G10 L995 P____ Q____

P: Set Operation Function

=1 Clear User Variables (Channel)

=101 Load User Variables (Channel)

=201 Save User Variables (Channel)

=2 Restore Mechanical Parameters to Factory Settings (Channel)

=102 Load Mechanical Parameters (Channel)

=202 Save Mechanical Parameters (Channel)

=3 Restore System Variables to Factory Settings (Channel)

=103 Load System Variables (Channel)

=203 Save System Variables (Channel)

=4 Clear Registers (Channel)

=104 Load Registers (Channel)

=204 Save Registers (Channel)

=11 Clear User Variables (Common)

=111 Load User Variables (Common)

=211 Save User Variables (Common)

=12 Restore Mechanical Parameters to Factory Settings (Common)

=112 Load Mechanical Parameters (Common)

=212 Save Mechanical Parameters (Common)

=13 Restore System Variables to Factory Settings (Common)

=113 Load System Variables (Common)

=213 Save System Variables (Common)

=14 Clear Registers (Common)

=114 Load Registers (Common)

=214 Save Registers (Common)

=19 Clear BUS Variables (Common)

=119 Load BUS Variables (Common)

=219 Save BUS Variables (Common)

=1001 Delete Current Program

=1002 Delete All Programs

Q: Specify Channel Number (When Q is not specified or specified as 0, it represents the current channel)

2.11.7 G10 L996

- **Instruction format**

G10 L996 P__ D__（Set the values of mechanical parameters）

P: Mechanical parameter number. P=0-9999 corresponds to MCM0000-MCM9999.

D: Setting value for the mechanical parameter.

D=-2^31~2^31

Note: This command takes effect during pre-processing. When executing this command, the internal variable corresponding to the mechanical parameter is changed, but the mechanical parameter itself is not modified.

The current functionalities that can be modified using the G10 L996 command are as follows：

Instruction	Function
G10 L996 P40 D___	Modify motion mode

Apologies for the mistake. The correct range for P is 0 to 9999

2.11.8 G10 L997

- **Instruction format**

G10 L997 P__ (Set the index of the master spindle)

P: Index of the master spindle

P=0-10 corresponds to Spindle 1Spindle 10

P=0 or 1, both indicate setting Spindle 1 as the master spindle.

2.11.9 G10 L998

- **Instruction format**

G10 L998 P__ D__ (Enable feedforward gain function)

P: Axis number for enabling feedforward gain function

P=1-32 corresponds to Axis 1Axis 32

P=0 or blank, indicates enabling for Axis 1

D: Feedforward gain percentage

D=0~10000

2.11.10 G10 L999

- **Instruction format**

G10 L999 P__ (Disable feedforward gain function)

P: Axis number for disabling feedforward gain function

P=1-32 corresponds to Axis 1Axis 32

2.11.11 G10 L1000

- **Instruction format**

G10 L1000 Axxx Bxxx Cxxx (Spindle Move by Distance, Non-stop Type G-Code, No Interpolation Cycle)

A: Spindle Specification

Range: 0~10

Unit: NULL

Explanation: If no letter A or 01=Spindle 1, 2=Spindle 2, 3=Spindle 3, ...

B: Distance Mode Setting

Range: 0~32

Unit: NULL

Explanation: 0=Current Spindle Command, 1=Feedback from Hardware Axis 1, 2=Feedback from Hardware Axis 2, ...

C: Distance Setting

Range: 0~+-2^31

Unit: PULSES

Explanation: Set the distance that the spindle needs to move.

2.12 G11, G12 Input/Output Point Control

- **Instruction format**

1. G11 P___& G11 P-___

G11 P___ specifies that the OUTPUT ___ should be turned ON.

G11 P-___ specifies that the OUTPUT ___ should be turned OFF.

When Reset is activated, all OUTPUTs that were turned ON by G11 P___ will be turned OFF.

- **Program Example**

N1 G11 P1 ;;; When executed, OUTPUT O1 is turned ON.

N2 G11 P-1 ;;; When executed, OUTPUT O1 is turned OFF.

Note: If the Reset button is pressed when executing N1, then at that moment OUTPUT O1 will be turned OFF.

2. G11 P1___ & G11P-1___

G11 P1___ specifies that OUTPUT ___ is turned ON.

G11 P-1___ specifies that OUTPUT ___ is turned OFF.

When Reset is triggered, the state of OUTPUTs that were turned ON by G11 P1___ remains unchanged.

- **Program Example**

N1 G11 P10001 ;;; When executed, OUTPUT O1 is turned ON.

N2 G11 P-10001 ;;; When executed, OUTPUT O1 is turned OFF.

Note: If the Reset button is pressed when executing N1, at that moment OUTPUT O1 will still be ON.

1. G11 P___ Q___

P: Specifies the OUTPUT point (range 1~2048). Positive values indicate turning ON the specified O point, while negative values indicate turning OFF the specified O point.

When P value is <10000, the G11 instruction clears the RESET state of the O point output.

When P value is >10000, the G11 instruction does not clear the RESET state of the O point output.

Q: Specifies the channel. (BIT00=1 for channel 1, BIT01 for channel 2, ...)

Q=-1 indicates all channels.

Q=0 or not specified indicates the current channel.

- **Program Example**

N1 G11 P10001 Q-1 ;;; When executed, OUTPUT O1 is turned ON for all channels.

N2 G11 P-10001 Q-1 ;;; When executed, OUTPUT O1 is turned OFF for all channels.

Note: If the Reset button is pressed when executing N1, at that moment OUTPUT O1 for all channels will still be ON.

For example, in a dual-channel system:

G11 P10 Q3 will turn ON OUTPUT O10 for both Channel 1 and Channel 2, and when reset, OUTPUT O10 will turn OFF.

G11 P-10 Q3 will turn OFF OUTPUT O10 for both Channel 1 and Channel 2, and when reset, OUTPUT O10 will remain OFF.

G11 P10010 Q3 will turn ON OUTPUT O10 for both Channel 1 and Channel 2, and when reset, the state of OUTPUT O10 will not change.

G11 P-10010 Q3 will turn OFF OUTPUT O10 for both Channel 1 and Channel 2, and when reset, the state of OUTPUT O10 will not change.

- **Instruction format**

2. G12 Pxxxx & G12 P-xxxx

G12 Pxxxx: When the specified INPUT xxxx is in the ON state, the next block is executed.

G12 P-xxxx: When the specified INPUT xxxx is in the OFF state, the next block is executed.

3. G12 Pxxxx Lxx & G12 P-xxxx Lxx

When the letter L has a value:

G12 Pxxxx Lxx: When a rising edge is detected on the specified INPUT xxxx, the next block is executed.

G12 P-xxxx Lxx: When a falling edge is detected on the specified INPUT xxxx, the next block is executed.

4. G12 Pxxxx Lxxx Bxxx

B represents the detection time, and an alarm will be triggered when the detection time is reached, in milliseconds.

Program format: G12 P__

The next block is executed only when the specified input point is in the ON state.

Example：N10 G00 X30. F1000

N20 G12 P08

N30 G00 X60

N40 G00 X100

N50 M30

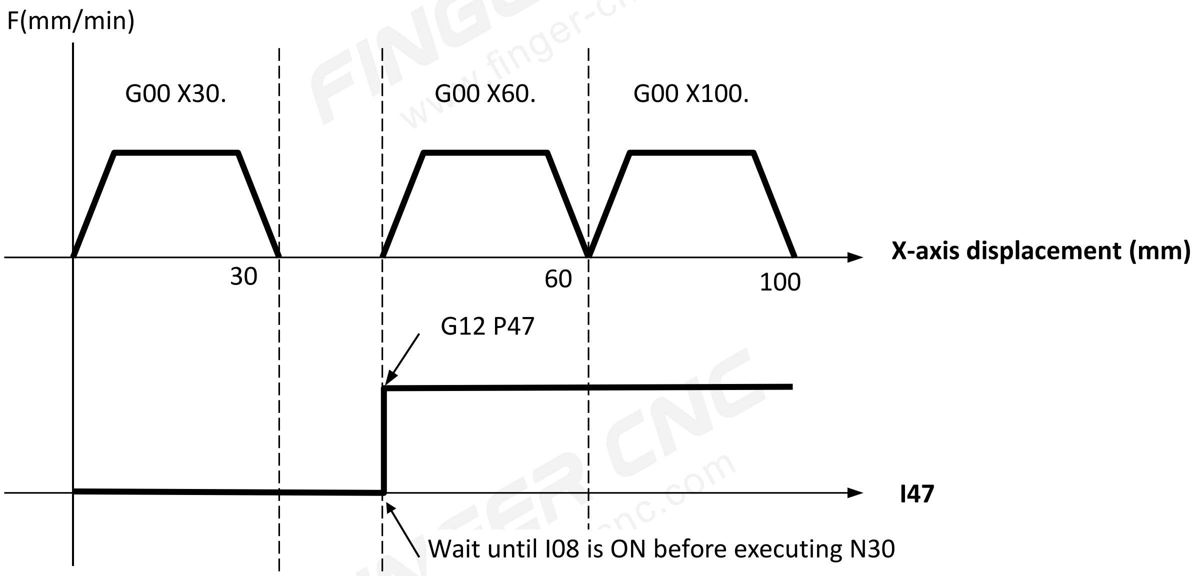


Figure 3-18

2.13 G12.1, G13.1 Polar Coordinate Interpolation

- **Function and Purpose**

Polar coordinate interpolation is a contour control method that converts programming instructions in Cartesian coordinate system into linear axis movement (tool movement) and rotational axis movement (workpiece rotation). It is primarily used in turning operations for facing cuts and grinding of camshafts.

- **Instruction Format**

G12.1 (G112): Enable polar coordinate interpolation

...

Perform linear and arc interpolation using the Cartesian coordinate system composed of linear axes and rotary axes (imaginary axes)

...

G13.1 (G113): Cancel polar coordinate interpolation

- **Detailed Explanation**

1. The axes for coordinate interpolation (linear axes and rotary axes) are pre-set through parameters. The G12.1 command is used to activate the polar coordinate interpolation mode. The two orthogonal axes, with the linear axis as the first axis and the orthogonal imaginary axis as the second axis, form the coordinate interpolation plane (polar coordinate interpolation plane).

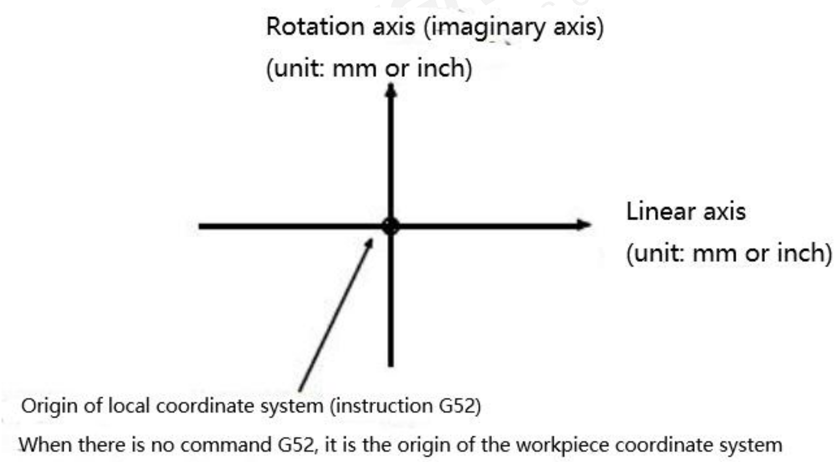


Figure 2.8.1 Polar Coordinate Interpolation Plane

2.

❖ The plane selection commands G17/G18/G19 used before the G12.1 command are cancelled and restored after the execution of the G13.1 command.

❖ After a system reset, the coordinate interpolation mode is cancelled and returns to the plane specified by G17/G18/G19.

❖ The second axis of the polar coordinate plane must be set as a rotational axis (imaginary axis).

❖ The positioning mode for the rotational axis of the second axis should be selected as the nearest positioning mode.

3. Values for Motion Command and Feedrate during Polar Coordinate Interpolation. Motion command value: The command units on the rotary axis and linear axis (mm/inch) are both set to zero when executing G12.1, meaning that polar coordinate interpolation starts with a rotation angle of 0; Feedrate: The feedrate specified by the F command represents the tangential velocity in the polar coordinate interpolation plane (Cartesian coordinate system).

4. G codes that can be used in polar coordinate interpolation:
G01: Linear interpolation
G02/G03: Circular interpolation (clockwise/counterclockwise)
G04: Dwell (pause)
G40/G41/G42: Cutter radius compensation (the system performs polar coordinate interpolation based on the compensated toolpath)
G65/G66/G67: User-defined macro command
G90/G91: Absolute/incremental programming
G98/G99: Feedrate per minute/feedrate per revolution

5. Circular Interpolation in Polar Coordinate Plane
For circular interpolation in the polar coordinate interpolation plane, the radius letter is determined based on the first axis (linear axis) of the plane. When the linear axis is the X-axis or an axis parallel to the X-axis, use IJ to specify. When the linear axis is the Y-axis or an axis parallel to the Y-axis, use JK to specify. When the linear axis is the Z-axis or an axis parallel to the Z-axis, use KI to specify. The arc radius can also be specified using R.

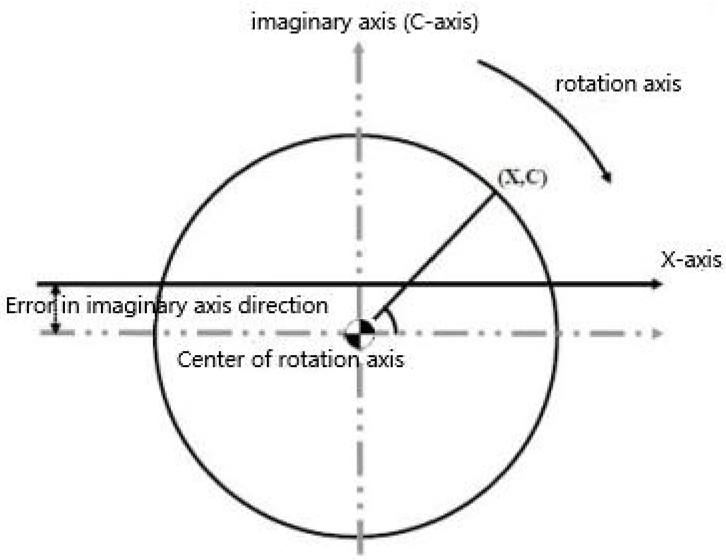
6. Coordinate Display: Actual coordinate values are displayed for the remaining coordinates within a program segment. Polar coordinate interpolation plane (Cartesian rectangular coordinates) displays the remaining coordinates.

7. Setting the Coordinate System in Polar Coordinates:
Before using G12.1, it's necessary to set a local coordinate system or workpiece coordinate system, with the center of rotation axis as the origin. In G12.1 mode, the

coordinate system must not be altered, and functions like G50/52/53/54/55/56/57/58/59, relative coordinate reset, etc., should not be used; otherwise, an alarm will be triggered.

8. Compensation in the Direction of the Imaginary Axis in Polar Coordinate Interpolation:

   In cases where there is an error in the direction of the imaginary axis in the polar coordinate interpolation plane, specifically when the center of rotation axis is not on the linear axis, you can use the compensation feature in the direction of the imaginary axis in polar coordinate interpolation. The system will automatically consider this compensation error before performing polar coordinate interpolation.



(XC)	The point in the XC plane (with the center of the rotation axis as the origin of the XC plane).
X	The X-axis coordinate value in the X-C plane
C	The coordinate value of the imaginary axis in the X-C plane
P	Error amount in imaginary axis direction

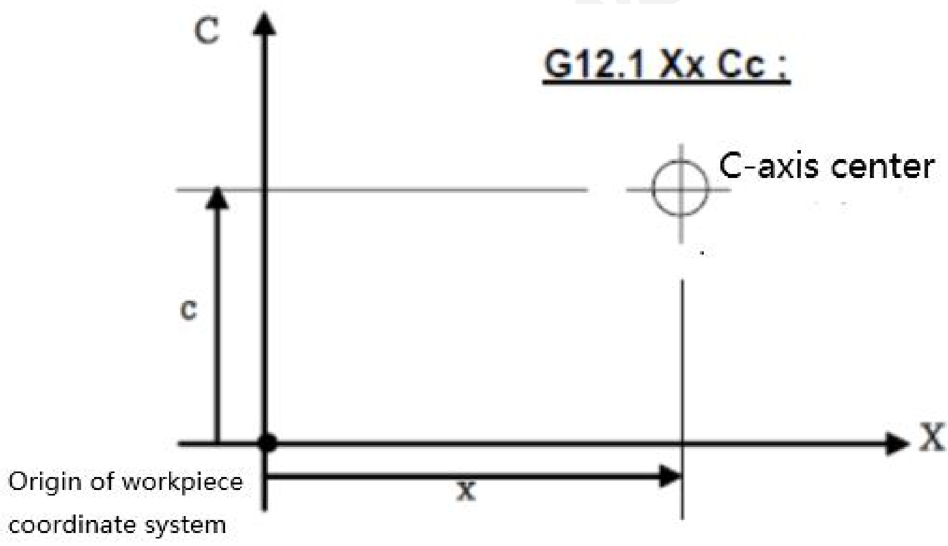
Figure 2.8.2 Polar Coordinate Interpolation

9. Coordinate system offset in polar coordinate interpolation mode:

   In polar coordinate interpolation mode, it is possible to use the work coordinate system offset, but the coordinate display shows the position coordinates before the offset.

   Setting the coordinate system offset can be done through parameter settings or using specific commands. Here are examples of command-based offset settings in polar coordinate interpolation mode:

G12.1 X_C_ ; (Polar coordinate interpolation based on X and C axes)
G12.1 Y_A_ ; (Polar coordinate interpolation based on Y and A axes)
G12.1 Z_B_ ; (Polar coordinate interpolation based on Z and B axes)



- **Limitations**

1. Limitations in changing coordinate systems
   In polar coordinate interpolation mode (G12.1), the coordinate system cannot be changed using commands such as G50, G52, G53, G54, G55, G56, G57, G58, G59, or relative coordinate reset commands.

2. Limitations in tool tip radius compensation
   In G41/G42 mode (tool tip radius compensation), switching between polar coordinate interpolation (G12.1/G13.1) and other modes is not allowed. The transition from G40 (canceling compensation) to G12.1/G13.1 can be performed.

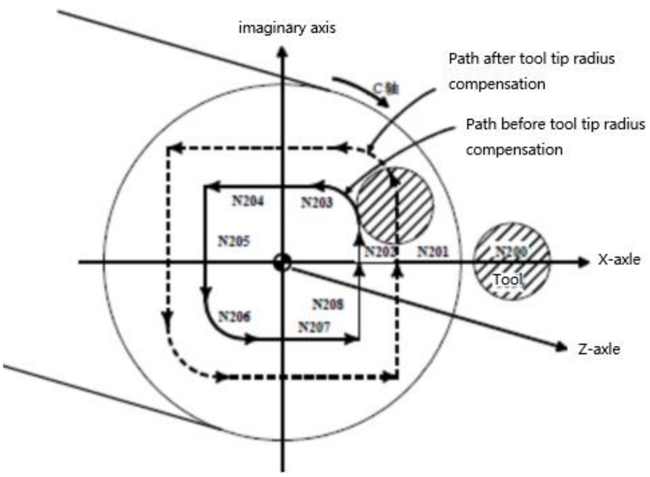
3. Program restart limitation
   During the execution of a program segment started with G12.1, the program restart (RS) command is not allowed to be set.

4. Automatic speed limitation
   In polar coordinate interpolation, the speed component of the rotational axis may exceed the maximum cutting feed rate range due to a decrease in polar radius. In such cases, the system internally limits the rotational axis speed and simultaneously

limits the linear axis speed to ensure safe operation.

- **Program Example**

Program Example for Polar Coordinate Interpolation in Cartesian Coordinates Based on the X-Axis (Linear Axis) and the Imaginary Axis.



X 轴为直径指定，C 轴为半径指定

```
O0001 ;
  .
  .
N010 T0101 ;
  .
  .
N0100 G90 G00 X120.0 C0 Z__ ;        向开始位置的定位
N0200 G12.1 ;                        开始极坐标插补
N0201 G42 G01 X40.0 F__ ;           ┐
N0202 C10.0 ;                       │
N0203 G03 X20.0 C20.0 R10.0 ;       │
N0204 G01 X-40.0 ;                  │
N0205 C-10.0 ;                      │  形状程序
N0206 G03 X-20.0 C-20.0 I10.0 J0 ;  ├─ （基于X轴和假想轴的笛卡尔坐标平面中的
N0207 G01 X40.0 ;                   │  值的程序）
N0208 C0 ;                          │
N0209 G40 X120.0 ;                  ┘
N0210 G13.1 ;                        取消极坐标插补
N0300 Z__ ;
N0400 X__C__ ;
  .
  .
N0900 M30 ;
```

2.14 G12.8 Three-Point Circular Command

- **Instruction format**

G12.8 X(U) Y(V) I J K D A B C F

X(U): Absolute/incremental coordinate of the third point (endpoint) along the X-axis of the arc.

Y(V): Absolute/incremental coordinate of the third point (endpoint) along the Y-axis of the arc.

D: When D = 0 or blank,

I: Absolute X-coordinate of the second point of the arc.

J: Absolute Y-coordinate of the second point of the arc.

When D = 1,

I: Incremental X-coordinate of the second point of the arc (relative to the first point, the starting point).

J: Incremental Y-coordinate of the second point of the arc (relative to the first point, the starting point).

A: Coordinate of the linear interpolation axis A.

B: Coordinate of the linear interpolation axis B.

C: Coordinate of the linear interpolation axis C.

- **Detailed explanation**

1. The use of PLC function to calculate a circular path using three points requires the rising edge of C0073 to be enabled for it to be effective.
2. The selected three points must not lie on the same straight line, otherwise the system will generate an alarm.
3. The selected three points must lie in the same plane.

2.15 G13.9 Quick Check of Variable Bit Status

- **Function and Purpose**

The G13.9 command is used to quickly check the status of any variable bit and perform corresponding actions based on the checked state. When the checked object's state satisfies the G13.9 instruction, the system immediately proceeds to the next program block. If the checked object's state does not satisfy the G13.9 instruction, the system remains at the G13.9 line, waiting for the checked object's state to meet the specified condition. This functionality is not affected by pre-buffering of the program and can dynamically respond to the status conditions of the specified object in real-time.

- **Instruction format**

G13.9 P___ A____ B___

P___: Specifies the address of the object to be checked.

When P has more than 6 digits, the higher digits represent the channel number. In other words, P=xx yyy yyy, where xx represents the channel number. When xx=0, it represents the current channel.

For example: P=1 represents Channel 1...; yyy yyy represents the address within the specified channel.

A___: Specifies the position of the object to be checked within the corresponding address. A variable address can store 32 bits of state. Once the variable address is determined based on the object to be checked, the specific bit position of the object within that variable can be determined.

A value is positive, the program will proceed only if the object's status is ON.

A value is negative, the program will proceed only if the object's status is OFF.

B___: Specifies the address type of P.

0 or blank: Absolute address

1: User variable of the channel

2: Mechanical parameter of the channel

3: System variable of the channel

4: Register of the channel

10: Common BUS data

11: Common user variable

12: Common mechanical parameter

13: Common system variable

14: Common register

- **Example**

G13.9 P10 A1 checks the status of BIT01 in the current channel's variable 10. If the status is ON, the program proceeds to the next block immediately. If the status is OFF, the system stays in the G13.9 block and waits until the status becomes ON before proceeding.

N1 G13.9 P10 A1 ;;;; Check BIT01 in variable 10, proceed to the next line if ON, wait if OFF.

2.16 G15 Servo Spindle Positioning Command

- **Instruction format**

G15 R____

R: Set the positioning point of the servo spindle.

- **Detailed explanation**

(1) This command is only applicable to servo spindles.

(2) The setting range is from 0.000° to 359.999°.

- **Example**

N1 G15 R90.000 < ------- Spindle positioned at 90 degrees.

2.17 G16 Cylindrical Interpolation Command

- **Function and Purpose:**

The G16 command is used to convert the movement amount of a rotary axis specified by angular instructions into the linear distance on the external surface, enabling linear or circular interpolation with other axes. After interpolation is completed, this distance is converted back into the movement amount of the rotary axis.

- **Instruction Format**

G16 Axxxx.xxx: Sets axis A as the cylindrical interpolation axis, where xxxx.xxx represents the cylindrical radius value.

G16 Bxxxx.xxx: Sets axis B as the cylindrical interpolation axis, where xxxx.xxx represents the cylindrical radius value.

- **Note**

1. When xxxx.xxx is non-zero, the cylindrical interpolation function is activated.
2. When xxxx.xxx = 0, the cylindrical interpolation function is terminated.
3. The plane selection using G17-G19 codes determines the linear axis specified as the rotary axis for that plane. The plane selection is shown in Figure 2-16 (taking axis A as the rotary axis, for example).
4. The feed rate specified in cylindrical interpolation mode corresponds to the speed on

the unwrapped surface of the cylinder.

In cylindrical interpolation mode, G02/G03 can only be used with the R parameter to specify the arc radius; the I, J, K parameters cannot be used.

Example: Arc interpolation command (performing arc interpolation with Z-axis and C-axis).

G17 Y_______________A_____

G02（03）Y_______________A_____R_____

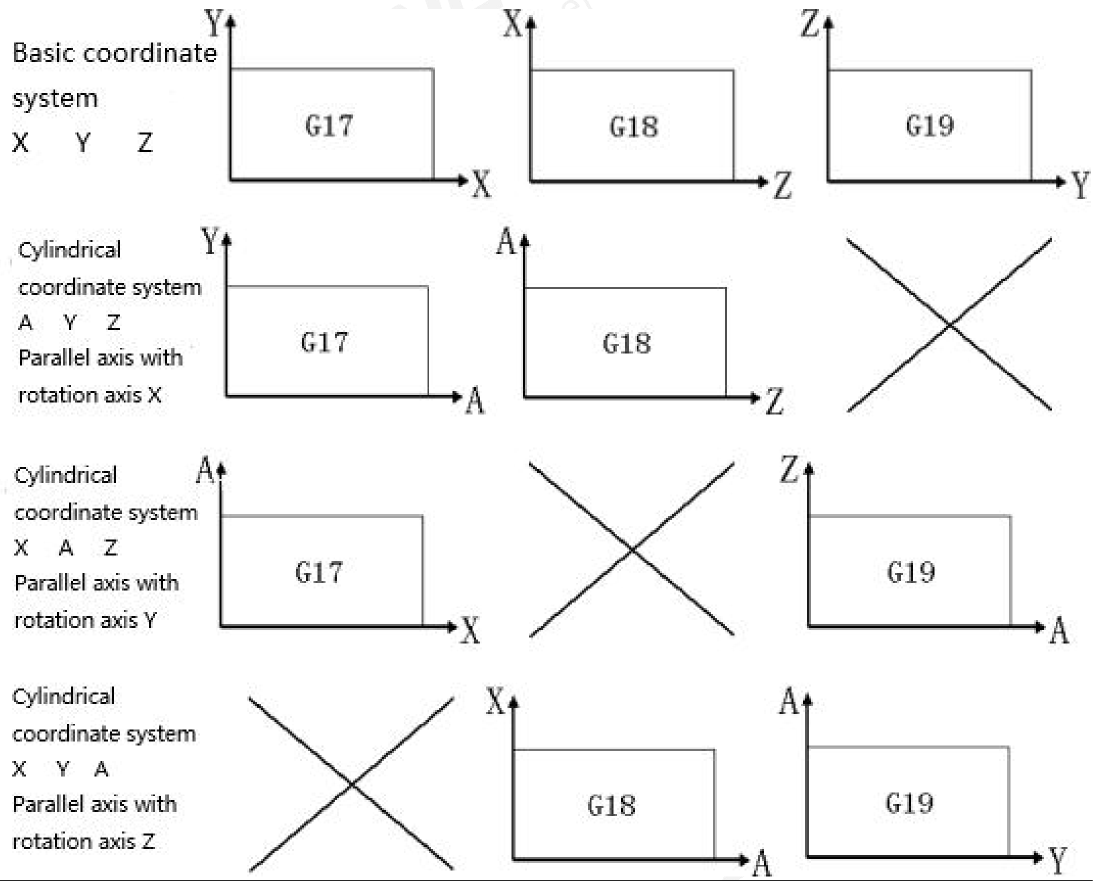


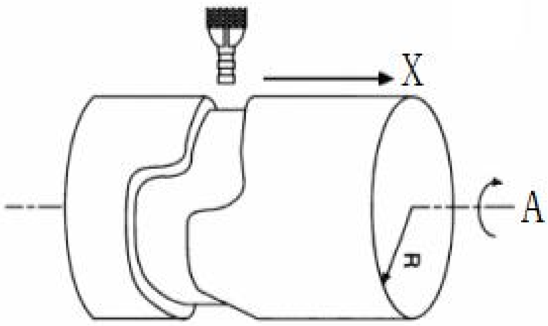
Figure 2-16: Cylindrical Interpolation - Plane Selection.

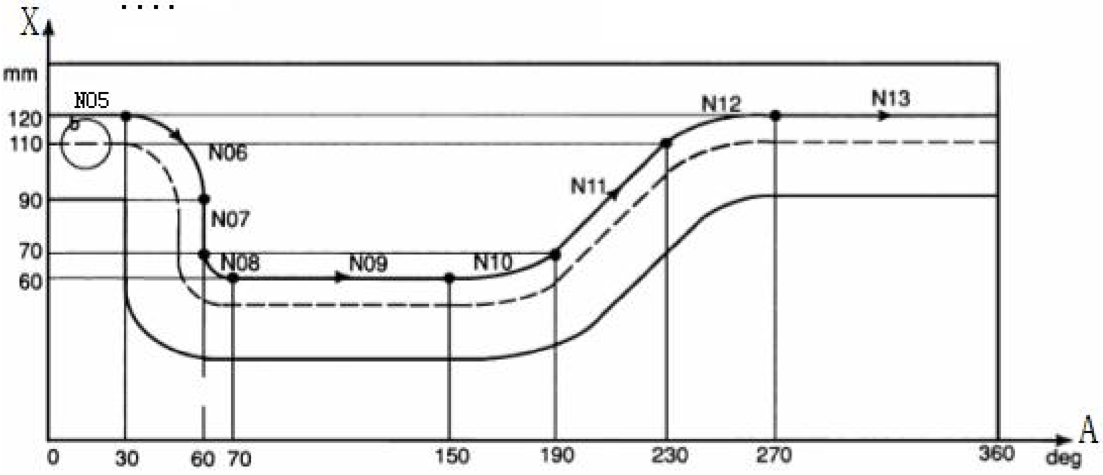
5. In cylindrical interpolation mode, tool tip radius compensation can be used. To perform tool radius compensation in cylindrical interpolation mode, any ongoing tool compensation mode should be cancelled before entering the cylindrical interpolation mode. Then, tool radius compensation can be started and ended within the cylindrical interpolation mode.
6. If cylindrical interpolation mode is initiated while tool radius compensation is already applied, arc interpolation may not be correctly performed within the cylindrical interpolation mode.

7. In cylindrical interpolation mode, the movement amount of the rotational axis specified by angular commands is internally converted into linear axis distance on the outer surface for linear or arc interpolation with other axes. After interpolation, this distance is converted back into angles. The movement amount is rounded to the smallest input increment unit for this conversion. When the cylindrical radius is small, the actual movement amount may differ from the specified amount, but this error does not accumulate.
8. The cylindrical interpolation function ends when a reset is performed.
   The cylindrical interpolation axis should be set as the rotational axis, and only one rotational axis can be specified.

- **Program Example**

```
O001
N01 G00 X100. A0
N02 G16 A20.
N03 G17 X0 A0
N04 G42 X120. F500
N05 G01 A30.
N06 G02 X90. A60. R30.
N07 G01 X70.
N08 G03 X60. A70. R10.
N09 G01 A150.
N10 G03 X70. A190. R75.
N11 G01 X110. A230.
N12 G02 X120. A270. R75.
N13 G01 A360.
N14 G40 X100.
N15 G16 A0
....
```





2.18 G17、G18、G19 Flat Planning

- **Function and Purpose**

These commands are used to select the control plane or the plane in which arcs are to be interpreted.

- **Detailed Explanation**

1. The horizontal and vertical axes corresponding to the G17, G18, G19 planes are specified by the I, J, K commands.

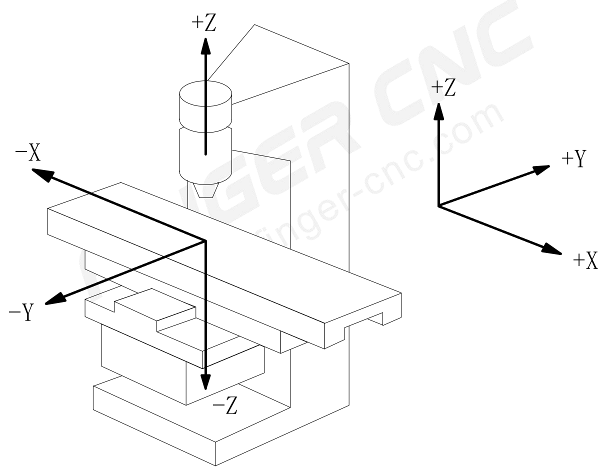
Command	Horizontal Axis	Vertical Axis
G17	I	J
G18	K	I
G19	J	K

Table 2.9.1 G17/G18/G19 Plane Selection Table

2. The axes corresponding to I, J, and K are set by Mcm1881, Mcm1882, and Mcm1883, respectively. The set axes are used as the default horizontal and vertical axes for G17, G18, and G19.

   For example, if Mcm1881=1 (X-axis as the base axis for I),
   Mcm1882=2 (Y-axis as the base axis for J),
   Mcm1883=3 (Z-axis as the base axis for K),
   then the default axes for the G17, G18, and G19 planes can be specified as follows:

   G17 or G17 Xxx Yxx - Horizontal axis X-axis, vertical axis Y-axis
   G18 or G18 Zxx Xxx - Horizontal axis Z-axis, vertical axis X-axis
   G19 or G19 Yxx Zxx - Horizontal axis Y-axis, vertical axis Z-axis



2.19 G20、G21 Imperial/Metric Measurement Mode

- **Function and Purpose**

The G20 and G21 codes are used to select the unit of measurement for input data, specifying whether it is in Imperial (inch) or Metric (mm) units.

- **Instruction format**

G20 - System measurement is in Imperial units (inch).

G21 - System measurement is in Metric units (mm).

2.20 G28~G30

- **Function and Purpose**

1. G28 Instruction
   Specifies that the axis returns to the first reference point (origin) using rapid feedrate (G00) (whether each axis is independent is to be determined, as it depends on the controller).
2. G29 Instruction
   Specifies that the axis independently returns to the specified position after passing through the intermediate point of G28 or G30, using rapid feedrate.
3. G30 Instruction
   Specifies that the axis returns to the second, third, or fourth reference point using rapid feedrate (G00).

2.20.1 G28

- **Instruction format**

G28 IP: Automatic Reference Point Return

IP: Instruction to specify the intermediate point, either in absolute or incremental format. The address codes for each axis can be omitted or completely omitted. Omitting a specific axis means that the axis will not return to the reference point, and omitting all axes means that the tool will not move.

MCM2000: G28 X-axis First Reference Point Setting

MCM2001: G28 Y-axis First Reference Point Setting

MCM2002: G28 Z-axis First Reference Point Setting

MCM2003: G28 A-axis First Reference Point Setting

MCM2004: G28 B-axis First Reference Point Setting

MCM2005: G28 C-axis First Reference Point Setting

......

G28 X___: Axis X return to reference point.

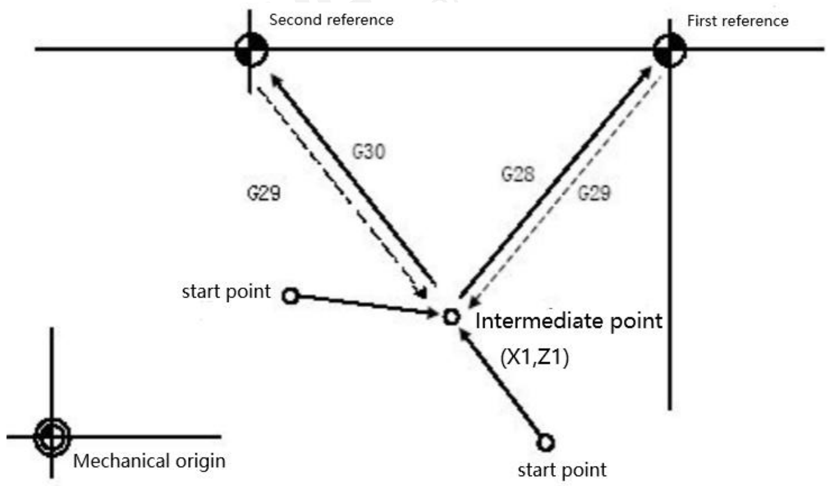
G28 X___ Y___: Axes X and Y return to reference point.

- **Note**

1. The position of the first reference point is set by the MCM parameters [G28 First Reference Point] for X and Z axes.
2. The values of X and Z in the instruction format indicate the intermediate points that the machine should pass through in the corresponding axes. If the G28 instruction is standalone, the tool will automatically return to the reference point position specified by the X and Z values in the MCM parameters. If the instruction includes both X and Z, the tool will automatically return to the intermediate points specified by the X and Z values in the MCM parameters before returning to the reference point position.
3. Tool compensation must be canceled before executing the G28 instruction.

- **Program Example**

G28 X1.0 Z1.0; // Return to the intermediate point (X1.0, Z1.0) for X and Z axes.



2.20.2 G29

1. The G29 command is used after using G28 to quickly move to a specified position via the intermediate point. It is important to note that the G29 command should not be used alone. Instead, it relies on the intermediate point specified in the preceding G28 command. Therefore, the G28 command must be executed before using the G29 command.
2. When using absolute value instructions, X, Y, and Z are the absolute coordinates of the target point. When using incremental value instructions, U, V, and W represent the incremental distances from the intermediate point to the target point.

- **Instruction format**

G29 IP__

IP: Specifies the coordinate value instruction for the return position, either in absolute or incremental values.

If IP is not specified, no action is taken.

- **Additional notes**

1. When executing the G29 command, the intermediate point position for the G29 return is based on the last executed G28 or G30 intermediate point. If no intermediate point is set in the preceding G28 or G30 commands, the G29 command will directly move to the position specified in the preceding G28 or G30 command.
2. In single block mode with intermediate point settings, executing G28, G30, or G29 commands will pause at the intermediate point. Pressing the start button will continue the execution to the reference point position.

- **Example 1**

N1 G00 X10.0 Z10.0; // Axes move to the specified position

N2 G28 X-10.0 Y-10.0 Z-10.0; // Move to the intermediate point (X-10.0, Y-10.0, Z-10.0), then to the reference point

...

N3 G29; // Stay at the current position

...

- **Example 2**

N1 G00 X10.0 Z10.0; // Axes move to the specified position

N2 G28 X-10.0 Y-10.0 Z-10.0; // Move to the intermediate point (X-10.0, Y-10.0, Z-10.0), then to the reference point

...

N3 G29 X30.0 Y20.0 Z10.0; // Axes move through the intermediate point, then to the specified position (X30.0, Y20.0, Z10.0)

...

2.20.3 G30

Similar to G28, the G30 command is used to specify that the axes return to the second reference point using rapid traverse.

- **Instruction format**

P2: Return to the second reference point.

P3: Return to the third reference point.

P4: Return to the fourth reference point.

IP: Instruction to specify the return position, either as absolute values or incremental values.

This command is used in a similar way as G28, but the coordinates of the reference points are set by MCM parameters.

- **Additional Explanation**

MCM2040: Set the second reference point for the X-axis in G30

MCM2041: Set the second reference point for the Y-axis in G30

MCM2042: Set the second reference point for the Z-axis in G30

MCM2043: Set the second reference point for the A-axis in G30

MCM2044: Set the second reference point for the B-axis in G30

MCM2045: Set the second reference point for the C-axis in G30

MCM2080: Set the third reference point for the X-axis in G30

MCM2081: Set the third reference point for the Y-axis in G30

MCM2082: Set the third reference point for the Z-axis in G30

MCM2083: Set the third reference point for the A-axis in G30

MCM2084: Set the third reference point for the B-axis in G30

MCM2085: Set the third reference point for the C-axis in G30

MCM2120: Set the fourth reference point for the X-axis in G30

MCM2121: Set the fourth reference point for the Y-axis in G30

MCM2122: Set the fourth reference point for the Z-axis in G30

MCM2123: Set the fourth reference point for the A-axis in G30

MCM2124: Set the fourth reference point for the B-axis in G30

MCM2125: Set the fourth reference point for the C-axis in G30

MCM2280: X-axis reference point movement speed

MCM2281: Y-axis reference point movement speed

MCM2282: Z-axis reference point movement speed

MCM2283: A-axis reference point movement speed

MCM2284: B-axis reference point movement speed

MCM2285: C-axis reference point movement speed

2.21 G31

2.21.1 G31 Skip Stop

- **Function and Purpose**

1. When a specified standard input signal or fast input signal is detected by the system, the system quickly stops the motion of the specified axis or all axes according to the requirements.
2. Allows switching between 6 sets of probe signals. SYS4540 to SYS4547 are parameters for G31 Group 1, SYS4550 to SYS4557 are parameters for G31 Group 2, SYS4560 to SYS4567 are parameters for G31 Group 3, and so on.

- **Instruction format**

G31 X/Y/Z__ P__ Q__ H__ E__ L__ D__ F__

X/Y/Zxxxx:

Definition: Specifies the target position for the axis of motion.

Range: -99999999 to 99999999

Unit: mm

Description: =0 means the axis does not move.

<> Blank indicates interpolating motion along the set target position.

P__:

Definition: Sets the probe input index.

Range: 0~16384

Explanation: = Blank, the system defaults to using the 5th pin of the MPG interface as the probe signal input.

<> Blank, the system uses the specified input as the probe signal.

=10000: Indicates using Fast Input 0 from the MPG interface as the external interrupt signal for G31.

=10001: Indicates using Fast Input 1 from the MPG interface as the external interrupt signal for G31.

=0~511: Indicates using the Input from the standard I/O board interface as the external interrupt signal for G31.

=16384: Resets the axis pre-position coordinates without motion or signal detection.

(Generally used in CNC programs when switching from spindle axis to servo axis and requiring repositioning, or when the system switches back from JOG mode to CNC execution mode and requires the axis positions to continue executing based on the specified program coordinates.)

Q__:

Definition: Sets the probing axis, i.e., the axis that should stop when a probe signal is encountered.

Range: 0~1031

Explanation: = Blank, all axes stop.

=N, the x-axis (0=X axis, 1=Y axis, ...) is designated as the probing axis, and when a probe signal is encountered, the axis stop command is issued.

(Q031: Regardless of whether the Q value specifies incremental coordinates or absolute coordinates, when the G31 signal is received, the remaining unexecuted coordinate movement distance is compensated in the next corresponding axis movement segment to ensure the final position is correct; Q10001031: If the specified axis is in incremental mode, when the G31 signal is received, the axis stops at the current position, and the system no longer compensates for the remaining G31 incremental commands in the next corresponding axis movement segment; if it is an absolute coordinate command, it behaves the same as Q0Q31.)

H__:

Definition: G31 group selection.

Range: 0~6

Explanation: Used to select the G31 probe signal group.

=0 or 1: Select G31 group 1.

=2: Select G31 group 2.

=3: Select G31 group 3.

...

E__:

Definition: Synthetic vector speed for non-probing axes to continue moving after encountering a probe signal.

Range: 0~99999999

Unit: mm/min

Explanation: = Blank: The system continues to move the remaining distance of non-probing axes at the synthetic vector speed specified in the current segment.

<> Blank: The system interpolates the remaining distance of non-probing axes at the speed specified by the letter.

L__:

Definition: During instruction execution, setting the output O point INDEX when a probe signal is encountered.

Range: 0~+511

Explanation: =0 or blank: The system defaults to not outputting the O point.

<> Blank: The system outputs the specified O point and continues until the specified time specified in MCM7229 is reached or the G31 instruction ends (the O point will also be closed if the next segment is G31). Note: The letter L only takes effect when the Q instruction is present.

D__:

Definition: Setting the INDEX of the O point to be output when the G31 instruction is entered.

Range: 0~+511

Explanation: =0 or blank: The system defaults to not outputting the O point.

<> Blank: The system outputs the specified O point and closes it when the

G31 signal is encountered or after the G31 instruction is completed.

Note:

1. The letter D only takes effect when the Q instruction is present.

2.If the instruction is completed and no G31 signal has entered, the system automatically closes the output of the O point.

3.This function is mainly used in the structure of G31 signals for spring machine cylinder types, where the G31 signal is detected through cylinder actions.

F__:

Definition: Specifies the speed of the motion axis's composite vector.

Range: -99999999~99999999

Unit: mm/min

Explanation: =Blank: The system uses the speed specified in the previous motion segment as the speed of the composite vector.

<> Blank: The system uses the speed specified by the letter as the speed for the axis's composite vector interpolation.

- **Program Example**

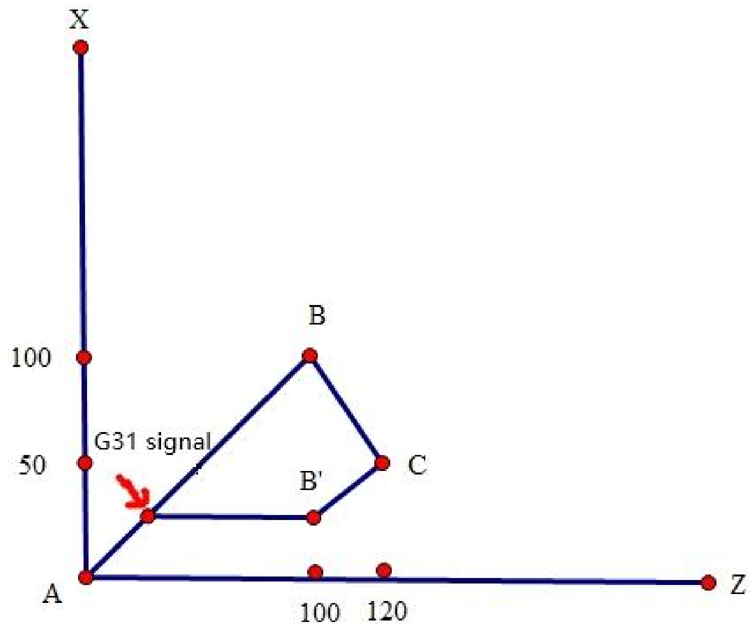
G00 X0. Z0.

G31 P1 X100. Z100. H1 Q2 F100. L1 D2 E10000.

G01 X50. Z120. F1000.

M30

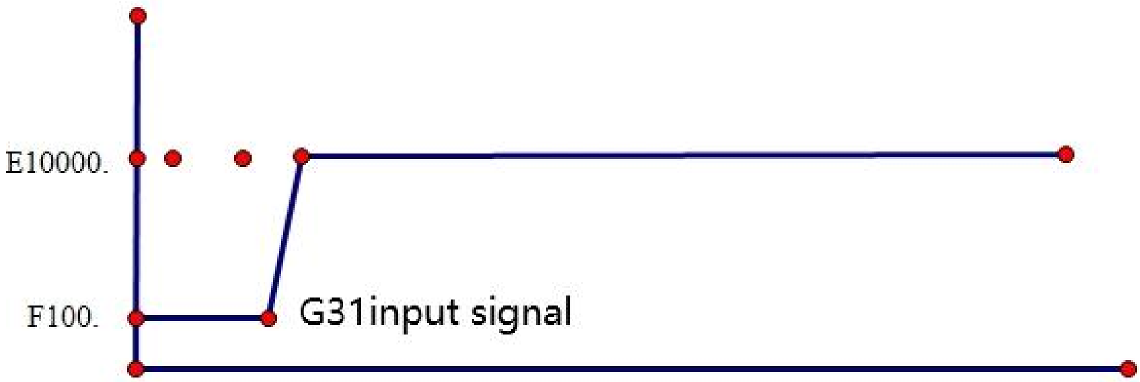
Motion trajectory analysis：



When the G31 signal is not entered: A -> B -> C

When the G31 signal is entered: A -> G31 signal point -> B' -> C

Velocity profile analysis:



When the G31 signal is not entered: The axis velocity is set to F100.

When the G31 signal is entered: The Z-axis stops, and the X-axis continues to feed at a velocity of E10000.

2.21.2 G31 Coordinate Reinitialization Command

- **Function and Purpose**

1. G31 P16384 X0/Y0/Z0... - Programmed coordinate reinitialization (if no axis is specified, it defaults to reinitializing all axes).
2. G31 P16384 X0/Y0/Z0...

   Definition: Reinitializes the programmed coordinates for the specified axis(es) without movement or signal detection (if no axis is specified, it defaults to reinitializing all axes).

- **Application Example**

1. G00 X0. Z0.

   G01 X100. Z100. F10000.

   M12; X-axis switched to spindle

   M03 S1000.

   G01 Z-20. F100.

   M05;

   M13; X-axis switched to servo axis

   G31 P16384 X0. // Reinitializing the programmed coordinate for X-axis to ensure the correct endpoint for the next motion segment (If the coordinate before switching to servo axis is 100., this command is not necessary as there is no additional movement beyond the specified instructions and the coordinate remains unchanged)

   G01 X120. Z120. F3000.

   M30

2. G00 X0. Z0.

   G01 X10. Z10. F30000.

   M01; Pause, switch to JOG or handwheel mode, only move X-axis manually

   G31 P16384 X0. // When switching back to CNC execution mode, reinitialize the X-axis coordinate to ensure correct execution of the next motion segment (If the coordinate after JOG stop is 10., this command is not necessary as there is no additional movement beyond the specified instructions and the coordinate remains unchanged).

   G01 X100. Z100. F30000.

   M30

2.22 G33 Thread Cutting

- **Instruction format**

G33 Z_ F_

Z: Represents the Z-axis coordinate of the cutting endpoint in absolute mode (G90)

Represents the axial length of the thread cutting in incremental mode (G91)

F: Represents the thread pitch or lead

2.23 G34 Circular Hole Pattern

- **Function and Purpose**

The G34 command is used to create a circular hole pattern. It drills a specified number of holes evenly distributed on a circle with a center point defined by X and Y coordinates. The circle has a radius of R. The drilling starts at an angle θ from the X-axis. The drilling actions follow a standard fixed cycle, and all movements between hole positions are performed in rapid positioning (G00) mode. Data is not saved after the G34 command is executed.

- **Instruction Format**

G34 X_ Y_ I_ J_ K_

X_: X-coordinate of the center position of the hole pattern, affected by G90/G91.

Y_: Y-coordinate of the center position of the hole pattern, affected by G90/G91.

I_: Radius (r) of the circle, specified as a positive value according to the selected unit.

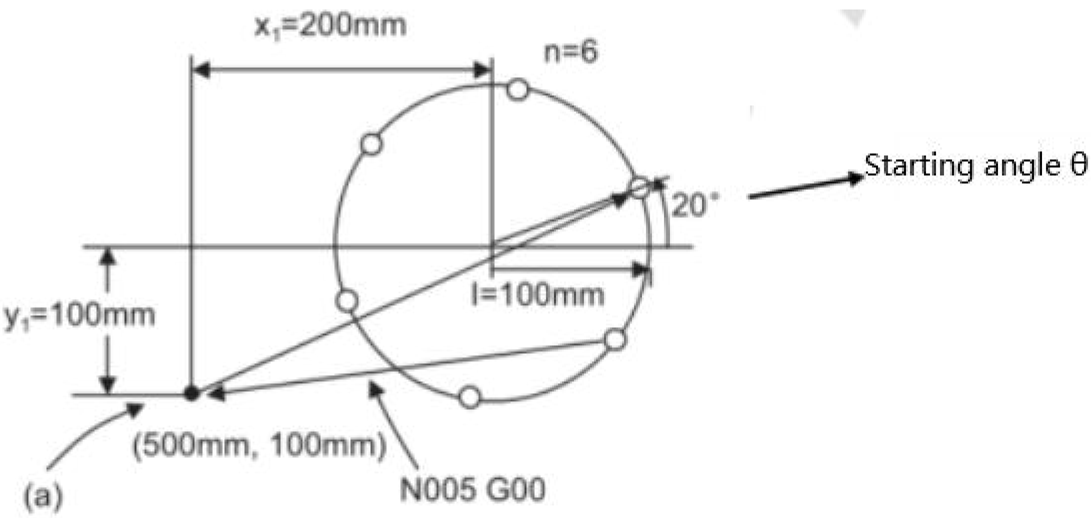
J_: Starting angle (θ) of the first drilling point. Positive values indicate a counterclockwise direction. If programmed without a decimal point, the value is in units of 0.001 degrees. If programmed with a decimal point, the value is in units of 1 degree.

K_: Number of holes (n) to be drilled. Valid range is 1 to 9999. The value 0 is not allowed. Positive values indicate counterclockwise rotation, while negative values indicate clockwise positioning. Decimal values are not valid.

- **Program Example**

1. N001 G90 G00 X500.Y100.Z10.
2. N002 G91 G99 G81 Z-10.000 R5.000 F200 (In B1 system, R direction is from the initial point to the bottom of the hole)
3. N003 G34 X200.000 Y100.000 I100.000 J20.000 K6
4. N004 G80
5. N005 G90 G00 X500.000 Y100.000

Drilling Illustration:



2.24 G35 Angle Linear Hole Cycle

- **Function and Purpose**

Starting from the position specified by X and Y, drill holes at intervals of d in the direction specified by angle θ, drilling a total of n holes. The drilling actions at each hole position follow a standard fixed cycle. The movements between hole positions are done in G00 mode. No data is saved after the G35 command is executed.

- **Instruction format**

G35 X_ Y_ I_ J_ K

X_Y_: Specifies the starting point coordinates, affected by G90/G91.

I_: Interval distance d. Depending on the input unit, when d is negative, drilling is done in the point symmetric direction with respect to the starting point.

J_: Angle θ. Defines the counterclockwise direction as the positive direction. When programming without a decimal point, the unit is 0.001 degrees. When programming with a decimal point, the unit is 1 degree.

K_: Number of holes n. The specified range can be from 1 to 9999, including the starting point.

- **Program Example**

N001 G90 G00 X-200.Y-100.

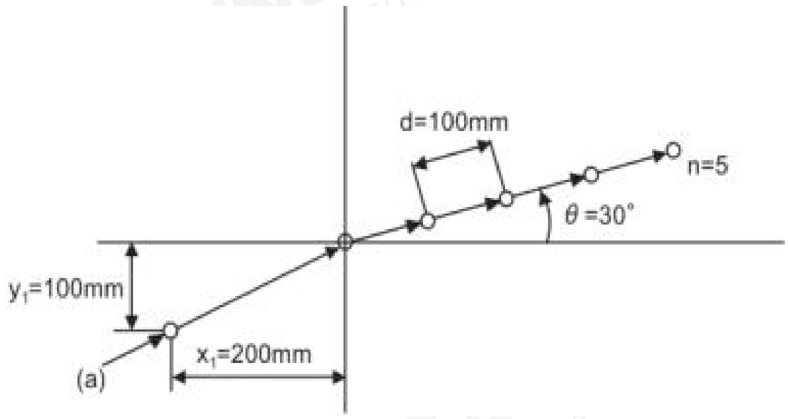
N002 G00 Z10.

N003 G91 G99 G81 Z-10.000 R5.000 L0 F100

N004 G35 X200.000 Y100.000 I100.000 J30.000 K5

N005 G80

Drilling Illustration：



2.25 G36 Arc Hole Cycle

- **Function and Purpose**

The G36 command is used to drill a series of holes at regular intervals along an arc on a circle with a center specified by X and Y coordinates. The arc starts at a point defined by an angle θ with respect to the X-axis, and the holes are drilled with a spacing of Δθ degrees. The drilling action for each hole follows a standard fixed cycle. All movements between hole positions are performed in G00 mode. No data is saved after the G36 command is executed.

- **Command Format**

G36 X_ Y_ I_ J_ P_ K_

X_Y_: Specifies the center coordinates of the arc, affected by G90/G91.

I_: Specifies the radius of the arc, given as a positive value depending on the input unit.

J_: Specifies the initial drilling point angle θ. The counterclockwise direction is defined as the positive direction. When programming without a decimal point, the unit is 0.001 degrees. When programming with a decimal point, the unit is 1 degree.

P_: Specifies the angular interval Δθ. A positive value indicates counterclockwise rotation, while a negative value indicates clockwise drilling. When programming without a decimal point, the unit is 0.001 degrees. When programming with a decimal point, the unit is 1 degree.

K_: Specifies the number of holes n to be drilled. The specified range is from 1 to 9999.

- **Program Example**

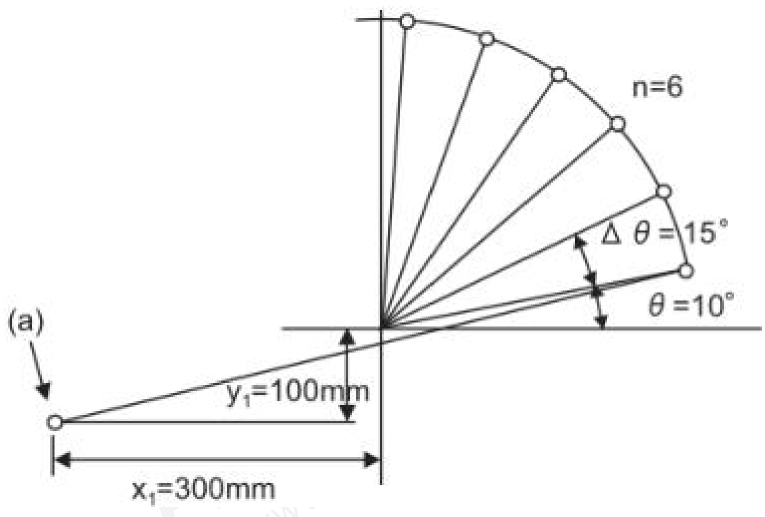
N001 G90 G00 X-300.Y-100.

N002 G00 Z10.

N003 G91 G99 G81 Z-10.000 R5.000 F100

N004 G36 X300.000 Y100.000 I300.000 J10.000 P15000 K6

N005 G80



2.26 G37.1 Checkerboard Hole Cycle

- **Command Explanation**

The G37.1 command is used to drill a series of holes in a checkerboard pattern starting from a specified position defined by X and Y coordinates. The holes are drilled along the X-axis with a parallel spacing of Δx, forming nx grid points. The drilling action for each hole follows a standard fixed cycle. All movements between hole positions are performed in G00 mode. No data is saved after the G37.1 command is executed.

- **Instruction format**

G37.1 X_ Y_ I_ J_ P_ K_

X_Y_: Specifies the starting point coordinates, affected by G90/G91.

I_: Specifies the spacing Δx along the X-axis. Depending on the input unit, a positive value indicates spacing in the positive direction from the starting point, while a negative value indicates spacing in the negative direction.

P_: Specifies the number of grid points nx along the X-axis. The specified range is from 1 to 9999.

J_: Specifies the spacing Δy along the Y-axis. Depending on the input unit, a positive value indicates spacing in the positive direction from the starting point, while a negative value indicates spacing in the negative direction.

K_: Specifies the number of grid points ny along the Y-axis. The specified range is from 1 to 9999.

- **Program Example**

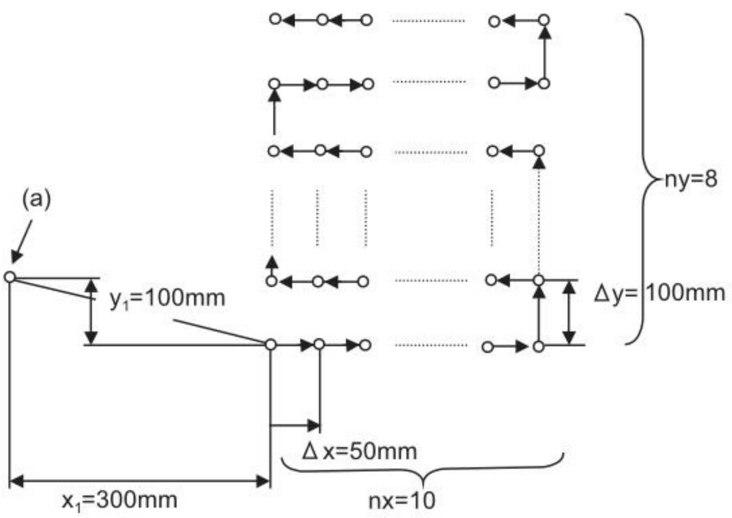
N001 G90 G00 X-300.Y100.

N002 G00 Z10.

N003 G91 G99 G81 Z-10.000 R5.000 F300

N004 G37.1 X300.000 Y-100.000 I50.000 P10 J100.000 K8

N005 G80



- **Notes**

1. The first positioning point before the special fixed cycle will perform the drilling action (point "a" in the diagram).
2. The R point of the drilling should be lower than the initial point, otherwise an alarm will be triggered.

2.27 G40, G41, G42 Tool Radius Compensation

- **Function and Purpose**

During arc cutting and taper cutting, there is a discrepancy between the programmed shape and the actual machined shape due to the circularity of the tool tip. The tool radius compensation function automatically calculates this error and applies compensation by setting the tool radius.

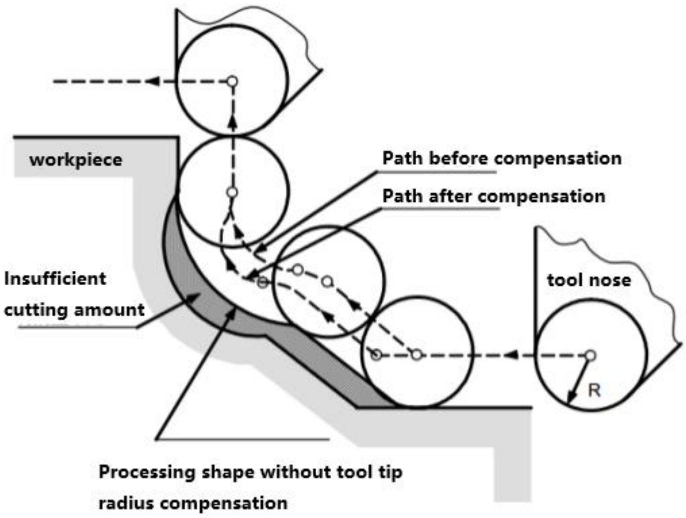


Figure 2.13.1 Tool Tip Path with Tool Radius Compensation

- **Instruction format**

T__ Call tool number for compensation

G41 (G42) X/U_ Z/W_ Compensation settings

G40 Cancel compensation

Note: When using tool radius compensation, a tool number must be specified.

- **Detailed Explanation**

1. Tool Offset Range:

G40: Cancels tool compensation and follows the programmed path without any offset.

G41: Moves to the left side of the programmed path in the feed direction.

G42: Moves to the right side of the programmed path in the feed direction.

The tool offset is on the opposite side of the workpiece.

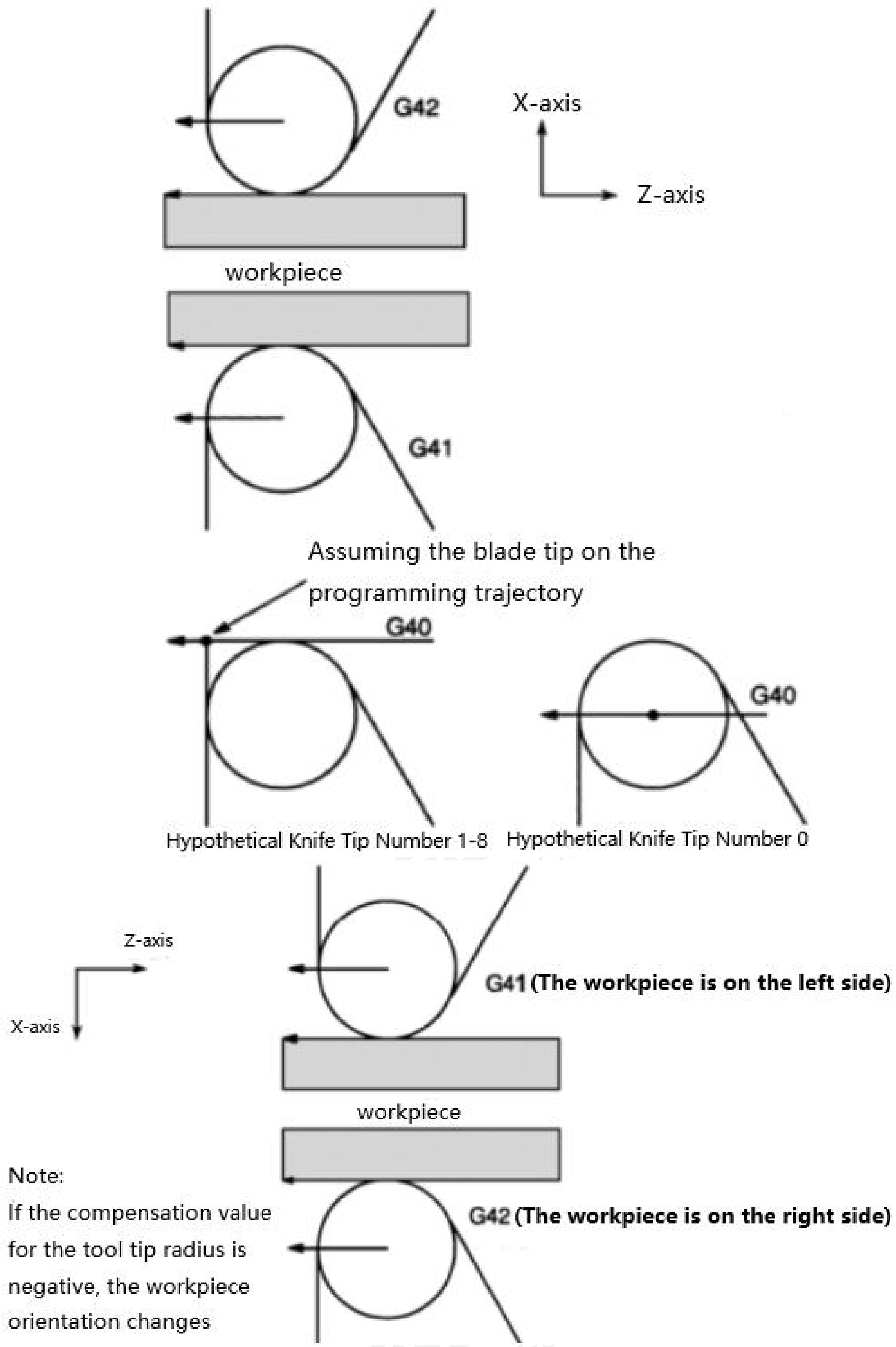


Figure 3.13.1

2. Tool Tip:

The imaginary tool tip is a theoretical point that does not physically exist. It is difficult to precisely align the actual tool tip radius center at the starting or reference position. Therefore, an imaginary tool tip is used.

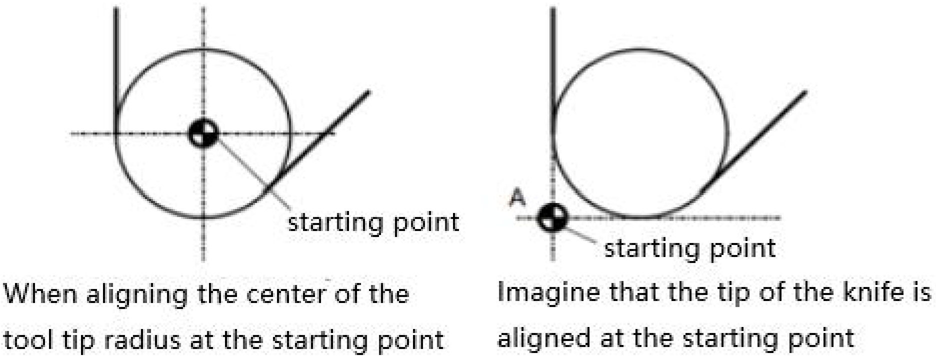


Figure 2.13.2

3. Tool Tip Direction:

The tool tip direction is the direction viewed from the tool tip radius center. It is determined by the orientation of the tool during the cutting process and must be specified in advance, just like the compensation value. The tool tip direction can be selected from the 8 options shown in the diagram below, along with the corresponding T code selection.

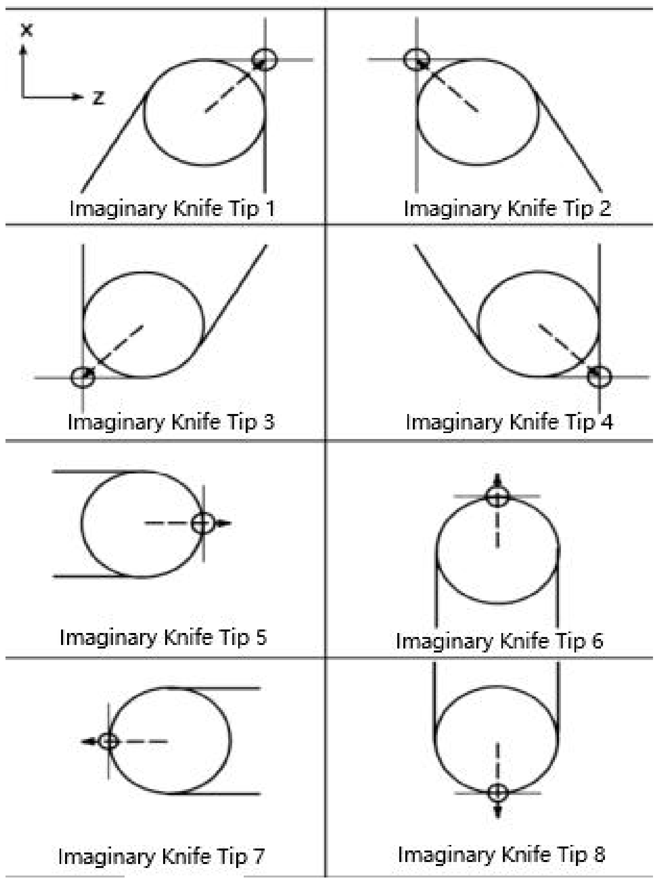


Figure 2.13.3

1. Tool Tip Radius Compensation Value:

The tool tip radius value (OFR) consists of two parts: tool tip length compensation value (OFGR) and tool tip wear compensation value (OFWR). OFR = OFGR + OFWR. When OFR is a positive value, G41 is left compensation, and G42 is right compensation. When OFR is a negative value, G41 is right compensation, and G42 is left compensation. The tool tip length compensation value (OFGR) is set by Sys2201.

2. Tool Tip Point and Compensation Operation:

The tool tip radius center is used for machining at the starting position.

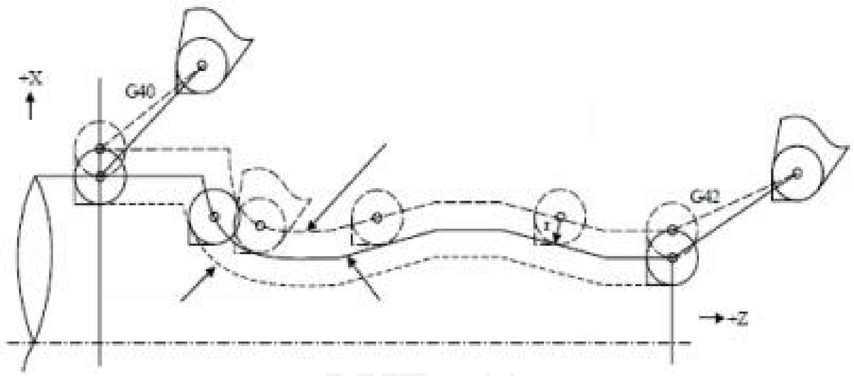


Figure 2.13.4

The tool tip point (TTP) is used for machining at the starting position

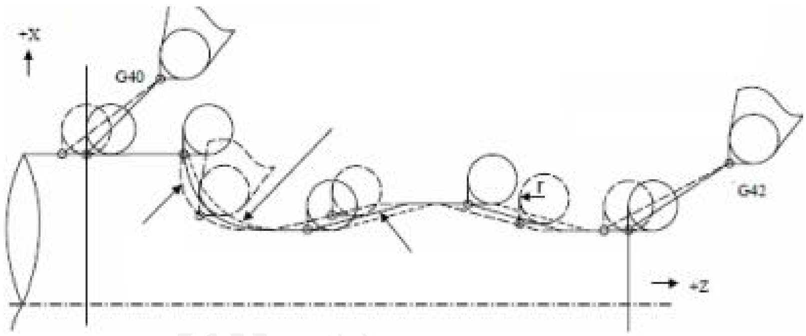


Figure 2.13.5

3. Preprocessing for Tool Tip Radius Compensation:

Regardless of whether it is in continuous or single-step execution mode, when compensation starts, it is necessary to continuously read N program segments for intersection calculation.

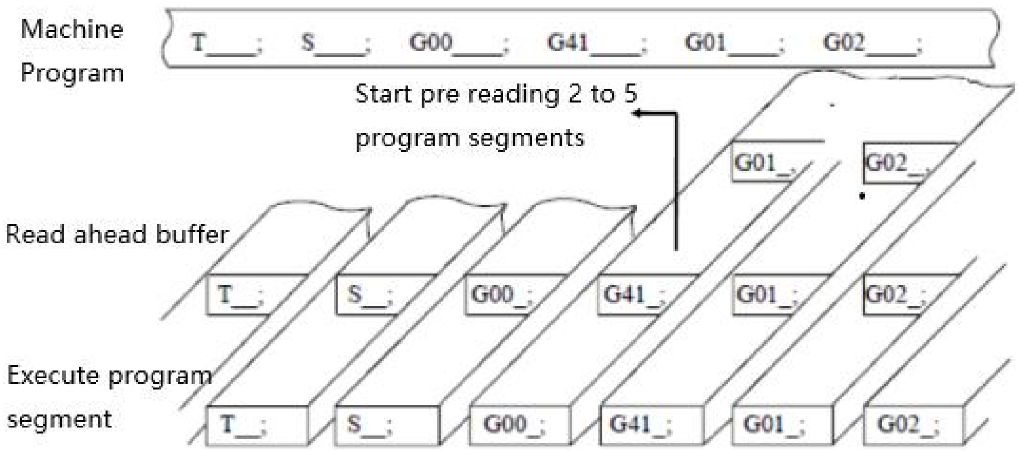


Figure 2.13.6

When encountering a non-movement program segment, the system performs the following actions:

1) During the start of compensation, continuous specification of N non-movement program segments is done, and no tool tip compensation is established at the non-program segment location.
2) During compensation, if N consecutive non-movement program segments are encountered, a vertical compensation vector is created at the end of the previous program segment.
3) When canceling compensation, a non-movement program segment is specified, i.e., compensation is canceled when G40 is in the same block as the non-movement program.

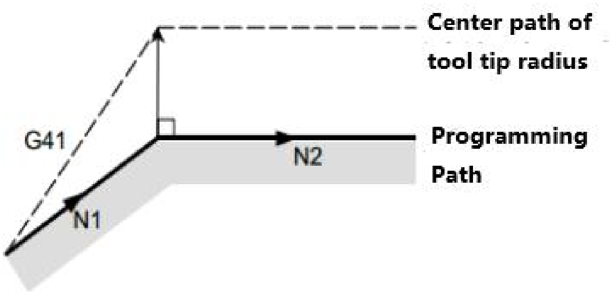
4. Types of Tool Tip Radius Compensation:

The types of tool tip radius compensation for the start and end can be classified into the following four types: Type A, Type B, Type C, and Type D, which are set by Mcm1701.

Start Programming:

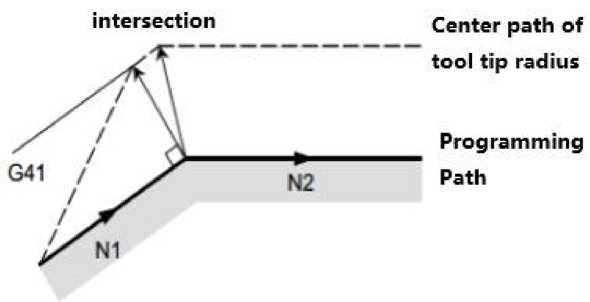
1) G41 is in the same block as the motion of the current plane axis.
2) G41 is in a separate block or the plane axis increment is 0.

Type A: A vertical compensation vector is output in the next program segment after the start.



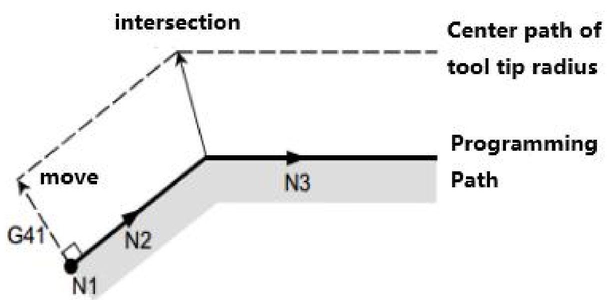
Programming the running trajectory of ① and ②

Type B: Outputs a compensation vector that is perpendicular to the program segment where the tool tip is engaged, as well as an intersection vector.



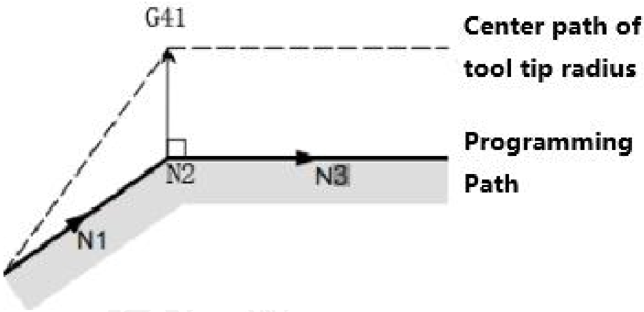
Programming the running trajectory of ① and ②

Type C: Outputs a compensation vector that is perpendicular to the program segment where the tool tip is engaged, as well as an intersection vector.



Programming the running trajectory of ① and ②

Type D: The system pre-reads the program segment before G41/G42 and establishes tool compensation at the end of the program segment without any perpendicular movement. Note: Program segment N2 is the G41 program segment.

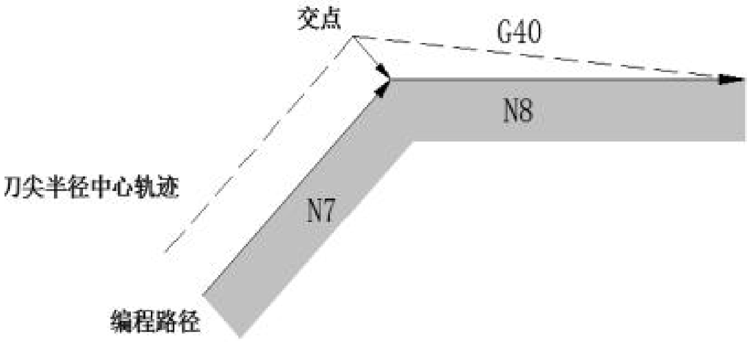


- **Ending Programming**

1) G40 is in the same program segment as the current plane axis motion.

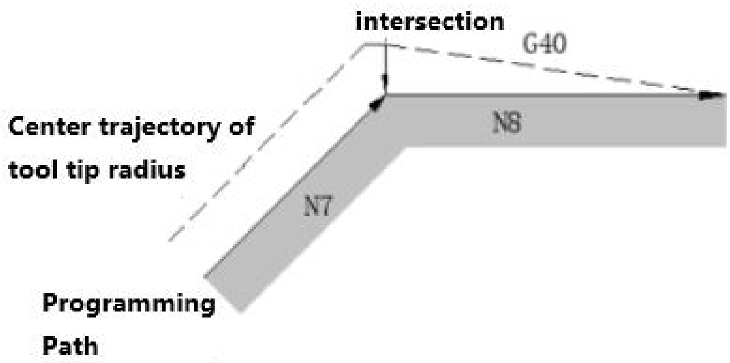
2) G40 is a standalone program segment or has a plane axis increment of 0.

Type A: Cancel the perpendicular compensation vector in the previous program segment upon cancellation.

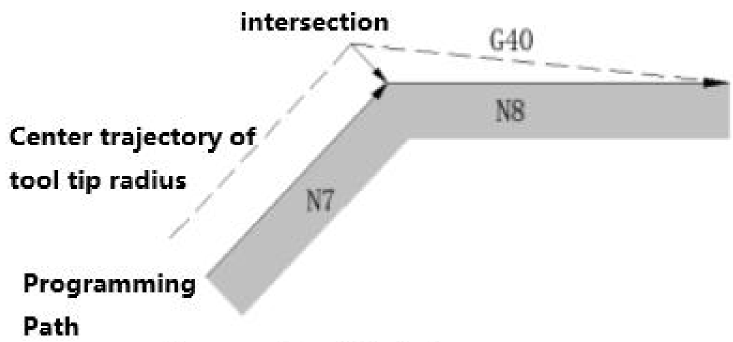


编程③、④轨迹

Type B: Output the compensation vector perpendicular to the canceled program segment, as well as the intersection vector.

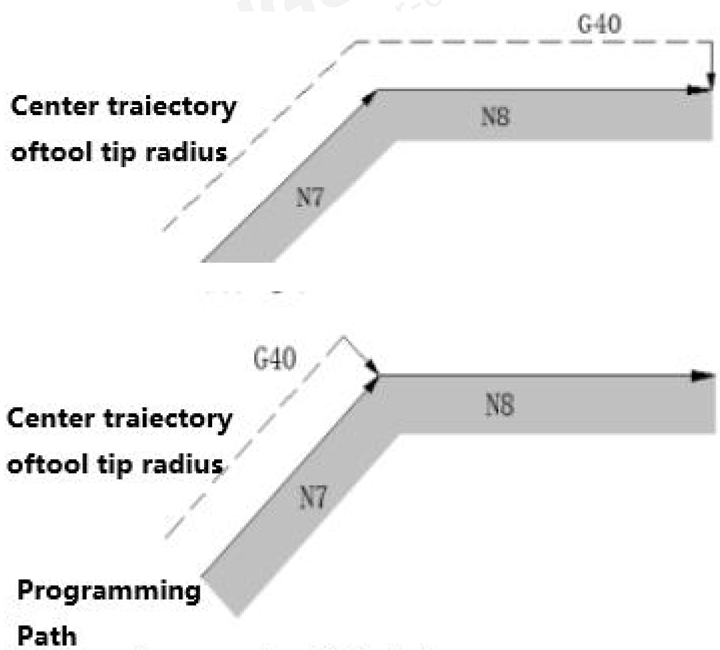


Programming ③ Trajectory



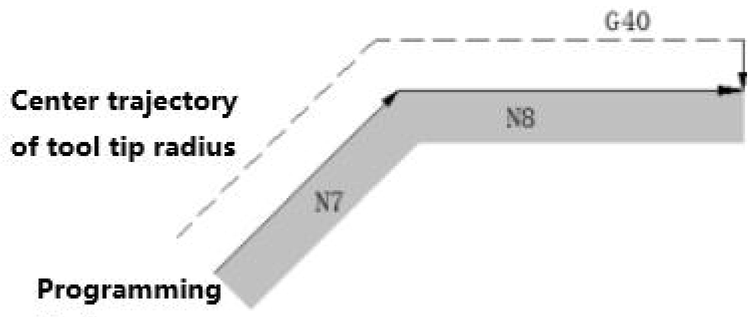
Programming ④ Trajectory

Type C: The tool moves along the direction perpendicular to the canceled program segment, equivalent to the amount of tool tip radius compensation.

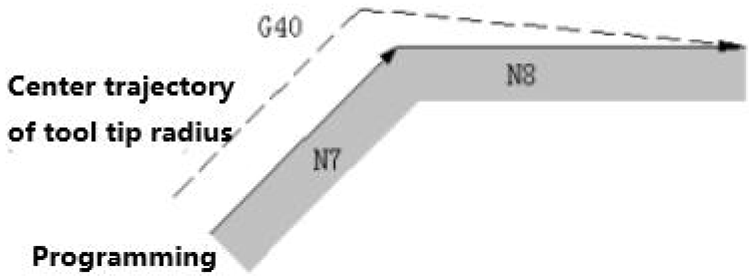


Programming ④ Trajectory

Type D: The system pre-reads the program segment before G40, and cancels the tool compensation at the end of the preceding program segment, without any perpendicular movement.



Programming ③ Trajectory



Programming ④ Trajectory

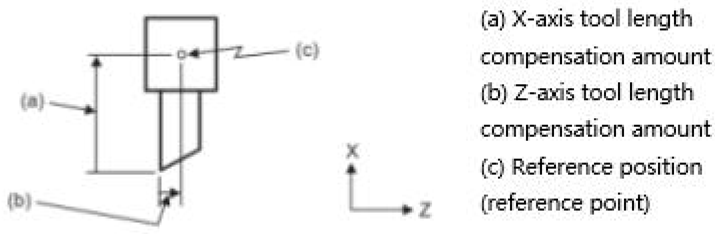
2.28 G43, G44, G49 Tool Length Compensation

- **Function and Purpose**

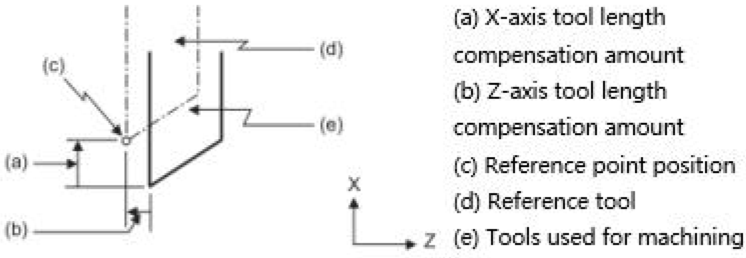
Perform tool length compensation relative to the program's reference position. The program reference position is typically located at the center position of the tool turret or the reference tool.

- **Tool Length Compensation Setting**

1. Setting at the center position of the tool turret.

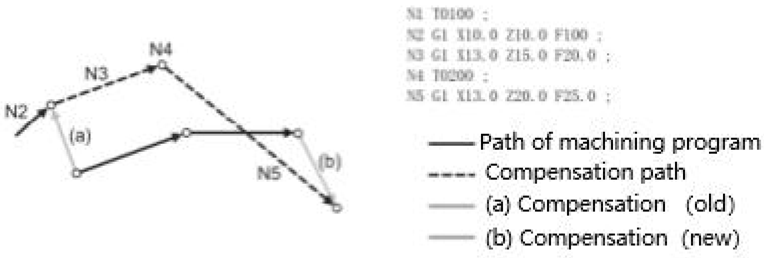


2. Setting at the tip position of the reference tool.



3. Changing tool length compensation number

When changing the tool number, the accumulated tool length compensation value corresponding to the new tool number is added to the movement amount of the machining program.

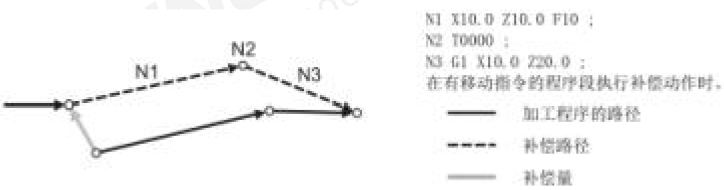


The diagram above shows the compensation action of changing the tool length compensation number during the execution of a movement command in the program segment.

- **Canceling tool length compensation**

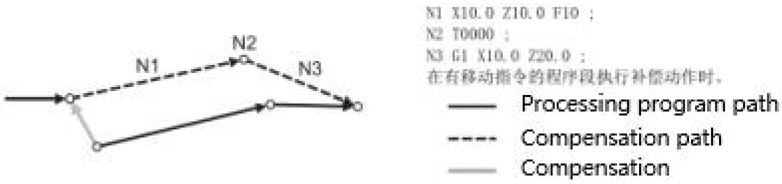
1. Specify compensation number T0:

When the tool length compensation number in the T command is set to 0, the compensation is canceled.



2. Specify a compensation value of 0:

When the compensation value for the tool length compensation number in the T command is set to 0, the compensation is canceled.



3. Resetting tool length compensation.

Program format:

G43 (G44) Z_____ H_____ - Length compensation setting

or G43 (G44) H_____ - Length compensation setting

G49 - Cancel length compensation

Z: Starting compensation coordinate.

H: Tool number for length compensation.

Explanation:

When machining each workpiece using a milling machine or machining center, multiple tools with different lengths are used, resulting in varying distances between the tool tip and the workpiece. If the program is executed without considering this, the difference in tool length before and after tool change will cause errors in the Z-axis position. The purpose of tool length compensation (G43/G44) is to correct the error in tool length by adjusting the Z-axis position. The setting of the length compensation value can be done using the following methods:

- **Setting of Length Compensation Value**

Method 1: Start from the mechanical origin of the Z-axis, manually move the tool downward until it touches the workpiece surface. Measure the distance it moved and input this value into the tool setting page on the operator interface, along with the tool number for length compensation in the H value field of the program command format.

Method 2: Choose a reference tool and set it as the basis. On the controller's operator interface, use the G54 work coordinate system for tool length calibration. Then, calculate the length compensation values for other tools based on the difference in length compared to the reference tool.

When using G43, the controller will take the specified compensation value and directly add it to the Z-axis compensation.

When using G44, after selecting the specified compensation value, the direction is reversed, and then added to the Z-axis compensation.

The direction of compensation is defined with respect to the Z-axis coordinate axis. If, after compensation, the tool moves in the positive direction of the Z-axis coordinate axis, it is called positive compensation. If it moves in the negative direction of the Z-axis coordinate axis, it is called negative compensation. Therefore, the positive and negative values of tool length compensation are determined by the direction of compensation under G43 and G44 instructions.

	MCM parameters positive value	MCM parameters negative value
G43	Positive Compensation	Negative Compensation
G44	Negative Compensation	Positive Compensation

- **Program Example**

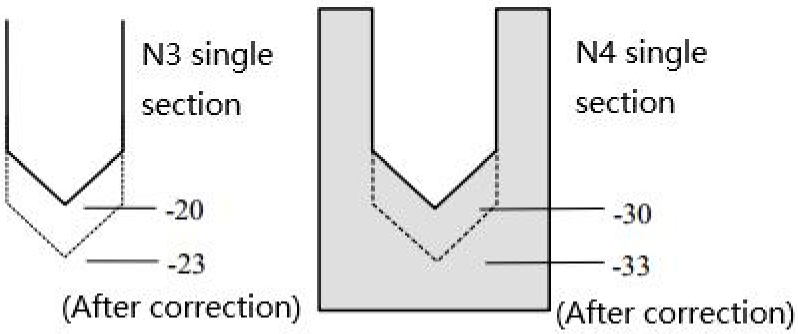
N1 G00 Z0.000

N2 G0 X1.000 Y2.000

N3 G43 Z-20.000 H10 (- Length compensation -3.000).

N4 G01 Z-30.000 F200

N5 G49 Z0.000



2.29 G51, G50 Scaling

- **Function and Purpose**

This command is used to scale the specified movement instructions within a specified range by multiplying them with a scaling factor, resulting in the desired size of the programmed shape.

- **Instruction Format:**

1. Scaling along all axes with the same scaling factor (MCM5840 BIT00=0)

G51 X_ Y_ Z_ P_: Start scaling with a proportional factor

...

G50: Cancel scaling

X_ Y_ Z_: Absolute instruction for the center coordinates of scaling (cannot be specified incrementally)

P_: Scaling factor (integer value, not affected by decimal point)

If P value is not specified, the value set in MCM5843 is used.

2. Scaling along each axis individually (MCM5840 BIT00=0)

G51 X_ Y_ Z_ I_ J_ K_: Start scaling with individual scaling factors for X, Y, and Z axes

...

G50: Cancel scaling

X_ Y_ Z_: Absolute instruction for the center coordinates of scaling (cannot be specified incrementally)

I_ J_ K_: Scaling factors for X, Y, and Z axes, respectively (integer values, not affected by decimal point)

If I, J, K values are not specified, the values set in MCM5844~MCM5879 are used. I, J, K can be specified as negative values, indicating mirroring.

Note: If the scaling center is not specified, the position of the tool when the G51 command is executed will be considered as the scaling center.

- **Function Introduction**

1. Scaling with the same factor along all axes

G51 X_ Y_ Z_ P_: P_ specifies the scaling factor. The smallest unit of the scaling factor is determined by MCM5842.

Conditions to enable scaling with the same factor along all axes:

a. MCM5840 BIT00=0 (individual scaling along axes not allowed).

b. MCM558 enables the corresponding axes for scaling.

c. Scaling factor is specified by P_ or MCM5843 (if P is not specified).

- **Example Program 1**

a. MCM5840=0: Scaling with the same factor enabled.

b. MCM558=5: X and Z axes scaling enabled.

c. MCM5842=0: Smallest unit of scaling factor is 0.001.

G00 X0. Z0.

G01 Z20. F3000.

G01 X20.

G01 Z0.

G01 X0.

G51 X0. Z0. P2000 ;;;; Enable scaling with a scaling factor of 2.

G01 Z20.

G01 X20.

G01 Z0.

G01 X0.

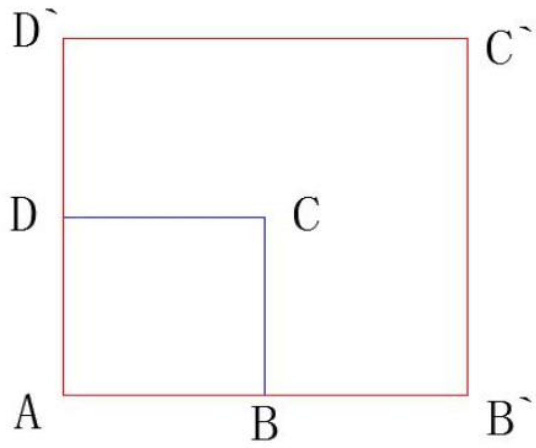
G50 ;;; Cancel scaling.

M30

The processing graphic is as follows:

Shape A→B→C→D represents the shape before scaling.

Shape A→B→C→D` represents the shape after scaling.



2. Individual scaling for each axis

G51 X_ Y_ Z_ I_ J_ K_:

I_ J_ K_ specifies the scaling factor for each axis. The smallest unit of the scaling factor is determined by MCM5842.

Conditions to enable individual scaling for each axis:

a. MCM5840 BIT00=1 (individual scaling along axes allowed).

b. MCM558 enables the corresponding axes for scaling.

c. Scaling factor is specified by I_ J_ K_ or MCM5844-MCM5879 (if I_ J_ K_ is not specified).

- **Example Program 2**

a. MCM5840=1: Allowing individual scaling along axes.

b. MCM558=5: Enabling scaling for X and Z axes.

c. MCM5842=0: Scaling factor minimum unit is 0.001.

G00 X0. Z0.

G01 Z20. F3000.

G01 X20.

G01 Z0.

G01 X0.

G51 X10. Z10. I2000 K3000 ;;; Enabling scaling with a scaling factor of 2

G01 Z20.

G01 X20.

G01 Z0.

G01 X0.

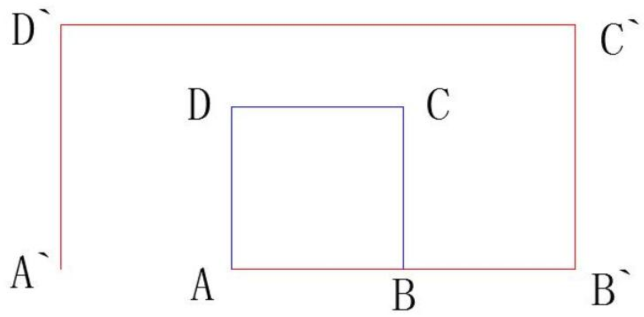
G50 ;;; Cancel scaling

M30

The processing graphics are as follows:

A→B→C→D: Original shape before scaling

A→B→C→D→A: Shape after scaling



3. Scaling in Circular Interpolation

Even when different scaling factors are applied to each axis in circular interpolation, the tool path will not become an ellipse.

a. The instructions:

G0 X100. Z0.

G51 X0. Z0. I1000 K2000

G02 X0. Z100. R100. F5000.

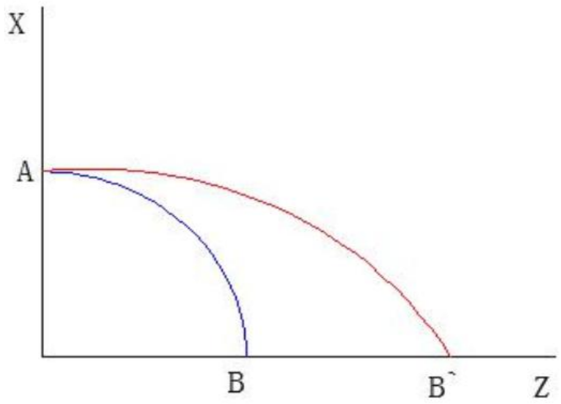
Are equivalent to:

G0 X100. Z0.

G02 X0. Z200. R200. F5000.

The circular arc AB represents the circular arc before scaling.

The circular arc AB` represents the circular arc after scaling.



b. The instructions:

G0 X100. Z0.

G51 X0. Z0. I1000 K2000

G02 X0. Z100. I-100. K0. F5000.

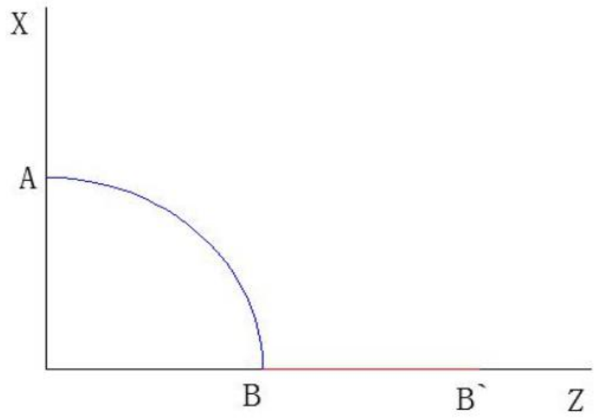
Are equivalent to:

G0 X100. Z0.

G02 X0. Z200. I-100. K0. F5000.

The circular arc AB represents the circular arc before scaling.

The arc A→B→B` represents the circular arc after scaling.



4. Tool Compensation

Scale scaling does not apply to tool radius compensation values, tool length compensation values, and tool offset values for tool position offset. Scale scaling is not applicable in the following cases:

a. Cutting depth Q and retraction distance d in deep hole drilling cycles (G83, G84).

b. In manual operations, scale scaling cannot be used to enlarge or shrink distances for tool compensation. Scale scaling, coordinate rotation, tool compensation, and scale scaling can be used in any order, but it is important to note that the resulting shapes may differ when applying scale scaling before coordinate rotation compared to applying coordinate rotation before scale scaling.

- **Example Program 3**

G0 X0. Z0.

G98

G18

T01

G68 X0. Z0. R45.

G51 X0. Z0. P2000

G42

G01 Z40. F3000.

G03 Z50. X10. R10.

G01 X40.

G01 X40. Z10.

G03 X30. Z0. R10.

G40

G01 X0.

G50

G69

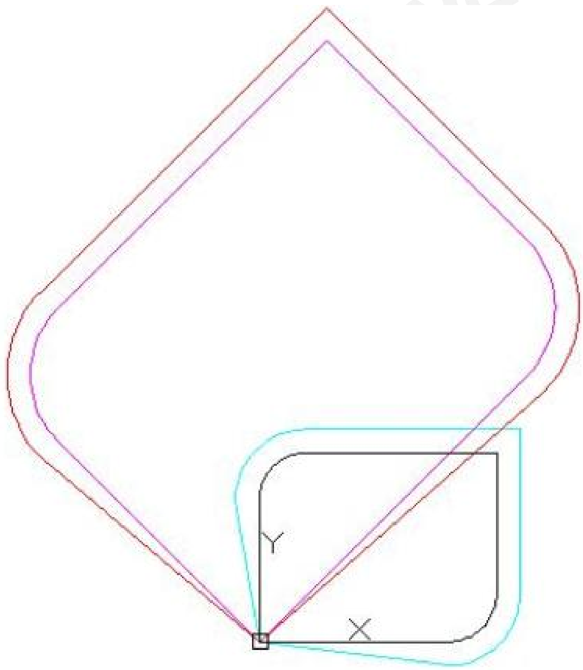
M30

Black line: Original shape without tool compensation.

Cyan line: Shape with tool compensation applied.

Magenta line: Shape with a 2x scale scaling and a 45° rotation applied.

Red line: Shape with a 2x scale scaling and a 45° rotation applied, including tool compensation.



Instructions related to Reference Point Return/Coordinate System

Under the scaling mode, G codes related to reference point return (G28, G29, G30, etc.) and commands that change the coordinate system (G52-G59, G50, etc.) should not be specified. When specifying these G codes, the scaling mode needs to be canceled first.

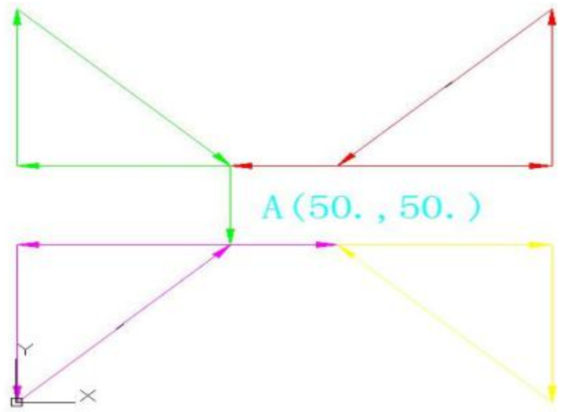
- **Example Program 4**

Subprogram:

```
G51-01.CNC
G00 X60. Z60.
G01 Z100. F3000.
G01 X100.
G01 X60. Z60.
M99
```

Main program:

```
G51-02.CNC
G98
G18
M98<G51-01.CNC>
G51 X50. Z50. I1000 K1000
M98<G51-01.CNC>
G51 X50. Z50. I1000 K-1000
M98<G51-01.CNC>
G51 X50. Z50. I-1000 K-1000
M98<G51-01.CNC>
G51 X50. Z50. I-1000 K1000
M98<G51-01.CNC>
G50
M30
```



The execution order is red line → green → magenta → yellow

2.30 G50 External Workpiece Coordinate System

- **Function**

Sets the external workpiece coordinate system.

- **Instruction Format**

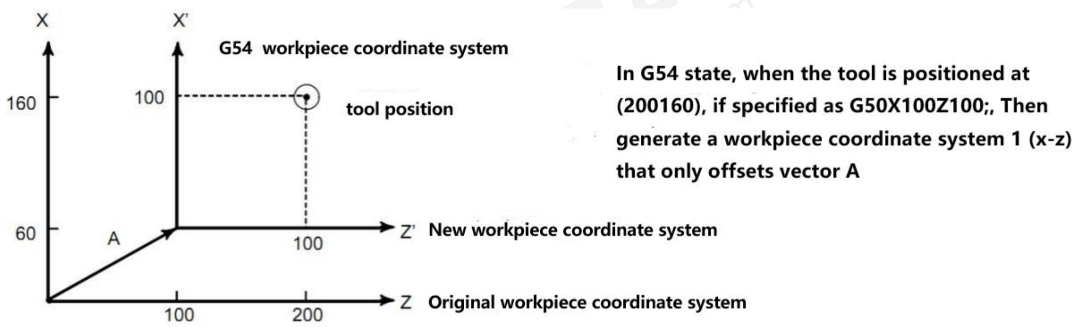
G50 IP_

IP_: Specifies the axis position, which is the position of the tool in the current workpiece coordinate system.

The IP value is written in real-time to the variables Sys2160~Sys2165.

G50 IP_: Set External Workpiece Coordinate System

1. Absolute Specification: The position specified by IP_ is the current position of the tool.



2. (2) Incremental Specification: The specified incremental value is added to the current tool coordinate values before the specification, resulting in the current position of the tool.

Note: G50 does not change the values of G54~G59 and G54.1Pxx work coordinate systems. It modifies the values of the external workpiece coordinate system, causing all work coordinate systems to be offset.

- **Example**

Program	Machine Coordinates	Program Coordinates	SYS2160 , SYS2162
G0X0.Z0.	0.000，0.000	0.000，0.000	0，0
G0X100.Z50.	100.000，50.000	100.000，50.000	0，0
G50X0.Z20.	100.000，50.000	0.000，20.000	100000，30000
G0X50.Z50.	150.000，80.000	50.000，50.000	100000，30000
G50U20.W-10.	150.000，80.000	70.000，40.000	80000，40000
G0X0.Z0.	80.000，40.000	0.000，0.000	80000，40000

2.31 G51.2, G50.2 Spindle Synchronization

- **Function and Purpose**

By using G codes or PLC control, the spindle synchronization feature enables the synchronization of the main spindle (tool spindle) and the reference spindle (workpiece spindle) at a set speed ratio and phase difference. This functionality is primarily used for workpiece alignment, polygon machining, and similar operations involving two spindles.

- **Instruction Format**

G51.2 H_ D_ P_ Q_ R_ : Enable spindle synchronization feature.

G50.2 : Cancel spindle synchronization feature.

H: 1 to n (total number of spindles). Sets the reference spindle. If not set, the value in parameter Sys3080 is used without decimal point influence. When set, Sys3080 is modified in real-time.

D: 1 to n (total number of spindles). Sets the synchronized spindle. If not set, the value in parameter Sys3081 is used without decimal point influence. When set, Sys3081 is modified in real-time.

P: 1 to 999. Sets the speed ratio of the reference spindle. Not affected by decimal point.

Q: (1 to 999, -999 to -1). Sets the speed ratio of the synchronized spindle. When set to a negative value, reverse synchronization is performed. Not affected by decimal point.

R: 0 to 359.999. Sets the phase difference between the synchronized spindle and the reference spindle, in the clockwise direction along the spindle. Affected by decimal point.

- **Spindle Synchronization Steps**

1. Enable the spindle synchronization command.
2. Start rotating the reference spindle (this step can be performed before step 1, but it must be specified by the user).
3. The synchronized spindle rotates at the specified speed ratio (P:Q) according to its own acceleration and deceleration time, reaching the designated speed.

   Note: The corresponding axis of the synchronized spindle needs to be enabled before the spindle synchronization command is enabled. If the synchronized spindle is not started before the synchronization command, the system will automatically start it.
4. Before entering phase synchronization, both spindles adjust their speeds according to their own acceleration and deceleration time. Once the synchronized spindle enters phase synchronization and completes it, its acceleration and deceleration are completely determined by the reference spindle.
5. After reaching the synchronized speed, the synchronized spindle begins phase synchronization, achieving it through deceleration.
6. Phase synchronization is considered complete when the phase difference between the synchronized spindle and the reference spindle reaches the set value, and the

speed of the synchronized spindle returns to the synchronized speed.

7. Cancel the spindle synchronization.

   Note: Throughout the entire synchronization process, the speed of the reference spindle can be changed at any time, and the synchronized spindle will adjust its speed proportionally and maintain synchronization. For precise phase synchronization, closed-loop synchronization functionality needs to be enabled.

- **Spindle Synchronization Commands**

G51.2 H_ D_ P_ Q_ R_ : Enable spindle synchronization.

G50.2: Cancel spindle synchronization.

Note:

When using the complete command, specify a new synchronization command while the previous synchronization command has not been canceled yet.

Ⅰ. If the reference spindle and synchronized spindle remain unchanged, the system will not generate an alarm and will proceed with the new synchronization command.

Ⅱ. If the reference spindle or synchronized spindle is changed, the system will generate an alarm.

Programming error: G0 X0. Z0.

......

G51.2 H1 D2 P1 Q1 R90.

......

G51.2 H2 D3 P1 Q1 R90.

......

G50.2

System Alarm: Spindle synchronization is already in progress. It is not allowed to start spindle synchronization again within the same channel.

Correct programming: G0 X0. Z0.

......

G51.2 H1 D2 P1 Q1 R90.

......

G50.2

G51.2 H1 D2 P1 Q1 R90.

......

G50.2

Correct programming： G0 X0. Z0.

......

G51.2 H1 D2 P1 Q1 R90.

......

G51.2 H1 D2 P1 Q2 R90.-------Change Synchronization Speed

......

G50.2

- **Spindle Synchronization Related Functions**

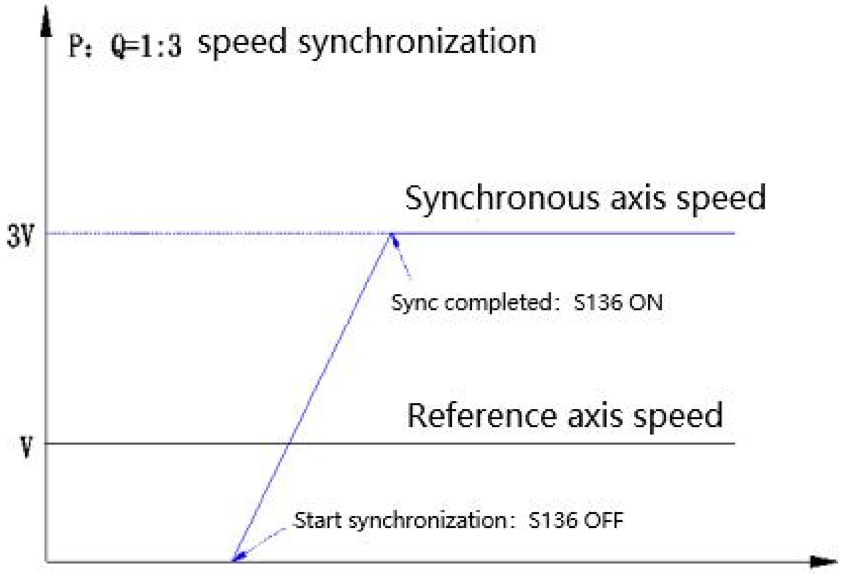
1. Speed Synchronization:

The reference spindle and the synchronized spindle rotate at a speed ratio of P:Q without phase synchronization.

Mcm04 BIT05 = 0: Speed synchronization is performed when R is specified as a negative value.

= 1: Speed synchronization is performed when R is not specified or specified as a negative value.

When Sys3084 is set to a value less than 0, PLC spindle synchronization performs speed synchronization.



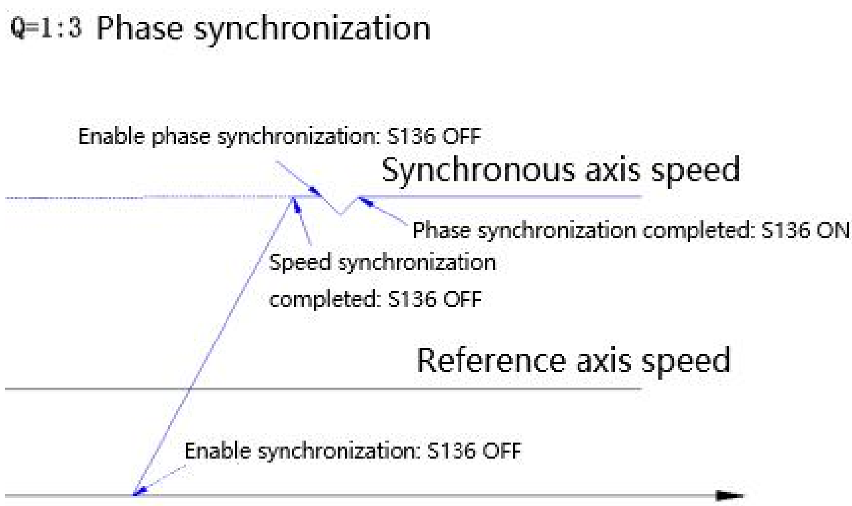
2. Phase Synchronization:

The reference spindle and the synchronized spindle rotate at a speed ratio of P:Q with phase synchronization.

Mcm04 BIT05 = 0: Phase synchronization is performed when R is specified as ≥ 0 or when R is not specified (R must not exceed the set range).

= 1: Phase synchronization is performed when R is specified as ≥ 0 (R must not exceed the set range).

When Sys3084 is set to a value ≥ 0, PLC spindle synchronization performs phase synchronization (R must not exceed the set range).



1. Semi-Closed Loop Phase Synchronization:

Semi-closed loop phase synchronization means that the system synchronizes based on commands. When the reference spindle and the synchronized spindle reach the set speed ratio of P:Q and maintain the set phase difference at the current speed, semi-closed loop phase synchronization is achieved. When the speed of the reference spindle changes, the phase difference between the reference spindle and the synchronized spindle will also change, and the system does not correct it. Semi-closed loop phase synchronization is mainly used in processes where the spindle speed does not change.

2. Closed Loop Phase Synchronization:

Closed loop phase synchronization means that the system synchronizes based on feedback. When the reference spindle and the synchronized spindle reach the set speed ratio of P:Q and have just completed phase synchronization, closed loop synchronization is activated. The system continuously corrects the synchronized spindle based on the real-time speed of the reference spindle to ensure a fixed phase difference between the synchronized spindle and the reference spindle. Closed loop phase synchronization should be used when Q:P is an integer. Mcm04 BIT07: Activate the closed loop phase synchronization function. Mcm20: Adjust the proportional gain of the position loop during closed loop phase synchronization. This parameter should be set to a value greater than 0; otherwise, the closed loop phase synchronization function cannot be enabled. Mcm21: Limit the synchronization position error during closed loop phase synchronization. When the system detects excessive position fluctuation of the synchronized axis after entering closed loop phase synchronization, an alarm is triggered, and the closed loop synchronization function is canceled for protection. Closed loop synchronization is generally used when the reference spindle speed can be modified at any time after phase synchronization and precise phase synchronization (to eliminate actual phase deviation caused by speed changes) is required.

3. It is used for precise phase synchronization to eliminate the actual phase deviation

caused by speed changes. Otherwise, it may not be necessary.

- **Other Functions Introduction**

1. When spindle synchronization is enabled but not yet synchronized (S136 OFF), the reference spindle and synchronized spindle operate with their own acceleration and deceleration.
2. When spindle synchronization is enabled and already synchronized (S136 ON), the reference spindle and synchronized spindle operate with the acceleration and deceleration of the reference spindle.
3. After spindle synchronization is completed, the synchronized spindle is controlled by the reference spindle. When the state of the reference spindle is changed, the synchronized spindle also changes according to the rotation ratio P:Q.
4. After spindle synchronization is completed, changes to the synchronized spindle's MFO, speed, rotation direction, and operation state during spindle synchronization will not affect the synchronized spindle, but the status will still be updated and take effect after spindle synchronization is canceled.
5. During spindle synchronization, constant speed control of the synchronized spindle is ineffective, but the status is still updated and takes effect after spindle synchronization is canceled.
6. The speed of the reference spindle is limited by the minimum and maximum speeds of the synchronized spindle. The system will set a new reference spindle speed based on the minimum or maximum speed of the synchronized spindle according to the rotation ratio P:Q.

For the given setup:

Reference Spindle: Maximum speed 3000r/min, minimum speed 10r/min

Synchronized Spindle: Maximum speed 2100r/min, minimum speed 10r/min

Program:

N1 G0 X0. Z0.

N2 M03 S800

N3 G51.2 H1 D2 P1 Q3 R90.

N4 G04 X5.

N5 G50.2

Since the maximum speed of the synchronized spindle is 2100r/min, after synchronization, the speed of the reference spindle will be limited to 700r/min.

The system will generate an alarm in the following cases:

a. .The minimum speed of the synchronized spindle is higher than the maximum speed of the reference spindle.

b. .The minimum speed of the reference spindle is higher than the maximum speed of the synchronized spindle.

   In both of these situations, the system cannot find a suitable synchronized speed between the minimum and maximum speeds, and the alarm message will be: "The reference spindle cannot achieve a suitable speed between the minimum and maximum speeds during the synchronization process."

7. Before the start of spindle synchronization, if the synchronized spindle is not activated, the system will automatically control the synchronized spindle to start according to the rotation ratio and synchronization direction, initiating the spindle synchronization function. (Note: The system does not control the enable control of the corresponding motor for the synchronized spindle, so the enable control of the synchronized spindle motor should be activated before the start of spindle synchronization.)
8. Only one synchronized spindle instruction is allowed within the same channel. Otherwise, the system will generate an alarm.
9. Two different channels cannot issue synchronization instructions for two identical spindles. Otherwise, the system will generate an alarm.
10. Multiple channels can simultaneously perform their own spindle synchronization, but the same synchronized spindle cannot be specified. The same reference spindle can be specified, but it must ensure that there are no conflicting speed limitations;

otherwise, synchronization cannot be achieved correctly.

11. The reference spindle and the synchronized spindle must be pulse-type position-controlled motors and drives. Otherwise, synchronization cannot be performed, and the system will generate an alarm.
12. Spindle synchronization can be canceled in five different states: G50.2, C136=0, reset, emergency stop, and power failure.
13. Resetting to cancel synchronization has two different scenarios:

    a. If synchronization has started but is not yet completed (S136 OFF), resetting will first cancel the synchronization, and the reference spindle and synchronized spindle will decelerate to 0 according to their respective acceleration and deceleration settings.

    b. If synchronization has started and is already completed (S136 ON), resetting will cause the synchronized spindle to decelerate to 0 based on the acceleration and deceleration settings of the reference spindle before releasing the synchronization.
14. When G50.2, C136, emergency stop, or power failure occurs, spindle synchronization is immediately canceled, and the reference spindle and synchronized spindle will operate according to their respective acceleration and deceleration settings after synchronization is canceled.
15. If a program pause occurs during the synchronization state, the spindle synchronization will still be maintained.

2.32 G51.4, G50.4 Axial Synchronous Control

- **Function and Purpose**

The axis synchronization control function allows multiple axes (up to the maximum axis count determined by the current channel's interpolation settings, Mcm1901) to move in synchronization. In this mode, one axis is designated as the master axis, while the remaining axes, known as slave axes, maintain synchronization with the master axis.

When a motion command is programmed for the master axis, the same command is automatically transmitted to the slave axes, ensuring coordinated movement across all axes.

- **Instruction Format**

G51.4 P_Q_ (L_): Activate axis synchronization.

G50.4 Q_: Deactivate axis synchronization.

P_: Specifies the master axis for axis synchronization.

Channel number * 100 + axis number. If the channel number is not specified, it represents the current channel.

P can be specified as a negative value, in which case the command for the slave axis is the opposite of the master axis.

Q_: Specifies the slave axis for axis synchronization.

Channel number * 100 + axis number. If the channel number is not specified, it represents the current channel.

L_: Hold command.

=0: No hold (cancel hold).

=1: Hold the master axis (cancel hold for slave axis).

=2: Hold the slave axis (cancel hold for master axis).

L can be omitted. When L is omitted, it is treated as L0.

- **Instruction Control**

1. When controlling axis synchronization, multiple axes can be specified as follow axes using G51.4, with one axis as the master axis.
2. Example:

G51.4 P1 Q2: Specifies the X axis as the master axis for the Y axis.

G51.4 P1 Q3: Specifies the X axis as the master axis for the Z axis.

...

G50.4 Q2: Cancels the X axis as the master axis for the Y axis.

G50.4 Q3: Cancels the X axis as the master axis for the Z axis.

1. When specifying P in the instruction control and it exceeds the valid number of axes in the system, the system will generate an alarm (Program Error -451-1).

   When specifying P in the instruction control and it does not correspond to a valid axis, the system will generate an alarm (Program Error -451-1).
2. When specifying Q in the instruction control and it exceeds the valid number of axes in the system, the system will generate an alarm (Program Error -451-2).

   When specifying Q in the instruction control and it does not correspond to a valid axis, the system will generate an alarm (Program Error -451-2).

   When using G50.4 Q_ and specifying Q that was not previously defined in the G51.1 instruction, the system will generate an alarm (Program Error -451-2). An alarm will also be generated if multiple G50.4 Q_ commands specify the same Q (Program Error -451-2).
3. During instruction control, when currently in axis synchronization and the master and slave axes are in the same block, the command of the master axis is automatically ignored, and only the command of the slave axis is executed. For example, with the command: G51.4 P1 Q3 G01 U100. W50. F2000., after executing this block, the X and Z axes will have moved by 100.000 and 50.000 respectively.
4. After canceling axis synchronization, it is necessary to issue G31P16384 to reinitialize the coordinate values of each axis.
5. When specifying P, positive and negative values can be used. When specifying a negative value, the command direction for the slave axis will be opposite to that of the master axis.

   Example:G51.4 P-1 Q03

   G01 U100. F5000.: When executing this command, the Z-axis will move with a displacement of -100.000.
6. When using the L parameter in the instruction control to determine whether the master or slave axis should be stopped, the following cases can be considered:

   Example:

   G51.4 P1 Q3 L2: The slave axis Z-axis is stopped, and the master axis X-axis stop is canceled.

...

G50.4 Q3: Cancel the synchronization status, and also cancel the stop of the slave axis Z-axis.

G51.4 P1 Q3 L1: The master axis X-axis is stopped, and the stop of the slave axis Z-axis is canceled.

G51.4 P1 Q2 L1: The master axis X-axis is stopped, and the stop of the slave axis Y-axis is canceled.

...

G50.4 Q3: Cancel the synchronization status of the slave axis Z-axis. Since there is still a slave axis Y-axis, the stop of the master axis X-axis is maintained.

G50.4 Q2: Cancel the synchronization status of the slave axis Y-axis. Since there are no more slave axes, the stop of the master axis X-axis is canceled.

Example:

N1 G98

N2 G08

N3 G01 Y100. F5000.

N4 G51.4 P2 Q3

N5 G01 Y200.

N6 G50.4 Q3

N7 Y100.

N8 Z200.

Explanation: Before starting the program, C71 is turned ON, and the Y and Z axes have stop enabled (Sys15030=6). Therefore, when executing the program, in line N3, the Y-axis stops, but in line N4, the G51.4 command does not specify stop for the relevant axes, so it is assumed that the stop is canceled. As a result, in line N7, the Y-axis does not stop either.

7. When both instruction control and PLC control coexist, with the setting Mcm5001=1:

   G51.4 P1 Q3

   G01 U50. F2000.: The changes in the X, Y, and Z axes are 50.000.

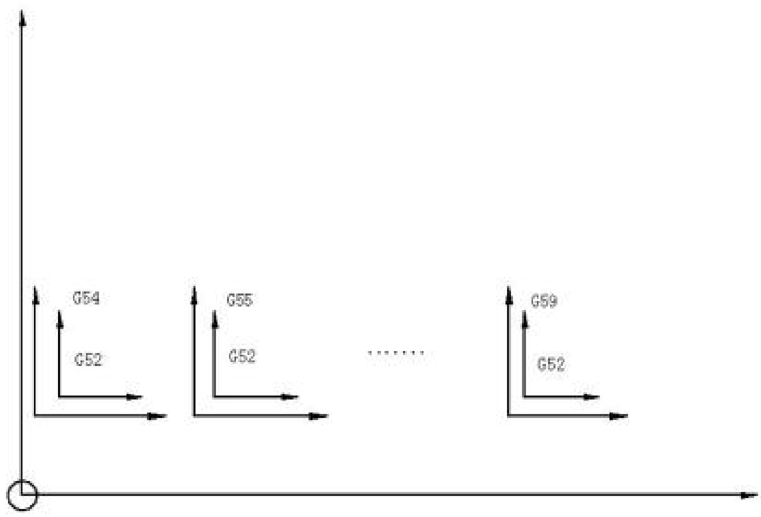
G50.4 Q3

G01 U50.: The changes in the X and Y axes are 50.000.

2.33 G52 Setting Local Coordinate System

- **Function and Purpose**

The G52 command is used to set a local coordinate system. This local coordinate system can be independently set within the workpiece coordinate systems G54~G59. When programming within a workpiece coordinate system, it may be convenient to define a sub-coordinate system within the workpiece coordinate system. This sub-coordinate system is referred to as the local coordinate system.



- **Instruction Format**

G52 IP_ - Set local coordinate system for specified axis.

IP - Specifies the axis for setting the local coordinate system.

G52 IP0 - Cancel the local coordinate system setting for the specified axis.

Note: The G52 command alone cannot cancel the local coordinate system setting.

Setting Local Coordinate System with G52:

1. Absolute Specification: When using absolute coordinate values, the local coordinate system is set at the specified coordinate position relative to the current local

coordinate system.

2. Incremental Specification: When using incremental coordinate values, the local coordinate system is offset from the current local coordinate system.

Note:

1. Local coordinate systems can be set within all workpiece coordinate systems (G54~G59, G54.1 P__) by specifying the IP_ axis. The origin of the local coordinate system is defined within the workpiece coordinate system.
2. Once a local coordinate system is set, it remains effective until it is canceled.
3. The local coordinate system remains active even when switching between workpiece coordinate systems (G54~G59).

- **G52 Canceling Local Coordinate System**

1. G52 IP__ - Cancel the local coordinate offset for the specified axis when setting IP to 0.

   Example:

   N1 G52 X2. Z2.

   N2 G0 X100. Z100.

   N3 G52 X0.

   In line N3, the local coordinate offset for the X-axis is canceled, while the Z-axis retains its local coordinate offset.

2. Canceling local coordinate offset during reset:

   MCM06 BIT00=1 - Set to cancel local coordinate offset during reset.

3. Canceling local coordinate offset at program start:

   Translation: Cancelling local coordinate offset at program start.

Sample Program:

Program	Machine Coordinate (X, Y, Z)	Program Coordinate(X, Y, Z)	SYS11620，SYS11621，SYS11622

G0X0.Y0.Z0.	0.000,0.000,0.000	0.000,0.000,0.000	0,0,0
G54	0.000,0.000,0.000	0.000,0.000,0.000	0,0,0
G52X10.Y20.Z30.	0.000,0.000,0.000	-10.000,-20.000,-30 .000	10000,20000,30000
G0X100.Y80.Z50.	110.000,100.000,80 .000	100.000,80.000,50. 000	10000,20000,30000
G55	110.000,100.000,80 .000	100.000,80.000,50. 000	10000,20000,30000
G0X-100.Y-100.Z-1 00.	-90.000, -80.000, -70.000	-100.000,-100.000,- 100.000	10000,20000,30000
G52U-50.V30.W0	-90.000, -80.000, -70.000	-50.000,-130.000,-1 00.000	-40000,50000,3000 0
G0X-50.Y-50.Z-50.	-90.000,0.000,-20.0 00	-50.000,-50.000,-50 .000	-40000,50000,3000 0
G52X0.Y0.	-90.000,0.000,-20.0 00	-90.000,0.000,-50.0 00	0,0,30000

G0X0.Y0.Z0.	0.000,0.000,30.000	0.000,0.000,0.000	0,0,30000

G00 X0. Y0. Z0.

G54

G52 X10. Y20. Z30. Setting the local coordinate system for X, Y, and Z axes.

G00 X100. Y80. Z50. Local coordinate system for X, Y, and Z axes is active.

G55 Local coordinate system for X, Y, and Z axes is active.

G00 X-100. Y-100. Z-100. Local coordinate system for X, Y, and Z axes is active.

G52U-50.V30.W0.

G0X-50.Y-50.Z-50.

G01 X0. Y0. Z0.

G52 X0. Y0. Canceling the local coordinate system for X and Y axes.

The local coordinate system for the Z axis remains active.

G0X0.Y0.Z0.

2.34 G53 Machine Coordinate Offset

- **Function and Purpose**

G53 instruction is used in combination with feed mode (G01 or G00) and followed by coordinate commands to move the tool to a specified position in the machine coordinate system (currently set as G00 rapid speed).

- **Instruction format**

G53 IP__

IP: Specifies the axis to move to the specified machine coordinate position (can be specified as incremental or absolute coordinates).

- **Detailed explanation**

1. When G53 specifies a position in the machine coordinate system, the tool moves to that position at a rapid speed.
2. G53 is only effective for the current instruction block.
3. Using G53 does not cancel tool radius compensation and does not cancel the current workpiece coordinate system.
4. IP_ can be specified as incremental or absolute coordinates.
5. G91 mode is effective for G53.

Example: G91 G53 X100., which means the X-axis moves in the machine coordinate system by an incremental distance of 100.

Sample Program：

Program	Machine Coordinate (X, Y, Z)	Program Coordinate (X, Y, Z)
G0X100,Z100.	100.000,100.000	100.000,100.000
G53X20,Z30.	20.000,30.000	20.000,30.000
G55	20.000,30.000	-30.000,-20.000
G01X80,000,Z80,000F50.	130.000,130.000	80.000,80.000
G53U-20,W-20.	110.000,110.000	60.000,60,000
T01	110.000,110.000	10.000,160.000
G53X0,Z0.	0.000,0.000	-100.000,50.000
G0X0Z0	100.000,-50.000	0.000,0.000

Note: The default workpiece coordinate system is G54, and T01 has tool length compensation values of (50.000, -100.000). G55 workpiece coordinate system is set to (50.000, 50.000).

2.35 G54~G59、G54.1__Workpiece Coordinate System

- **Function and Purpose**

The workpiece coordinate system is a coordinate system based on the reference point of the workpiece, used to simplify the programming of the workpiece. It can be changed in relation to the machine coordinate system by using the G50, G10 L02, G10 L20 commands, or manually modifying the corresponding parameters. The current workpiece coordinate system can be changed using G54 to G59, G54.1P1 to G54.1P48.

- **Instruction Format**

G54 to G59 IP_: Selects the G54 to G59 workpiece coordinate system.

IP: Specifies the position in the workpiece coordinate system to move the axis (can be specified in incremental or absolute coordinates).

G54.1P_IP_: Selects the G54.1P1 to G54.1P48 workpiece coordinate system.

P: 1 to 48, specifies a group within G54.1P1 to G54.1P48.

IP: Specifies the position in the workpiece coordinate system to move the axis (can be specified in incremental or absolute coordinates).

Note: When IP is not empty, the workpiece coordinate system is selected first, and then the movement command is executed at the specified position (using the speed of the previous movement command).

G50 IP_: Modifies the external workpiece coordinate system.

P: Specifies the axis code and position to modify, can be specified in incremental or absolute coordinates.

Incremental Specification: Adds the specified increment value to the previous tool coordinate value to determine the tool position in the current workpiece coordinate system.

Absolute Specification: The position specified by IP__ becomes the tool position in the current workpiece coordinate system.

Note: G50 does not change the values of G54 to G59 and G54.1Pxx work coordinate systems. It modifies the values of the external workpiece coordinate system (Sys2160 to

Sys2199), causing all work coordinate systems to be offset.

- **Detailed explanation**

1. The selection of the workpiece coordinate system is done through G54 to G59 or G54.1P1 to G54.1P48 commands. Once a workpiece coordinate system is specified using a command, it can be changed using other commands except for that specific command.

   For example, consider the following program:

   N01 G54

   N02 G00 X100. Z100.

   N03 G55

   In this case, the workpiece coordinate system before N03 is G54, and after N03, it is G55.

2. When the workpiece coordinate system and axis movement command are specified in the same line, the workpiece coordinate system is set first, and then the axis is moved to the specified position within that coordinate system. The axis movement speed will inherit the speed of the previous movement command (G00 or G01).

   For example:

   G00 X100.

   G55 X50.

   This is equivalent to:

   N01 G55

   N02 G00 X50.

Sample Program:

Program	Machine Coordinate (X, Y, Z)	Program Coordinate (X, Y, Z)
G0X0,Z0.	0.000,0.000	0.000,0.000
G54X100,Z100.	100.000,10.000	100.000,100.000

G55	100.000,10.000	50.000,50.000
G01X0,Z0,F500.	50.000,50.000	0.000,0.000
G59U50,W-50.	100.000,0.000	120.000,30.000
G0X0,Z0.	-20.000,-30.000	0.000,0.000

Note: G54 workpiece coordinate system is set to (0.000, 0.000), G55 workpiece coordinate system is set to (50.000, 50.000), and G59 workpiece coordinate system is set to (-20.000, -30.000).

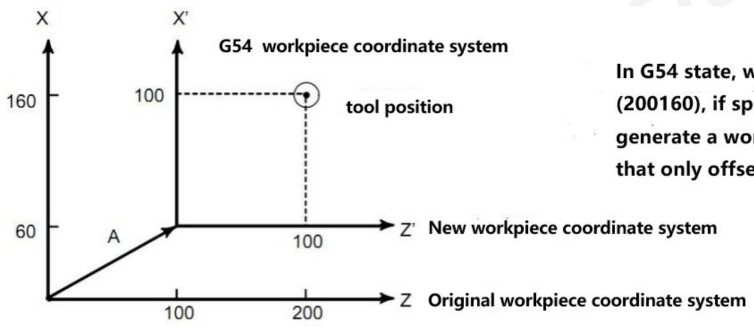
3. The workpiece coordinate system can be set using the G10 L02, G10 L20 commands, or manually modified. When using the command to modify the workpiece coordinate system, you can use either incremental or absolute values. For manual modification, you directly modify the corresponding values in Sys0000 to Sys2159.

   For incremental modification, you add the increment value to the specified workpiece coordinate system values. For absolute modification, you directly modify the specified workpiece coordinate system values based on the set data.

Sample Program:

Program	Machine Coordinate (X, Y, Z)	Program Coordinate (X, Y, Z)
G0X0,Z0.	0.000,0.000	0.000,0.000
G54	0.000,0.000	0.000,0.000
G10L2P0X100.Z50.	0.000,0.000	-100.000,-50.000
G0X0,Z0	100.000,50.000	0.000,0.000
G10L2P1X50,Z-30.	100.000,50.000	-50.000,30.000
G0X0,Z0.	150.000,20.000	0.000,0.000
G10L2P1U30,W50.	150.000,20.000	-30.000,-50.000
G0X0,Z0.	180.000,70.000	0.000,0.000

4. The external workpiece coordinate system refers to a coordinate system between the machine coordinate system and the workpiece coordinate system. Changing the external workpiece coordinate system will result in changes to the positions of all workpiece coordinate systems within the machine coordinate system.



❖ The external workpiece coordinate system can be modified using the G50 or G10 L02 commands or manually. The modified values are stored in Sys2160 to Sys2199. Modification can be done either incrementally or absolutely.

❖ Incremental modification: Adds the incremental modification values to the current external workpiece coordinate system.

❖ Absolute modification: Directly changes the values of the external workpiece coordinate system, with the set data determining the new values.

Sample Program:

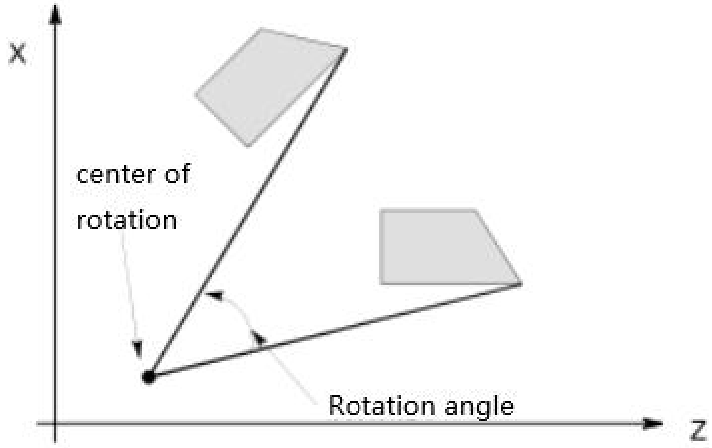
Program	Machine Coordinate (X, Y, Z)	Program Coordinate (X, Y, Z)	SYS2160, SYS2162
G0X0,Z0.	0.000,0.000	0.000,0.000	0,0
G0X100,Z50.	100.000,50.000	100.000,50.000	0,0
G50X0,Z20.	100.000,50.000	0.000,20.000	100.000,30.000
G0X50,Z50.	150.000,80.000	50.000,50.000	100.000,30.000
G5OU20,W-10.	150.000,80.000	70.000,40.000	80.000,40.000
G0X0,Z0.	80.000,40.000	0.000,0.000	80.000,40.000

5. When the system is powered on, it will default to a Workpiece Coordinate System, which is set by Mcm05. For example, if Mcm05=54, the default workpiece coordinate system on power-on is G54. If Mcm05=5401, the default workpiece coordinate system on power-on is G54.1P1.
6. You can use Mcm06 BIT01 to set the behavior of the workpiece coordinate system during a reset. When Mcm06 BIT01=0, the current workpiece coordinate system is retained after a reset. When Mcm06 BIT01=1, the workpiece coordinate system is reset to the default coordinate system set on power-on. For example, if Mcm05=54 and the current workpiece coordinate system is G55, after a reset with Mcm06 BIT01=0, the current workpiece coordinate system remains as G55. But after a reset with Mcm06 BIT01=1, the current workpiece coordinate system changes to the default G54 coordinate system.
7. You can use Mcm07 BIT00 to set the behavior of the workpiece coordinate system during program startup. When Mcm07 BIT00=0, the current workpiece coordinate system is retained when a program starts. When Mcm07 BIT00=1, the workpiece coordinate system is reset to the default coordinate system set on power-on when a program starts. For example, if Mcm05=54 and the current workpiece coordinate system is G55, after starting a program with Mcm07 BIT00=0, the current workpiece coordinate system remains as G55. But after starting a program with Mcm07 BIT00=1, the current workpiece coordinate system changes to the default G54 coordinate system.
8. The currently active workpiece coordinate system can be displayed using Sys10256.

2.36 G68, G69 Plane Rotation

- **Function and Purpose**

When machining complex shapes on a rotated position, programming can be done based on the shape before rotation. By using coordinate rotation instructions, the rotation angle can be specified to machine the shape after rotation.



- **Instruction Format**

G17 G68 X_ Y_ R_: G17 plane-specific rotation

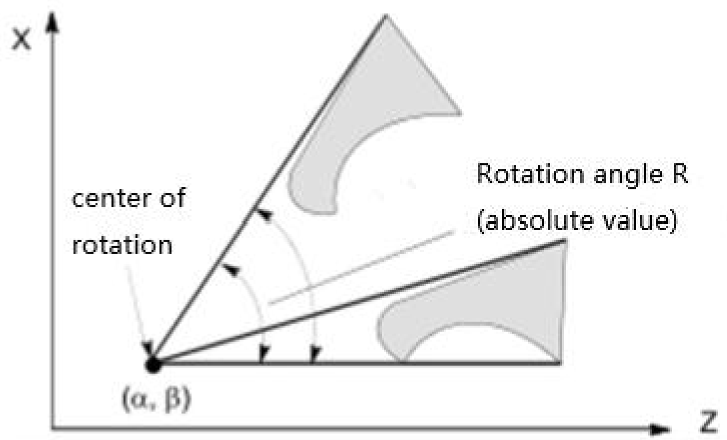
G18 G68 Z_ X_ R_: G18 plane-specific rotation

G19 G68 Y_ Z_ R_: G19 plane-specific rotation

G69: Cancel coordinate rotation

X_, Y_, Z_: Specify the rotation center, which can only be specified in absolute coordinates and is affected by the radius.

R_: Specify the rotation angle with a fixed radius, where a positive value represents counterclockwise rotation. Unit: 0.001°, Range: -360° to 360°.



- Detailed Explanation:

1. The rotation center must be specified in absolute coordinates.
2. The G17, G18, G19 planes cannot be specified within the rotation plane.
3. The G28, G29, G30 commands can only be specified after canceling the rotation plane function.
4. The G50, G54-G59, T, and other commands cannot be specified within the rotation plane.
5. Coordinate rotation cannot be used in single-type fixed cycles, compound fixed cycles, or drilling fixed cycles.
6. In the movement commands following the coordinate rotation instruction G68 or the coordinate rotation cancellation instruction, absolute values must be specified to avoid movement abnormalities.
7. G68 and G69 can be specified in tool tip radius compensation.
8. If the rotation center is not specified, the current point serves as the rotation center.

- **Program Example**

G17;

G00 X0. Y0.

G68 X0. Y0. R45.

G01 X20. Y20. F1500

X40.

Y40.

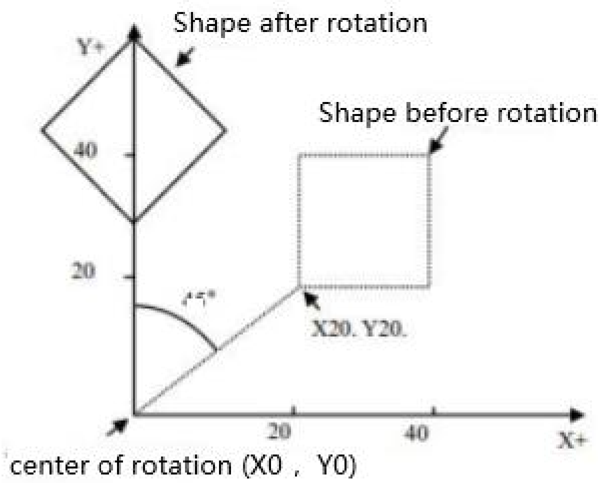
X20.

Y20.

G69

M30

The trajectory is as follows：



2.37 G73 High-Speed Deep Hole Drilling Fixed Cycle

- **Function and Purpose**

This cycle is used for high-speed deep hole drilling operations. The cycle performs intermittent cutting feed to reach the bottom of the hole. During the drilling process, a small distance is retracted to clear the metal chips from the hole.

- **Parameter Overview**

SYS 3020

Definition: Specifies the G73 high-speed deep hole drilling cycle D value.

Range: 0 to 9999999

Unit: μm

Default Value: 0

Effective Mode: Real-time

SYS 3019 BIT02

Definition: Handling of K value in the high-speed deep hole drilling cycle.

Range: Bit type

Unit: null

Default Value: 0

Effective Mode: Activated by C4

- **Explanation**

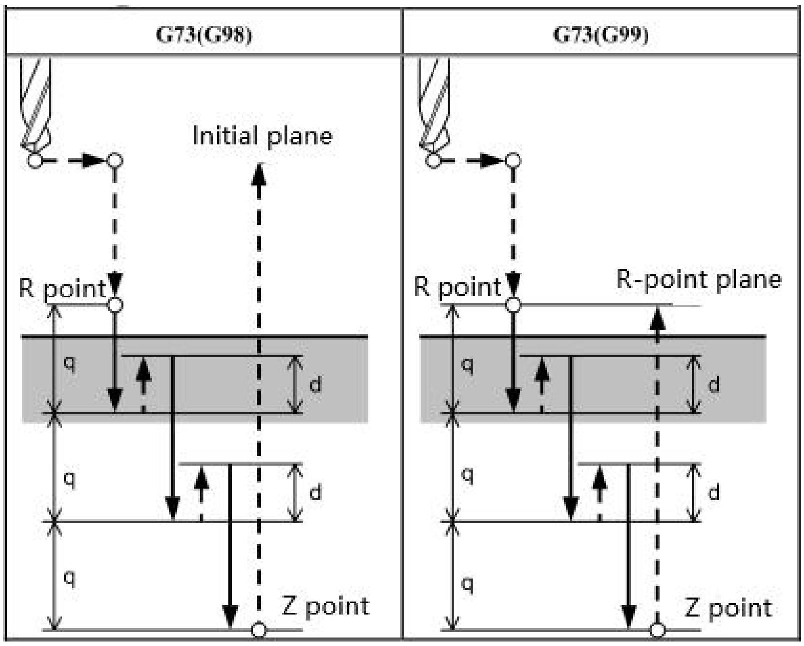
BIT02=0: When specifying K0, the drilling/boring/tapping operation is not performed, only the drilling/boring/tapping data is stored.

BIT02=1: When specifying K0, the drilling operation is performed only once.

- **Instruction format**

G73 X_ Y_ Z_ R_ Q_ P_ F_ K_

X Y	Specification of the Initial Point of the Cycle (G90 Absolute/G91 Incremental)
Z	Distance from the R-point to the Bottom of the Hole
R	Distance from the Initial Plane to R, G90 Absolute/G91 Incremental, Modal Instruction
Q	Specification of the Cut-in Amount for Each Pass, Incremental, Modal Instruction, Unit: um
P	Specification of the Pause Time at the Bottom of the Hole, Equivalent to G04 P_, Modal, Unit: ms
F	Specification of the Drilling Feed Rate, Modal Instruction
K	Specification of the Number of Repeats, Not Set Defaults to 1, Not a Modal Instruction (Parameter can be set to handle k=0 case)



- **Note**

1. When specifying M code on the same line as G73 and having a K value, the M code will only execute once.
2. Tool length compensation (G43/G44/G49) takes effect when positioning to the R point.

- **Program Example**

M3 S30000

G90 G00 X0.Y0.

G90 G73 G99 X30.Y20.Z-150. R-50.Q15.F240 ; Positioning, drill hole 1, then return to R point

Y30. ; Positioning, drill hole 2, then return to R point

Y40. ; Positioning, drill hole 3, then return to R point

Y50. ; Positioning, drill hole 4, then return to R point

G98 Y60. ; Positioning, drill hole 5, then return to initial plane

G80 G28 G91 X0 Y0 Z0 ; Cancel G73 modal, return to reference point

M5 ; Stop spindle.

2.38 G76 Fine Boring Fixed Cycle

- **Function and Purpose**

This cycle is used for high-precision boring operations. The spindle reaches the bottom of the hole and stops accurately, and the tool quickly retracts from the workpiece surface.

- **Parameter Overview:**

SYS 3019 BIT02

Definition: Handling of K value in the drilling/boring/tapping cycle

Range: 0 to 2^31

Unit: Bit type

Default value: 0

Effective mode: Activated by C4

Explanation:

BIT02=0: When specifying K0, no drilling/boring/tapping operation is performed, only the drilling/boring/tapping data is stored.

BIT02=1: When specifying K0, only one drilling operation is performed.

SYS 3014

Definition: Specifies the direction of OSS (eccentricity)

Range: -3 to 3

Unit: N/A

Default value: 0

Effective mode: Activated by C4

- **Explanation**

#S3014	Specifies the direction of OSS (eccentricity)
0/1	+X
-1	-X
2	+Y
-2	-Y
3	+Z
-3	-Z

SYS 3170~ SYS 3179

Definition: Specifies the M code for spindle stop during boring

Range: 0 to 99999999

Unit: N/A

Default value: 19

Effective mode: Activated by C4

Explanation:

SYS3170 = Specifies the M code for the first spindle stop

SYS3171 = Specifies the M code for the second spindle stop

……

- **Instruction format**

G76 X_ Y_ Z_ R_ Q_ P_ F_ K_

X, Y: Specification of the initial point of the cycle (G90 absolute/G91 incremental)

Z: Distance from R-point to the bottom of the hole

R: Distance from the initial plane to R, using G90 absolute/G91 incremental (modal command)

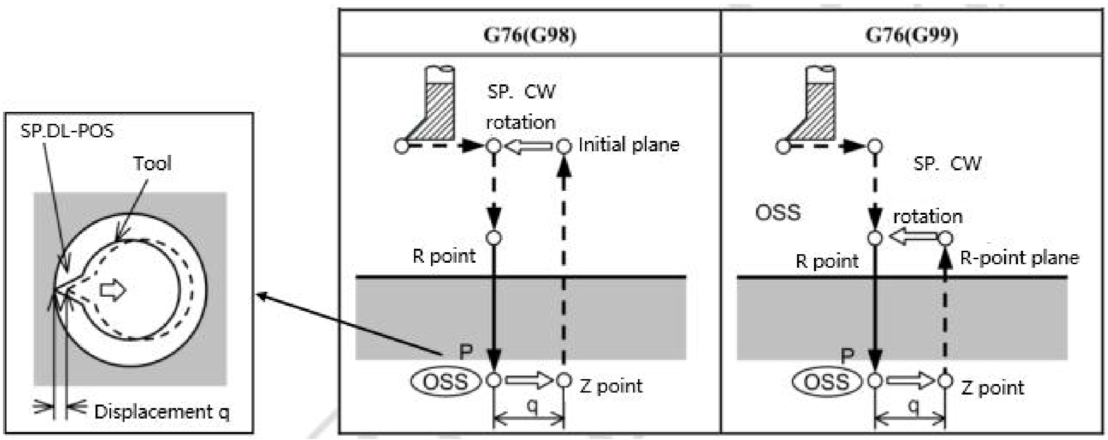
Q: Hole bottom displacement (axial displacement controlled by parameter SYS 3014), modal command, positive value (negative sign ignored)

P: Specifies the pause time at the bottom of the hole, equivalent to G04 P_ (modal), unit: ms

F: Specifies the feed rate for boring, modal command

K: Specifies the number of repetitions, assumed to be 1 if not specified, not a modal command (system processing when parameter k=0)

- **Boring Hole Illustration**



- **Action Explanation**

1. XY positioning.
2. Positioning to the R point (Ensure the spindle is functioning properly before boring).

3. Boring action from the R point to the bottom of the hole.
4. Pause for P milliseconds.
5. Spindle stops at the bottom of the hole (The system outputs the spindle stop M code first, then the spindle align M code specified by parameter #S3170).
6. Tool displacement by the distance of eccentricity Q.
7. Rapid retraction of the tool.
8. Tool retracts in the opposite direction by the distance of eccentricity Q (i.e., -Q).
9. Spindle resumes normal rotation (Outputs Mxx based on the spindle's forward rotation M code #S3140).

- **Note**

1. When specifying M code on the same line as G76 and having a K value, the M code will only execute once.
2. If there is no X, Y, Z, R, or other axial motion specified in the modal group, the boring operation will not be performed.
3. Tool length compensation (G43/G44/G49) takes effect when positioning to the R point.
4. G41/G42/G40 are not valid during the boring operation.
5. Other motion G codes on the same line as G8x will cancel G8x.

- **Program Example**

M3 S800

G90 G00 X300.Y400.

G90 G76 G99 X30. Y20. Z-150. R-50. Q1. F240 ; Positioning, perform boring hole 1, then return to R point

Y30. ; Positioning, perform boring hole 2, then return to R point

Y40. ; Positioning, perform boring hole 3, then return to R point

Y50. ; Positioning, perform boring hole 4, then return to R point

G98 Y60. ; Positioning, perform boring hole 5, then return to initial plane

G80 G28 G91 X0 Y0 Z0 ; Cancel G76 modal, return to reference point

M5 ; Stop spindle.

2.39 G81 Drilling Cycle

- **Function and Purpose**

This cycle is used for drilling operations.

- **Instruction Format**

G81 X_ Y_ Z_ R_ F_ K_

X_ Y_: Specifies the initial point of the cycle (G90 for absolute coordinates/G91 for incremental coordinates).

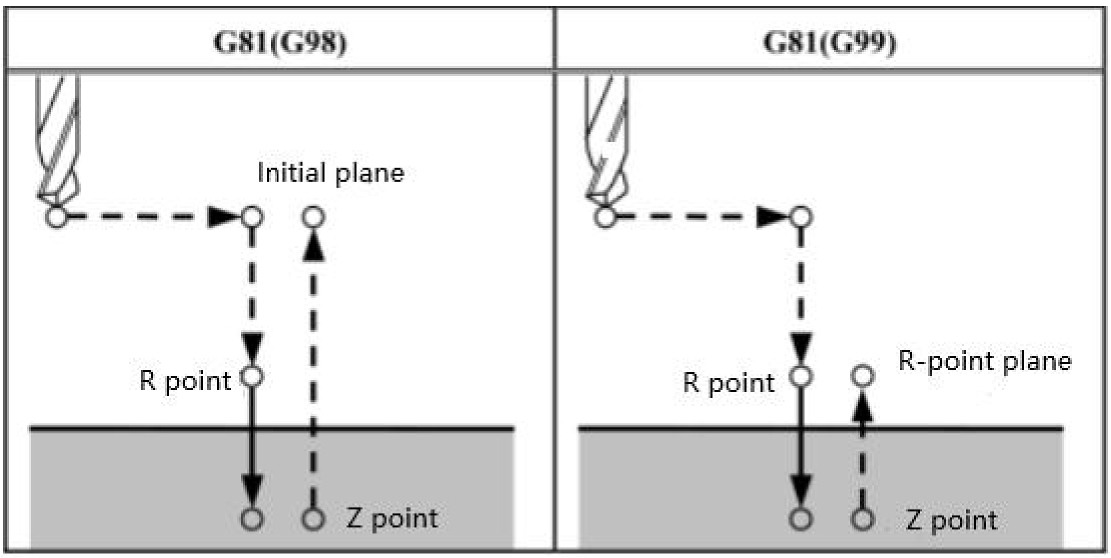
Z: Distance from the R point to the bottom of the hole.

R: Distance from the initial plane to the R point, specified using G90 for absolute coordinates or G91 for incremental coordinates (modal command).

F: Specifies the drilling feed rate, modal command.

K: Specifies the number of repetitions. If not specified, the default is 1. It is not a modal command (system behavior can be defined when k=0).

- **Drilling Diagram**



- **Note**

1. When specifying M code on the same line as G81 and having a K value, the M code

will only execute once.

2. If there are no X, Y, Z, R, or other axis values in the modal group, drilling will not be performed (to be confirmed).
3. Tool length compensation (G43/G44/G49) takes effect when positioning to the R point.
4. Adding a Q value in the program is invalid.
5. Adding a P value in the program is invalid.
6. Please set SYS3019 BIT3=1 and SYS3026 BIT0=1. R can be specified as absolute or incremental based on G90/G91.

- **Program Example**

M3 S30000

G90 G00 X0.Y0.

G90 G81 G99 X30. Y20. Z-150. R-50. F240 ; Positioning, drill hole 1, then return to R point

Y30. ; Positioning, drill hole 2, then return to R point

Y40. ; Positioning, drill hole 3, then return to R point

Y50. ; Positioning, drill hole 4, then return to R point

G98 Y60. ; Positioning, drill hole 5, then return to initial plane

G80 G28 G91 X0 Y0 Z0 ; Cancel G81 modal, return to reference point

M5 ; Stop spindle.

2.40 G82 G82 Dwell Drill Cycle

- **Instruction format**

G82 X_ Y_ Z_ R_ P_ F_ K_

X Y: Specification of initial point of the cycle (G90 absolute/G91 incremental).

Z: Distance from R point to the bottom of the hole.

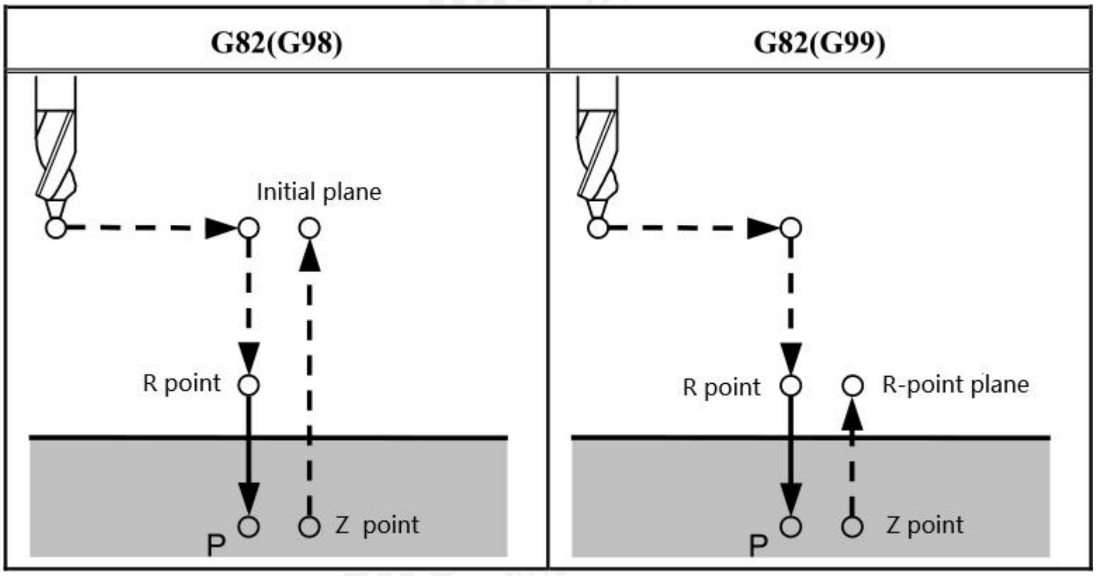
R: Distance from initial plane to R point, G90 absolute/G91 incremental, modal command.

P: Specifies the dwell time at the bottom of the hole, equivalent to G04 P_ (modal), unit: ms.

F: Specifies the drilling feed rate, modal command.

K: Specifies the number of repetitions, assumed to be 1 if not specified, not a modal command (system's handling can be set when k=0).

- **Drilling Diagram**



- **Notes**

1. When specifying M code on the same line as G82 and having a K value, the M code will only execute once.
2. In modal state, if there is no specification of X, Y, Z, R, or other axes, drilling will not be performed.
3. Tool length compensation G43/G44/G49 takes effect when positioning to the R point.
4. The Q value is ineffective in programming.

- **Program Example**

M3 S30000

G90 G00 X0. Y0.

G90 G82 G99 X30. Y20. Z-150. R-50. F240 ; Positioning, drill hole 1, then return to R point

Y30. ; Positioning, drill hole 2, then return to R point

Y40. ; Positioning, drill hole 3, then return to R point

Y50. ; Positioning, drill hole 4, then return to R point

G98 Y60. ; Positioning, drill hole 5, then return to initial plane

G80 G28 G91 X0 Y0 Z0 ; Cancel G82 modal, return to reference point

M5 ; Stop spindle

The difference between G82 and G81 instructions is that G82 includes a dwell time (P) after drilling to the bottom of the hole before retracting the tool. The dwell time is specified in milliseconds as an integer input.

2.41 G83 and G87 Deep Hole Drilling Cycles

- **Instruction format**

G83 X/U_C/H_Z/W_Rr_Qq_Pp_Ff_Kk_Mm_Ii_;

or G87 Z/W_C/H_X/U_Rr_Qq_Pp_Ff_Kk_Mm_Ii_;

G83: Face Deep Hole Drilling Cycle, modal command

G87: Longitudinal Deep Hole Drilling Cycle, modal command

X/U: Specification of initial point (absolute/incremental)

Z/W: Specification of initial point (absolute/incremental)

Z/W_ or X/U_: Specification of hole bottom position

C/H: Angle of the C-axis

Rr_: Distance from initial plane to R, in regular format: incremental, radius, ignore symbol; in special format: can specify semi-diameter, incremental programming, modal command

Qq_: Specification of feed per revolution, specified as incremental radius value (in Fanuc, for drilling it's non-modal, for tapping it's modal), unit: micrometer (um)

Pp: Pause time at the bottom of the hole, equivalent to G04 P_ effect (modal), unit:

milliseconds (ms)

Ff_: Specification of drilling feed rate, modal command

Kk_: Specification of repeat count, not specified defaults to 1, not a modal command (system behavior can be set when k=0)

Mm_: C-axis clamping M code, not a modal command

Ii_: Specification of drilling axis in special format, when I is specified, other axes are positioning axes (effective only when using special format)

- **Detailed Explanation**

1. The drilling cycle is divided into three types: regular fixed drilling cycle, deep hole fixed drilling cycle, and high-speed deep hole fixed drilling cycle. When Q is not specified, it is a regular fixed drilling cycle. When Q is specified, the deep hole fixed drilling cycle or high-speed deep hole fixed drilling cycle is determined based on the setting of Sys3019 BIT00.
2. The instruction for specifying the initial point of the hole position is non-modal. When continuously issuing G83 (G87) commands, the initial point of the hole position needs to be specified in each program segment.
3. When the specified Q value is greater than the total cutting feed, it follows the regular fixed drilling cycle. In this case, the cutting feed reaches the bottom of the hole, and the tool rapidly retracts from the bottom of the hole.
4. When K is specified, the M code is executed only when initially positioning to the initial plane. It is not executed during repetitions.
5. There is a positioning check function for the hole bottom and returning to the R point or starting point.
6. Regarding Qq_ expectation as a modal command, it is relatively safer for the tool. It is possible that some customers may misunderstand it as a modal command and not specify Q, resulting in a single feed.

- **Basic control elements for specifying the plane axis and drilling axis in a fixed drilling cycle are the positioning plane and the drilling axis.**

1. **General Format:**

Sys3019 BIT03 = 0

The basic axis I/J/K and parallel axis are specified by parameters like Mcm1880.

Assuming I/J/K as the basic axes for X/Y/Z,

For face drilling (G83), the first plane axis is assumed to be the drilling axis, and the other axes are assumed to be positioning axes.

SYS 3019	Plane Selection	Drilling Axis
BIT01=1	G17 Xp-Yp	Xp
	G18 Zp-Xp	Zp
	G19 Yp-Zp	Yp
BIT01=0	——————————	Zp

For side drilling (G87), the second plane axis is assumed to be the drilling axis, and the other axes are assumed to be positioning axes.

SYS 3019	Plane Selection	Drilling Axis
BIT01=1	G17 Xp-Yp	Yp
	G18 Zp-Xp	Xp
	G19 Yp-Zp	Zp
BIT01=0	——————————	Xp

G17 Plane Command Format:

G83 Y/V_C/H_X/U_Rr_Qq_Pp_Ff_Kk_Mm_Ii_;

Or G87 X/U_C/H_Y/V_Rr_Qq_Pp_Ff_Kk_Mm_Ii_;

G18 Plane Command Format:

G83 X/U_C/H_Z/W_Rr_Qq_Pp_Ff_Kk_Mm_Ii_;

Or G87 Z/W_C/H_X/U_Rr_Qq_Pp_Ff_Kk_Mm_Ii_;

G19 Plane Command Format:

G83 Z/W_C/H_Y/V_Rr_Qq_Pp_Ff_Kk_Mm_Ii_;

Or G87 Y/V_C/H_Z/W_Rr_Qq_Pp_Ff_Kk_Mm_Ii_;

2. Special Format:

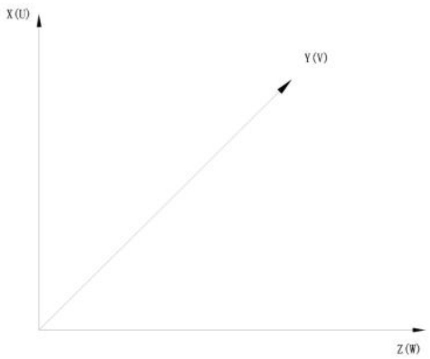
Sys3019 BIT03 = 1

The basic axis I/J/K and parallel axis are specified by parameters like Mcm1880.

Assuming I/J/K as the basic axes for X/Y/Z,

The positioning plane is determined by the plane selection command G17, G18, or G19. The drilling axis is the axis perpendicular to the selected plane (X, Y, Z, or their parallel axes).

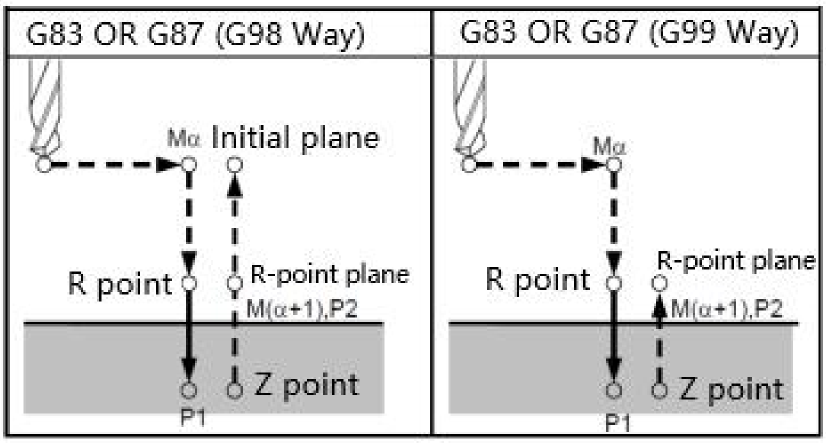
For example, if G17 is specified for the XY plane, then XY is the positioning plane, and the Z axis is the drilling axis.



- **Diagram Explanation of Drilling Fixed Cycle Action**

1. Drilling Fixed Cycle - Regular Drilling Cycle:

If the cutting amount Q is not specified, the drilling process follows the regular drilling cycle. The cutting feed reaches the bottom of the hole, pauses for P1 seconds, and then the tool rapidly retracts from the hole bottom.



Mα: M code for braking the C-axis.

M(α+1): M code for releasing the C-axis brake.

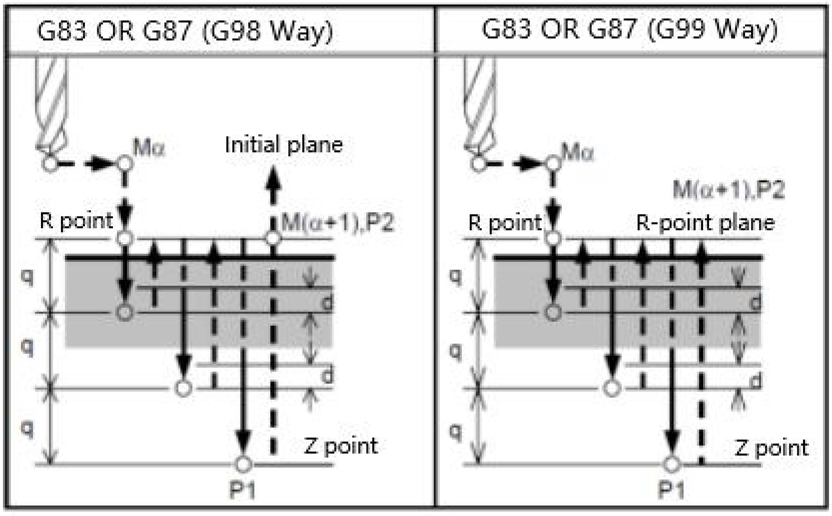
P1: Pause specified in the program.

P2: Pause time set by parameters (ensuring completion of positioning check and releasing of main spindle brake).

Note:

Regarding the retract point: For lathes, the final retraction is to the starting point; for milling machines, it is controlled by G98 or G99.

3. Deep hole drilling cycle



Mα: M code for braking the C-axis.

M(α+1): M code for releasing the C-axis brake.

P1: Pause specified in the program.

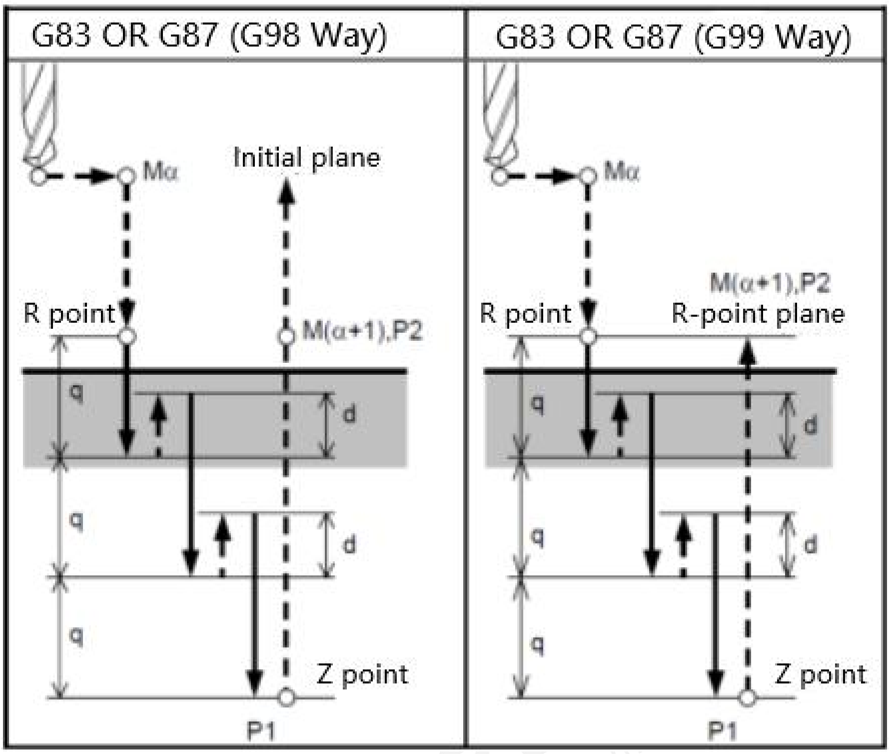
P2: Pause time set by parameters (ensuring completion of positioning check and releasing of main spindle brake).

Note:

Regarding the retract point: For lathes, the final retraction is to the starting point; for milling machines, it is controlled by G98 or G99.

4. High-speed deep hole drilling cycle

This cycle is used for high-speed deep hole machining operations. In this drilling method, the following cycle is repeated: intermittent cutting feed with a specified retract amount at high speed before reaching the bottom of the hole, while continuously evacuating chips from the hole.



Mα: M code for braking the C-axis.

M(α + 1): M code for releasing the C-axis brake.

P1: Program-specified pause.

P2: Parameter-defined pause time (ensures positioning check and release of main spindle brake are completed).

d: Parameter-defined retract distance.

P3: Pause time for each feed (pause in special format).

Note:

Regarding the retract point, in a lathe, the final tool retracts to the starting point, while in a milling machine, it is controlled by G98 and G99.

Detailed Explanation and Precautions of Drilling Fixed Cycle Actions

Part 1: Positioning Preparation

Steps	Explanation
N01：	Positioning to the starting point of the rotation hole (Initial Plane) using

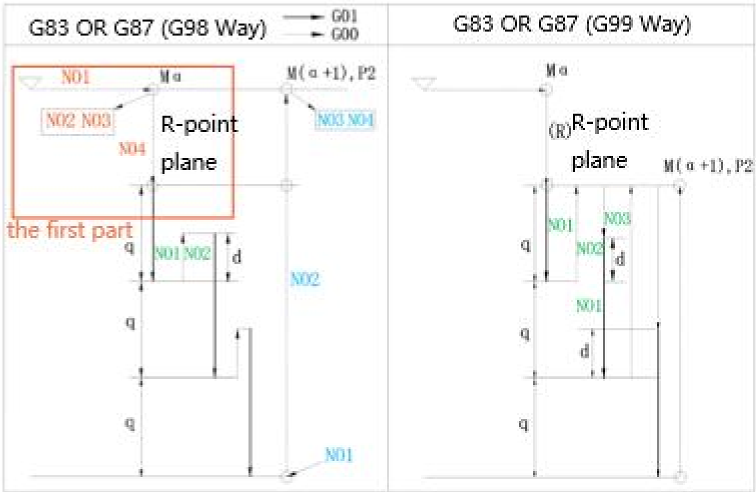
	G00.
N02:	The C-axis is positioned to the specified angle. This step is skipped if no C value is specified.
N03:	Applying the main spindle brake with the specified M-code. This step is skipped if no M-code is specified.
N04:	Positioning to the safe plane at point R. If there is no R value specified, this step is skipped, and no positioning occurs (G00).

Note:

When executing G83 or G87 continuously, this part does not include NO1.

System functionality processing for each step.

Step	N01	N02	N03	N04
Execute Single Block	Can	Can	Can	Can
Pause	Can	Can	Can	Can
Handwheel Prediction	Can	Can	Can	Can
Change MFO	Can	Can	Can	Can
Non-Stop Mode Function	No	No	Yes	Yes
Wait for Grid Point?	No Waiting	No Waiting	No Waiting	No Waiting
G92 Acceleration/Deceleration Time?	No Use	No Use	No Use	No Use



Part 2: Feed Preparation

1. When Q value is not 0 or Q is less than the hole depth:

Steps	Explanation
N01：	Feed the cutter to [nq+R]. If [nq+R] exceeds the endpoint Z, then feed to Z and conclude the second part of the process.
N02:	During high-speed deep hole drilling cycles: Rapid retract the tool to a distance d from the current point. In deep hole drilling cycles, retract the tool to point R.
N03:	During high-speed deep hole drilling cycles: Execute N01. In deep hole drilling cycles: Rapid approach to position [n*q+R-d], then execute N01. Repeat this process until reaching the bottom of the hole, concluding the second part of the process.

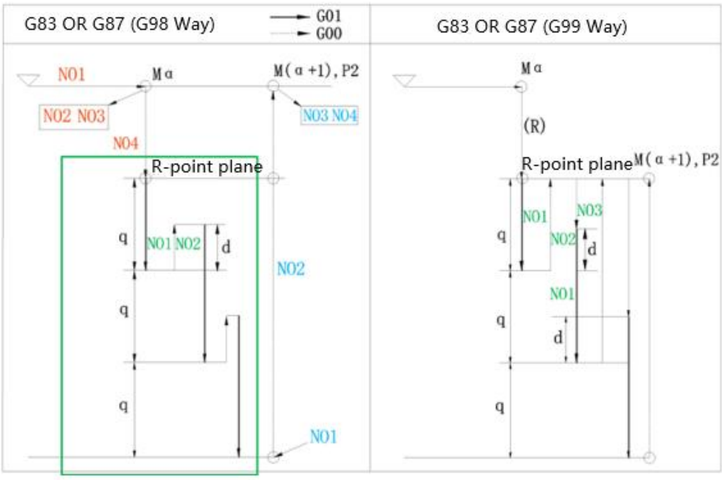
2. When Q value is 0 or Q is greater than the hole depth, or Q is not specified:

Steps	Explanation
N01：	Feed to Z using G01 to conclude the second part of the process.

Processing of system functions for each step:

Step	N01	N02	N03

Execute Single Block	Can't	Can't	Can't
Pause	Can't	Can't	Can't
Handwheel Prediction	Can't	Can't	Can't
Change MFO	Can	Can	Can
Non-Stop Mode Function	No	No	No
Wait for Grid Point?	No Waiting	No Waiting	No Waiting
G92 Acceleration/Deceleration Time?	No Use	No Use	No Use



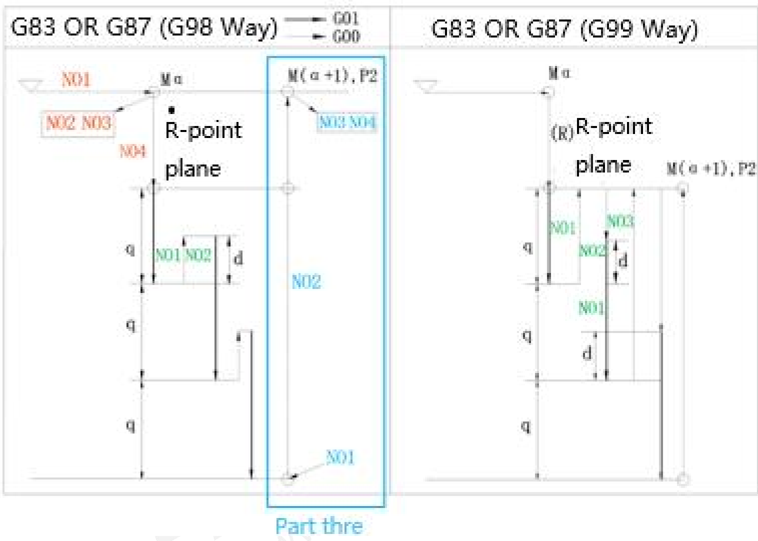
Part 3: Retraction Preparation

System functionality processing for each step.

Steps	Explanation
N01：	Pause at hole bottom using G04 P1.
N02:	When in G98 mode, rapid feed to initial plane using G00. In G99 mode, rapid feed to R point using G00.
N03:	If Mα exists, execute M(α+1).
N04:	If P2 is set, then perform dwell using G4 P2, ending the third part.

Step	N01	N02	N03	N04
Execute Single Block	Can't	Can	Can	Can
Pause	Can't	Can't	Can	Can

Handwheel Prediction	Can	Can	Can	Can
Change MFO	Can	Can	Can	Can
Non-Stop Mode Function	No	No	No	No
Wait for Grid Point?	No Waiting	No Waiting	No Waiting	No Waiting
G92 Acceleration/Deceleration Time?	No Use	No Use	No Use	No Use



- **Program Example**

M3 S30000

G90 G00 X0.Y0.

G90 G83 G99 X30. Y20. Z-150. R-50. F240 ;Positioning, drill hole 1 and return to R point

Y30. ;Positioning, drill hole 2 and return to R point

Y40. ;Positioning, drill hole 3 and return to R point

Y50. ;Positioning, drill hole 4 and return to R point

G98 Y60. ;Positioning, drill hole 5 and return to initial plane

G80 G28 G91 X0 Y0 Z0 ;Cancel G83 modal, return to reference point

M5 ;Stop spindle

2.42 G84, G74 Tapping Cycle

1. Types of Tapping Cycles:

Tapping cycles can be classified into two types based on their working principles:

- (1) Flexible tapping, which includes high-speed rigid tapping or deep hole tapping.
- (2) Rigid tapping, which includes high-speed rigid tapping or deep hole tapping.

Flexible Tapping:

In flexible tapping, the spindle can rotate or stop while moving along the tapping axis. Tapping is performed using auxiliary functions (M03, M04, M05). During the deceleration of the spindle at the bottom of the hole, the tapping axis experiences stretching action. Therefore, a floating tap holder or a tension-compression tap holder should be used. If a rigid tap holder is used, it may cause the tap to break.

Rigid Tapping:

In rigid tapping, the spindle motor is controlled in the same manner as a servo motor (i.e., the spindle uses position control). Tapping is performed by the spindle motor and interpolation between the tapping axis and the spindle. For each rotation of the spindle, the tapping axis advances a certain distance (lead). Rigid tapping does not require the use of a floating tap holder or a tension-compression tap holder. A standard tap holder can be used.

2. Tapping Cycle Specification:

Rigid Tapping:

- **M code specifies rigid tapping, MxxSxxxx**

Note:

a. The M code is specified by Sys3034 (default is M29 when Sys3034=0).

b. The tapping command M code can be set before the tapping instruction or within the same program block as the tapping instruction.

c. M29 specifies rigid tapping, and it is canceled when encountering G80 or G01. To specify rigid tapping again, M29 needs to be redefined.

❖ **Setting Sys3031 BIT00=1 indicates that M code instructions are not required, and G84 (or G88) is fixed as rigid tapping.**

❖ **Instruction Format for Specifying Rigid Tapping (with parameter settings):**

a. Specify M29 before the G84 (or G88) tapping program block:

M29

G84 X C R P F K (M);

X C;

G80

b. Specify M29 within the G84 (or G88) tapping program block; in this case, do not specify the M code for clamping the C-axis within the G84/G88 program block:

G84 X_ C_ Z_ R_ P_ F_ K_ M29 S_;

X_ C_;

G80;

c. Use G84/G88 as the G code for rigid tapping (configured by Sys3031 BIT00); in this case, G84/G88 can only be used for rigid tapping and cannot be used for regular tapping:

G84 X_ C_ Z_ R_ P_ F_ K_ M ;

X C_;

G80

Note: It is not allowed to specify S or axis movement instructions between M29 and the G84/G88 program block, as it may result in an alarm.

● **Instruction Formats:**

a. Forward Tapping (Spindle Rotating in the Forward Direction):

G84 X/U_C/H_Z/W_Rr_Qq_Pp_Ff_Kk_Mm_Jj_Dd

b. Reverse Tapping (Spindle Rotating in the Reverse Direction):

G74 Z/W_C/H_X/U_Rr_Pp_Qq_Ff_Kk_Mm_Jj_Dd

1. Letter Definitions

G84(G74)	G84 - Forward Deep Hole Tapping Cycle G74 - Reverse Deep Hole Tapping Cycle
X/U(Z/W) _C/H_	Specification of Initial Point (Absolute/Incremental Values) Angle of C-axis
Z/W(X/U)_	Position of Hole Bottom (Absolute Value/Incremental Value from R-point) If specified as Z_, it represents the absolute coordinate value from the starting point to the hole bottom position. If specified as W_, it represents the incremental value from the R-point to the hole bottom position.
Rr_	Specification of R-point (Modal with Incremental Radius Value) If no R-value is set, the R-point plane coincides with the initial plane.
Qq_	Specification of Cutting Feed per Pass (Modal with Incremental Radius Value, ignoring sign)
Pp_	Specification of Pause Time at the Hole Bottom, equivalent to G04 P_ (Modal) Unit: milliseconds Real-time modification of the value of Sys3022
Ff_	Specification of Feed Speed (Modal)
Kk_	Specification of Repeat Count 0-9999 (Default: 1) (Non-modal)
Mm_	Specification of Spindle Brake Command (Non-modal)
Dd	Specification of Spindle Used in Tapping Cycle (Modal) (Range: 1 to the number of spindles)
Jj	Specification of Spindle Speed during Retraction (only valid in rigid tapping), remains effective until the cycle is canceled

2. Detailed Explanation of Letters

a. Positioning Hole Location

X(U)/Z(W) C(H)_: Set the current hole positioning location, specified as radius/diameter.

b. Hole Bottom Position

1) Z(W)/X(U): Set the distance from the R-point to the hole bottom, specified as radius/diameter.
2) Z/X Absolute Specification: Represents the absolute coordinate value from the starting point to the hole bottom position.
3) W/U Incremental Specification: Represents the incremental value from the R-point to the hole bottom position.

c. R-point plane

R_: The distance from the initial plane to the R-point, specified as a radius, modal (only effective when specified in the tapping command), can include decimal points.

When R=0, the R-point plane coincides with the initial plane, and the R-point setting is aligned with the tapping direction (independent of the positive or negative value of R).

Program Example:

G84 X5. Z-50. R-5. P10000. Q10. ← Tapping cycle with R-point plane

G84 X5. Z-60. ← Tapping cycle with R-point plane

X5. Z-65. ← Tapping cycle with R-point plane

G80 ← Cancel tapping cycle

G84 X5. Z70. ← Tapping cycle with initial plane

G80 ← Cancel tapping cycle

d. P_ dwell at the bottom

P_: Dwell time at the bottom of the hole, without decimal points, used in the same way as the G04 command, set by Sys3022.

Unit: ms

Relationship between Sys3023 and the specified P.

Hole Bottom positioning check Sys3032 BIT03	R-Point positioning check Sys3032 BIT02	Hole bottom dwell time
0	0	P value and Sys3023 specified together, whichever is larger takes effect. When both are set to 0, there is no dwell at the hole bottom.
0	1	P value and Sys3023 specified together, whichever is larger takes effect. When both are set to 0, there is no dwell at the hole bottom.
1	0	P value and Sys3023 specified together, whichever is larger takes effect. When both are set to 0, there is no dwell at the hole bottom.
1	1	P value and Sys3023 specified together, whichever is larger takes effect. When both are set to 0, the dwell time is 10ms.

0: Positioning check adjustment function is disabled.

1: Positioning check adjustment function is enabled.

Program example:

G84 X5. Z-50.R-5.P10000Q10. ← Dwell at the hole bottom for 10 seconds

G84 X5. Z-60. ← Dwell at the hole bottom for 10 seconds

X5. Z-65. ← Dwell at the hole bottom for 10 seconds

G80 ← Cancel fixed cycle

e. Thread pitch F value

F_: F value specifies the thread pitch. In feed per minute mode, feed speed ÷ spindle speed = thread pitch.

F_: In feed per revolution mode, feed speed = thread pitch.

The following table provides annotations for specifying the F value.

F	Programming F: F value
Sγ	Specify spindle speed (set speed in the program)
S’	Current spindle speed (actual speed)
Sα	Spindle speed from the previous operation or startup speed
F’	Spindle speed = 1000r/min * F
Fmax	Maximum feed speed of the tapping axis
Fγ	Actual speed of the tapping axis
Fβ	Spindle speed Sγ * F
Fα	Sα* F

G95

❖ **Rigid tapping:**

Tapping axis speed: Fγ = S' * F

1) Assuming no specified spindle speed Sγ

   Fα ≤ Fmax

   S' = Sα

   If calculated: Fα > Fmax ---------> System alarm

2) Assuming specified spindle speed Sγ

   Fβ ≤ Fmax

   S' = Sγ

   If calculated: Fα > Fmax ---------> System alarm

● **Rigid tapping interpolation:**

The F value is the thread pitch, tapping axis speed Fγ = S' * F

1) Assuming no specified spindle speed Sγ

If F' ≤ Fmax, system default: S' = 1000r/min

If F' > Fmax, system automatically handles: S' = Fmax / F

2) Assuming specified spindle speed Sγ
   If Fβ ≤ Fmax, S' = Sγ

   If calculated: Fβ > Fmax, then S' = Fmax / F

f. Feed per revolution

Q_: Q value specifies the feed per revolution, radius designation, modal, unsigned.

Based on the Q value, threading can be classified into three types of threading cycles (standard threading cycle, deep hole threading cycle, high-speed deep hole threading cycle).

1) If Q axis is not specified or Sys3031 BIT01 = 1, it is a standard threading cycle.
2) If Q value is specified, Sys3031 BIT01 = 0, and Sys3031 BIT02 = 1, it is set as a deep hole threading cycle.
3) If Q value is specified, Sys3031 BIT01 = 0, and Sys3031 BIT02 = 0, it is set as a high-speed deep hole threading cycle.

- **Example Program**

Setting Sys3031 BIT02 = 0

G84 X5. Z-50.R-5.P10000Q10. ← High-speed deep hole threading cycle

G84 X5. Z-60. ← High-speed deep hole threading cycle

X5. Z-65. ← High-speed deep hole threading cycle

G80 ← Cancel fixed cycle

G84 X5. Z70. ← Standard threading

G80 ← Cancel fixed cycle

G84 X5. Z75.Q0 ← Standard threading

g. Repeat Count

1. K_: K value specifies the repeat count, ranging from 0 to 9999. Sys3019 BIT02 sets the CNC working status with K=0.
2. When performing cyclic drilling on a set of holes separated by equal distances

(G80-G89), the position of the first hole needs to be specified incrementally. The number of repetitions is specified using K.

3. If the position of the hole is specified in absolute mode, the drilling will repeat at the same position.
4. K is only effective within the program block where it is specified.

- **Example Program**

G00 X10. Z100.

G84 X5. Z-50.R-5.P10000 Q10. K2 ← High-speed deep hole threading cycle, repeated 2 times

G84 X5. Z-60. ← High-speed deep hole threading cycle, not repeated

X5. Z-65. ← High-speed deep hole threading cycle, not repeated

G80 ← Cancel fixed cycle

G84 U5. W-50.R-5.P10000 Q10. K2 ← High-speed deep hole threading cycle, processing two holes with equal spacing

-> Hole 1 (10.000, 100.000)

-> Hole 2 (15.000, 100.000)

h. C-axis Clamp and Release

The M-code used in the program to clamp and release the C-axis can be specified through Sys3015.

Note:

❖ When the workpiece spindle and threading spindle are coordinated for threading, there is no need to specify the C-axis clamp and release M-code.

❖ If the specified M-code in the program is inconsistent with the parameter setting, the parameter value will not be modified.

❖ If the specified M-code in the program is inconsistent with the parameter setting or if the program does not specify an M-code, the C-axis brake and clamp actions will not be activated.

❖ If the specified M-code in the program is consistent with the parameter setting, the C-axis brake and clamp actions will be activated.

Please note: Do not use M00, M01, M02, M30, M98, M99.

i. Threading Spindle

D_: D-value specifies the threading spindle used in the threading cycle (Mitsubishi system programming).

Range: 1 to the number of spindles.

When the D command is omitted, the current Master spindle is used as the threading spindle.

j. Retraction Amount

The retraction amount, denoted as 'd', is specified by Sys3033. It is specified as a radius and cannot exceed the set R-value.

k. C-axis Release Pause

1) The C-axis release pause is specified by Sys3023 and is used in the same way as G04. It pauses after retracting to the R-point and then performs the C-axis release. It is possible to specify a mechanical fix/release of the C-axis using an M-code. Two types of M-codes can be output by adding them in the program segment of G84(G74) for clamping.

l. Feedrate Setting (Only applicable in rigid tapping)

The feedrate for feed-in (G01) and spindle speed are fixed at 100% in rigid tapping. The feedrate for retraction can be planned by Sys3031 BIT03, BIT04, BIT05, and Sys3036. When Sys3031 BIT03 is set to 1 (effective for pull-back rate), specifying J in the G84(G74) program segment or setting a value in Sys3036 will make it effective. Otherwise, it will be ineffective.

The actual effective magnification achieved through parameter settings and instructions is shown in the following table：

<table>
<tr><td></td><td></td><td colspan="2">SYS 3031
BIT03=1 （Pull-out ratio is effective）</td><td colspan="2">SYS 3031
BIT03=0 （Pull-out ratio is invalid）</td></tr>
<tr><td></td><td></td><td>When H command is set.</td><td colspan="2">When the J command is set invalid.</td><td rowspan="3">100% (magnification value)</td></tr>
<tr><td colspan="2">Spindle speed command when pull-out specified by address "J" exists.</td><td>Program instruction (specified by J command).</td><td colspan="2" rowspan="2">SYS 3036 (multiplier value).</td></tr>
<tr><td colspan="2">Spindle speed command when pull-out specified by address "J" does not exist.</td><td>SYS 3036 (multiplier value).</td></tr>
</table>

- **Magnification signal**

When Sys3031 BIT10 (Feedrate magnification signal and magnification cancellation signal in rigid tapping) is set to "1" (active), the following magnification can be applied to the cutting/pulling actions in rigid tapping:

❖ Apply magnification through the feedrate magnification signal

❖ Cancel magnification through the magnification cancellation signal

The relationship between the magnification and each action is as follows:

❖ When the magnification cancellation signal = "0" during cutting: Magnification specified by the magnification signal = "1" (magnification adjustment canceled): 100%

❖ When the magnification cancellation signal = "0" during pulling: Magnification specified by the magnification signal = "1" (magnification adjustment canceled): 100% if pulling magnification is not valid, or the value specified by the pulling magnification if it is valid.

m. Jj sets the pulling speed.

The Jj command specifies the pulling speed and is only effective when Sys3031 BIT03 = 1.

3. Plane planning:

a. General format:

G84/G74 Command for Thread Tapping Axis Setting (Milling Machine)

SYS 3019 BIT01	Plane Selection	Drilling Axis
1	G17 Xp-Yp Plane	Zp
	G18 Zp-Xp Plane	Yp
	G19 Yp-Zp Plane	Xp
0		Zp

Xp: Parallel axis to the X-axis or X-axis itself.

Yp: Parallel axis to the Y-axis or Y-axis itself.

Zp: Parallel axis to the Z-axis or Z-axis itself.

b. Special Format:

The special format is applicable for non-standard machine customers. It is only effective when Sys3019 BIT03 is set to 1. The selection of the threading axis in the special format is determined by the following parameters:

- ❖ Sys3019 BIT03: Enable or disable the special format.
- ❖ Sys3024 BIT03: Axis selection for G80G89 special format drilling. 0 = Determined based on the plane, 1 = Determined based on the instruction Ii, 2 = Fixed to Z-axis.
- ❖ Sys3025 BIT00: Selectable drilling axis for X-axis in the G80G89 special format.
- ❖ Sys3025 BIT01: Selectable drilling axis for Y-axis in the G80~G89 special format.

SYS 3024	Plane Selection	Drilling Axis
0	G17 Xp-Yp Plane	Zp

	G18 Zp-Xp Plane	Yp
	G19 Yp-Zp Plane	Xp
1		Ii letter specifies the tapping axis.
2		Zp

......

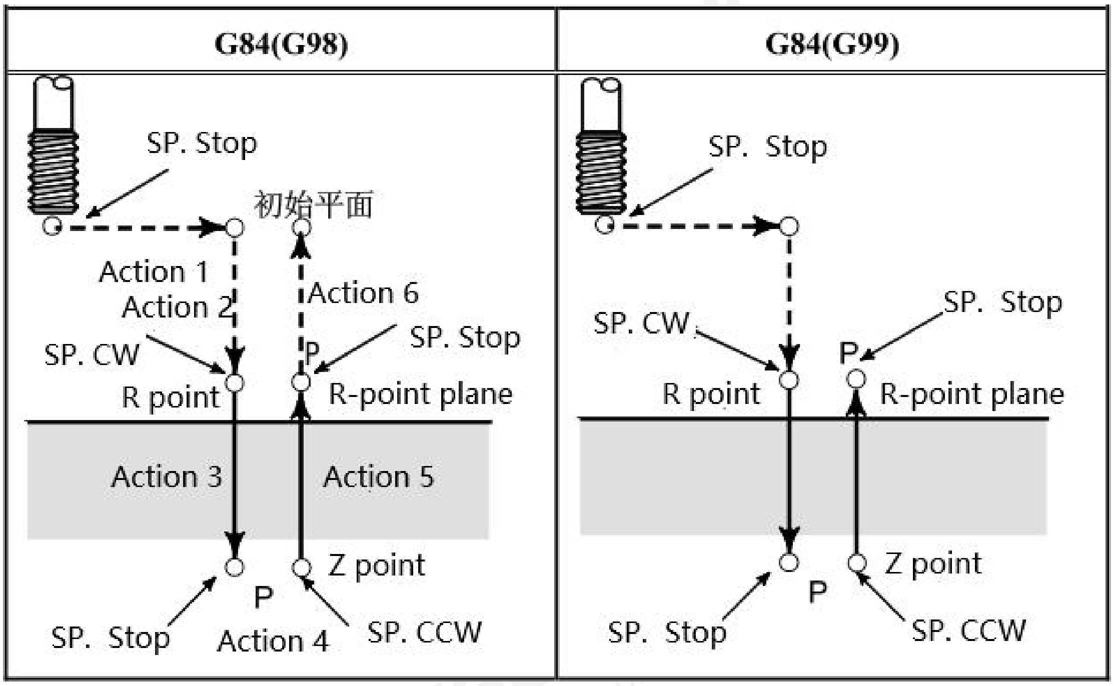
Xp: X-axis or parallel axis to the X-axis.

Yp: Y-axis or parallel axis to the Y-axis.

Zp: Z-axis or parallel axis to the Z-axis.

4. Threading Fixed Cycle Action Diagram

Single Pass Threading (No Q value specified, complete threading in one pass)



- **Action Description**

Action 1: X/Y G00 positioning (M-code can be output)

Action 2: Z G00 positioning to the R-point (skip if no spindle positioning is required)

Action 3: Forward threading feed, Z descends to the specified bottom position

Action 4: Pause programmed by address P

Action 5: Reverse retract to the R-point plane

Action 6: Ascend with G00 to the initial point (G98) or program R-point (G99)

- **The action commands are as follows**

Before specifying the threading command, the spindle rotation command must be specified (specify the first spindle or other spindles rotation based on the requirement).

1) Stop spindle (M05).
2) If there is a positioning command to the hole position, perform the positioning action (G00), otherwise skip.
3) If the R-point is specified, perform the positioning action to the R-point (G00). If there is a spindle positioning action, perform the spindle positioning. If no spindle positioning is required, continue execution.
4) Start spindle rotation (M03), the threading axis performs threading operation in cutting feed mode until reaching the bottom of the hole. The spindle and threading axis stop simultaneously at the bottom.
5) If a pause time (P) is specified, pause execution.
6) Output auxiliary function M04 for spindle reversal and wait for the completion signal of the M-code.
7) Upon receiving the completion signal, retract the threading tool in cutting feed mode until reaching the R-point (G01). The spindle and threading axis stop simultaneously at the R-point.
8) Return to the starting point in rapid traverse mode (G00) (this action is performed only in G98 mode).

   If a repetition count is specified, repeat from step 1).
9) Restore the spindle to its rotation state before threading.

- **Program Example**

M3 S500

G1 Z10.

G84 X10. Y10.Z-20.R0.P500 F500

X10.

G80

M5 S0

M30

5. General Peck Threading

Specify a non-zero Q value and set the parameters for deep-hole rigid tapping.

- **Action Description**

Action 1: X/Y G00 positioning (optional M-code output).

Action 2: Z G00 positioning to the R-point (skip if no spindle positioning is required).

Action 3: Start spindle rotation (M03) and perform threading feed in the positive direction. Z-axis descends by the depth of one cutting increment Q.

Action 4: Pause for the specified delay time (P).

Action 5: Retract the threading tool in the reverse direction until reaching the R-point plane.

Action 6: Pause for the specified delay time (P), then start spindle rotation (M03) and perform threading feed in the positive direction. Z-axis descends by the depth of one cutting increment Q relative to the current threading depth.

Action 7: Pause for the specified delay time (P).

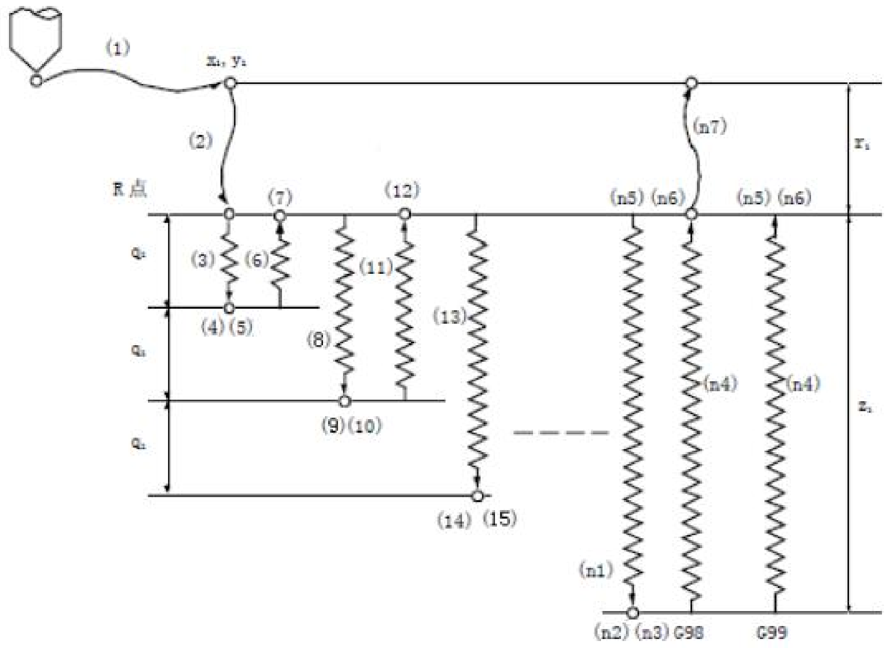
Action 8: Retract the threading tool in the reverse direction until reaching the R-point plane.

Action 9: Repeat the above threading actions until reaching the Z-point at the bottom of the hole.

Action 10: Pause for the specified delay time (P).

Action 11: Retract the threading tool in the reverse direction until reaching the R-point plane.

Action 12: Ascend in rapid traverse mode (G00) to the initial point (G98) or program R-point (G99).



- **Execution process as shown in the diagram:**

1) Start of machining: The tool moves to the designated (X, Y) position using G00.
2) Descend to the R-plane at the speed of G00.
3) Perform threading feed in the downward direction using G01 until reaching position Q1. Stop the spindle.
4) Pause for the specified time P.
5) Reverse the spindle rotation.
6) Rise to the R-plane using G01. Stop the spindle.
7) Start the spindle rotation.
8) Perform threading feed in the downward direction using G01 until reaching position Q2. Stop the spindle.
9) Pause for the specified time P.
10) Reverse the spindle rotation.
11) Rise to the R-plane using G01. Stop the spindle.
12) Start the spindle rotation.
13) Perform threading feed in the downward direction using G01 until reaching position Q3. Stop the spindle.

...

n1) Perform threading feed in the downward direction using G01 until reaching the bottom of the hole. Stop the spindle.

n2) Pause for the specified time P.

n3) Reverse the spindle rotation.

n4) Rise to the R-plane using G01. Stop the spindle.

n5) Pause for the specified time P.

n6) Start the spindle rotation.

n7) Rise to the initial point using G00 (G98) or program R-point (G99).

- **Program Example**

M3 S500

G1 Z10.

G84 X10. Y10.Z-20.R0.P500 Q10000 F500

X10.

G80

M5 S0

M30

6. High-speed peck tapping

- **Action description**

Action 1: X/Y rapid positioning (M-code output possible).

Action 2: Z rapid positioning to the R-plane (skip spindle positioning if not required).

Action 3: Forward rotation tapping feed, Z descends by a cutting depth Q.

Action 4: Pause with delay specified by command P.

Action 5: Reverse the spindle rotation, G01 upward movement by a retraction amount d.

Action 6: Pause with delay specified by command P. Forward rotation tapping feed, Z descends to a depth Q below the current tapping depth.

Action 7: Pause with delay specified by command P.

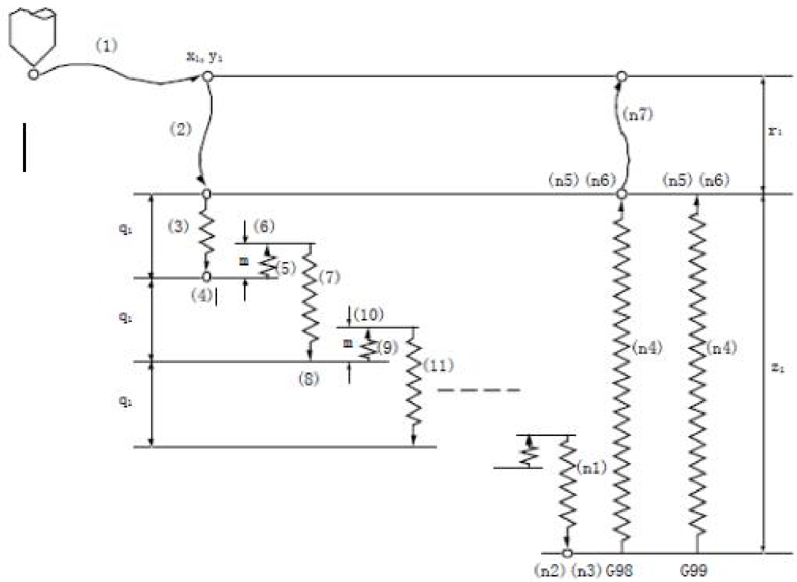
Action 8: Reverse the spindle rotation, G01 upward movement by a retraction amount d.

Action 9: Repeat the above tapping actions until reaching the bottom of the hole at Z point.

Action 10: Pause with delay specified by command P.

Action 11: Reverse the retraction to the R-plane.

Action 12: Rapid upward movement using G00 to the initial point (G98) or program R-point (G99).



(When Q value is specified (non-zero) and parameters are set for high-speed deep-hole rigid tapping)

- **As shown in the above diagram, the execution process is as follows:**

1) The machining begins by moving the tool to the designated (X, Y) position using G00.
2) Descend at the rapid speed of G00 to the R-plane.
3) Descend with G01 to the position Q1 for tapping, and stop the spindle.
4) Pause for the specified time P, and reverse the spindle rotation.
5) Ascend with G01 by the retract distance d.
6) Resume forward rotation of the spindle.
7) Descend with G01 to the position Q2.
8) Pause for the specified time P, and reverse the spindle rotation.
9) Ascend with G01 by the retract distance d.
10) Resume forward rotation of the spindle.
11) Descend with G01 to the position Q3.

...

n1) Descend with G01 to the bottom position of the hole for tapping, and stop the spindle.

n2) Pause for the specified time P.

n3) Reverse the spindle rotation.

n4) Ascend with G01 to the R-plane, and stop the spindle.

n5) Pause for the specified time P.

n6) Resume forward rotation of the spindle.

n7) Ascend using G00 to the initial point (G98) or program R-point (G99).

- **Program Example**

M3 S500

G1 Z10.

G84 X10. Y10.Z-20.R0.P500 Q10000 D2000 F500

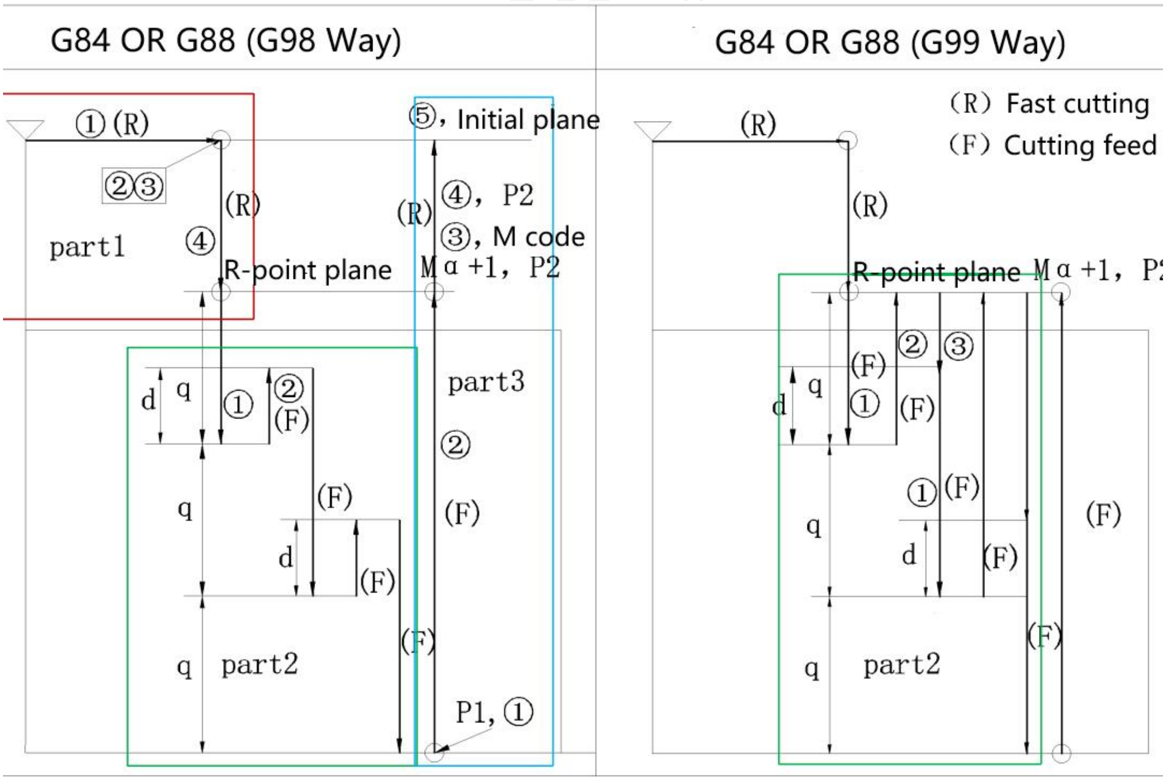
X10.

G80

M5 S0

M30

7. Explanation of Tapping Fixed Cycle Actions and Considerations



Part 1 (preprocessing): Tool approach preparation.

Step	Explanation
1	Position to the starting point of the hole (initial plane) using G00.
2	If specified, position the C-axis to the designated angle. This step is skipped if no C-value is specified. (Note: C-axis positioning should ideally be done during step 1.)
3	Execute Mα. This step is skipped if no Mα is specified.
4	Position to the safe plane R using G00.

Part 2 (Cycle): Threading preparation

	When the Q value is not 0
Step	Explanation
1	Feed to position [nq+R] for cutting. If [nq+R] exceeds the final Z position, feed to the Z position and end this section.
2	In high-speed deep hole drilling cycle mode, retract to a distance d from the current point using G01. In deep hole drilling cycle mode, retract to the R point.
3	Feed to position [n*q+R-d] using G01. This step is only present in deep hole drilling cycle mode.
	When the Q value is not 0
Step	Explanation
1	Feed to the Z position using G01 and end this section of processing

Part 3 (Retract): Retraction Preparation

Step	Explanation
1	Pause with G04 P1.
2	Rapid feed to the R point with G01.
3	If Mα is specified, execute M(α+1); otherwise, skip this step.
4	Pause with G04 P2.
5	Retract to the initial plane (in milling mode) with G98 mode.

2.43 G85 Boring Cycle

- **Instruction Explanation**

This cycle is used for boring operations.

- **Parameter Overview**

SYS 3019 BIT02

Definition: Handling of K values in drilling/boring/tapping cycle.

Range: Bit type.

Unit: null

Default Value: 0

Effective Mode: Enabled with C4.

Explanation:

BIT02=0: When specifying K0, the drilling/boring/tapping operation is performed only once.

BIT02=1: When specifying K0, the drilling/boring/tapping operation is not performed, and only the drilling/boring/tapping data is stored.

- **Instruction Format**

G85 X_ Y_ Z_ R_ F_ K_

X, Y: Specifies the initial point of the cycle (G90 for absolute, G91 for incremental).

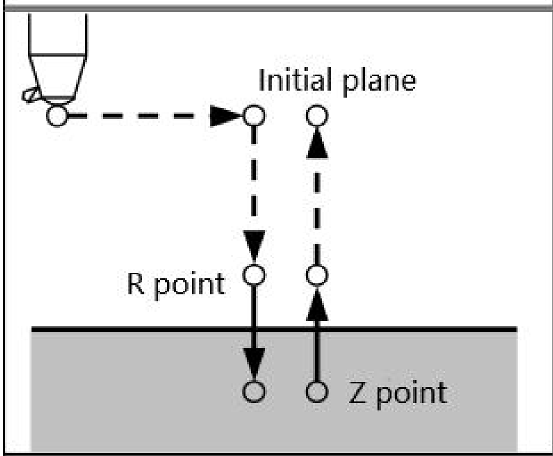
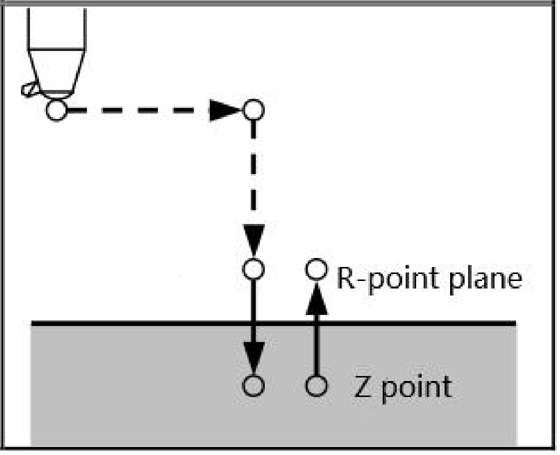
Z: Distance from R-point to the bottom of the hole.

R: Distance from the initial plane to the R-point (G90 for absolute, G91 for incremental), modal command.

F: Specifies the feed rate for boring, modal command.

K: Specifies the number of repetitions. If not specified, it defaults to 1. Not a modal command (system behavior can be defined when k=0).

- **Boring Cycle Illustration**

G85(G98)	G85(G99)
	

Action Explanation:

1. XY positioning.
2. Positioning to the R-point (Ensure the spindle is operating normally before boring).
3. Boring action from the R-point to the bottom of the hole.
4. Retract the tool to the R-point using G01.
5. Retract the tool to the R-point or the initial plane using G00.

- **Note**

1. When specifying M codes on the same line as G85 with a K value, the M code is executed only once.
2. In modal mode, if there are no X, Y, Z, R, or other axis movements, the boring operation is not performed.
3. Tool length compensation with G43/G44/G49 is effective during positioning to the R-point.
4. Tool radius compensation with G41/G42/G40 is not valid.
5. The R-point is not executed when K=0, only recorded.
6. Other motion G codes on the same line as G8x are canceled

- **Example Program**

M3 S30000

G90 G00 X300. Y400.

G90 G85 G99 X30. Y20. Z-150. R-50. F240 ; Positioning and perform boring operation 1, then return to the R-point

Y30. ; Positioning and perform boring operation 2, then return to the R-point

Y40. ; Positioning and perform boring operation 3, then return to the R-point

Y50. ; Positioning and perform boring operation 4, then return to the R-point

G998Y60. ; Positioning and perform boring operation 5, then return to the initial plane

G80 G28 G91 X0 Y0 Z0 ; Cancel G85 mode and return to the reference point

M5 ; Stop the spindle.

2.44 G86 Boring Fixed Cycle

- **Instruction Description**

This cycle is used for boring operations.

- **Parameter Overview**

SYS 3019 BIT02

Definition: Handling of K value in drilling/boring/tapping cycle.

Range: Bit type.

Unit: null

Default value: 0

Effective mode: C4 startup.

Explanation:

BIT02=0 When specifying K0, the drilling operation is executed only once.

BIT02=1 When specifying K0, the drilling/boring/tapping operation is not performed, only the data is stored.

- **Instruction Format**

G86 X_ Y_ Z_ R_ P_ F_ K_

X Y: Specifies the initial point of the cycle (G90 absolute/G91 incremental).

Z: Distance from R-point to the bottom of the hole.

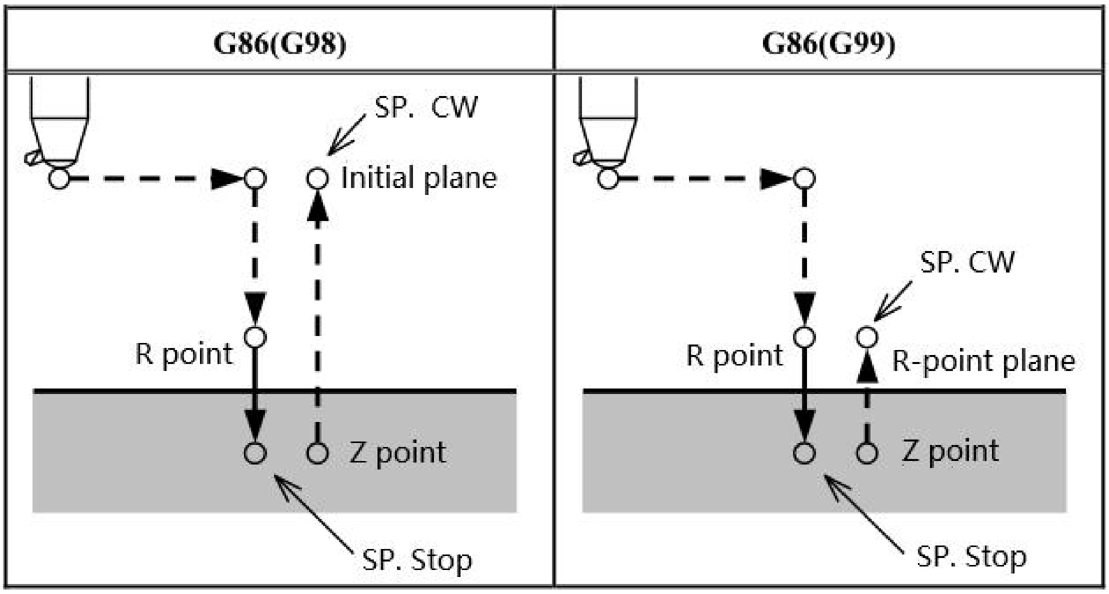
R: Distance from the initial plane to R-point, G90 absolute/G91 incremental, modal instruction.

P: Specifies the pause time at the bottom of the hole, equivalent to G04 P_ (modal), unit: ms (available in Mitsubishi, not described in Fanuc).

F: Specifies the feed rate for boring, modal instruction.

K: Specifies the repetition count, assumed to be 1 if not specified, not a modal instruction (system handling when k=0 can be set).

- **Boring Cycle Illustration**



- **Action Explanation**

1. XY positioning.
2. Positioning to the R-point (Ensure proper operation of the spindle before boring).
3. Boring action from the R-point to the bottom of the hole.
4. Stop the spindle at the bottom of the hole (Application engineer, please note that C32 should be clamped, and Mxx should be output according to the spindle stop M code #S3140).
5. Pause.
6. Rapid retraction.
7. Restore the spindle to forward rotation (Output Mxx according to the spindle forward

rotation M code #S3140).

- **Note**

1. When specifying M code in the same line as G86 and there is a K value, the M code is executed only once.
2. In modal mode, if X, Y, Z, R, or other axes are not included, boring operation is not performed (To be determined).
3. Tool length compensation G43/G44/G49 takes effect when positioning to the R-point.

- **Program Example**

M3 S30000

G90 G00 X0. Y0.

G90 G86 G99 X30. Y20. Z-150. R-50. F240 ; Positioning and perform boring 1, then return to R-point

Y20. ; Positioning and perform boring 2, then return to R-point

Y20. ; Positioning and perform boring 3, then return to R-point

Y20. ; Positioning and perform boring 4, then return to R-point

G98 Y20. ; Positioning and perform boring 5, then return to initial plane

G80 G28 G91 X0 Y0 Z0 ; Cancel G86 modal, return to reference point

M5 ; Stop the spindle

2.45 G87 Back Boring Fine Cycle

- **Instruction Explanation**

This cycle is used for high-precision back boring operations.

- **Parameter Overview**

SYS 3019 BIT02

Definition: Handling of K value in drilling/boring/tapping cycle.

Range: 0 to 2^31.

Unit: Bit type.

Default Value: 0.

Activation Method: Enabled by C4.

Explanation:

BIT02=0 When specifying K0, drilling/boring/tapping operations are not performed, only the data is stored.

BIT02=1 When specifying K0, drilling operation is performed only once.

SYS 3014

Definition: Specifying the direction of eccentricity (OSS).

Range: -3 to 3.

Unit: null.

Default Value: 0.

Activation Method: Enabled by C4

Explanation:

0/1	+X
-1	-X
2	+Y
-2	-Y
3	+Z
-3	-Z

SYS 3170 ~ SYS 3179

Definition: Specifies the M code for spindle stop during boring.

Range: 0 to 99999999.

Unit: null.

Default Value: 19.

Activation Method: Enabled by C4.

Explanation:

SYS 3170 = Specifies the M code for the first spindle stop.

SYS 3171 = Specifies the M code for the second spindle stop.

……

- **Instruction Format**

G87 X_ Y_ Z_ R_ Q_ P_ F_ K_

Explanation:

X, Y: Specifies the initial position of the cycle (G90 for absolute, G91 for incremental).

Z: Distance from R point to the bottom of the hole.

R: Distance from the initial plane to R point (G90 for absolute, G91 for incremental, modal command).

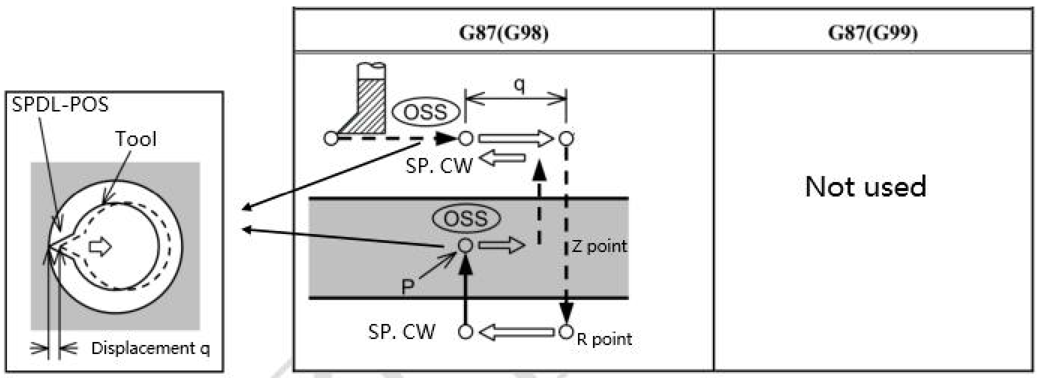
Q: Eccentricity at the bottom of the hole (axial direction controlled by parameter SYS 3014), modal command. Positive values, negative sign ignored.

P: Specifies the dwell time at the bottom of the hole, equivalent to G04 P_ (modal), unit: ms.

F: Specifies the boring feed rate, modal command.

K: Specifies the number of repetitions. If not specified, it defaults to 1. Not a modal command (system handles k=0 case).

- **Boring Cycle Illustration**



Action Explanation:

1. XY positioning (Ensure that the spindle is operating normally before boring).
2. Spindle preparatory stop (The system outputs the spindle stop M code first, then the spindle preparatory stop M code specified by parameter #S3170).
3. Tool offset by eccentricity distance Q.

4. G00 positioning to the R point.
5. Tool offset by reverse eccentricity distance -Q.
6. Spindle clockwise rotation.
7. G01 rise to the Z point.
8. Dwell for P milliseconds.
9. Spindle preparatory stop (The system outputs the spindle stop M code first, then the spindle preparatory stop M code specified by parameter #S3170).
10. Tool offset by eccentricity distance Q.
11. Positioning to the initial plane.
12. Tool offset by reverse eccentricity distance -Q.

- **Note**

1. When specifying M codes on the same line as G87 and there is a K value, the M code will only be executed once.
2. If the modal group does not contain X, Y, Z, R, or any other axis, the boring operation will not be performed.
3. Tool length compensation (G43/G44/G49) takes effect when positioning to the R point.
4. G99 is invalid.
5. G41/G42/G40 are ineffective during the boring operation.
6. If there are other motion G codes on the same line as G8x, the G8x code will be canceled.

- **Example program**

M3 S800

G90 G00 X300. Y400. Z0.

G90 G87 G98 X30. Y20. Z-10. R-50. Q1. F240 ; Positioning and bore hole 1, then return to the initial plane

Y30. ; Bore hole 2, return to the initial plane

Y40. ; Bore hole 3, return to the initial plane

Y50. ; Bore hole 4, return to the initial plane

Y60. ; Bore hole 5, return to the initial plane

G80 G28 G91 X0 Y0 Z0 ; Cancel G87 mode and return to the reference point

M5 ; Stop the spindle.

2.46 G88 Semi-Automatic Fine Boring Cycle

- **Instruction Description**

This cycle is used for high-precision fine boring operations.

- **Parameter List**

SYS 3019 BIT02

Definition: Handling of K value in drilling/boring/tapping cycle loop.

Range: 0 to 2^31

Unit: Bit type

Default: 0

Effective Mode: C4 Start

Explanation:

When BIT02=0 and K0 is specified, drilling/boring/tapping operations are not executed, only the data is stored.

When BIT02=1 and K0 is specified, the drilling operation is performed only once.

SYS 3014

Definition: Direction specification of OSS (eccentricity).

Range: -3 to 3

Unit: Null

Default: 0

Effective Mode: C4 Start

Explanation:

#S3014	OSS (Eccentricity) Direction
0/1	+X
-1	-X
2	+Y

-2	-Y
3	+Z
-3	-Z

SYS 3170~ SYS 3179

Definition: Specification of M code for spindle alignment stop in boring operation.

Range: 0 to 99999999

Unit: Null

Default: 19

Effective Mode: C4 Start

Explanation:

SYS3170 = Specifies the M code for the 1st spindle alignment stop.

SYS3171 = Specifies the M code for the 2nd spindle alignment stop.

……

- **Instruction format**

G88 X_ Y_ Z_ R_ P_ F_ K_

X, Y: Specifies the initial point of the cycle (G90 absolute/G91 incremental)

Z: Distance from R point to the bottom of the hole

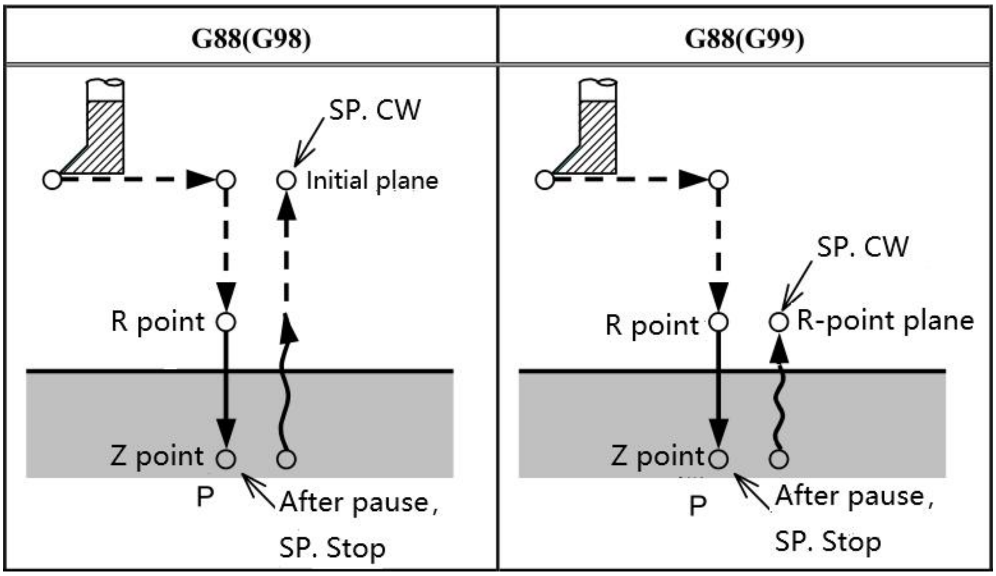
R: Distance from the initial plane to R, modal command with G90 absolute/G91 incremental

P: Specifies the pause time at the bottom of the hole, same effect as G04 P_ (modal), unit: ms

F: Specifies the boring feed rate, modal command

K: Specifies the number of repetitions, assumed to be 1 if not set, not a modal command (parameter can be set to handle k=0)

- **Boring Cycle Illustration**



Action explanation

1. XY positioning (Please ensure the spindle is operating normally before boring).
2. Positioning to the R point.
3. Performing the boring action from the R point to the bottom of the hole.
4. Pause for P milliseconds.
5. Stop the spindle at the bottom of the hole.
6. System outputs M0 for pause.
7. User switches to JOG mode - User should ensure safety, any JOG movement is allowed.
8. User restarts.
9. System returns to G98/G99 plane.
10. Start the spindle in the forward direction.

- **Note**

1. When specifying M codes in the same line with G88 and there is a K value, the M code is executed only once.
2. If there are no X, Y, Z, R, or other axis specifications in the modal group, the boring operation will not be performed.
3. Tool length compensation G43/G44/G49 takes effect when positioning to R.
4. G41/G42/G40 are ineffective during the boring operation.
5. If other motion G codes appear in the same line as G8x, the G8x is canceled.

- **Program Example**

M3 S800

G90 G00 X300. Y400. Z0.

G90 G88 G98 X30. Y20. Z-10. R-50. F240 ; Positioning, perform boring 1 and return to the initial plane

Y30. ; Boring 2, return to the initial plane

Y40. ; Boring 3, return to the initial plane

Y50. ; Boring 4, return to the initial plane

Y60. ; Boring 5, return to the initial plane

G80 G28 G91 X0 Y0 Z0 ; Cancel G88 modal group, return to the reference point

M5 ; Stop the spindle.

2.47 G89 Boring Cycle

- **Instruction Explanation**

This cycle is used for boring operations.

- **Parameter Overview**

SYS 3019 BIT02

Definition: K value handling in drilling/boring/tapping cycles.

Range: Bit type.

Unit: null

Default: 0

Effective: Activated by C4.

Explanation:

BIT02=0 When specifying K0, drilling is performed only once.

BIT02=1 When specifying K0, drilling/boring/tapping operations are not executed, only the drilling/boring/tapping data is stored.

- **Instruction format**

G89 X_ Y_ Z_ R_ P_ F_ K_

X, Y: Specifies the initial point of the cycle. It can be specified in absolute coordinates (G90) or incremental coordinates (G91).

Z: Specifies the distance from R to the bottom of the hole.

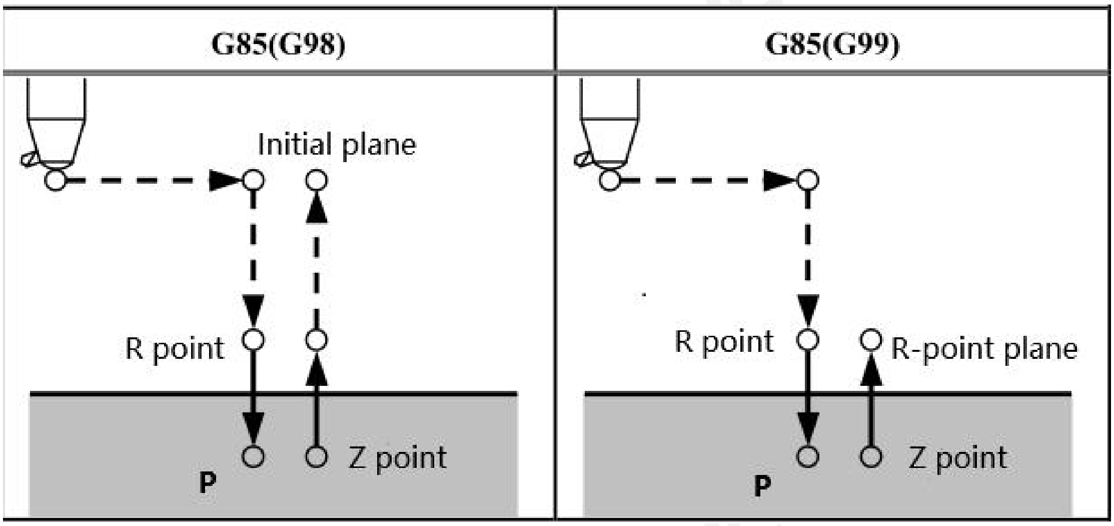
R: Specifies the distance from the initial plane to R. It can be specified in absolute coordinates (G90) or incremental coordinates (G91) as a modal command.

P: Specifies the dwell time at the bottom of the hole, similar to the effect of G04 P_. The unit is milliseconds and it is a modal command.

F: Specifies the feed rate for the boring operation as a modal command.

K: Specifies the number of repetitions. If not specified, it is assumed to be 1. It is not a modal command (the system can handle k=0).

- **Boring Cycle Illustration**



Action explanation:

1. XY positioning.
2. Positioning to the R point (Please ensure that the spindle is operating correctly before boring).
3. Boring operation from the R point to the bottom of the hole.
4. Pause for P milliseconds.
5. Retract the tool to the R point using G01 motion.
6. Retract the tool to the R point or the initial plane using G00 motion.

- **Note**

1. When specifying M code in the same line as G89 and there is a K value, the M code is executed only once.
2. If the modal command does not include X, Y, Z, R, or other axes, the boring operation is not performed (to be confirmed).
3. Tool length compensation with G43/G44/G49 takes effect when positioning to the R point.
4. Tool radius compensation with G41/G42/G40 is not valid.
5. The operation specified by R is not executed when K value is 0, it is only recorded.
6. If there are other motion G codes on the same line as G8x, the G8x command is cancelled.

- **Program Example**

M3 S30000

G90 G00 X300. Y400.

G90 G89 G99 X30. Y20. Z-150. R-50. F240 ; Positioning, bore hole 1, then return to R point

Y30. ; Positioning, bore hole 2, then return to R point

Y40. ; Positioning, bore hole 3, then return to R point

Y50. ; Positioning, bore hole 4, then return to R point

G998 Y60. ; Positioning, bore hole 5, then return to the initial plane

G80 G28 G91 X0 Y0 Z0 ; Cancel G89 mode, return to the reference point

M5 ; Stop the spindle.

2.48 G90, G91 Absolute or Incremental Coordinate Setting

- **There are two ways to specify absolute or incremental coordinate values**

1. Mode specification - using the G90 and G91 commands (G90 for absolute coordinate value setting, G91 for incremental coordinate value setting).

2. Incremental specification - using the U, V, W commands.

By default, when a milling machine from the Finger series is powered on, it is automatically set to absolute coordinates. In the program, the G90 and G91 commands can be used to set the coordinate system as either absolute or incremental. The U, V, W address codes are only effective in the G90 mode (absolute coordinate mode) and are invalid in the G91 mode (incremental coordinate mode).

In the G91 mode, the X, Y, and Z values represent incremental values.

- **Program Example 1:**

G90 Absolute Coordinate Value Setting

N1 G90

N2 G1 X20.000 Y15.000 P0 to P1

N3 X35.000 Y25.000 P1 to P2

N4 X60.000 Y30.000 P2 to P3

- **Program Example 2:**

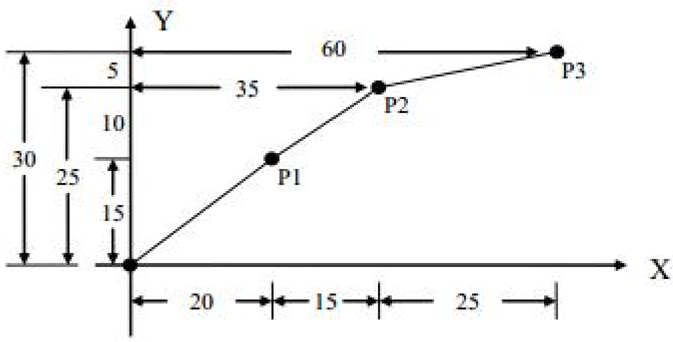
G91 Incremental Coordinate Value Setting

N1 G91

N2 G1 X20.000 Y15.000 P0 to P1

N3 X15.000 Y10.000 P1 to P2

N4 X25.000 Y5.000 P2 to P3



Mixing Absolute and Incremental Coordinate Usage

Here are three G01 program examples, each using a different coordinate mode but following the same path for linear cutting:

Program Example 1: Absolute Coordinate Mode (Figure 3-2)

N1 G90

N2 G01 X25.000 Y20.000 Z10.000 F100.00 ... P1

N3 X60.000 Y50.000 Z40.000 ... P2

Program Example 2: Incremental Coordinate Mode (Figure 3-2)

N1 G91

N2 G01 X25.000 Y20.000 Z10.000 F100.00 ... P1

N3 X35.000 Y30.000 Z30.000 ... P2

Program Example 3: Absolute Coordinate Mode with Incremental Values (Figure 3-2)

N1 G90

N2 G01 U25.000 Y20.000 Z10.000 F100.00 ... P1

N3 X60.000 V30.000 W30.000 ... P2

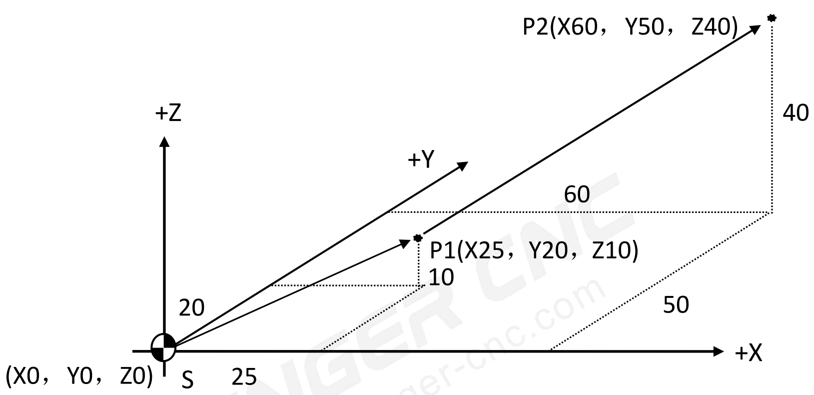


Figure 3-2: G01 Program Examples

2.49 G94, G95 Feed Rate Setting

- **Instruction Explanation**

G94: Feed Per Minute (mm/min)

G95: Feed Per Revolution (mm/rev)

In the finger milling machine series, the feed rate (F) is determined by the values set with G94 and G95. The conversion formula between the two is as follows:

Fm = Fr * S

Fm: Feed rate per minute, mm/min.

Fr: Feed rate per revolution, um/rev.

S: Spindle speed, rev/min.

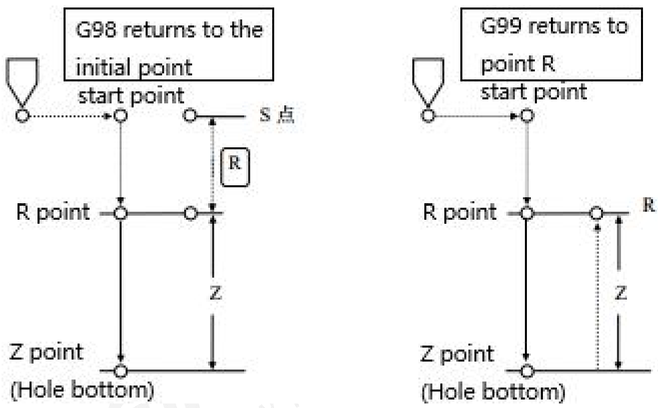
......

2.50 G98, G99 Return to Initial Point in Drilling Cycle

- **Instruction Explanation**

G98: Set Return to Initial Point

G99: Set Return to R-plane



- **Program Example**

M3 S1000 ; Start spindle clockwise

G00 X10.Y10.Z10. ; Move to initial point

```
G99                              ; Set return to R-plane
G81 X10. Y10.Z-30.R0.Q10.F500    ; G81 drilling cycle, first point (return to R-plane)
X20.Y20.                         ; Second point (return to R-plane)
X30.                             ; Third point (return to R-plane)
G98 X40.                         ; Fourth point (return to initial point)
G80                              ; End of drilling cycle
M5                               ; Stop spindle
M30                              ; End of program
```

2.51 C, R, A Instructions Explanation

- **Function and Purpose**

The C, R, A instructions are used to achieve chamfering, filleting, and angle cutting functionalities in machining operations.

1. ,C Chamfering

- **Function and Purpose**

With two consecutive segments, the first segment can be used to perform chamfering using the ",C_" instruction. ",C_" indicates the length from the imaginary chamfer starting point to the chamfer ending point.

- **Instruction Format**

N1 G0x X_Z_,C_

N2 G0x X_Z_

G0x: Can be either G00, G01, G02, or G03.

,C_: Indicates the length from the imaginary chamfer starting point to the chamfer ending point.

- **Program Example**

1) Straight Line to Arc

Absolute Coordinate Instructions

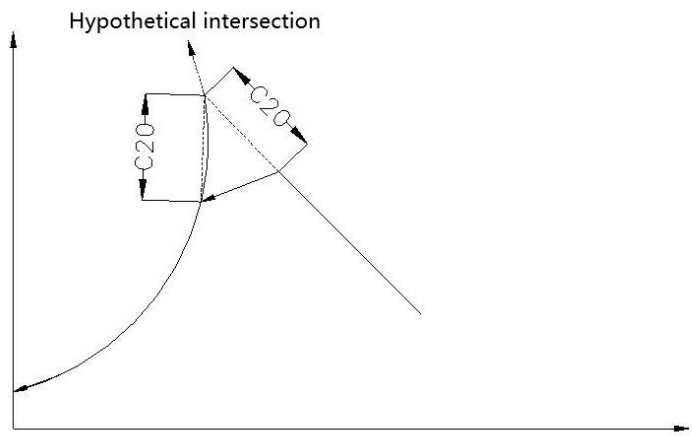
N1 G28 X0 Z0

N2 G00 X50. Z100.

N3 G01 X150. Z50. ,C20. F100.

N4 G02 X50. Z0. I0. K-50.

N5 M30



2) Arc to Arc

Incremental Value Instruction

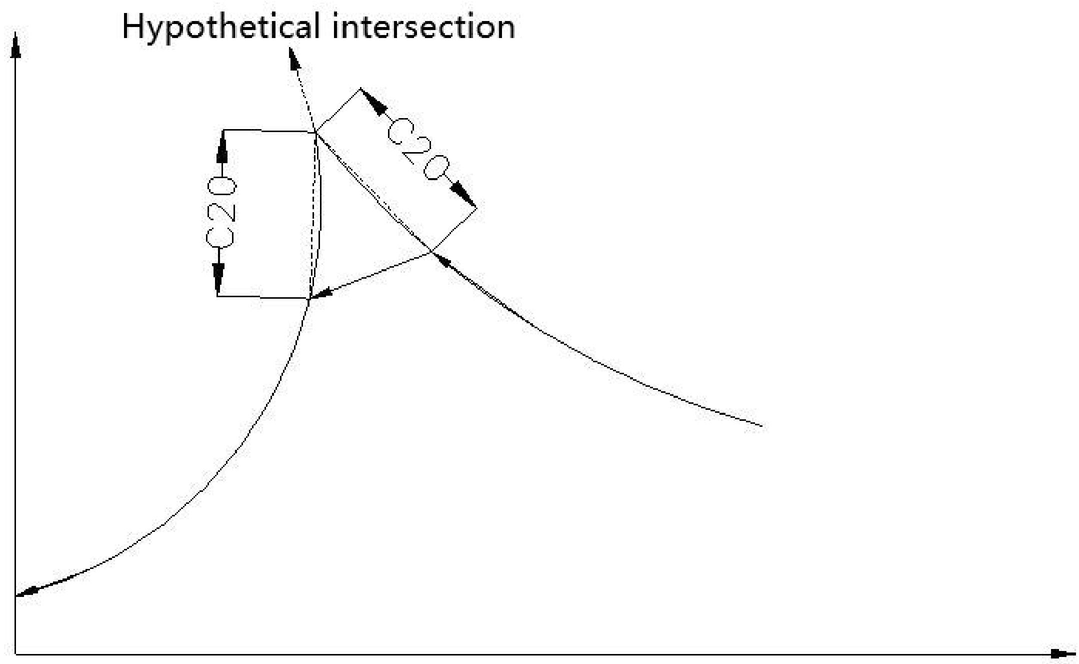
N1 G28 X0 Z0

N2 G00 U10. W140.

N3 G02 U40. W-80. I100. K0. ,C20. F100.

N4 G02 U-20. W-60. I180. K-60.

N5 M30



2. ,R Instruction Explanation:

- **Function and Purpose**

In a consecutive sequence of two blocks, the ",R_" instruction is used in the first block to execute fillet rounding. ",R_" indicates the radius of the fillet rounding.

- **Instruction Format**

N1 G0x X_Z_,R_

N2 G0x X_Z_

G0x: Can be G00, G01, G02, or G03

,R_: Indicates the radius of the fillet rounding.

- **Program Example**

1) Line to Arc

Absolute Value Instruction

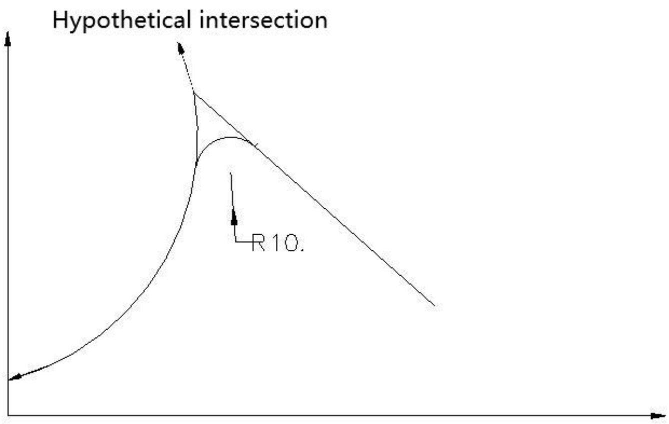
N1 G28 X0 Z0

N2 G00 X60. Z100.

N3 G01 X160. Z50. ,R10. F100.

N4 G02 X60. Z0. I0. K-50.

N5 M330



2) Arc to Arc
Incremental Value Instruction

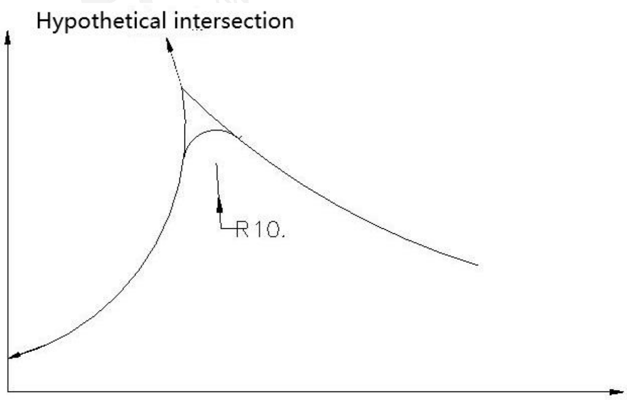
N1 G28 X0 Z0

N2 G00 U30. W100.

N3 G01 U50. W-50. I50. K0. ,R10. F100.

N4 G02 U-50. W-50. I0. K-50.

N5 M30



- **Automatic calculation of fillet intersection points.**

Condition：

1) Connection between a line and an arc, where the line is shorter and the arc is larger.
2) The set corner rounding radius R is relatively large, and the system automatically calculates the tangent point.

- **Program Example**

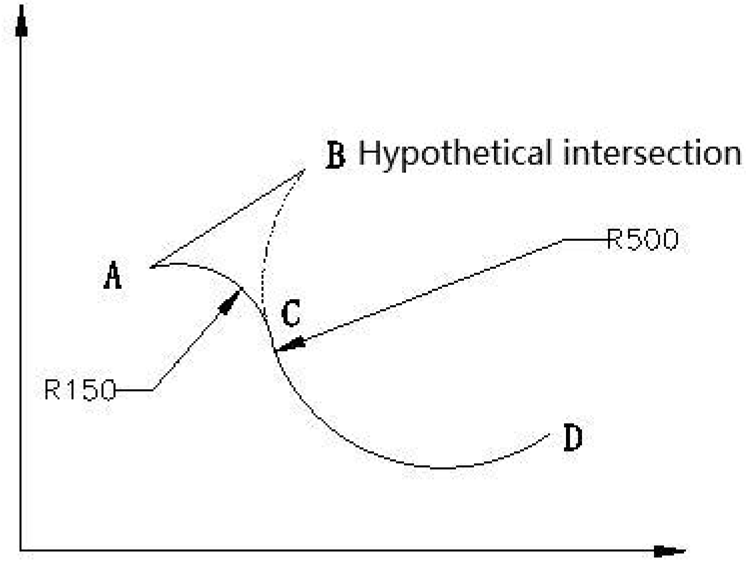
N1 G28 X0 Z0

N2 G00 X50. Z20.

N3 G01 X80. Z80. ,R300. F100.

N4 G03 X25. Z200.R500.

N5 M30



3. A Command for Angular Movement

- **Function and Purpose**

By providing the angle of a straight line and the endpoint coordinates of one axis, the endpoint coordinates of the other axis are automatically calculated.

- **Instruction Format**

N1 G01 X_(Z_),A_

- **Angle Representation**

The angle is measured starting from the positive direction of the first axis in the plane, in the counterclockwise direction (CCW) as positive, and in the clockwise direction (CW) as negative.

- **Angle Range**

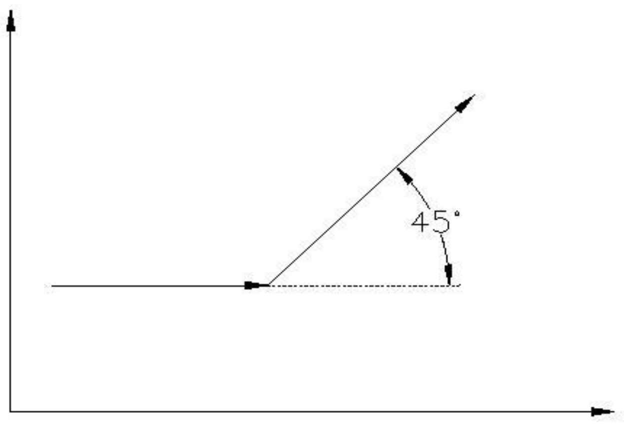
The angle range is -360.000° to 360.000°. If the angle exceeds 360.000°, the remainder after dividing by 360.000° is taken.

- **Program Example**

N1 G00 X50. Z0.

N2 G01 X50. Z50. F1000.

N3 G01 Z100. ,A45.0



4. Combining ,C, ,R, and ,A Instructions

- **Function and Purpose**

When used in combination, the ,C, ,R, and ,A instructions enable the execution of chamfers and fillets, simplifying programming and automatically calculating intersection points.

5. ,A →A ,A Combination

- **Function and Purpose:**

If it is difficult to determine the intersection point of two straight lines, the endpoint coordinates and inclination of the second straight line can be specified along with the inclination of the first straight line. The system will automatically determine the endpoint of the first straight line and control the movement path accordingly.

- **Instruction format**

G01 ,A_

G01 X_ Z_ ,A_

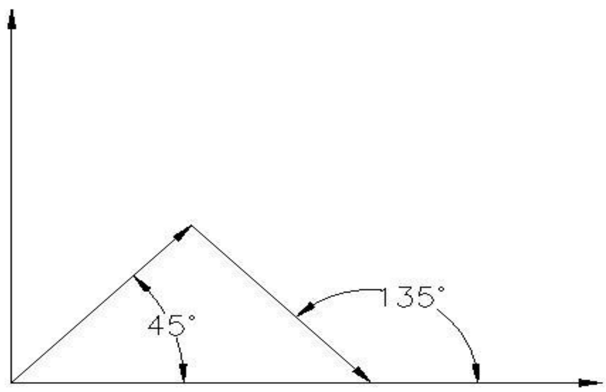
- **Program Example**

N01 G00 X0. Z0.

N02 G01 ,A45. F1000.

N03 G01 Z90. X0. ,A135.0

N04 M30



Note:

1) When using relative coordinates for the endpoint of the second segment, an error occurs.
2) If the two straight lines do not intersect or the intersection angle is less than 1 degree, an error occurs.

6. ,A →C ,C Combination

- **Instruction format**

G01 X_ Z_ ,A_ ,C_

G0x X_ Z _I_ J_

Or

G01 X_ Z_ ,A_ ,C_

G01 X_ Z _,A

- **Program Example**

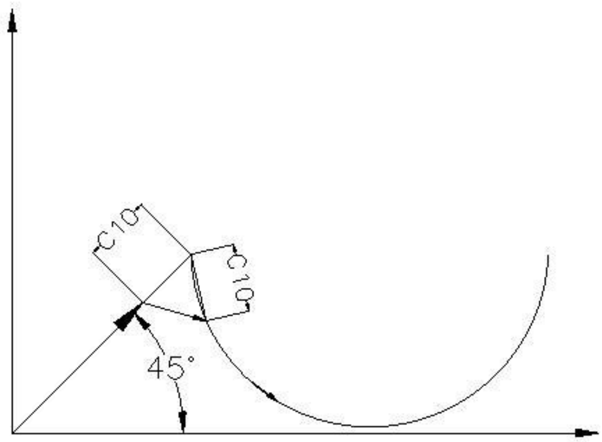
1)

N1 G00 X0. Z0.

N2 G01 X50.,A45.0 F1000. ,C10.

N3 G03 X50. Z1 00. R50.

N4 M30



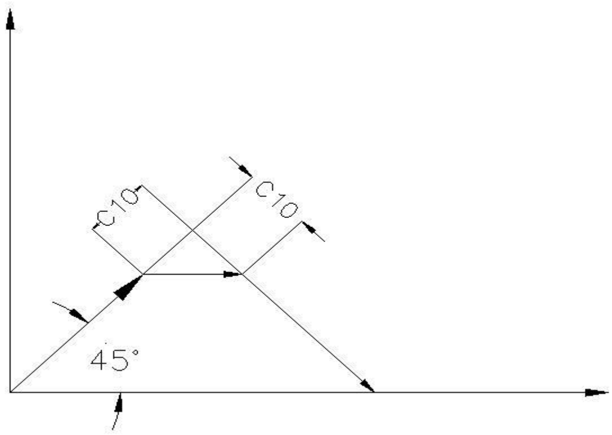
2)

N1 G00 X0. Z0.

N2 G01 ,A45. ,C10.

N3 G01 X0. Z90. ,A135.

N4 M30



7. ,A →,R Combination

- **Instruction format**

G01 X_ Z_ ,A_ ,R_

G0x X_ Z _I_ J_

Or

G01 X_ Z_ ,A_ ,R_

G01 X_ Z _,A

- **Program Example**

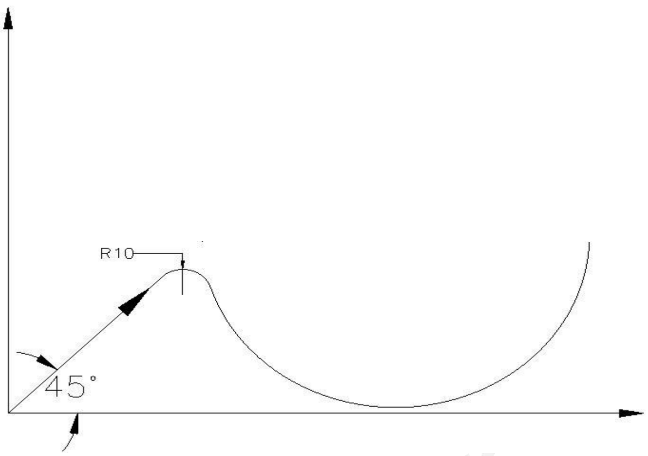
1)

N1 G00 X0. Z0.

N2 G01 X50.,A45.0 F1000. ,R10.

N3 G03 X50. Z100. R50.

N4 M30



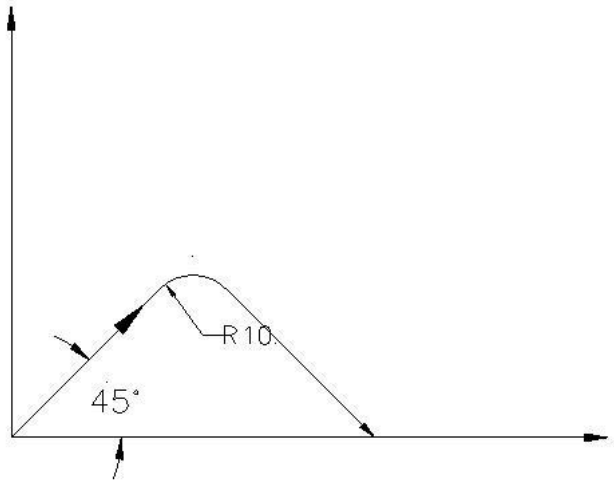
2)

N1 G00 X0. Z0.

N2 G01 ,A45. ,R10.

N3 G01 X0. Z90. ,A135.

N4 M30



8. Automatic Calculation of Intersection between a Line and an Arc

- **Function and Purpose**

In situations where the intersection point between a line and an arc is not specified, the system automatically calculates the intersection point and controls the movement path accordingly.

- **Instruction format**

G01 ,A_,R(,C)

G02(G03) X_ Z_ P_ Q_ H_

- **P, Q, X, Z Axis Absolute Coordinate Values for Arc Centers**

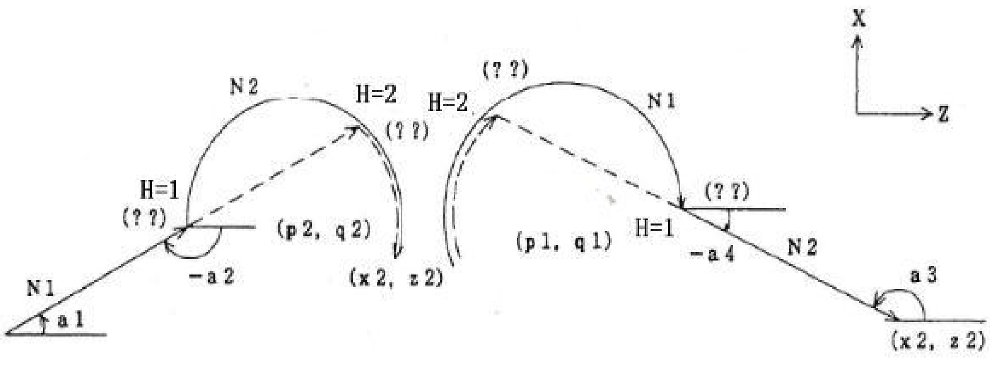
H: Intersection Point Selection between a Line and an Arc

=1: Select the intersection point based on the shorter line segment.

=2: Select the intersection point based on the longer line segment.

Note:

1) If the second segment is not an absolute coordinate value, an error will be reported.
2) If the second segment is an arc and P, Q are not specified, an error will be reported.
3) If there is no intersection point between the line and the arc, an error will be reported.



- **Motion Planning Alerts**

1) Segment C or R is not a line or an arc (1-1-1).
2) Next segment after C or R is not a line or an arc (1-1-2).
3) Incorrect value set for C or R (1-1-3).
4) The preceding and following lines of C or R are parallel (1-1-4).
5) Segment A is not a line (1-1-7).
6) Next segment after A is incorrect (1-1-8).
7) Incorrect value set for A (1-1-9).

Part 3. M Code Instruction Explanation

Code value	Function definition	Code value	Function definition
00	Program pause	40	
01	Select stop	41	
02	Program end	42	
03	First spindle clockwise rotation	43	
04	First spindle counterclockwise rotation	45	
05	First spindle stop	46	
08	Cutting fluid on	50	Convert the first spindle to servo axis
09	Cutting fluid off	51	Restore the first spindle as main spindle
10		60	Convert the second spindle to standard axis
11		61	Restore the second spindle as main spindle
12		63	Second spindle clockwise rotation
13		64	Second spindle counterclockwise rotation
15	Increment machining count	65	Second spindle stop
16	Reset machining count	70	Convert the third spindle to standard axis
17		71	Restore the third spindle as main spindle
18		73	Third spindle clockwise rotation
19		74	Third spindle counterclockwise rotation
20		75	Third spindle stop
27		85	
30	Program end and return to program start	98	Call subroutine

		99	Subroutine end and return

M-codes are composed of the letter "M" followed by a two-digit number. They represent different actions and are commonly used in CNC machining. However, the following M-codes specifically pertain to the Finger milling machine and are not intended for external use by customers:

M00 Program pause

At this point in the program, all machining instructions will stop (spindle rotation will stop, cutting fluid will be turned off). Pressing the CYCST key will allow the program to continue.

M01 Optional stop

In automatic mode, when the optional stop is enabled, the program will pause when encountering the M01 code. The program will remain halted until the start button is pressed, and then it will continue to the next block of the program.

M02 Program end

When the CNC reaches the M02 code at the end of the main program, the machine will stop all actions.

M03 Spindle clockwise rotation

The M03 code is used to rotate the spindle in a clockwise direction. It is used in conjunction with the S code to set the spindle speed for clockwise rotation.

M04 Spindle counterclockwise rotation

The M04 code is used to rotate the spindle in a counterclockwise direction. It is used in conjunction with the S code to set the spindle speed for counterclockwise rotation.

M05 Spindle stop

M06 Tool change

The M06 code is used to perform a tool change. This code does not specify the tool to be used and must be used in conjunction with the T code.

M08/M09 Coolant on/off

M08 code is used to turn on the coolant.

M09 code is used to turn off the coolant.

M30 Program end and return to program start

When the CNC reaches the M30 code, the program ends and returns to the starting position.

M98 Call subroutine.

3.1 Format of usage

1. M98 Pxxxxx Hxxxx Lxxxx Dxxxx
2. M98 <filename> Hxxxx Lxxxx Dxxxx

3.2 Meaning of letters

M98: Subprogram call command

P: File name of the subprogram, with the file extension ".CNC"

It can only be a numeric value, up to 8 digits.

H: Sequential number (Nxxxx) within the subprogram to be called. Up to 5 digits.

If omitted, it starts from the beginning of the subprogram.

Example: M98 P123 H10 will start from sequence N10 in the file 123.CNC.

L: Number of iterations for the subprogram. If omitted, it defaults to L1 (one iteration).

Example:

M98 P123 L3

M98 P123 L1

M98 P123 L1

M98 P123 L1

<filename>:

1. The file name of the subprogram to be called. When the file extension is omitted, it defaults to ".CNC".
2. The file name can contain alphanumeric characters and can be up to 32 characters long.

Example:

M98 <AA123> will call the subprogram AA123.CNC.

M98 <TEST.CN> will call the subprogram TEST.CN.

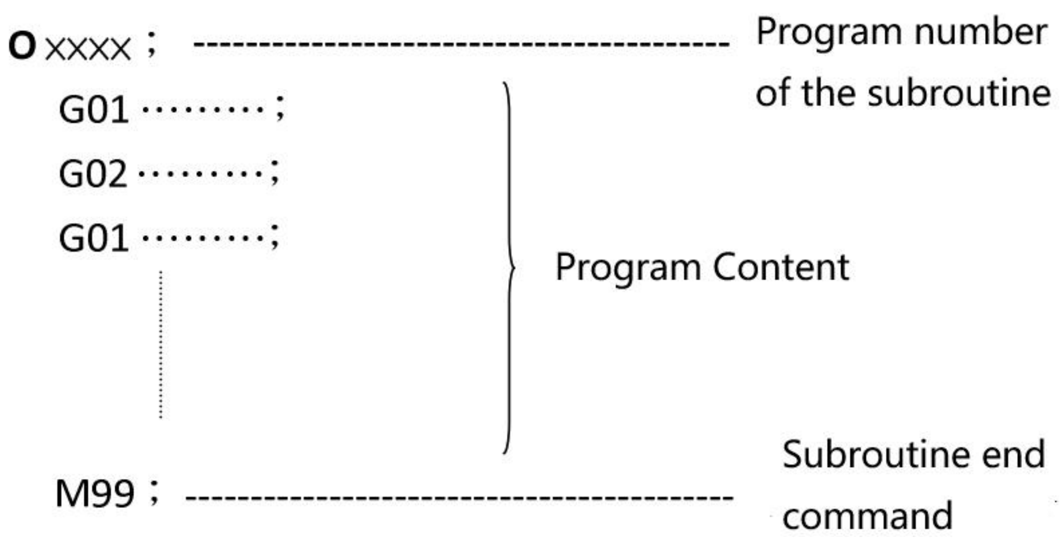
Note:

1. Examples:
   M98 P123 will call the subprogram 123.CNC.
   M98 P00123 will call the subprogram 123.CNC (not 00123.CNC).
   M98 P0 will call the subprogram 0.CNC.
2. Examples:
   Calling program 00123.CNC: M98 <00123>
3. Examples:
   When both P and <filename> are omitted, it calls the current program.

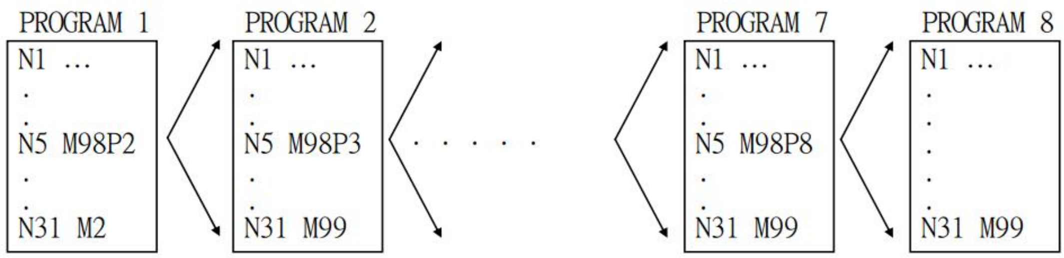
D: Subprogram call folder, range from 0 to 10

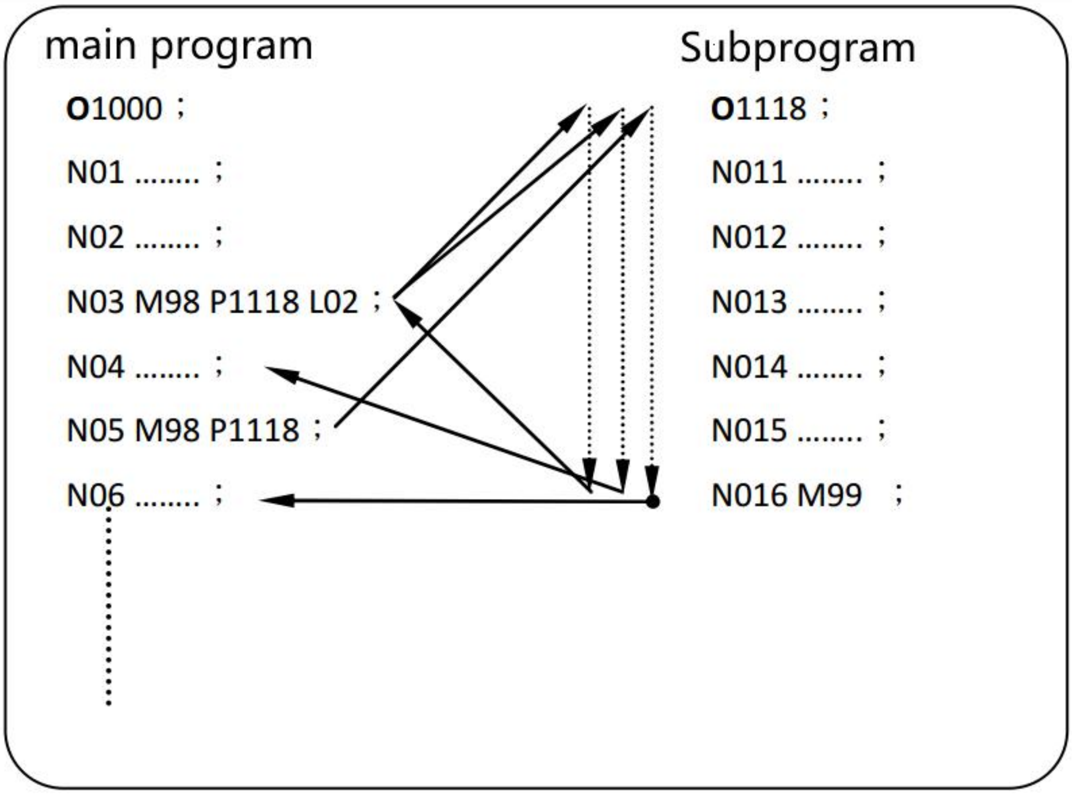
1. When D is not specified, it calls the subprogram from the current folder. Example: M98 P123
2. When D=1: M98 P123 D1 calls the subprogram from the /home/root/finger/usr/sys0001/program/M98 folder. If 123.CNC is not found in the M98 folder, an alarm will be raised.

- **The format of the subprogram is as follows**



Execution sequence of the main program in conjunction with subprogram call instructions





3.3 M99

1. In the main program, writing M99 denotes the end of the main program. If there is no P specified after M99, it automatically loops back to the beginning of the main program and starts execution again. If there is a P value specified after M99, it jumps to the line number specified by P and continues execution from there.
2. If it's a subprogram and you write M99, it signifies the end of the subprogram, and it will automatically return to the main program to continue execution.

M99: Subprogram return command.

P: Sequential number of the program to return to.

Note:

1. In the main program, it returns to the single block with the corresponding number in

the main program.

2. In a subprogram, it returns to the single block with the corresponding number in the calling program.

For example: M99 P200

This would return to line N200 in the program.

Example:

O000.CNC program (main program):

N1 G98

N2 G0X10.Z10

N3 G01U10.W10.F1000.

N4 M98P<O001>L3H3

N5G0X-100.Z-100.

M30

O001.CNC program (subprogram):

N1 G0Y0.

N2 G0Y100.

N3 G0Y-100.

N4 G0V10.

M99P5

When executing the O000.CNC program, at line N4, it will not immediately execute the O001.CNC program three times. Instead, it will directly jump to line N5 in the main program and continue execution.

3.4 T-code: Tool Calling

- **Function description**

The T-code functionality is primarily used for tool selection. It is commonly used in conjunction with the tool change command (M06) to select a specific tool for tool changing or to identify the current tool number for the main spindle. This allows for automatic tool changes based on tool variations.

- **Instruction format**

M06 Txx

- **Example program**

M06 T1	... Call tool number 1 as the current tool for the main spindle and perform machining with this tool
M03 S10000	... Start the main spindle with a speed of 10000
G01 X100. Y100.	... Position to X100. Y100.
...	... Perform cutting operations
M05	... Stop the main spindle
M06 T5	... Call tool number 5 as the current tool for the main spindle and perform machining with this tool
M03 S8000	... Start the main spindle with a speed of 8000
G1 X-100. Y50.	... Position to X-100. Y50.
...	... Perform cutting operations
M05	... Stop the main spindle
M30	... Program completion

3.5 F-code: Feedrate

- **Instruction format**

F

- **Description**

When cutting a workpiece, the speed at which the specified tool moves in the machining program is called the feedrate. There are two methods to set the feedrate: feedrate per minute (G94) and feedrate per revolution (G95). In the G94 mode, a feedrate of 1000mm/min can be directly specified as F300. In the G95 mode, F0.5 represents a feedrate of 0.5mm/rev.

- **Program examples**

G94 G01 X100. Y100. F1000;

// Tool moves in a straight line cutting at a feedrate of 1000mm/min

G95 G01 X100. Y100. F0.5;

// Tool moves in a straight line cutting at a feedrate of 0.5mm/rev

3.6 S-code: Spindle Speed

- **Instruction format**

S_

- **Description**

The S-code is used to command the spindle speed, specifying the rotational speed of the spindle per minute.

- **Program examples**

M03 S500; // Sets the spindle speed to 500 revolutions per minute

3.7 Automatic Plane Switching

- **Function Description**

1. Automatically switches the machining plane when programming without explicitly specifying the plane (assuming there is no 3-axis interpolation or IJ specification).
2. Related Parameter: MCM1879

3. BIT00=1: Enables automatic arc plane detection functionality.

- **Program Example**

G0 X200. Y40. Z0.

G03 X140. Y100. R60. F3000; ;;;The system automatically recognizes the arc command even if G17 is not used in the current or previous line.

G02 X120. Y60. R50; ;;;The system automatically recognizes the arc command even if G17 is not used in the current or previous line.

G0 X200. Z40.

G03 X140. Z100. R60. F3000; ;;;The system automatically recognizes the arc command even if G18 is not used in the current or previous line.

G02 X120. Z60. R50; ;;;The system automatically recognizes the arc command even if G18 is not used in the current or previous line.

M30

FINGER CNC

Guangzhou Finger Technology Co.,Ltd

Hotline：020-39389901
Repair Helpline：18127931302
Fax：020-39389903
Postal Code：511495
E-mail：finger@fingercnc.com
Website：www.finger-cnc.com
Address：201,No. 8, Chengding Street, Zhongcun Street, Panyu District, Guangzhou City, Guangdong Province

Official Website

Official Wechat