# Bergerda
# I5M5 I5M5E
# CNC Controller for Milling Machine
# User Manual V1.23


## Hangzhou Bergerda Automation Technology Co., LTD

**Read this manual carefully before installing / debugging/ using the product**

In this user manual we have tried to describe the matters concerning the operation of CNC system to the greatest extent. However, it is impossible to give particular descriptions for all unnecessary or unallowable operations due to length limitation and products application conditions; Therefore, the items not presented herein should be regarded as "impossible" or "unallowable".

# PREFACE

Your Excellency:

We are deeply honored that you have chosen the CNC system for your milling!

This manual is the "Programming and Operation Manual" section. Please read this manual carefully to provide a detailed introduction to the programming and operation methods of the CNC system of the milling.

Improper operation will cause accidents, and qualified personnel are required to operate this system.

Please read this user manual carefully before operation!

Special reminder: The system power supply installed on the chassis (inside) is a dedicated power supply only provided for CNC systems manufactured by our company. Users are prohibited from using this power supply for other purposes. Otherwise, there will be great danger!

This manual is collected by the end user.

All specifications and designs are subject to change without prior notice from the company.

Sincere thanks - for your friendly support in using our company's products.

# Safety warnings and precautions

## Instructions for warnings, precautions, and comments

This manual contains safety precautions to protect users and prevent machine tool damage. These precautions are divided into warnings and precautions based on their safety nature. Supplementary information is provided as a commentary. Please carefully read the warnings, precautions, and notes before operating the machine tool.

**warn**

If the specified operating methods or steps are not followed, it is possible to harm the user or damage the equipment.

**Caution**

If the specified operating methods or steps are not followed, it is possible to damage the equipment.

**annotation**

Annotations are used to indicate supplementary information in addition to warnings and precautions.

# Safety responsibility

## Manufacturer's safety responsibility

--The manufacturer shall be responsible for any hazards that have been eliminated and/or controlled in the design and structure of the provided CNC system and accompanying accessories.

--The manufacturer shall be responsible for the safety of the provided CNC system and accompanying accessories.

--The manufacturer shall be responsible for the usage information and suggestions provided to the user.

## User's safety responsibility

--Users should learn and train in the safe operation of CNC systems, and be familiar with and master the content of safe operation.

--Users shall be responsible for the safety and hazards caused by adding, changing, or modifying the original CNC system and accessories themselves.

--Users shall be responsible for the hazards caused by failure to operate, adjust, maintain, install, and store products in accordance with the instructions in the user manual.

# CONTENTS


Chapter I Overview....1
1.1 Product introduction....1
1.2 technical specifications....1
Chapter II Composition of Parts Program....
2.1 Program composition.... 4
2.1.1 Program Name.... 4
2.1.2 code word.... 5
2.2 General Structure of Program.... 6
2.2.1 Subroutine writing....7
2.2.2 Subprogram Call....7
2.2.3 End of program.... 8
Chapter III Programming Basics....
3.1 Control axis....9
3.2 Axis name....9
3.3 Coordinate System....9
3.3.1 machine coordinates.... 9
3.3.2 Reference point.... 9
3.3.3 Workpiece Coordinate System....10
3.3.4 Absolute coordinate programming and relative coordinate programming....11
3.4 Modal and Non Modal.... 12
3.5 Decimal point programming.... 13
Chapter IV Preparation Function G Code....14
4.1 Type of Preparation Function G Code.... 14
4.2 Simple G Code.... 17
4.2.1 Fast Positioning G00.... 18
4.2.2 Linear Interpolation G01....18
4.2.3 Circular (Helical) Interpolation G02/G03.... 18
4.2.4 Absolute Value/Incremental Value Programming G90/G91.... 23
4.2.5 Pause (G04).... 23
4.2.6 Single Direction Positioning (G60).... 24
4.2.7 Online Change of System Parameters (G10)....25
4.2.8 Workpiece Coordinate Systems G54~G59....26
4.2.9 Additional Workpiece Coordinate Systems....27
4.2.10 Selection of Machine Coordinate System G53.... 28
4.2.11 Floating Coordinate System G92....28
4.2.12 Plane Selection G17/G18/G19....30
4.2.13 Polar Coordinate Beginning/Cancellation G16/G15.... 31
4.2.14 Scaling In The Plane G51/G50.... 32
4.2.15 Coordinate System Rotation G68/G69.... 35
4.2.16 Skip Function G31....39
4.2.17 Imperial/Metric Conversion of G20/G21....40
4.2.18 Any Angle Chamfer/Corner Arc....40
4.3 Reference Point G Code.... 41
4.3.1 Return To Reference Point G28....42
4.3.2 Return To Reference Points 2, 3 and 4 (G30).... 43
4.3.3 Automatic Return From A Reference Point (G29).... 43
4.3.4 Return To A Reference Point For Testing (G27).... 44
4.4 Fixed Cycle (G Code)....44
4.4.1 Groove Rough Milling Inside Circle (G22/G23).... 50
4.4.2 Finish Milling Cycle Inside Full Circle (G24/G25).... 52
4.4.3 Finish Milling Cycle Outside Circle (G26/G32).... 54
4.4.4 Rectangular Groove Rough Milling (G33/G34)....55
4.4.5 Finish Milling Cycle Inside Rectangular Groove (G35/G36)....57
4.4.6 Finish Milling Cycle Outside Rectangle (G37/G38).... 58
4.4.7 High-Speed Deep Hole Machining Cycle (G73).... 59
4.4.8 Drilling Cycle and Point Drilling Cycle (G81)....61
4.4.9 Drilling Cycle and Boring Cycle (G82).... 62

4.4.10 Chip removal drilling cycle (G83) ........ 64
4.4.11 Tapping Cycle (G74 Or G84) ........ 65
4.4.12 Precision Boring Cycle (G76) ........ 68
4.4.13 Boring Cycle (G85) ........ 69
4.4.14 Boring Cycle (G86) ........ 71
4.4.15 Boring Cycle and Back Boring Cycle (G87) ........ 72
4.4.16 Boring Cycle (G88) ........ 73
4.4.17 Boring Cycle (G89) ........ 75
4.4.18 Left Rigid Tapping (G74) ........ 76
4.4.19 Right Rigid Tapping (G84) ........ 78
4.4.20 Deep-Hole Tapping (Chip Removal) Cycle ........ 79
4.4.22 Fixed cycle cancellation G80 ........ 81
4.5 Tool compensation (G code) ........ 85
4.5.1 Tool Length Compensation (G43, G44, G49) ........ 85
4.5.2 Tool Radius Compensation (G40/G41/G42) ........ 87
4.5.3 Detailed Description of Tool Radius Compensation ........ 93
4.5.4 Corner Offset Circular Interpolation (G39) ........ 107
4.5.5 Input of Tool Compensation Value and Compensation Number With Program (G10) .. 108
4.6 Feed (G Code) ........ 108
4.6.1 Feed Mode (G64/G61/G63) ........ 108
4.6.2 Automatic Corner Override (G62) ........ 109
4.7 Macro Function (G Code) ........ 111
4.7.1 User Macro Program ........ 111
4.7.2 Macro Variables ........ 112
4.7.3 User Macro Program Call ........ 117
4.7.4 User Macro Program - Function A ........ 117
**Chapter V Auxiliary Function M Code**
5.1 M Code Controlled By PLC ........ 124
5.1.1 Spindle Rotation Cw and Ccw Commands (M03, M04) ........ 124
5.1.2 Spindle Stop Code Command (M05) ........ 124
5.1.3 Cooling On and Off (M07,M08, M09) ........ 124
5.1.4 A-Axis Unclamping and Clamping (M10, M11) ........ 124
5.1.5 Tool Control - Unclamping and Clamping (M16, M17) ........ 124
5.1.6 Spindle Orientation and Cancellation (M18, M19) ........ 124
5.1.7 Tool library zeroing instruction (M20) ........ 125
5.1.8 Tool library forward and backward code instructions (M23, M24) ........ 125
5.1.9 Tool library retrieval and tool library return operations (M25, M26) ........ 125
5.1.10 Rigid tapping opening and closing (M29, M28) ........ 125
5.1.11 Lubrication on/off (M32, M33) ........ 125
5.1.12 Chip conveyor on/off (M35, M36) ........ 125
5.1.13 Automatic tool change buckle, exchange, return (M45, M46, M47) ........ 125
5.1.14 Second spindle forward rotation, reverse rotation, stop (M63, M64, M65) ........ 125
5.2 M Code For Program Control ........ 125
5.2.1 Program End and Return (M30, M02) ........ 126
5.2.2 Program Halt (M00) ........ 126
5.2.3 Selective Halt of Program (M01) ........ 126
5.2.4 Code Command For Program Calling Subprogram (M98) ........ 126
5.2.5 Program End and Return (M99) ........ 126
**Chapter VI Spindle Function S Code**
6.1 Spindle Analog Control ........ 127
6.2 Constant Surface Cutting Speed Control G96/G97 ........ 127
**Chapter VII Feed Function F code**
7.1 Fast Movement ........ 130
7.2 Cutting Speed ........ 130
7.2.1 Feed Per Minute (G94) ........ 130
7.2.2 Feed Per Revolution (G95) ........ 131
7.3 Tangential Speed Control ........ 131
7.4 Feed Speed Override Key ........ 132
7.5 Automatic Acceleration and Deceleration ........ 132

7.6 Acceleration and Deceleration Processing At Program Segment Corner ....................................132
Chapter VIII Tool Functions
8.1 Tool Functions ....................................134

## VOLUME II OPERATING DESCRIPTION

Chapter I Operating Panel
1.1 Panel division ....................................136
1.2 Panel Function Description ....................................136
1.2.1 LCD (Liquid Crystal Display) display area ....................................136
1.2.2 Edit keyboard area ....................................137
1.2.3 Introduction to screen operation keys ....................................138
1.2.4 Machine tool control area ....................................138
Chapter 2 System Power On, Power Off, and Safe Operation
2.1 System power on ....................................142
2.2 turn off a machine ....................................142
2.3 safe operation ....................................143
2.3.1 Reset ....................................143
2.3.2 ESP ....................................143
2.4 Cycle Start and Feed Hold ....................................144
2.5 Overstroke Protection ....................................144
2.5.1 Software Overstroke Protection ....................................144
2.5.2 Software Overstroke Protection ....................................145
2.6 Stroke Inspection ....................................145
Chapter 3 Interface Display and Data Modification and Setting
3.1 Position display ....................................148
3.1.1 Five ways to display location pages ....................................148
3.1.2 Clear the number of completed items, set the required total number of items, and clear the cutting time ....................................152
3.1.3 Steps for resetting machine coordinates ....................................152
3.2 Program display ....................................152
3.2.1 program display ....................................153
3.2.2Program(directory)display ....................................154
3.2.3Program(USB)display ....................................155
3.3 tool compensation display, modification, and settings ....................................156
3.3.1 tool repair display ....................................156
3.3.2 [Coordinate System] Interface Operation Instructions ....................................156
3.3.3common variable interface ....................................157
3.3.4System Variables interface ....................................158
3.4 Set Display ....................................158
3.4.1 Set Display ....................................158
3.4.2 Password modification and setting ....................................159
3.4.3 Datarestoration ....................................160
3.4.4 Data backup ....................................161
3.4.5 Graphical ....................................162
3.5 Parameter Display ....................................163
3.5.1 Parameter page ....................................163
3.6 diagnostic display ....................................164
3.6.1Diagnostic data display ....................................164
3.6.2 PLC signal display ....................................165
3.6.3 IO monitoring display ....................................165
3.6.4 User M code display ....................................166
3.6.5 Version information ....................................167
3.6.6 CNC assistance ....................................167
3.7 Ladder diagram display and parameter modification ....................................169
3.7.1 ladder diagram ....................................169
3.7.2 PLC parameters ....................................171
3.7.3 PLC signal ....................................172
3.7.4 PLC information ....................................172

## Definition and Connection of Interface III

1.1 X. Connection of Y-axis and Z-axis and A-axis interfaces....175
1.1.1 Driver interface definition....175
1.1.2 Command pulse signal and command direction signal ....175
1.1.3 Drive unit alarm signal nALM....176
1.1.4 Axis enable signal ENn....176
1.1.5 Zero point signal PCn....176
1.1.6 Connection with servo drive....178
1.2 Connection of spindle encoder interface ....179
1.2.1 Definition of spindle encoder interface....179
1.2.2 SVC signal description....179
1.2.3 ALM Description....180
1.2.4 signal description....180
1.2.5 Spindle encoder interface connection....181
1.3 Connection of hand pulse interface....181
1.3.1 Hand pulse interface definition....181
1.3.2 signal description....182
1.4 communication interface....184
1.5 I/O interface definition....185
1.5.1 Input signal....186
1.5.2 Output Signal....188
1.5.3 Fifth axis signal....191
1.5.4 Output signal 2....191
1.5.5 Input signal 2....192
1.6 Fixed cycle cancellation G80....193
1.6.1 Fixed cycle cancellation G80....193
1.6.2 External cycle start and feed hold....194
1.6.3 Control of spindle counterclockwise and clockwise rotation....195
1.6.4 Spindle orientation and position switching function....195
1.6.5 Lubrication control....196
1.6.6 Main spindle elastic tool function....197
1.6.7 Rigid tapping debugging....198
1.6.8 Machining center tool magazine....200
1.6.8.1 Parameter 001=1, CNC milling cutter....200
1.6.8.2 Parameter 001=2, Douli tool Library....204
1.6.8.3 Parameter 001=3, disk manipulator tool library....209
1.6.8.4 Parameter 001=4, servo bucket hat tool library....213

1.6.9 EtherCAT bus wiring instructions....216

# Summary

This manual consists of the following parts:

I programming describes the composition of the I5M5 system program and the basic knowledge of programming, as well as the functions of each code, the code format, characteristics, and limitations of programming in NC language.

Operation II describes the various interfaces and settings of I5M5 CNC system, the operation and automatic running of the machine tool, the input/output and editing of programs, system communication, and other related content.

Definition and Connection of Interface III

Described the various functional wiring definitions and debugging instructions of the 980MFi CNC system.

# Chapter 1 Overview

## 1.1 Product Introduction

The system adopts a brand new 32-bit high-performance CPU processor and a super large scale programmable device FPGA, with large operating memory and storage space. The advanced hardware platform and advanced control algorithms ensure high efficiency at the system's um level accuracy. The editable PLC makes the logic control function more flexible and powerful. Support 23 bit absolute encoder servo motor, support real-time mechanical position memory function after power failure, high accuracy, no return to zero


- Number of control axes: 5 feed axes, 1 channel of 0V~10V analog voltage, 1 channel of pulse spindle
- Number of linkage axes: 5 linear axes, 3 circular axes
- Support any tool library function,
- Open PLC, supporting secondary development
- G10 instruction for online modification of tool compensation, parameters, and other operations
- Support fixed cycle, drilling cycle, and rigid tapping
- Program preprocessing enables smooth and seamless connection between program segments
- The maximum fast moving speed can reach 60m/min, and the maximum cutting speed can reach 30m/min
- 430M program storage space, capable of storing 400 programs, with a maximum of 10M per program

## 1.2 technical norms

| | |
|---|---|
| **motion control function** | Control axis: X-axis, Z-axis, Y-axis, A-axis, B-axis; |
| | Interpolation method: positioning (G00), straight line (G01), arc (G02, G03) |
| | Position instruction range: metric: -99999999.99mm~999999.99mm, minimum instruction unit: 0.0001mm<br>English: -999.9999inch to 9999.9999inch, minimum instruction unit: 0.00001inch<br>Note: When editing the diameter, the X-axis is reduced by half |
| | Maximum feed speed: linear 8000mm/min<br>Feed rate: 0-150%, 15 level real-time adjustment |
| | Maximum fast speed: 60000 mm/min<br>Fast magnification: F0, 25%, 50%, 100% four level real-time adjustment |
| | Feed per revolution: 0.01 mm/r~500mm/r (1024P/r or 1200P/r spindle encoder needs to be installed) |
| | Acceleration/deceleration mode: front acceleration/deceleration (linear type, S type), rear acceleration/deceleration (linear type, Exponential type type) |
| | Electronic gear ratio: frequency doubling 1~65535, frequency division 1~65535 |
| | Hand pulse feed: 0.001, 0.01, 0.1mm three gears; Single step feed: 0.001, 0.01, 0.1, 1mm four gears |
| **display interface** | ❖ The system adopts a resolution of 800 × 600 color 8-inch LCD display<br>❖ Processing trajectory display |
| **G function** | ❖ System A instruction format is adopted, with 30 G-code in total, including fixed cycle code and compound cycle code<br>❖ Support for statement based macro programs<br>❖ Support for 5-level subroutine calls and user macro program calls |
| **Tool function** | ❖ Tool length compensation (32 sets)<br>❖ Tool wear compensation (32 sets)<br>❖ Blade tip radius compensation (C-type)<br>❖ G54-G59 coordinate system |

| | |
|---|---|
| **S principal axis function** | ❖ S2 digits (I/O gear control)/S5 digits (analog output)<br>❖ Spindle encoder: The number of encoder lines can be set (100-5000p/r)<br>❖ Encoder to spindle transmission ratio: (1-255): (1-255)<br>❖ Spindle magnification: 50%~120%, a total of eight levels of real-time adjustment<br>❖ 1 channel 0V~10V analog voltage output |
| **M auxiliary function** | ❖ Specify with address M and 2 digits, M function can be customized<br>❖ System internal M instruction (non redefining): program end M02, M30; Program stop M00; Select to stop M01; Subprogram call M98; Subprogram end M99<br>❖ Cooling liquid start/stop, lubrication start/stop, MDI/automatic control, chuck clamping/loosening, control tailstock in/out |
| **program edit** | ❖ Program capacity: 400MB, 4000 programs<br>❖ Format: Relative/Absolute Mixed Programming<br>❖ Subprogram: editable, supports five fold subroutine nesting |
| **Operation** | ❖ Method selection: edit, automatic, MDI, zero return, manual, single step, manual pulse<br>❖ Operation control: single segment, skip segment, idle operation, auxiliary lock, program restart, manual pulse interrupt, single step interrupt, manual intervention, machine lock, interlock, feed hold, cycle start, emergency stop, external reset signal |
| **PLC** | ❖ PLC processing speed: 1 us/step; Up to 8000 steps; 10 basic instructions and 35 functional instructions;<br>❖ I/O unit input/output: 16/16, expandable |
| **Security** | ❖ emergency stop<br>❖ Hardware travel limit<br>❖ Data backup and recovery |
| **communication** | ❖ USB: USB file operation, direct processing of USB files, support for PLC program and system software USB upgrade |

# Chapter II Composition of Parts Program

## 2.1 **Program Component**

**A program is composed of multiple program segments, which in turn are composed of words. Each program segment ends the code with a program segment, which is represented by the character ";" in this manual.**

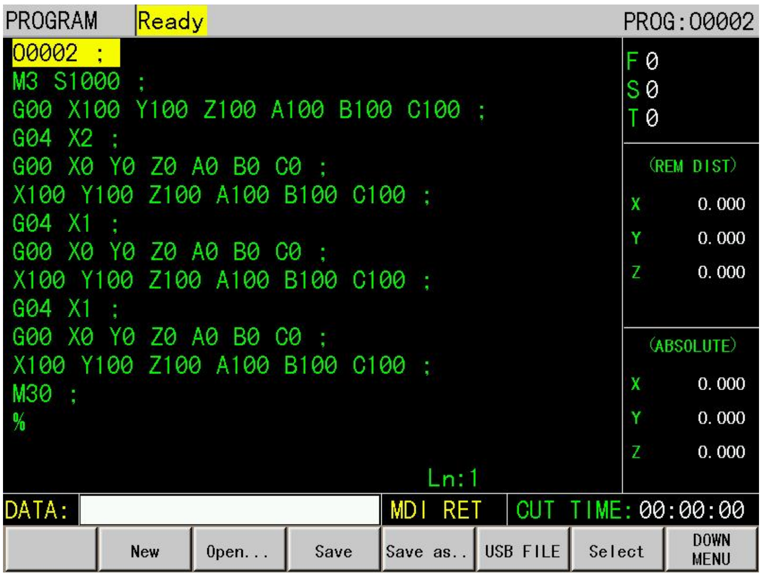



Figure 2-1-1 Structure of program

The collection of code series that controls CNC machine tools to complete part processing is called a program. After inputting the pre written program into the CNC system, the system controls the tool to move along a straight line or arc, or to rotate or stop the spindle according to the code. In the program, these codes should be written based on the actual motion sequence of the machine tool. The structure of the program is shown in Figure 2-1-1.

### 2.1.1 **Program Name**

In this system, multiple programs can be stored in the system's memory. To distinguish these programs from each other, a sequence of letters and numbers or a Chinese name can be used to form the program name, and the program name suffix is NC .TXT.CNC

For example: Bergerda01.NC

贝格达 01.CNC

If the program is to be called (such as M98), the program name should be composed of the address O and the following four digits, as shown in Figure 2-1-1-1.

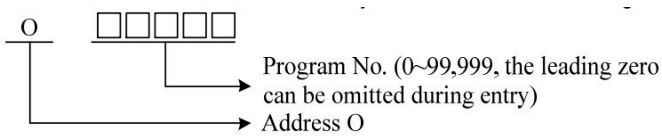



Figure 2-1-1-1Composition of program name

## 2.1.2 **Code Word**

The code word (Figure 2-1-2-1) is the element that makes up a program segment. It consists of the address and following numbers (which are sometimes preceded by symbols such as +, -).

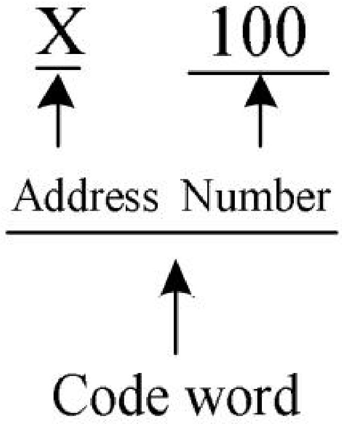



Figure 2-1-2-1 Composition of code word

The address is one of the English letters (A to Z). It defines the meaning of the following numbers. In this system, the address that can be used and its meaning and value range are shown in Table

| Address | Value range | Function meaning |
|---|---|---|
| O | 0～99999 | Program name |
| N | 0～99999 | Sequence No. |
| G | 00～99 | Preparation functions |
| X | -99999.9999～99999.9999（mm） | X-axis coordinate address |
| | 0～9999.999（S） | Specify pause time |
| Y | -99999.9999～99999.9999（mm） | Y-axis coordinate address |
| Z | -99999.9999～99999.9999（mm） | Z-axis coordinate address |
| R | -99999.9999～99999.9999（mm） | Arc radius/angular displacement/corner value |
| | -99999.9999～99999.9999（mm） | R plane in fixed cycle |
| I | -99999.9999～99999.9999（mm） | The arc center relative to origin is in the X-axis vector (circular/helical interpolation, scaling) |
| J | -99999.9999～99999.9999（mm） | The arc center relative to origin is in the Y-axis vector (circular/helical interpolation, scaling) |
| K | -99999.9999～99999.9999（mm） | The arc center relative to origin is in the Z-axis vector (circular/helical interpolation, scaling) |
| F | 0～99999（mm/min） | Feed per min |
| | 0.001～500(mm/r) | Feed per revolution |
| S | 0～99999（r/min） | Spindle speed specified |
| | 00～04 | Multi-speed spindle output |
| T | 0～9999 | Tool functions |
| M | 00～99 | Auxiliary function output, program execution flow, subprogram call |
| P | 0～99999.9999（ms） | Pause time |
| | 1～99999 | Called subprogram number |
| Q | -99999.9999～99999.9999（mm） | Cutting depth or hole bottom offset in fixed cycle |
| H | 01～99 | Operators in G65 |

| Address | Value range | Function meaning |
|---|---|---|
| | 00～256 | The length offset number (H0 defaults to 0 which the user cannot set or modify.) |
| D | 00～256 | The radius offset number, D0 defaults to 0 which the user cannot set or modify |

Please note that all of the limits shown in Table 3-1-3-1 are for CNC devices and not for machine tools. Therefore, in programming, please refer also to the machine manufacturer's user manual in addition to this manual so as to write programs based on the understanding of programming limits.

## 2.2 **General Structure of Program**

Programs are divided into main programs and subprograms. Usually, the CNC moves as instructed by the main programs. If codes of calling subprograms are present in main programs, the CNC moves as instructed by the subprograms. If codes of returning to main programs are present in subprograms, the CNC returns to the program segment following the segment where the main programs call the subprograms and continues execution. The sequence of program actions is shown in Figure 2-2-1.

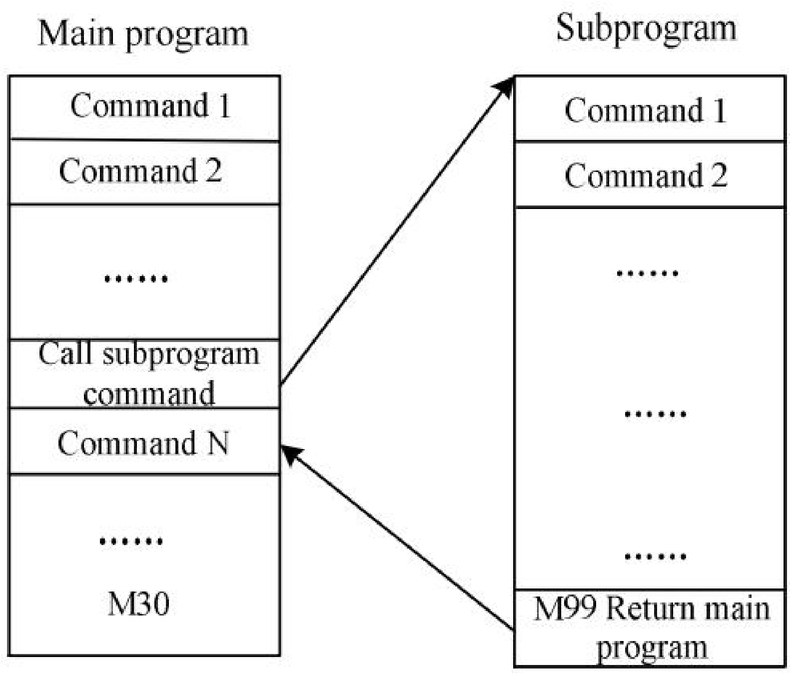



Figure 2-2-1

Main programs and subprograms share the same composition.

When a fixed sequence exists and repeats in a program, it can be stored as a subprogram in memory without having to write it repeatedly to simplify the program. The subprogram can be called in the automatic mode, generally called with M98 in the main program, and the called subprogram can also call another subprogram. The subprogram called from the main program is called one-fold subprogram, and four levels of subprograms can be called (see Figure 2-2-2). At the last segment of the subprogram the main program is returned with the M99 code, and the program following the subprogram segment is called to continue execution. (If at the last segment the subprogram is ended with the M02 or M30 code with the same function as M99 to return to the main program, the program following the subprogram segment is called to continue execution.)

When the main program ends with M99, the program is executed repeatedly.

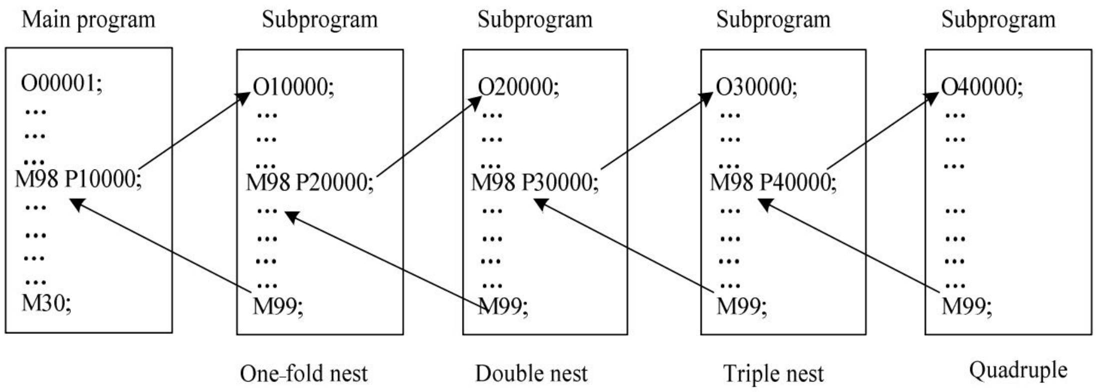



**Figure 2-2-2 Four levels of subprogram nesting**

A subprogram call code can be used to call the same subprogram continuously and repeatedly, up to 9,999 times.

## 2.2.1 **Subprogram Writing**

A subprogram is written in the following format:

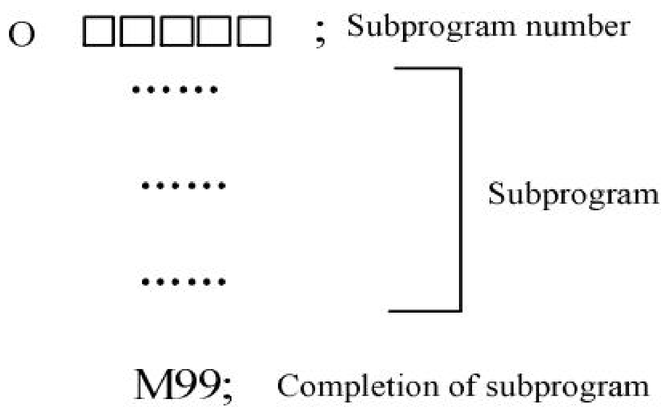



Fig.**2-2-1-1**

A subprogram starts with address O, followed by subprogram number, and ends with M99 code (the M99 is written as shown above).'

## 2.2.2 **Subprogram Call**

The subprogram is called by the main program or subprogram call code for execution. The format of a code for calling a subprogram is as follows:

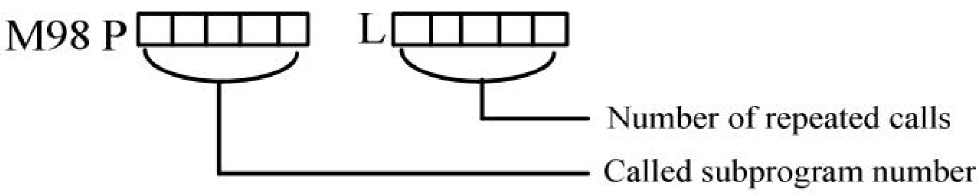



Fig.**2-2-2-1**

- If the number of repeats is omitted, it is considered to be 1.

  (Example) M98 P1002L5; (It indicates the subprogram with the number 1002 is called 5 times in succession.)

- The sequence in which the subprograms are called from the main programs for execution

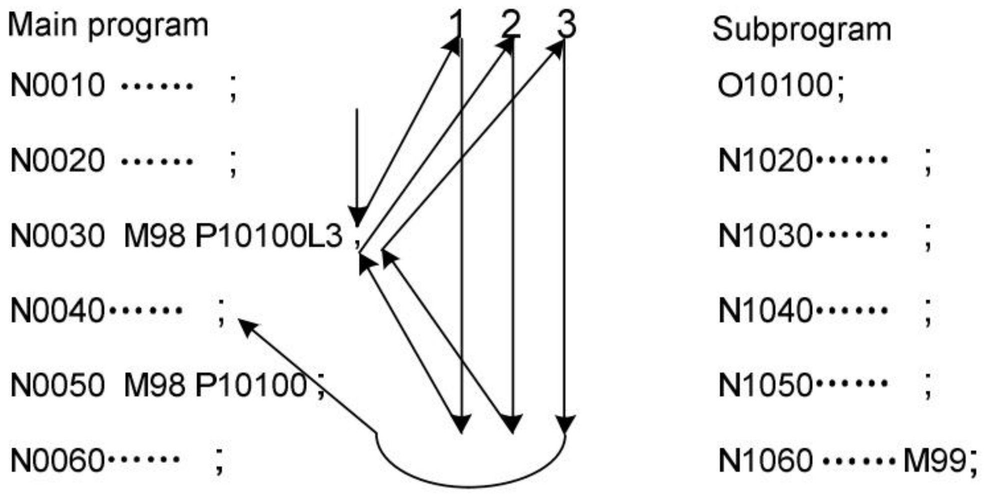



Fig.**2-2-2-2**

Calling a subprogram from a subprogram is the same as calling a subprogram from a main program.

**Note 1**: An alarm is generated when the subprogram number specified by the address P cannot be retrieved.

**Note 2**: The subprograms No.90000~99999 are the system retained programs. When the user calls such subprograms, the system can execute the subprogram content but will not display it.

**Note 3**: Subprogram calls can be nested up to four levels.

## 2.2.3 **End of Program**

A program starts with the program name and ends with M02, M30 or M99 (see Figure 2-2-2-2). In the execution of a program, if the program end code M02, M30 or M99 is detected, the program ends if it ends with code M02 or M30 and the system is in reset state; for M30, the position parameter **N0:33#4** can be used to control whether to return to the program header, and for M02, the position parameter **N0:33#2** can be used to control whether to return to the program header. If it ends with code M99, it will return to the program header and the program loops. If M99, M02, and M30 are at the end of a subprogram, it will return to the program calling the subprogram and continue to execute the following program segment.

# Chapter III Programming Basics

## 3.1 **Control Axis**

Table **3-1-1**

| **Item** | 980MFi |
|---|---|
| Number of basic control axes | 3 (X, Y, Z) |
| Number of extended control axes (total) | Up to 5 |

## 3.2 **Axis Name**

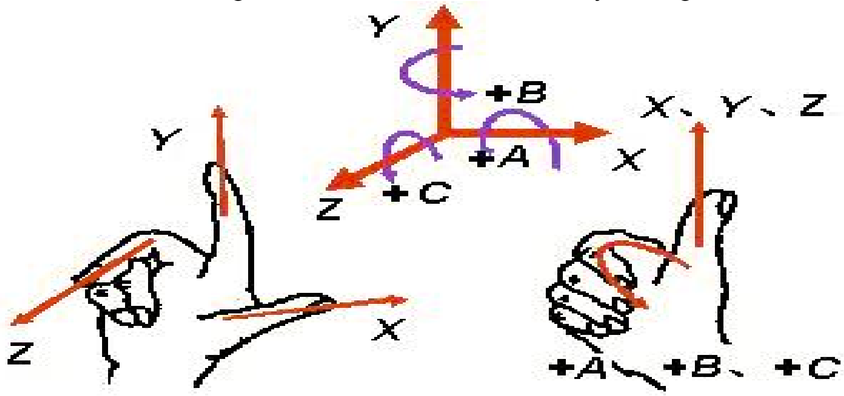
The names of the three basic axes are always X, Y, and Z.
Set the number of axes using parameter - [feed axis parameter] - P100

## **3.3 Coordinate System**

### 3.3.1 **Machine Coordinate System**

The specific point on the machine that is used as the machining reference is called the machine zero. The machine manufacturer sets the machine zero for each machine. The coordinate system set with the machine zero as the origin is called the machine coordinate system. After power is turned on, a manual return to the machine zero is performed to establish the machine coordinate system. Once the machine coordinate system is set, it remains unchanged until the power is turned off, the system is restarted, or the emergency stop is pressed.

The system adopts the right-hand Cartesian coordinate system. The vertical movement in the spindle direction is Z-axis movement. Viewing from the spindle to the workpiece, the spindle box is in Z-axis negative movement approaching workpiece and in Z-axis positive movement leaving workpiece. The remaining directions are determined by the right-hand Cartesian coordinate system.




### 3.3.2 **Reference Point**

On **CNC** machines, there is a special position where tool change usually takes places or the coordinate system is set, called the reference point. It is a fixed point in the machine coordinate system set by the machine manufacturer. With the reference point return function the tool can be easily moved to this position. Generally, the reference point of the CNC milling machine system coincides with the machine zero, and the machining center reference point is usually the tool change point.

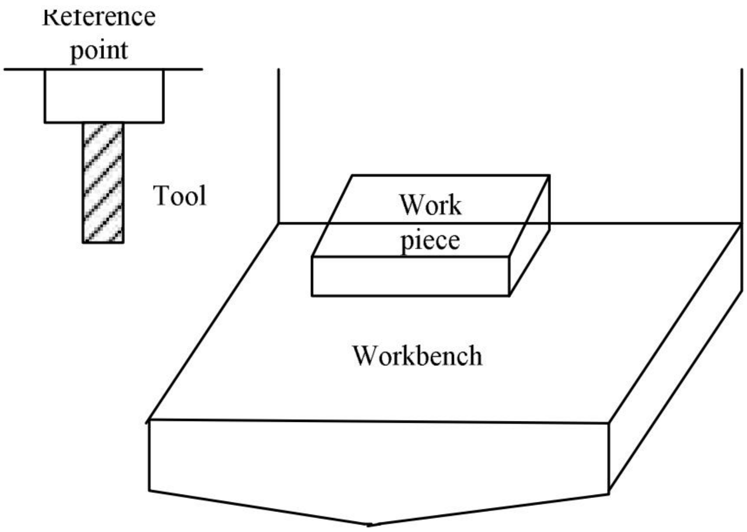



Fig.**3-3-2-1**

To move the tool to the reference point, there are two ways:
1. Manually return to the reference point (see “Chapter IX Zeroing Operation”);
2. Automatically return to the reference point.

## 3.3.3 **Workpiece Coordinate System**

The coordinate system used when machining a workpiece is called workpiece coordinate system (also called part coordinate system). The workpiece coordinate system is preset by the **CNC** (setting the workpiece coordinate system).

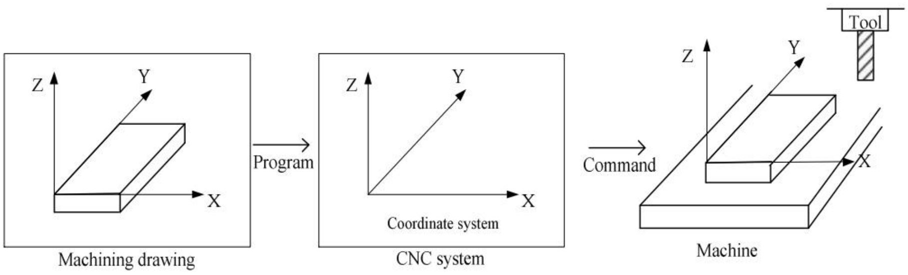



Fig.**3-3-3-1**

The tool cuts the workpiece into the shape as indicated on the drawing on the **CNC** command workpiece coordinate system according to the command program of the programmed coordinate system on the machining drawing. The relative relationship between the machine coordinate system and the workpiece coordinate system must be determined. The method of determining the relative relationship between those two coordinate systems is called alignment. Different methods can be used depending on the shape of parts and the number of parts to be machined.

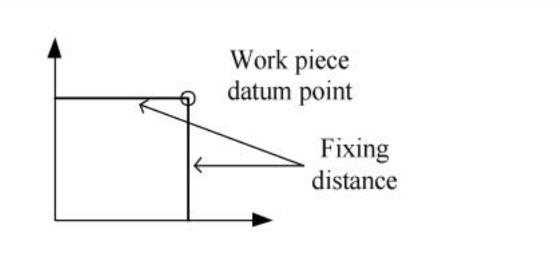
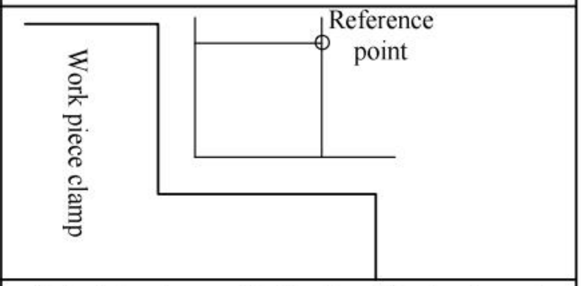
| I) Use the part datum point | II) When the part is directly mounted on the clamp |
|---|---|
|  |  |
| Move the tool center and align it with the part datum point. In this position, use the CNC command to set the work piece coordinate system. At this moment, the work piece coordinate system is coincided with the programming coordinate system. | As the tool center cannot be directly positioned on the work piece datum point, it is necessary to position the tool to a position with known distance to the datum point (which can be the reference point), and use the known distance to set the work piece coordinate system of the CNC command. (Such as G92) |

Fig.**3-3-3-2**

A workpiece coordinate system is set for each machining program (selecting a workpiece coordinate system). The set workpiece coordinate system can be changed by moving its origin. There are two ways to set the workpiece coordinate system:

1. G92. For details, see 4.2.11 of this chapter.
2. G54 to G59. For details, see 4.2.8 of this chapter.

## 3.3.4 **Absolute Coordinate Programming and Relative Coordinate Programming**

There are two methods of defining the axis movement amount: absolute value and relative value. The absolute value definition is a method of programming with the coordinates of the end position of the axis movement, called absolute coordinate programming. The relative value definition is a method of directly programming with the axis movement amount, called relative coordinate programming (also called incremental coordinate programming).

1) Absolute coordinates

The coordinates of the target position in the specified workpiece coordinate system, that is, the coordinate position to which the tool is moved.

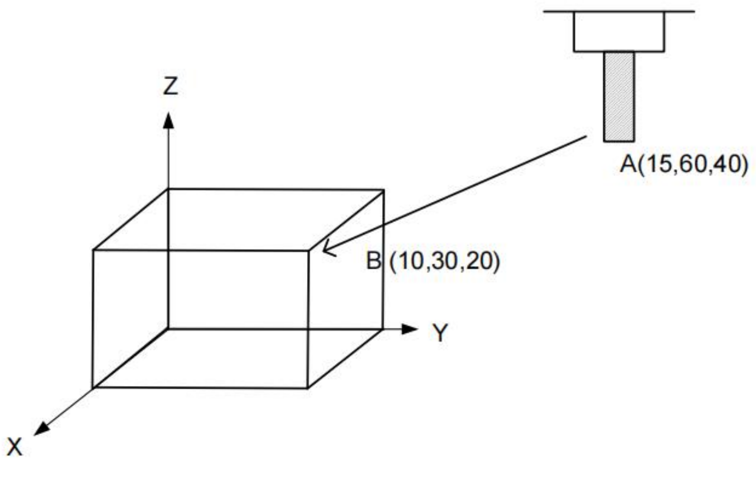



Fig.**3-3-4-1**

The tool moves from point A to point B. The coordinates of point B are used in the G54 workpiece coordinate system. The code is as follows:

G90 G54 X10 Y30 Z20;

2) Incremental coordinates

The coordinates of the target position relative to the current position using the current position as the coordinate origin.

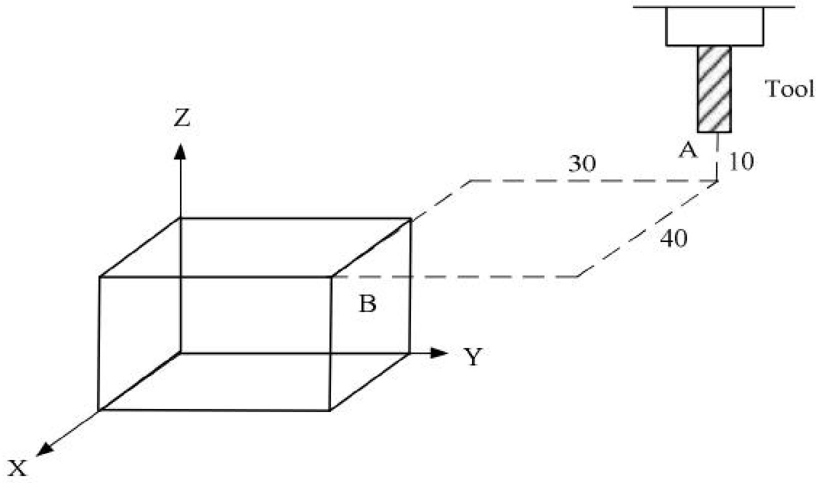



Fig.**3-3-4-2**

The tool moves fast from point A to point B. The code is as follows:
G0 G91 X40 Y-30 Z-10;

## 3.4 **Modal and Non-Modal**

Modal means that the value of an address remains valid as soon as it is set until it is reset. Another meaning of modal is that after a function word is set, if the same function is used in a later program segment, it is not necessary to input the field.

- For example, the following programs:
  G0 X100 Y100; (Fast positioning to X100 Y100)
  X20 Y30; (Fast positioning to X20 Y30, G0 is modal specified and is not required to input)
  G1 X50 Y50 F300; (Linear interpolation to X50 Y50, feed speed 300mm/min G0→G1)
  X100; (Linear interpolation to X100 Y50, feed speed 300mm/min, G1, Y50, F300 are modal specified and are not required to input)
  G0 X0 Y0; (Fast positioning to X0 Y0)

  The initial state refers to the default modal after the system is powered on. See Table 4-1-2 for details.

- For example, the following programs:
  O00001
  X100 Y100; (Fast positioning to X100 Y100, G0 is the initial state of the system)
  G1 X0 Y0 F100; (Linear interpolation to X0 Y0, feed per minute at a speed of 100mm/min)

Non-modal means that the value of the corresponding address is valid only in the program segment where this code is written, and must be reassigned if it is used in the later program segment. For example, Group 00 Code G in Table 4-1-2.

See Table 3-4-1 for the modal and non-modal descriptions of the function words.

Table 3-4-1 Modal and non-modal of function codes

| | | |
|---|---|---|
| **Modal** | Modal G function | A set of G functions that can end each other and, once executed, remain valid until they are ended by the G function in the same group. |
| | Modal M function | A set of M functions that can end each other and remain valid until they are ended by another function in the samegroup |
| **Non-modal** | Non-modal G function | Valid only in the specified program segment, and ended at the end ofthe program segment |
| | Non-modal M function | Valid only in the program segment where this code is written |

## 3.5 Decimal point programming

Numerical values can be entered with a decimal point. When entering distance, time, or speed, a decimal point can be used. The following address can specify the decimal point:

**X、Y、Z、A、B、C、I、J、K、R、P、Q、F。**

**explain:**

Decimals smaller than the minimum input increment unit are rounded off.

**example**

X9.87654； When the minimum input increment unit is 0.001mm, process it as X 9.876.
When the minimum input increment unit is 0.0001mm, process it as X 9.8765.

# Chapter IV Preparation Function G Code

## 4.1 Type of Preparation Function G Code

The preparation function is indicated in G code followed by numbers, specifying the meaning of the program segment where it is located. There are two types of G codes:

Table **4-1-1**

| Type | Meaning |
|---|---|
| Non-modal G code | Valid only in the program segment where it is instructed |
| Modal G code | Always valid before other G codes in the same group |

(Example) G01 and G00 are modal G codes in the same group

G01 X __ ;

Z ______ ; G01 is valid

X ______ ; G01 is valid

G00 Z____; G00 is valid

**Note:** Please refer to the system parameter table for specific system parameters

Table 4-1-2 G code and function

<table>
<tr><th>G code</th><th>Group</th><th>Code form</th><th>Function</th></tr>
<tr><td>*G00</td><td rowspan="4">01</td><td>G00 X_Y_Z_</td><td>Positioning (fast movement)</td></tr>
<tr><td>G01</td><td>G01 X_Y_Z_F_</td><td>Linear interpolation (cutting feed)</td></tr>
<tr><td>G02</td><td rowspan="2">G02 G03 X_Y_ R_ I_J_ F_;</td><td>Circular interpolation CW (clockwise)</td></tr>
<tr><td>G03</td><td>Circular interpolation CCW (counterclockwise)</td></tr>
<tr><td>G04</td><td rowspan="3">00</td><td>G04 P_ 或 G04 X_</td><td>Pause, exact stop</td></tr>
<tr><td>G10</td><td>G10 L_N_P_R_</td><td>Programmable data input</td></tr>
<tr><td>*G11</td><td>G11</td><td>Cancel programmable data input mode</td></tr>
<tr><td>*G12</td><td rowspan="2">16</td><td>G12 X_Y_Z_ I_J_K_</td><td>Stored travel detection function connected</td></tr>
<tr><td>G13</td><td>G13</td><td>Stored travel detection function disconnected</td></tr>
<tr><td>*G15</td><td rowspan="2">11</td><td>G15</td><td>Cancel polar coordinate code</td></tr>
<tr><td>G16</td><td>G16</td><td>Polar coordinate code</td></tr>
<tr><td>*G17<br>G18<br>G19</td><td>02</td><td>They are written in the program segments and used in circular interpolation and tool radius compensation.</td><td>XY plane selection<br>ZX plane selection<br>YZ plane selection</td></tr>
<tr><td>G20</td><td rowspan="2">06</td><td rowspan="2">The command must be specified in a separate program segment</td><td>Imperial data input</td></tr>
<tr><td>*G21</td><td>Metric data input</td></tr>
<tr><td>G22</td><td>09</td><td>G22 X_Y_Z_R_I_L_W_Q_V_D_F_ K</td><td>Groove rough milling inside circle (CCW)</td></tr>
</table>

| | | | | | |
|---|---|---|---|---|---|
| G23 | | G23 X_Y_Z_R_I_L_W_Q_V_D_F_K | | | Groove rough milling inside circle (CW) |
| G24 | | G24 X_Y_Z_R_I_J_D_F_K_ | | | Finish milling cycle inside full circle (CCW) |
| G25 | | G25 X_Y_Z_R_I_J_D_F_K_ | | | Finish milling cycle inside full circle (CW) |
| G26 | | G26 X_Y_Z_R_I_J_D_F_K_ | | | Finish milling cycle outside circle (CCW) |
| G27 | | G27 | | | Reference point return detection |
| G28 | | G28 | | | Reference point return |
| G29 | 00 | G29 | X_Y_Z_ | | Return from reference point |
| G30 | | G30Pn | | | Return to reference points 2, 3 and 4 |
| G31 | | G31 | | | Skip function |
| G32 | | G32 X_Y_Z_R_I_J__D_F_K_ | | | Finish milling cycle outside circle (CW) |
| G33 | | G33X_Y_Z_R_I_J_L_W_Q_V_U_D_F_K | | | Rectangular groove rough milling (CCW) |
| G34 | | G34X_Y_Z_R_I_J_L_W_Q_V_U_D_F_K | | | Rectangular groove rough milling (CW) |
| G35 | 09 | G35 X_Y_Z_R_I_J_L_ U_D_F_K_ | | | Finish milling cycle inside rectangular groove (CCW) |
| G36 | | G36 X_Y_Z_R_I_J_L_ U_D_F_K_ | | | Finish milling cycle inside rectangular groove (CW) |
| G37 | | G37 X_Y_Z_R_I_J_L_ U_D_F_K_ | | | Finish milling cycle outside rectangle (CCW) |
| G38 | | G38 X_Y_Z_R_I_J_L_ U_D_F_K_ | | | Finish milling cycle outside rectangle (CW) |
| G39 | 00 | G39 I_J_； I_K_； J_K_或 G39 | | | Corner offset circular interpolation |
| *G40 | | G17 | G40 G41 G42 | D_X_Y_ | Cancel tool radius compensation |
| G41 | 07 | G18 | | D_X_Z_ | Left tool radius compensation |
| G42 | | G19 | | D_Y_Z_ | Right tool radius compensation |
| G43 | | G43 | | | Positive direction tool length compensation |
| G44 | 08 | G44 | H_Z_ | | Negative direction tool length compensation |
| *G49 | | G49 | | | Cancel tool length compensation |
| *G50 | 12 | G50 | | | Cancel scaling |
| G51 | | G51 X_ Y_ Z_ P_ | | | Scaling |
| G53 | 00 | It is written in programs | | | Selection of machine coordinate system |
| *G54 | | | | | Workpiece coordinate system 1 |
| G55 | | | | | Workpiece coordinate system 2 |
| G56 | 05 | They are written in the program segments, usually at the beginning of the programs. | | | Workpiece coordinate system 3 |
| G57 | | | | | Workpiece coordinate system 4 |
| G58 | | | | | Workpiece coordinate system 5 |

| | | | |
|---|---|---|---|
| G59 | | | Workpiece coordinate system 6 |
| G65 | 00 | G65 H_P# i Q# j R# k | Macro program code |
| G68 | 13 | G68 X_ Y_ R_ | Coordinate rotation |
| *G69 | | G69 | Cancel coordinate rotation |
| G73 | 09 | G73 X_Y_Z_R_Q_F_; | High-speed deep hole machining cycle |
| G74 | | G74 X_Y_Z_R_P_F_; | Left tapping cycle |
| G76 | | G76 X_Y_Z_R_P_F_K_; | Precision boring cycle |
| *G80 | | It is written with other programs in the program segments | Cancel fixed cycle |
| G81 | | G81 X_Y_Z_R_F_; | Drilling cycle (point drilling cycle) |
| G82 | | G82 X_Y_Z_R_P_F_; | Drilling cycle and boring cycle |
| G83 | | G83 X_Y_Z_R_Q_F_; | Chip removal drilling cycle |
| G84 | | G84 X_Y_Z_R_P_F_; | Right tapping cycle |
| G85 | | G85 X_Y_Z_R_F_; | Boring hole cycle |
| G86 | | G86 X_Y_Z_R_F_; | Boring hole cycle |
| G87 | | G87 X_Y_Z_R_Q_P_F_; | Back boring cycle |
| G88 | | G88 X_Y_Z_R_P_F_; | Boring hole cycle |
| G89 | | G89 X_Y_Z_R_P_F_; | Boring hole cycle |
| *G90 | 03 | They are written in program segments | Absolute value programming |
| G91 | | | Incremental value programming |
| G92 | 00 | G92 X_Y_Z_ | Floating coordinate system setting |
| *G94 | 04 | G94 | Feed per minute |
| G95 | | G95 | Feed per revolution |
| G96 | 15 | G96S_ | Constant peripheral speed control (cutting speed) |
| *G97 | | G97S_ | Cancel constant peripheral speed control (cutting speed) |
| *G98 | 10 | They are written in program segments | Return to the initial plane in a fixed cycle |
| G99 | | | Return to the R point plane in a fixed cycle |

**Note 1**: If the modal codes are in the same segment as the non-modal codes, the non-modal codes take precedence and change to the corresponding modes according to other modal codes in the same segment, but they are not executed.

**Note 2**: For G codes with * mark, when the power is turned on, the system is in the state of these G codes. (Some G codes are determined by the position parameter NO: 31#0~7)

**Note 3**: The G codes in group 00 are non-modal G codes except for G10, G11, and G92.

**Note 4**: If G codes not listed in the G code table are used, an alarm occurs, or instructing G codes without selection function also leads to an alarm.

**Note 5**: G codes from several different groups can be instructed in the same program segment. In principle, two or more G codes from the same group cannot be instructed in the same program segment. If the alarm is set to not occur for codes from the same group being in the same segment, the G code that appears later will apply.

**Note 6**: When the codes from groups 01 and 09 are in the same segment, the codes from group 01 will apply. In the fixed cycle mode, if the G codes from the group 01 are instructed, the fixed cycle is automatically canceled and the system is in the G80 state.

**Note 7**: The G codes are represented by group number depending on their types. It is set by position parameters NO: 35#0~7 and NO: 36#0~7 whether to clear the G codes of each group during reset or emergency stop.

**Note 8**: When the rotating and scaling codes are in the same segment as codes from group 01 or 09, the rotating and scaling codes will apply, and the mode of the codes from group 01 or 09 will be changed. The system will give an alarm when the rotating and scaling codes are in the same segment as the codes from group 00.

# 4.2 **Simple G Code**

## 4.2.1 Fast Positioning G00

**Format: G00 X_Y_Z_**

**Function:** With the G00 code, the tool moves fast to the position in the workpiece coordinate system specified by the absolute value code or incremental value code. The position parameter **N0:12#1** is used for setting, and one of the following two tool paths is selected (see Figure 4-2-1-1).

1. Linear interpolation positioning: The tool path is the same as the linear interpolation (G01), and the tool moves fast to the specified position in the shortest time without exceeding the speed of each axis.

2. Non-linear interpolation positioning: The tool moves fast at the speed of each axis, and the tool path is generally non-linear.

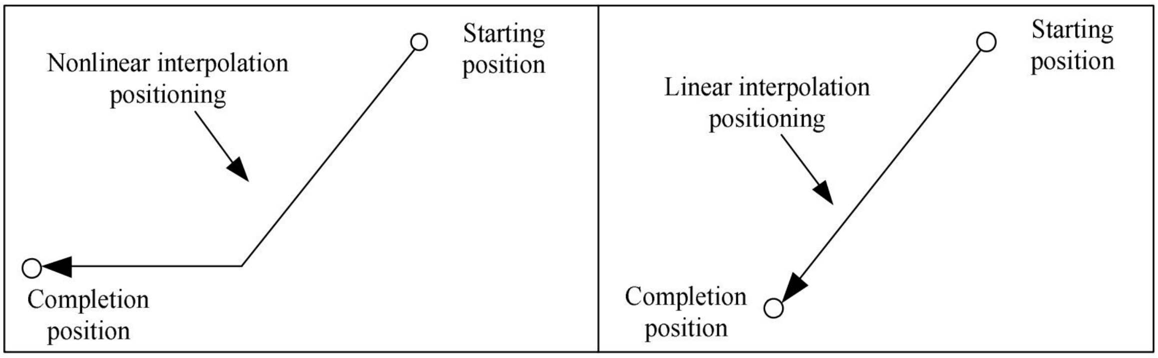



Fig.**4-2-1-1**

**Description:**

1、 Without specifying the positioning parameter, the tool does not move, and the system only changes the mode of the current tool movement to G00.

2、 G00 and G0 are equivalent formats.

3、 X. The speed of the Y and Z axes G0 is set by the parameter - [feed axis parameter] P026-P028.

**Restrictions:**

The fast movement speed is set by parameters. For example, F speed set in the G0 code is the cutting feed speed of the following machining segment.
For example:
G0 X0 Y10 F800; Fast feed at a speed set by system parameters
G1 X20 Y50; Feed at the speed of F800
The fast positioning speed is adjusted by the keys on the operation panel (as shown in Figure 4-2-1-2) to F0, 25%, 50%, 100%; the speed corresponding to F0 is set by data parameter [Feed axis parameters] P052 and applies to each axis.


Figure **4-2-1-2** Fast feed override key

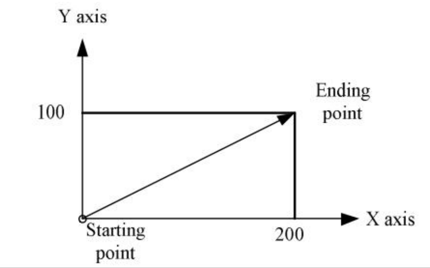
**Note:** During programming, pay attention to the position of table and workpiece to prevent tool collision.

## 4.2.2 **Linear Interpolation G01**

**Format: G01 X_ Y_ Z_ F_**
**Function:** The tool moves in the straight line to the specified position at the feed speed (mm/min) specified by parameter F.
**Description:**

1. X_ Y_ Z_ is the ending coordinate. For the concept of coordinate system, please refer to sections3.3.1～3.3.3
2. The F specified feed speed remains valid until a new F value is specified. The feed speed instructed by the F code is calculated by interpolation along the linear path. If the F code does not instruct in the program, the feed goes at the feed speed of the default F value when the system is powered on. [Feed axis parameters] P051

Program instance (Figure 4-2-2-1).




G01 X200 Y100 F200 ;

Note: The speed in each axis direction is as follows:
G01 Xα Yβ ZγFf ;
In this program segment:

Speed in X axis direction: $F_X = \frac{\alpha}{L} \times f$

Speed in Y axis direction: $F_Y = \frac{\beta}{L} \times f$

Speed in Z axis direction: $F_Z = \frac{\gamma}{L} \times f$

$L = \sqrt{\alpha^2 + \beta^2 + \gamma^2}$

Fig. **4-2-2-1**

**Note:**

1. The code parameters except for F are all position parameters. The upper limit of the cutting feed speed F is set by the data parameter [Feed axis parameters] P060. If the actual cutting speed (feed speed after the override is applied) exceeds the upper limit, it is limited to the upper limit. The unit is mm/min. The lower limit of the cutting feed speed F is set by the data parameter P97. If the actual cutting speed (feed speed after the override is applied) is lower than the limit, it is limited to the lower limit. The unit is mm/min.
2. The tool does not move when the position parameter is not specified after G01. The system just changes the mode of the current tool movement to G01. By changing the value of the system position parameter N0:31#0, it can be set whether the default mode is G00 (when the parameter value is 0) or G01 (when the parameter value is 1) when the power is turned on.

## 4.2.3 Circular (Helical) Interpolation G02/G03

**A. Circular interpolation G02/G03**

**G02 and G03 specification:**

The circular interpolation in a plane completes the circular path in the specified plane running from the start point to the end point in the specified direction of rotation and along the radius (or circle center). As knowing the start and end points cannot completely determine the circular path, it is necessary to give:

- The direction of rotation of the circular arc (G02, G03)
- The plane of the circular interpolation (G17, G18, G19)
- The center coordinates or radius, from which two code formats are derived, together with center coordinates I, J, K or radius R programming.

Only when the above three points are confirmed, interpolation operation can be done in the coordinate system.

The circular interpolation can be performed with the following codes, and the tool can move along the circular arc as follows:

Arc on the XY plane

$$G17 \begin{Bmatrix} G02 \\ G03 \end{Bmatrix} X\_Y\_ \begin{Bmatrix} R\_ \\ I\_J\_ \end{Bmatrix} F\_;$$

Arc on the ZX plane

$$G18 \begin{Bmatrix} G02 \\ G03 \end{Bmatrix} X\_Z\_ \begin{Bmatrix} R\_ \\ I\_K\_ \end{Bmatrix} F\_;$$

Arc on the YZ plane

$$G19 \begin{Bmatrix} G02 \\ G03 \end{Bmatrix} Y\_Z\_ \begin{Bmatrix} R\_ \\ J\_K\_ \end{Bmatrix} F\_;$$

Table **4-2-3-1**

| Item | Specified content | | Command | Meaning |
|---|---|---|---|---|
| 1 | Specify plane | | G17 | Specify arc on XY plane |
| | | | G18 | Specify arc on ZX plane |
| | | | G19 | Specify arc on YZ plane |
| 2 | Turning direction | | G02 | Turn CW |
| | | | G03 | Turn CCW |
| 3 | End position | G90 mode | Two axes of X, Y, Z | End position coordinates in the workpiece coordinate system |
| | | G91 mode | Two axes of X, Y, Z | Coordinates of end point relative to start point |
| 4 | Vector from start point to circle center | | Two axes of I, J, K | Position coordinates of circle center relative to start point |
| | Arc radius | | R | Arc radius |
| 5 | Feed speed | | F | angential speed of arc |

The CW and CCW are viewing the XY plane (ZX plane or YZ plane) from the positive direction of the Z-axis (Y-axis or X-axis) to its negative direction in the right-hand Cartesian coordinate system, as shown in Figure 4-2-3-1.

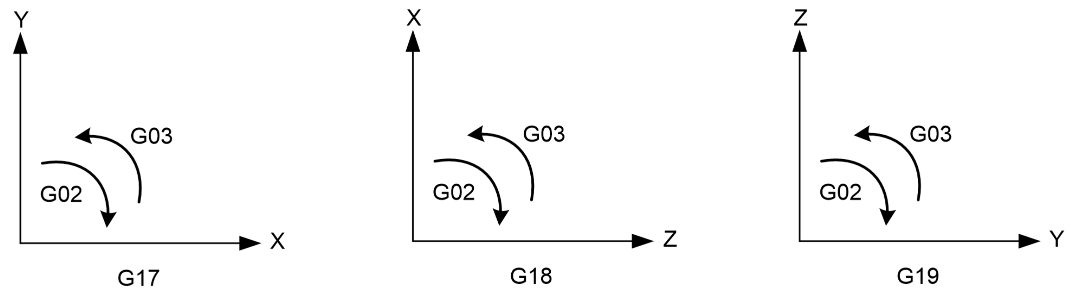



Fig. **4-2-3-1**

By setting the position parameters N0:31#1 and #2, the default plane modal information at power-up can be specified.

The end point of the arc is specified by the parameter word X, Y or Z. The end point is expressed in absolute value when corresponding to the G90 command, and in incremental value when corresponding to G91. The incremental value is the coordinate of the end point relative to the start point. The center of the arc is specified by the parameter words I, J, and K, which correspond to X, Y, and Z, respectively. The I, J, and K parameter values are the coordinates of the circle center relative to the start point of the arc, whether they are in the absolute mode G90 or the relative mode G91 (simply understood as the coordinates of the circle center with the start point being the coordinate origin temporarily), and are incremental values containing symbols, as shown in Figure 4-2-3-2.

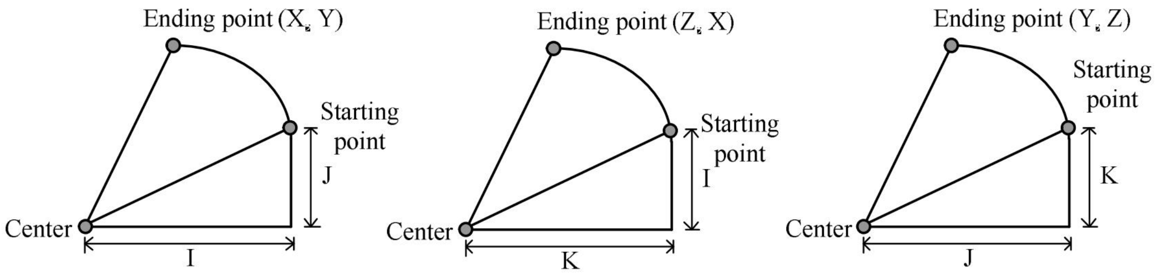



Fig.**4-2-3-2**

I, J, K contain symbols according to the direction of the circle center relative to the start point. In addition to I, J, and K, the center of the arc can also be specified by radius R. This is shown as follows:

G02 X_ Y_ R_ ；

G03 X_ Y_ R_ ；

1. Now the following two circular arcs can be drawn, one larger than 180° and the other smaller than 180°. For an arc greater than 180°, the radius is specified by a negative value. (For example, Figure 4-2-3-3) When the arc of ①is smaller than 180°

G91 G02 X60 Y20 R50 F300 ；

When the arc of ②is greater than 180°

G91 G02 X60 Y20 R-50 F300 ；

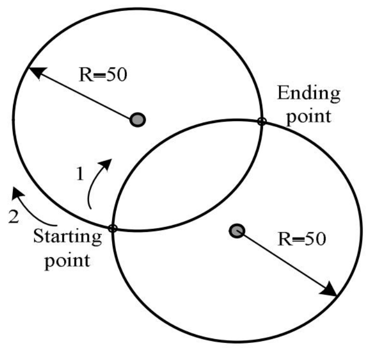



Fig.**4-2-3-3**

2.For an arc equal to 180°, I, J, and K as well as R can be used for programming:

Example: G90 G0 X0 Y0；G2 X20 I10 F100；

Equivalent to G90 G0 X0 Y0；G2 X20 R10 F100

Or G90 G0 X0 Y0；G2 X20 R-10 F100

**Note:**For an arc of 180°, positive and negative values of R do not affect the running path of the arc.

3.For an arc equal to 360°, only I, J, and K can be used for programming.

(Program example)：

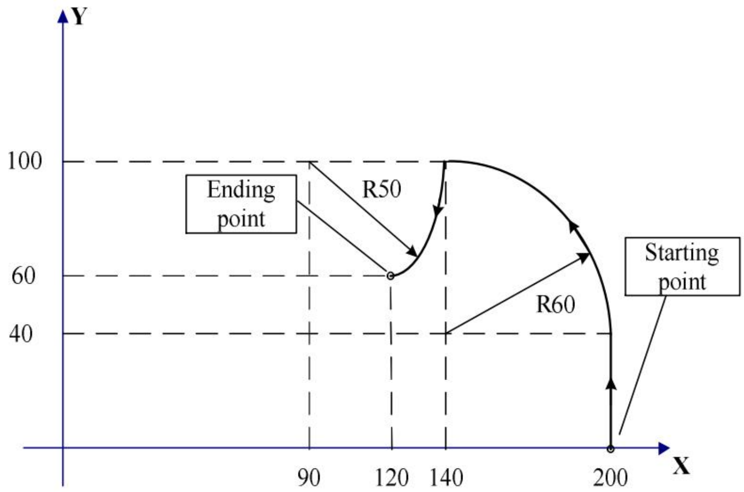



Fig.**4-2-3-4**

The tool path in Figure 4-2-3-4 is programmed as follows:

1)Absolute value programming

G90 G0 X200 Y40 Z0；

G3 X140 Y100 R60 F300；

G2 X120 Y60 R50；

Or

G0 X200 Y40 Z0；

G90 G3 X140 Y100 I-60 F300；

G2 X120 Y60 I-50；

2) Incremental value programming

G0 G90 X200 Y40 Z0；

G91 G3 X-60 Y60 R60 F3000；

G2 X-20 Y-40 R50；

Or

G0 G90 X200 Y40 Z0；

G91 G3 X-60 Y60 I-60 F300；

G2 X-20 Y-40 I-50；

**Restrictions:**

1.If the program specifies the addresses I, J, K, and R at the same time, the arc specified by the address R takes precedence, and the others are ignored.

2. If the arc radius parameter and the parameters from the start point to the arc center are not specified, the system will give an alarm.

3. For full circle interpolation, it is allowed to only specify the parameters I, J, and K from the start point to the arc center, rather than R.

4. Note the setting for selection of coordinate plane during circular interpolation.

5. If X, Y, and Z are all omitted, that is, the start and end positions are the same, and when R is specified (e.g.: G02R50;), the tool does not move.

## B. Helical interpolation

## Code format: G02/G03

Arc on the XY plane

$$G17\left\{\begin{matrix} G02 \\ G03 \end{matrix}\right\} X_p\_Y_p\_Z_p\_ \left\{\begin{matrix} I\_J\_ \\ R\_ \end{matrix}\right\} F\_$$

Arc on the ZX plane

$$G18\left\{\begin{matrix} G02 \\ G03 \end{matrix}\right\} X_p\_Y_p\_Z_p\_ \left\{\begin{matrix} I\_K\_ \\ R\_ \end{matrix}\right\} F\_$$

Arc on the YZ plane

$$G19\left\{\begin{matrix} G02 \\ G03 \end{matrix}\right\} X_p\_Y_p\_Z_p\_ \left\{\begin{matrix} J\_K\_ \\ R\_ \end{matrix}\right\} F\_$$

Fig.**4-2-3-5**

**Function:** Moving the tool to the specified position in a helical path from the current point at the feed speed specified by parameter F.

**Description:**

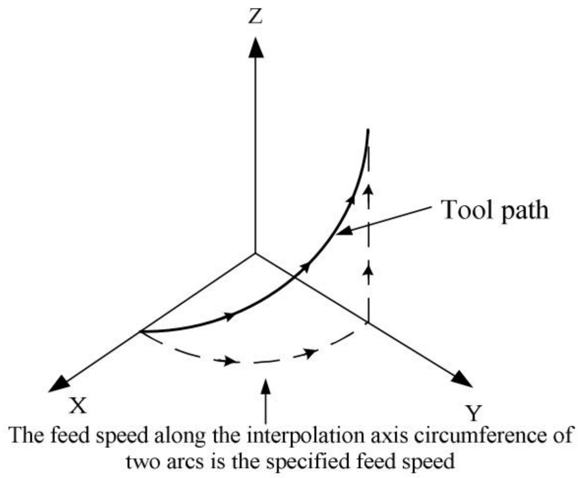



Fig.**4-2-3-6**

The first two numbers of the code parameters are the position parameters. The parameter word is the number of the two axes (X, Y or Z) in the current plane. Those two position parameters specify where the tool shall move in the current plane. The third parameter word of the code parameters is the linear axis other than the circular interpolation axis. Its parameter value is the helix height. The specific meanings and restrictions of other code parameters are the same as circular interpolation.

If the system cannot machine a circle based on a given code parameter, the system returns an error message. After execution, the system changes the mode of the current tool movement to G02/G03 mode.

The feed speed along the circumference of the two circular interpolation axes is specified. The instruction method simply adds a movement axis other than a circular interpolation axis. The F code specifies the feed speed along the arc. Therefore, the feed speed of the linear axis is as follows:

$$F_C = F \times \frac{\text{Linear axis length}}{\text{Arc length}}$$

The feed speed is determined in the way that the feed speed of the linear axis does not exceed any limit.

**Restrictions:**

Note the setting for selection of coordinate plane during helical interpolation.

## 4.2.4 **Absolute Value/Incremental Value Programming G90/G91**

**Format:G90/G91**
**Function:** There are two methods of coding axis movement amount: absolute value coding and incremental value coding.
The absolute value coding is a method of programming with the coordinates of the end position of the axis movement. As the end position involves concept of coordinate system, please refer to 2.4.1 to 2.4.4.
The incremental value coding is a method of directly programming with the axis relative movement amount. The increment value has nothing to do with the coordinate system. Just
give the movement direction and distance of the end position relative to the start position.
The absolute value code and the incremental value code are G90 and G91, respectively.

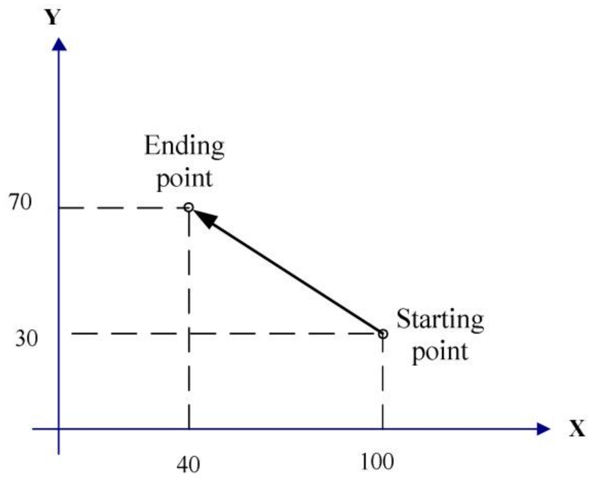



Fig.**4-2-4-1**

The movement from the start point to the end point in Figure 4-2-4-1 is programmed with the absolute value code G90 and the incremental value code G91 as follows:
G90 G0 X40 Y70;
Or G91 G0 X-60 Y40;
The same action can be done in both ways, and the operator can use them as needed.

**Description:**
- No code parameters. It can be written with other codes in the program segments.
- G90 and G91 are the modal values in the same group. When G90 is specified, G90 mode (default mode) applies before G91 is specified. For G91, it remains valid before G90 mode is specified.

**System parameters:**
Setting position parameter **N0:31#4** can specify whether the default position parameter at power-up is G90 mode (when parameter is 0) or G91 mode (when parameter is 1).

## 4.2.5 Pause (G04)

**Format: G04 X_ or P_**
**X, P: specified time.**
**Function:** G04 performs a pause action and executes the next program segment at the specified time delay. In addition, for the cutting method, in the G64 mode a pause can be specified

for an accurate stop check. Setting the position parameter No. 34#0 can specify a pause for the feed mode G95 per revolution.

Table 4-2-5-1 Range of command values for pause time (with X command)

| | | | |
|---|---|---|---|
| G04 | X<br>P | 0～9999.999<br>0～99999.9999 | X counter -s<br>P counter -ms |

**explain:**

1.G04 is a non-modal code and is valid only on the current line.
2.When the X and P parameters appear simultaneously, the X value is valid.
3. When the X and P values are set to negative values, an alarm will be given.
4. When neither X nor P is specified, the system does not perform a pause.

### 4.2.6 Unidirectional positioning (G60) (this function system is temporarily not open)

**Format: G60 X_ Y_ Z_**

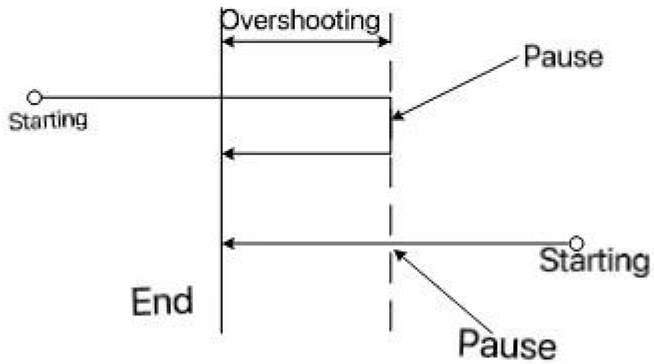



Fig.**4-2-6-1**

**Function:** When precise positioning is required to eliminate machine tool backlash, G60 can be used for accurate positioning from one direction.

**explain:**

G60 is a non modal G code that is only valid in specified program segments.

Parameters X, Y, and Z represent the coordinate values of the endpoint in absolute value programming, and the distance traveled by the tool in incremental value programming. When using one-way positioning under tool bias, the trajectory of one-way positioning is the trajectory after tool compensation.

In the above figure, the marked overshoot can be set according to the system parameters, the pause time can be set according to the parameters, and the positioning direction can be determined by the positive or negative value of the set overshoot. Please refer to the system parameters for specific details.

**Example 1:**

G90 G00 X-10 Y10；

G60 X20 Y25； (1)

If the corresponding system parameters are modified, for statement (1), the tool trajectory is AB → pause for 1 second → BC

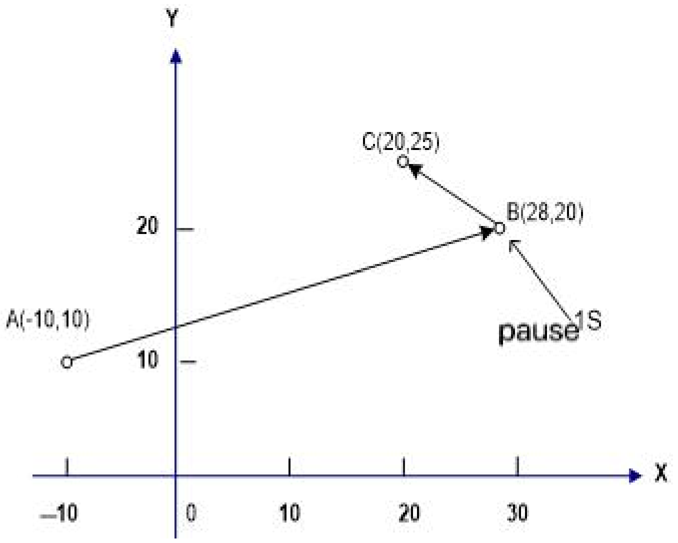



Fig.**4-2-6-2**

**Note 1:** The symbol of the parameter represents the direction of unidirectional positioning, and the value of the parameter represents the overtravel.
**Note 2:** If the overtravel is greater than 0, the positioning direction is positive.
**Note 3:** If the overtravel is less than 0, the positioning direction is negative.
**Note 4:** If the overtravel is 0, one-way positioning will not be performed.

## 4.2.7 **Online Change of System Parameters (G10)**

**Function:** This function is used to set or modify the tool radius, length offset, external zero offset, workpiece zero offset, additional workpiece zero offset, data parameters, position parameter, etc. in the programs.

**Format:**

G10 L50 N_P _R_; Set or modify a position parameter
G10 L51 N_ R_; Set or modify a data parameter
G11; Cancel parameter input mode

**Parameter definition:**

N: Parameter No. Parameter serial number to be modified.
P: Parameter bit number. Parameter bit number to be modified.
R: Modified value. Used to specify the modified value of a parameter.
The specified values can also be modified with the following code. Refer to the relevant sections for details:
G10 L2 P_X_Y_Z_A_B_; Set or modify the external zero offset orworkpiece zero offset
G10 L10 P_R_; Set or modify the length offset
G10 L11 P_R_; Set or modify the length wear value
G10 L12 P_R_; Set or modify the radius offset
G10 L13 P_R_; Set or modify the radius wear value
G10 L20 P_ X_Y_Z_A_B_; Set or modify the additional workpiece zero offset

**Note 1**: In the parameter input mode, other NC statements cannot be specified except for annotative statements.
**Note 2**: The G10 program segment must be instructed separately, or an alarm will occur. After using G10, remember to use G11 to cancel the parameter input mode, so as not to affect the normal use of the program.
**Note 3**: The parameter value modified by G10 must be within the range of the system parameters, or an alarm will occur.
**Note 4**: The modal code of the fixed cycle must be canceled before running G10, or the system will alarm.
**Note 5**: Parameters that are valid only after power-off and restart shall not be modified by G10.
**Note 6**: G20 and G21 cannot be modified online by G10.

**Note 7**: The G10 command is used to online modify the external zero offset, the workpiece zero offset, the additional workpiece zero offset or the tool offset. When modifying in the G91 mode, the system adds the command offset with the current offset; when modifying in the G90 mode, it is made by the command offset.

**Note 8**: The G10 mode is canceled when M00, M01, M02, M30, M99, M98, and M06 are executed.

## 4.2.8 **Workpiece Coordinate Systems G54~G59**

**Function:** Specifying the current workpiece coordinate system and selecting the workpiece coordinate system by specifying the workpiece coordinate system G code in the program.

**Format: G54~G59**

**Description:**

1. No code parameters.
2. The system itself can set six workpiece coordinate systems, and any one of the coordinates can be selected through codes G54~G59.
   G54 ----------------Workpiece coordinate system 1
   G55 ----------------Workpiece coordinate system 2
   G56 ----------------Workpiece coordinate system 3
   G57 ----------------Workpiece coordinate system 4
   G58 ----------------Workpiece coordinate system 5
   G59 ----------------Workpiece coordinate system 6
3. At power-up, the system displays the workpiece coordinate systems G54~G59, G92 or the additional workpiece coordinate systems that were executed before power-off.
4. When different workpiece coordinate systems are called in the program segments, an axis moved as instructed will be positioned to the coordinate point in the new workpiece coordinate system; ...for an axis not moved as instructed, its coordinates will skip to the corresponding coordinates in the new workpiece coordinate system, without any change to the actual machine position.

Example: The machine coordinate corresponding to the G54 coordinate system origin is (10, 10, 10) The machine coordinate corresponding to the G55 coordinate system origin is (30, 30, 30) When the programs are executed sequentially, the absolute coordinates of the end point and the machine coordinates are displayed as follows:

Table **4-2-8-1**

| Program | Absolute coordinate | Machine coordinate |
|---|---|---|
| G0 G54 X50 Y50 Z50 | 50，50，50 | 60，60，60 |
| G55 X100 Y100 | 100，100，30 | 130，130，60 |
| X120 Z80 | 120，100，80 | 150，130，110 |

5. G10 can be used to change the external workpiece zero offset value or the workpiece zero offset value. Methods as below:
   Using code G10 L2 Pp X_Y_Z_
   P=0: External workpiece zero offset value (base offset).
   P=1 to 6: Workpiece zero offset of workpiece coordinate systems 1 to 6.
   X_Y_Z_: For the absolute value code (G90), the workpiece zero offset value of each axis.
   For incremental value code (G91), the offset of each axis added to the set workpiece zero (the added result as the new workpiece zero offset).

Using the G10 command, each workpiece coordinate system can be changed separately.

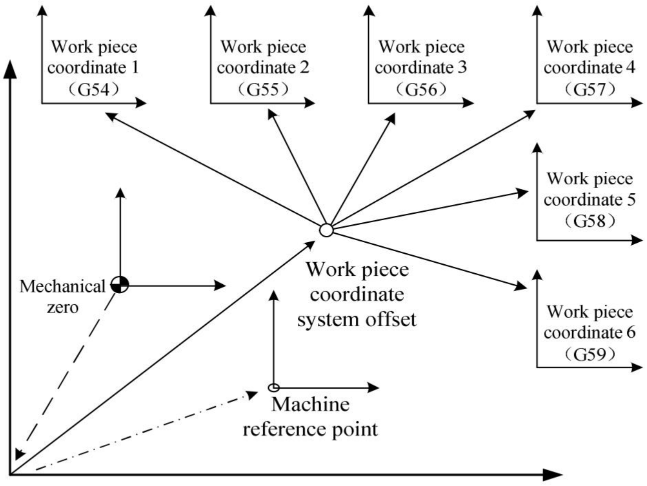



Fig.**4-2-8-1**

As shown in Figure 4-2-8-1 above, after the machine is started, it is manually returned to the mechanical zero from which the machine coordinate system is established, thereby generating the machine reference point and determining the workpiece coordinate system. The workpiece coordinate system offset data parameters **P10~P13** correspond to the overall offsets of the six workpiece coordinate systems. The origins of the six workpiece coordinate systems can be specified by inputting the coordinate offset in the input mode or by setting the data parameters **P15~P43**. The six workpiece coordinate systems are set according to the distance from the mechanical zero to the respective coordinate system zero.

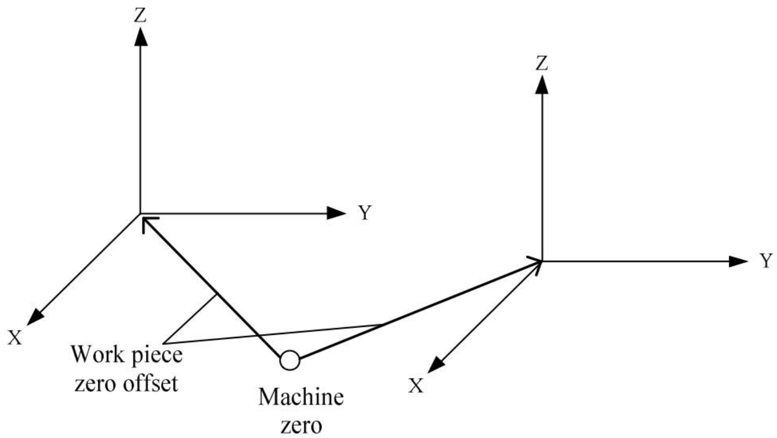



Fig.**4-2-8-2**

**Example:**

N10 G55 G90 G00 X100 Y20;
N20 G56 X80.5 Z25.5;

In the above example, when the execution of N10 program segment begins, fast movement is made to the position of the workpiece coordinate system G55 (X=100, Y=20). When the execution of N20 program segment begins, fast movement is made to the position of the workpiece coordinate system G56. The absolute coordinate value automatically becomes the coordinate value in the G56 workpiece coordinate system (X=80.5, Z=25.5).

## 4.2.9 **Additional Workpiece Coordinate Systems**

In addition to the six workpiece coordinate systems (G54 to G59 coordinate systems), 50 additional workpiece coordinate systems can be used.

**Format:** G54 Pn

Pn: Code specifying additional workpiece coordinate systems. The range of Pn is 1~50.

The setting and restriction of the additional workpiece coordinate systems are consistent with the workpiece coordinate systems G54~G59.
The workpiece zero offset values can be set with G10 in the additional workpiece coordinate systems. Methods as below:
Code: G10 L20 Pn X_Y_Z_;
n=1 to 50: Codes of additional workpiece coordinate systems.
X_Y_Z_ : Axis address and offset value setting workpiece zero offset.
For the absolute value code (G90), the specified value is the new offset value.
For the incremental value code (G91), the specified value is added to the current offset value to obtain the new offset.
Using the G10 command, each workpiece coordinate system can be changed separately.

## 4.2.10 **Format: G53 X_ Y_ Z_**

**Function:** Fast positioning the tool to the corresponding coordinate in the machine coordinate system.
**Description:**

1. When G53 is used in the program, the following code coordinates will be the coordinate values in the machine coordinates, and the machine will fast move to the specified position.
2. G53 is a non-modal code and is valid only in the current segment. It does not affect the previously defined coordinate system.

**Restrictions:**

Selection of machine coordinate system G53
When a position on the machine coordinate system is instructed, the tool moves fast to that position. The G53 used to select the machine coordinate system is a non-modal G code; that is,
it is valid only in the program segment where the machine coordinate system is instructed. The absolute value G90 shall be specified for G53, and the G53 command is ignored when the incremental value (G91) is specified. When the tool is instructed to move to a special position on the machine, for example, the tool change position can be set at that point by using the movement program written with G53.

**Note:** When G53 is specified, tool radius compensation and tool length offset are temporarily canceled and will be restored in the next compensated axis program segment that is cached.

## 4.2.11 Floating Coordinate System G92

**Format: G92 X_ Y_ Z_**
**Function:** Setting the floating workpiece coordinate system. The three code parameters specify the absolute coordinate value of the current tool in the new floating workpiece coordinate system. This code does not result in movement of the motion axis.

**Description:**

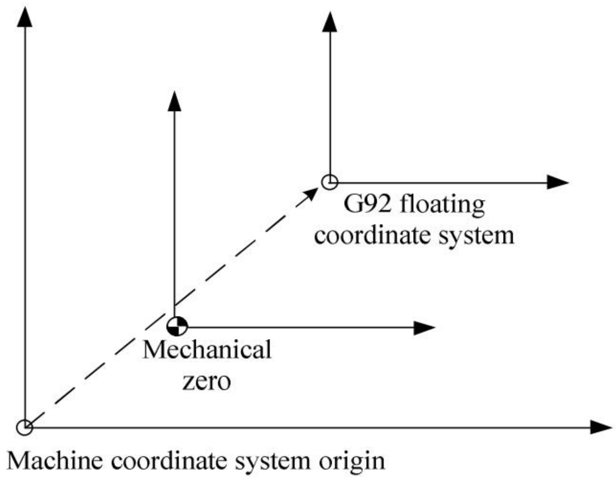



Fig.**4-2-11-1**

1. As shown in Figure 4-2-11-1, the origin of the G92 floating coordinate system is the value in the machine coordinate system, and has no relation with the workpiece coordinate system. For the validity of G92 after setting, it is valid:
   1) Before calling the workpiece coordinate system
   2) Before the machine zeroing
   The G92 floating coordinate system is usually used for alignment during temporary workpiece machining. It usually runs at the beginning of the program or as instructed in MDI mode before automatically running the program.
2. There are two ways to determine the floating coordinate system:
   1) Determine the coordinate system by tool tip

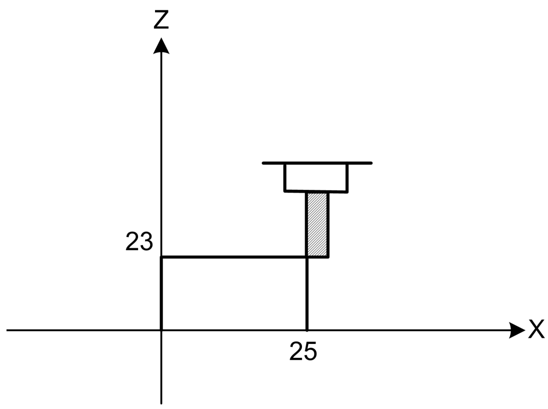



Fig **4-2-11-2**

As shown in Figure 4-2-11-2, G92 X25 Z23, the position of the tool tip is used as the point (X25, Z23) in the floating coordinate system.

2）Use a fixed point on the tool holder as the reference point coordinate system:

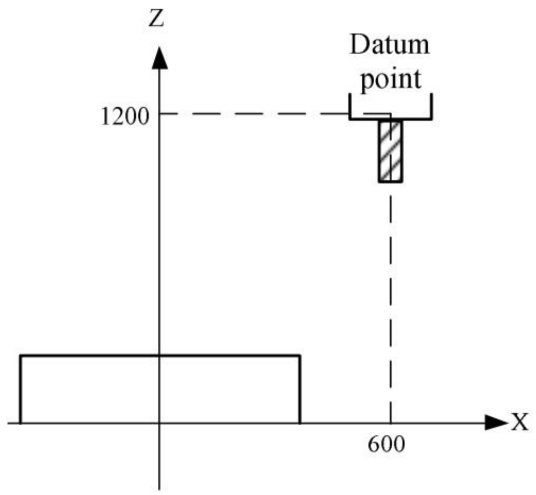



Fig **4-2-11-3**

As shown in Figure 4-2-11-3, G92 X600 Z1200 is used to instruct the setting of coordinate system (when a reference point on the tool holder is used as cutting point). Using a reference point on the tool holder as the start point, if movement is made by the absolute value code in the program, the reference point is moved to the instructed position, adding the tool length compensation, which is the difference from the reference point to the tool tip.

**Note 1**: If the coordinate system is set by G92 in the tool offset, it is the coordinate system set by G92 before the tool offset is added to the tool length compensation.

**Note 2**: Tool radius compensation must be canceled before using the G92 code.

## 4.2.12 Plane Selection G17/G18/G19

**Format: G17/G18/G19**

**Function:** For circular interpolation, tool radius compensation or drilling, boring, plane selection is required. Now the plane is selected through G17/G18/G19.

**Description:** If without command parameters, the system defaults to the G17 plane at power-up. The default plane after power-up can also be determined by setting the position parameters **N0:31#1, #2, #3**. Correspondence between code and plane:

G17-------------XY plane
G18-------------ZX plane
G19-------------YZ plane

If G17, G18, and G19 are in the program segments that are not instructed, the plane does not change.

**Example:** G18 X_ Z_; ZX plane

G0 X_ Y_; plane remains unchanged (ZX plane)

In addition, the movement code is not related to plane selection. In the case of the following code, the Y-axis does not exist on the ZX plane, so the Y-axis movement is not related to the ZX plane.

G18Y_;

**Hint:** At present, only the fixed cycle under the G17 plane is available. When programming, to be normal and strict, it is better to specify the plane in the corresponding program segment, especially when several people share the same system. This will avoid accidents or exceptions caused by programming errors.

## 4.2.13 **Polar Coordinate Beginning/Cancellation G16/G15**

**Format: G16/G15**

**Function:**

G16 specifies the beginning of the polar coordinate representation of the position parameters.
G15 specifies the cancellation of the polar coordinate representation of the position parameters.

**Description:**

No command parameters.

Setting G16 enables input of coordinate values with polar coordinate radius and angle. The positive direction of the angle is the counterclockwise turn of the first axis on the selected plane, and the negative direction is the clockwise turn. Both the radius and the angle can be in absolute value code or incremental value code (G90, G91).

After G16 appears, of the position parameter of the tool movement command the first axis represents the polar radius in the polar coordinate system, and the second axis represents the polar angle in the polar coordinate system.

By setting G15, the polar coordinate mode can be canceled to return the coordinate value to input with Cartesian coordinates.

**Requirement for polar coordinate origin:**

1. In the G90 absolute mode, when the G16 mode command is used, the workpiece coordinate system zero is taken as the polar coordinate origin.

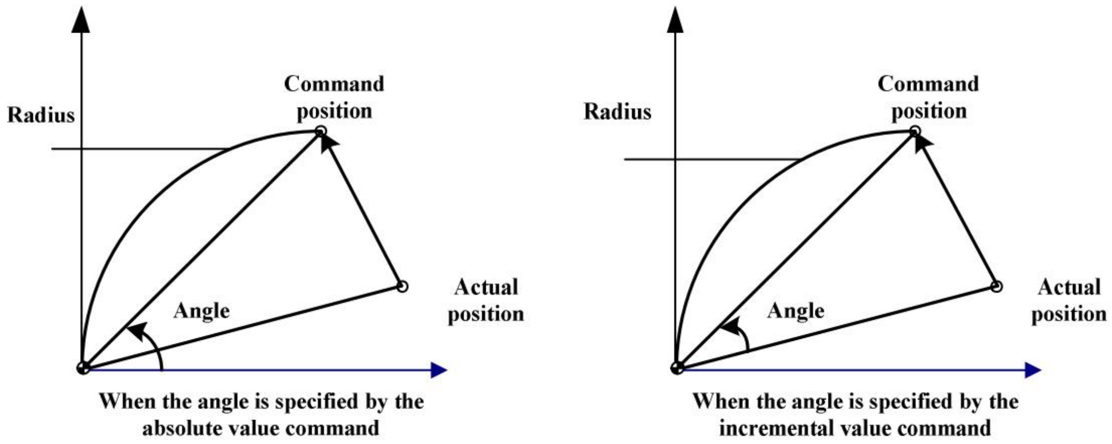



Fig **4-2-13-1**

2. In the G91 incremental mode, when the G16 mode command is used, the current point is taken as the polar coordinate origin.

**For example:** Bolt hole circle (the zero of the workpiece coordinate system is set as the polar coordinate origin, the X-Y plane is selected)

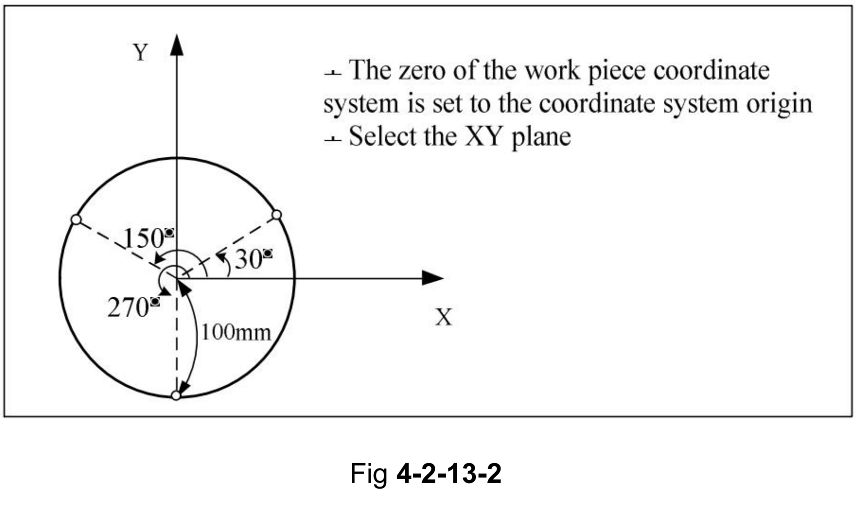



Fig **4-2-13-2**

- Specifying angle and radius by absolute values
  G17 G90 G16; specify the polar coordinate code and select the XY plane, set the zero of the workpiece coordinate system as the origin of the polar coordinate system
  G81 X100 Y30 Z-20 R -5 F200; specify a distance of 100mm and an angle of 30°
  Y150; specify a distance of 100mm and an angle of 150°
  Y270; specify a distance of 100mm and an angle of 270°
  G15 G80; cancel polar coordinate code
- Instructing angle by incremental value and polar radius by absolute value
  G17 G90 G16; specify the polar coordinate code and select the XY plane, set the zero of the workpiece coordinate system as the origin of the polar coordinate system
  G81 X100 Y30 Z-20 R -5 F200; specify a distance of 100mm and an angle of 30°
  G91 Y120; specify a distance of 100mm and an incremental angle of +120°
  Y120; specify a distance of 100mm and an incremental angle of +120°
  G15 G80; cancel polar coordinate code

In addition, when programming with polar coordinates, attention shall be paid to the setting of the current coordinate plane. The polar coordinate plane is related to the current coordinate plane. For example, in G91, if the current coordinate plane is G17, the X and Y-axis component of the current tool position is taken as the origin. If the current coordinate plane is G18, the Z and X-axis component of the current tool position is taken as the origin.
If the position parameter of the first hole cycle command is not specified after G16, the system uses the current position of the tool as the default position parameter of the hole cycle. The first fixed cycle code must be complete after the current polar coordinates. Otherwise, the tool movement is incorrect.
After G16, in addition to the hole cycle, the parameter word of the tool movement command position parameter is related to the specific plane selection mode. When the movement code follows immediately after canceling the polar coordinates with the G15 code, the current tool position is considered the start point of this movement code by default.

## 4.2.14 **Scaling In The Plane G51/G50**

**Format:**

**G51 X_ Y_ Z_ P_** (X.Y.Z: The absolute value code of the scaling center coordinate value, P: Each axis is scaled equally)
**…** Scaled machining program segment
G50 Scaling cancellation

**Or** **G51 X_ Y_Z_ I_ J_ K_** (the axes are scaled differently (I, J, K))
**…** Scaled machining program segment
**G50** Scaling cancellation

**Function:**

G51 enables the programmed shape to be centered at the specified position, zooming in and out at the same or different scales. It shall be noted that G51 is recommended to specify a separate program segment (otherwise unexpected conditions may occur, causing damage to the workpiece and personal injury) and cancellation made with G50.

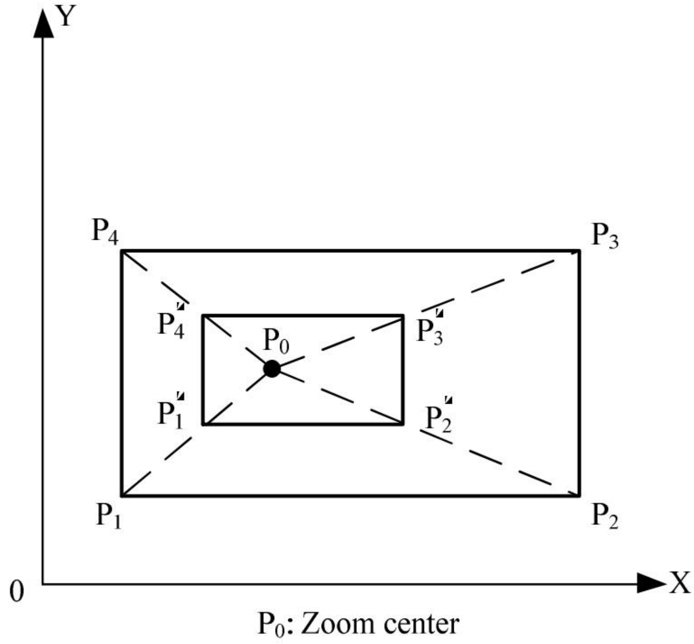



Figure 4-2-14-1 Scale (P1P2P3P4→ P1'P2'P3'P4')

**Description:**

1. Zoom center: G51 can carry 3 position parameters X_Y_Z_, which are optional. The position parameters are used to specify the zero of G51. If no position parameter is specified, the system sets the current position of the tool as the zero. Regardless of whether the current positioning mode is absolute or incremental, the zero is specified by absolute positioning. In addition, in the polar coordinate G16 mode, the parameters in the G51 code are also expressed in a Cartesian coordinate system.
   Example: G17 G91 G54 G0 X10 Y10;
   G51 X40 Y40 P2; incremental mode, the zero is still the absolute coordinate in the G54 coordinate system (40, 40)
   G1 Y90; parameter Y is still incremental
2. Scaling: Regardless of whether the current mode is G90 or G91, the scaling is always expressed in absolute terms.
   In addition to being specified in the programs, scaling can also be set in the parameters. The data parameter P330 sets the scaling ratio for each axis, and the data parameters **P331~P333** correspond to the scaling ratios of the first, second, and third axes, respectively. If there is no scaling ratio code, when the position parameter N0:47#6 is set to 0, scaling is performed by the set value of the data parameter P330; when the position parameter N0:47#6 is set to 1, scaling is performed by the set value of data parameters P331~P333.
   If the value of the parameter P or I, J, K is specified as negative, the corresponding axis is mirrored.
3. Scaling settings: Position parameter **No: 60#5** sets whether the scaling function is used, position parameter **N0:47#3** sets whether the first axis scaling is valid, position parameter **N0:47#4** sets whether the second axis scaling is valid, position parameter **N0:47#5** sets whether the third axis scaling is valid, and position parameter **N0:47#6** sets the way how the scaling ratio is specified for each axis (0: P command for each axis; 1: I, J, or K command for each axis).
4. Scaling cancellation: When the movement code follows immediately after canceling the scaling with the G50 code, the coordinate scaling is canceled, and the tool position is considered the start point of this movement code.
5. In scaling status, it is not allowed to instruct the G codes (G27~G30, etc.) for return to the reference point and G codes (G53~G59, G54P1~G54P50, G92, etc.) for coordinate systems. If those G codes must be specified, they shall be specified after canceling the scaling function, otherwise the system will alarm.
6. Even if different scaling ratios are specified for circular interpolation and each axis, the tool does not draw an elliptical path.

When the scaling ratio of each axis is different, and the circular interpolation is programmed with the radius R, the interpolation figure is shown in Figure 4-2-14-2 (in the following example, the ratio is 2 for the X axis and 1 for the Y axis).

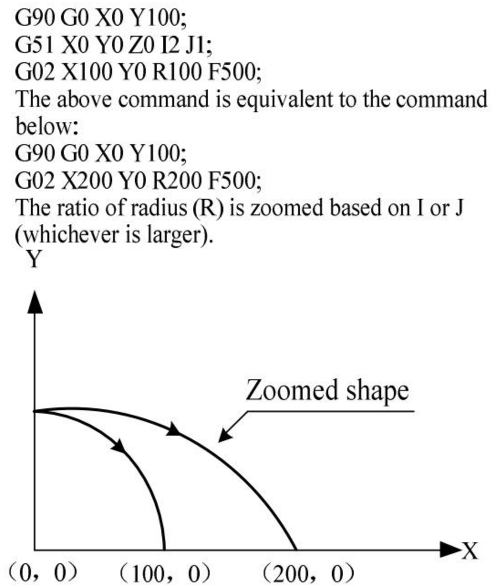



Figure 4-2-14-2 Scaling of Circular Interpolation 1

When the scaling ratio of each axis is different, and the circular interpolation is programmed with I, J, or K, the system will alarm if the arc is not established.

7. Scaling is invalid for tool offset values, as shown in Figure 4-2-14-3.

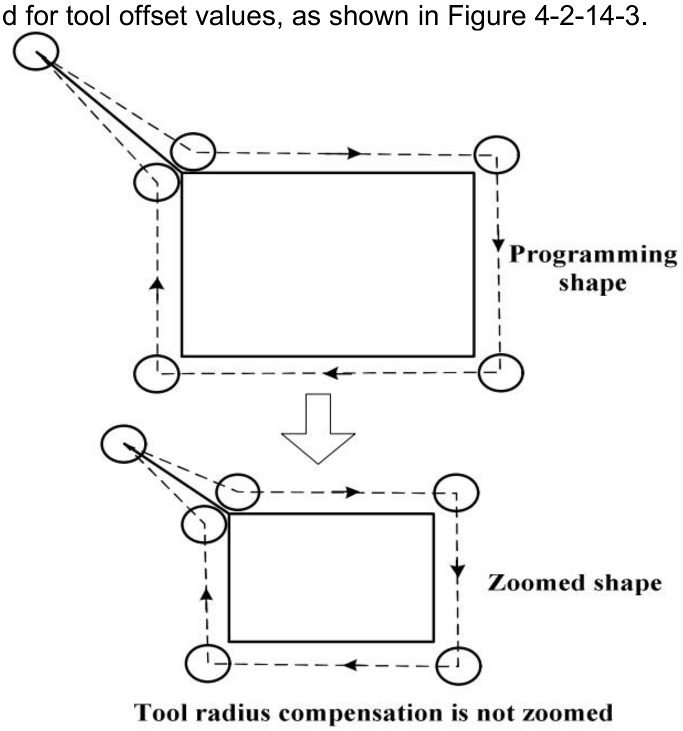



Figure 4-2-14-3 Scaling for tool radius compensation

**Examples of mirrored programs:**

**Main program**

G00 G90；

M98 P9000；

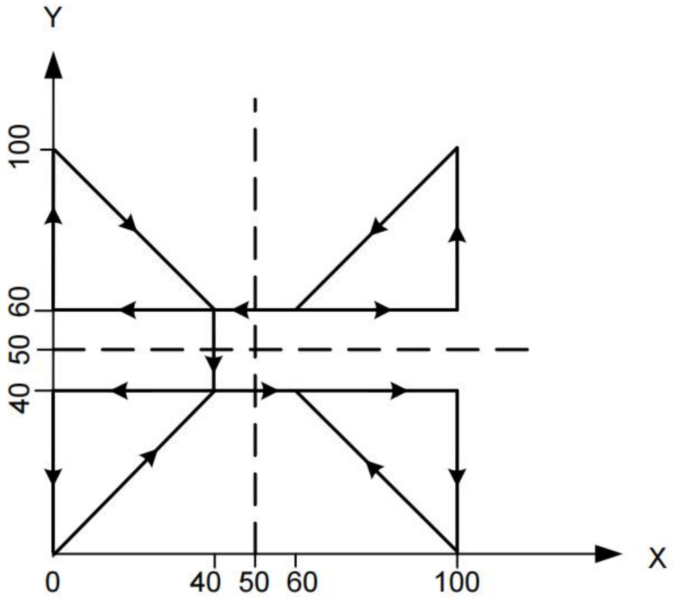
G51 X50.0 Y50.0 I-1 J1；
M98 P9000；

G51 X50.0 Y50.0 I-1 J-1；

```
M98 P9000；
G51 X50.0 Y50.0 I1 J-1；
M98 P9000；
G50；
M30；
```

**Subprogram:**

```
O9000；
G00 G90 X60.0 Y60.0；
G01 X100.0 F100；
G01 Y100；
G01 X60.0 Y60.0；
M99；
```




**Restrictions:**

1. In the fixed cycle, the movement and scaling are invalid for Z-axis cut-in value Q, Z depth, return value d, Z-axis first cutting depth W and fast entry distance V.
2. When running manually, the movement distance cannot be increased or decreased with the scaling function.

**Note 1**: The position shows the scaled coordinate value.
**Note 2**: When there is an axis performing mirroring on the specified plane, the result is as follows:

1) Arc code............ ....... ....... ....... .... Reverse direction of rotation
2) Tool radius compensation C.......... Reverse offset direction
3) Coordinate system rotation.............Reverse rotation angle
4) Change the direction of cutting feed

## 4.2.15 **Coordinate System Rotation G68/G69**

When the machining workpiece consists of many patterns of the same shape, it can be programmed with the coordinate rotation function by simply writing subprograms for the pattern elements and calling the subprograms through the rotation function.

**Code format:** **G17 G68 X_ Y_ R_；**
**Or G18 G68 X_ Z_ R_；**
**Or G19 G68 Y_ Z_R_ ；**
**G69；**

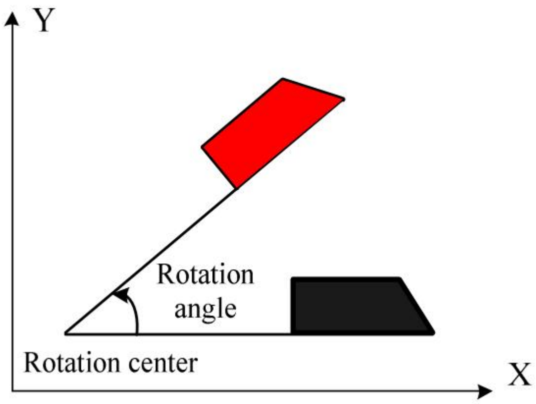



Figure **4-2-15-1**

**Function:** G68 enables the in-plane programmed shape to rotate around the specified center as the origin. G69 is used to cancel the rotation of the coordinate system.

**Description:**

1. G68 can carry 2 position parameters, which are optional. The position parameters are used to specify the center of the rotation. If no center of rotation is specified, the system uses the current tool position as the center of rotation. The position parameters are related to the current coordinate plane. X and Y are selected in G17; Z and X are selected in G18; Y and Z are selected in G19.
2. When the current positioning mode is absolute, the system uses the specified point as the center of rotation. When the positioning mode is relative, the system specifies the current point as the center of rotation. G68 can also carry a command parameter R, whose parameter value is the angle of rotation with positive value representing counterclockwise rotation. The angle of rotation is expressed in degrees. The angle of rotation used when there is no rotation angle code in the coordinate rotation is set by the data parameter P329.
3. In the G91 mode, the system uses the current tool position as the center of rotation; whether increment is performed on the angle of rotation is set by the position parameter **NO:47#0** (angle of coordinate rotation, 0: absolute code, 1: G90/G91 code).
4. When the system is in the rotation mode, the plane selection operation is not allowed, or an alarm will occur. Care shall be taken when programming.
5. In the coordinate system rotation mode, it is not allowed to instruct the G codes (G27~G30, etc.) for return to the reference point and G codes (G53~G59, G54P1~G54P50, G92, etc.) for coordinate systems. If those G codes must be specified, they shall be specified after canceling the rotation function, otherwise the system will alarm.
6. After the coordinate system rotation, tool radius compensation, tool length compensation, tool offset and other compensation operations can be performed.
7. When the coordinate system rotation code is executed in the scaling mode (G51), the coordinate value of the rotation center is also scaled, but the angle of rotation is not scaled. When the movement code is issued, the scaling is performed, followed by coordinate rotation.

**Example 1**: Rotation:

```
G92 X-50 Y-50 G69 G17;
G68 X-50Y-50 R60;
G90 G01 X0 Y0 F200;
G91 X100;
G02 Y100 R100;
G3 X-100 I-50 J-50;
G01 Y-100;
G69;
M30;
```

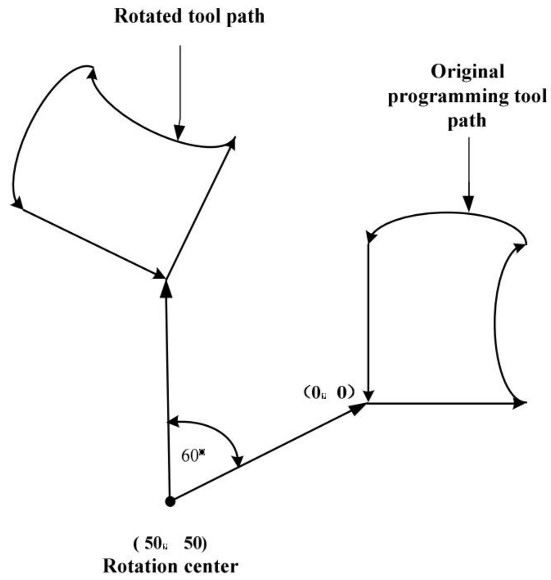



Figure **4-2-15-2**

**Example 2**: Scaling + rotation:

```
G51 X300 Y150 P0.5；
G68 X200 Y100 R45；
G01 G90 X400 Y100；
G91 Y100；
X-200；
Y-100；
X200；
G69 G50；
M30；
```

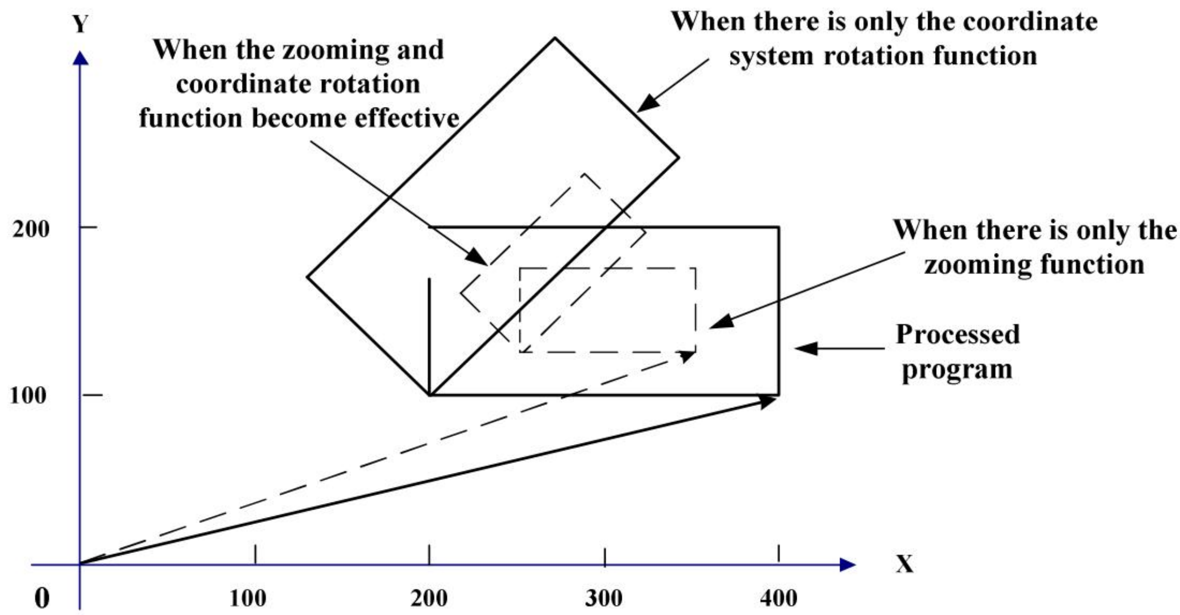



Figure **4-2-15-3**

**Example 3**: Repeating G68

Based on program (main program)

```
G92 X0 Y0 Z20 G69 G17；
M3 S1000；
G0 Z2；
G51 X0 Y0 I1.2 J1.2；
G42 D01；                  (Tool offset setting)
M98   P2100(P02100)；      (Subprogram call)
M98   P2200L7；            （Call 7 times）
G40；
G50；
G0 G90 Z20；
X0Y0；
M30；
Subprogram 2200
O2200
G91
G68 X0 Y0 R45.0；           (Relative rotation angle)
G90；
M98 P2100；                 (Subprogram O2200 calls subprogram O2100)
M99；
Subprogram 2100
O2100 G90 G0 X0 Y-20；  (Right tool compensation mode is established)
G01Z－2 F200；
X8.284；
X14.142 Y-14.142；
M99；
```

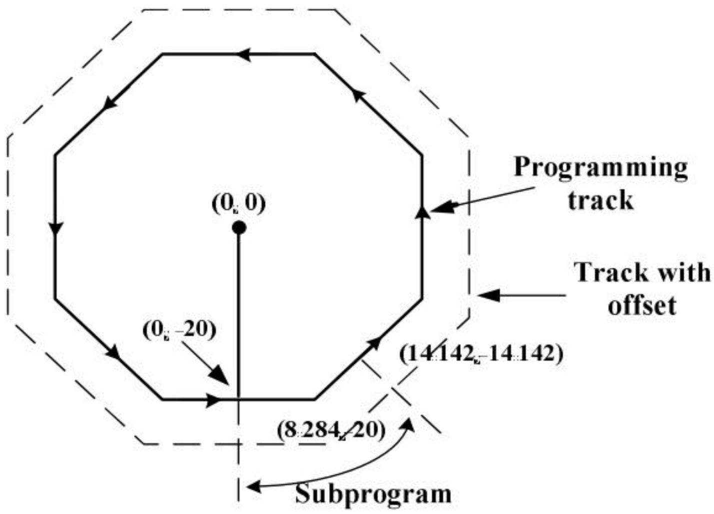



Figure **4-2-15-4**

# 4.2.16 **Skip Function G31**

**Format: G31 X_Y_Z_**

**Function:** After the G31 code, as with G01 linear interpolation can be instructed. If an external skip signal is input during the execution of the code, the execution of the code is interrupted for the execution of the next program segment. The skip function is used when the machining end point is not programmed, but is specified with a signal from the machine tool. For example, it is used for grinding. The skip function is also used to measure the workpiece size.

**Description:**

1. G31 is a non-modal G code and is valid only in the specified program segment.
2. When applying the tool radius compensation, if the G31 code is issued, the alarm will appear. The tool radius compensation shall be canceled before the G31 code.

**Example:**

The program segment following G31 is the single-axis movement instructed by the incremental value, as shown in Figure 4-2-16-1:

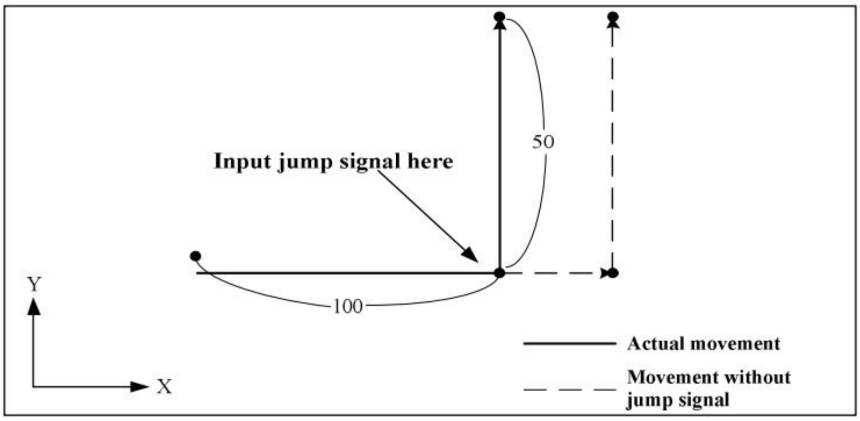



Figure 4-2-16-1 The program segment following is the single-axis movement instructed by the incremental value

The program segment following G31 is the single-axis movement instructed by the absolute value, as shown in Figure 4-2-16-2.

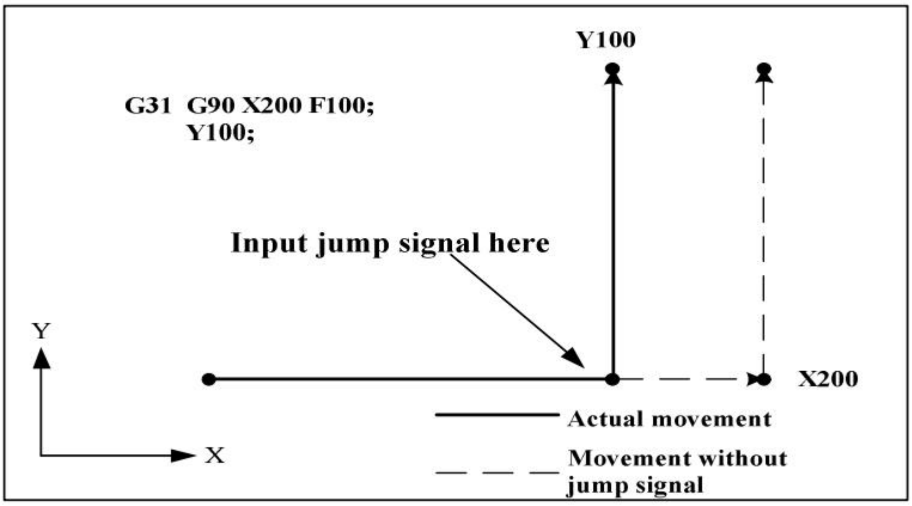



Figure 4-2-16-2 The program segment following is the single-axis movement instructed by the absolute value

The program segment following G31 is the two-axis movement instructed by the absolute value, as shown in Figure 4-2-16-3:

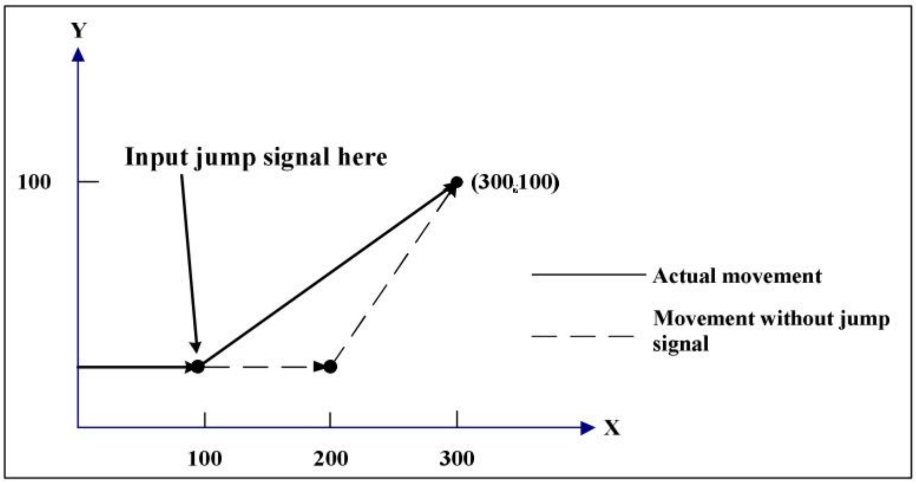



Figure 4-2-16-3 The program segment following is the two-axis movement instructed by the absolute value

**Note:** DNC and MDI single segment methods do not support program redirection.

## 4.2.17 **Imperial/Metric Conversion of G20/G21**

**Format: G20**: Imperial input
**G21**: Metric input
**Function:** It enables imperial/metric conversion of program input.
**Description:**
After the imperial/metric conversion, the units of the following values are changed:
F code instructed feed speed, position code, workpiece zero offset value, tool compensation value, scale unit of the manual pulse generator, and movement distance in incremental feed.
At power-on the G code is in the same state as what is before power-off.

**Note:** 1. When the imperial is converted to metric or the other way around, the tool compensation value must be preset based on the least input increment.
2. When the imperial is converted to the metric or the other way around, the operation of the first G28 code from the middle point is the same as the manual return to reference point.
3. When the least input increment and the least command increment are different, the maximum error is half of the least command increment, and this error does not accumulate.
4. The program input imperial/metric can be set by position parameter **N0:00#2**.
5. G20 or G21 must be specified in a separate program segment.

## 4.2.18 **Any Angle Chamfer/Corner Arc**

**Format: L_:** Chamfer
**R_:** Corner arc transition
**Function:** When the above code is added at the end of the linear interpolation (G01) or circular interpolation (G02, G03) program segment, the chamfer or transition arc is automatically added to the corner during machining. Program segments for chamfer or corner arc transition can be specified continuously.
**Description:**
1. Chamfer is behind L, specifying the distance from the virtual inflection point to the start and end points of the corner. The virtual inflection point is the actual corner point if it is assumed chamfer is not executed. Shown as follows:

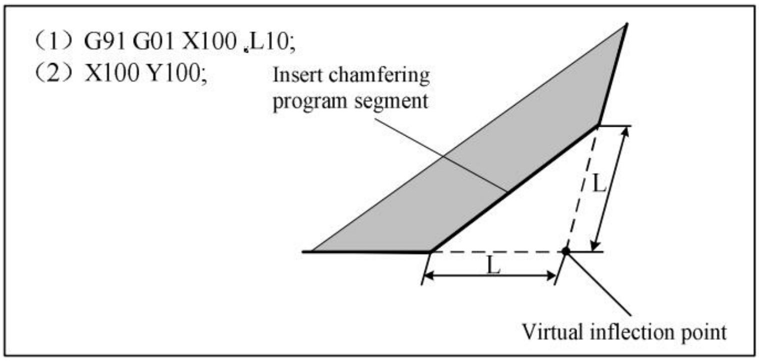
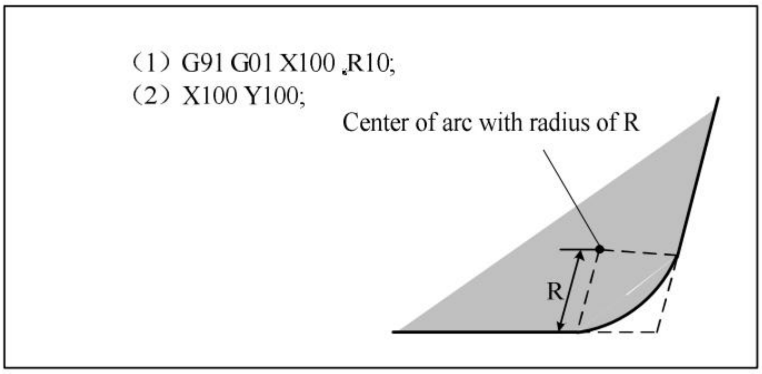



Fig **4-2-18-1**

2、Corner arc transition is behind R, specifying the radius of the corner arc, as shown below:




**Restrictions:**

1. Chamfer and corner arc can only be executed in the specified plane. These functions cannot be executed on parallel axes.
2. If the inserted chamfer or arc transition program segment causes the tool to exceed the original range of interpolation movement, an alarm will occur.
3. The corner arc transition cannot be specified in the thread machining program segments.
4. When the instructed chamfer value and corner value are negative, the system takes their absolute values.

## 4.3 **Reference Point G Code**

The reference point is a fixed point on the machine. With the reference point return function the tool can be easily moved to this position.

For the reference point, there are three code operation modes. As shown in Figure 4-3-1, G28 enables the tool to move through the middle point and automatically move to the reference point along the specified axis in the code. G29 enables the tool to move from the reference point, through the middle point, and automatically move to the specified point along the specified axis in the code.

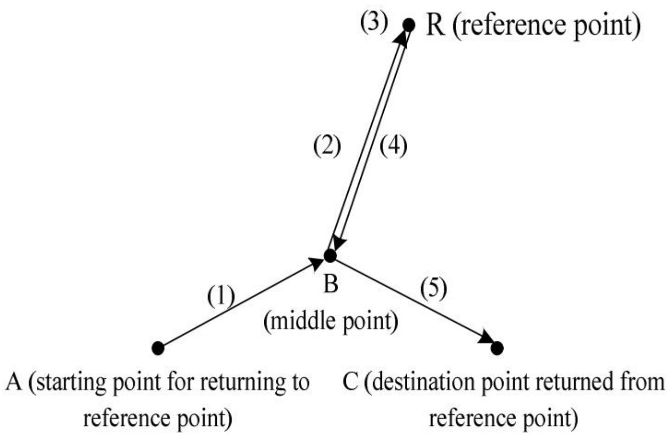



Fig **4-3-1**

## 4.3.1 **Return To Reference Point G28**

**Format: G28 X_ Y_ Z_**

**Function:** The G28 code is used to perform an operation that returns a reference point (a specific position on the machine) through a middle point.

**Description:**

**Middle point:**

The middle point is specified by the code parameter in G28 and can be represented by an absolute value code or an incremental value code. When this program segment is executed, the coordinate value of the middle point on the code axis is also stored for use by the G29 (return from reference point) code.

**Note:**

The coordinates of the middle points are stored in the **CNC**, but only the coordinate values on the axes instructed by G28 are stored at a time, and for other axes that are not instructed, the coordinate values on the axes previously instructed by G28 are used. Therefore, when the user
uses G28 command, if the default middle point in the current system is unknown, it is best to specify each axis. Please consider with the N5 program segment in Example 1 below.

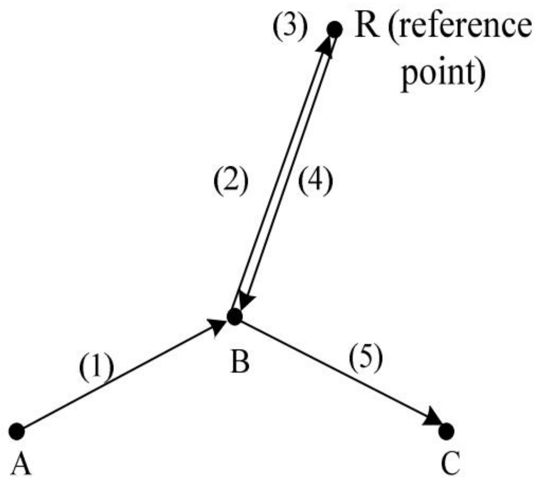



Fig **4-3-1-1**

1. The action of the G28 program segment can be decomposed into the following steps (see Figure **4-3-1-1**):

(1) Move fast from the current position to the position of the middle point on the instructed axis (point A → point B).

(2) Move fast from the middle point to the reference point (point B → point R).

2. G28 is a non-modal code and is valid only for the current segment.
3. Return to reference point is available in a combined way of single-axis or multi-axis movement. During the workpiece coordinate transformation, the coordinates of the middle point are stored in the CNC.

**Example:**

N1 G90 G54 X0 Y10;
N2 G28 X40; Set the middle point on the X axis to X40 in the G54 workpiece coordinate system, and return to the reference point through the point (40, 10), that is, return to reference point on the separate X axis.
N3 G29 X30; Return from the reference point to the point (30, 10) through the point (40, 10), that is, return to target point on the separate X axis.
N4 G01 X20;
N5 G28 Y60; Middle point Y60.
N6 G55; When the workpiece coordinate system is transformed, the middle point is changed from the point (40, 60) in the G54 workpiece coordinate system to the point (40, 60) in the G55 workpiece coordinate system.
N7 G29 X60 Y20; Return from the reference point to the point (60, 20) through the middle point (40, 60) in the G55 workpiece coordinate system.

G28 will automatically cancel the tool compensation. But this code is generally used during automatic tool change (that is, tool change at the reference point after returning to the reference point), so when using this code, in principle, the tool radius compensation and tool length compensation must be canceled first. 1st reference point setting
See data parameters P45~P49

### 4.3.2 Return To Reference Points 2, 3 and 4 (G30)

There are 4 reference points in the coordinate system, but for systems without absolute position detector, it is possible to return to Reference Points 2, 3 and 4 only when Automatic Return to Reference Point (G28) or Manual Return to Reference Point is executed.

**Format:**

**G30 P2 X_ Y_ Z_;** return to Reference Point 2 (P2 can be omitted)
**G30 P3 X_ Y_ Z_;** return to Reference Point 3
**G30 P4 X_ Y_ Z_;** return to Reference Point 4

**Function:** Execute G30 for return to the designated reference point through the intermediate point specified in G30.

**Description:**

1. X_ Y_ Z; specify the intermediate position (absolute value/incremental value code)
2. G30 shares the same setting and limitations with G28. Please see Data Parameters **P50 - P63** for setting of Reference Points 2, 3 and 4.
3. G30 can also be used together with G29 (return from a reference point) code, sharing the same setting and limitations with G28.

### 4.3.3 Automatic Return From A Reference Point (G29)

**Format: G29 X_ Y_ Z_**

**Function:** Execute G29 for return from a reference point (or the current point) through the intermediate point specified in G28 and G30.

**Description:**

1. The action in the G29 program segment can be decomposed into the following steps (see Figure 4-3-1-1):
(1) Locate to the intermediate point specified in G28 and G30 from a reference point (or the current point) in a fast moving manner (Point R → Point B).
(2) Locate to the designated point from the intermediate point in a fast moving manner (Point B → Point C).
2. G29 is non-modal information and only applicable to the current segment. In general, after definition of G28 and G30, you should immediately specify the code for return from a

reference point.

3. The optional parameters X, Y and Z in the G29 code format are used to specify the target point for return from a reference point (i.e. Point C in Figure 4-3-1-1), and can be expressed with absolute value code or incremental value code. Program the incremental value and use the code value to specify the incremental value leaving the intermediate point. The case that some axle is not specified means that there is no movement of the axle relative to the intermediate point. G29 only followed by one-axle command means one-axle return and the other axles remain inactive.

**Example:**

G90 G0 X10 Y10;
G91 G28 X20 Y20; return to a reference point via the intermediate point (30, 30)
G29 X30; return from a reference point to the point (60, 30) via the intermediate point (30, 30), and attention should be paid to the incremental programming mode and the component of the X axis should be 60.

The intermediate point specified by G29 is assigned by G28 and G30. Definition, specification and system defaults of an intermediate point are detailed in the description of G28.

## 4.3.4 Return To A Reference Point For Testing (G27)

**Format: G27 X_ Y_ Z_**
**Function:** Execute G27 for return to a reference point for testing and use X_ Y_ Z_ to specify the reference point.
**Description:**

1. G27 enables the tool to locate in a fast moving manner. If the tool reaches the reference point, a signal of successful return will be triggered; if the tool arrives at a position other than the reference point, an alarm will be triggered.
2. For a machine tool in locked state, even if the G27 code is specified and the tool has automatically returned to the reference point, the signal of successful return will not be triggered.
3. In the offset mode, the position where the tool reaches as specified by G27 is a result of addition of the offset value. Therefore, if the position plus the offset value is not the reference position, the signal will fail and the alarm will be triggered. Usually, the tool offset should be canceled before using G27.
4. The position defined by the X, Y, and Z coordinate points specified by G27 is the position of the machine tool in the coordinate system.

## 4.4 Fixed Cycle (G Code)

A fixed cycle uses a program segment containing G functions to enable processing activities which may have been realized by multiple program segments, so as to simplify the programming. And it also makes it easy for programmers to program (this system only has a fixed cycle of the G17 plane).

**General process of a fixed cycle:**

A fixed cycle usually consists of 6 activities in sequence, as shown in Figure 4-4-1.

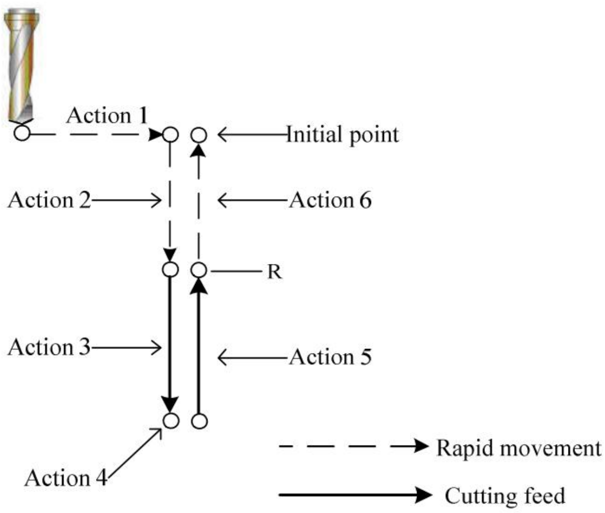



Fig **4-4-1**

Activity 1: positioning in the X and Y axis (it may also include another axis)
Activity 2: quickly move to Point R.
Activity 3: hole machining
Activity 4: operations at the bottom of the hole
Activity 5: return to Point R
Activity 6: quickly move to the initial point

Positioning is conducted in the XY plane, and hole machining is performed in the Z-axis direction. Specified activities in a fixed cycle depend on three ways. They are specified by G code.

1) Data form
   G90 absolute value; G91 incremental value
2) Return to the point plane
   G98 initial point plane; G99 R point plane
3) Hole machining method
   G73, G74, G76, G81 - G89

**Initial point Z plane and R point plane**

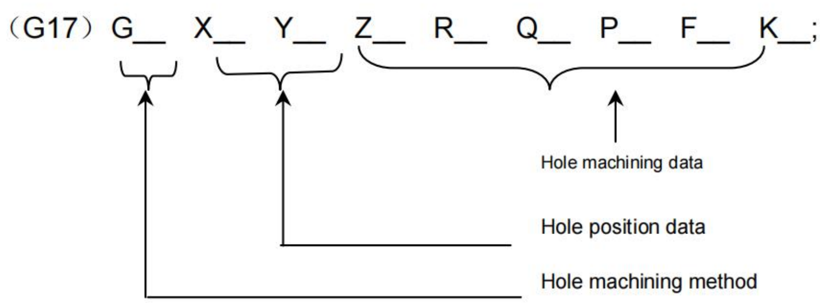
**Initial point plane:** It is the absolute position of the tool in the Z-axis direction before start of the fixed cycle

**R point plane:** Also known as safety plane, it is the position in the Z-axis direction when shifting from rapid movement to cutting feed in a fixed cycle, usually set at a certain distance above the surface of a workpiece to prevent the tool from hitting the workpiece and ensure sufficient distance to complete the acceleration process.

All data related to the fixed cycle (including data about hole position, hole machining and number of repetitions) are specified by G73/G74/G76/G81 - G89 to form a program segment.

**Z, R:** In case of lack of any of Parameters Z and R when drilling is conducted to form the first hole, the system will only change its mode and do not perform the Z-axis activities.

**The format for hole machining method is as follows:**




The basic meanings of hole position data and hole machining data are shown in Table 4-4-1.

Table **4-4-1**

| Specified content | Parameter | Description |
|---|---|---|
| Hole machining method | G | Please refer to Table **4-4-3** and pay attention to the above restrictions. |
| Hole position data | X，Y | Use absolute or incremental value to specify the hole position, and use the same control as G00 in terms of positioning. |
| Hole machining data | Z | As shown in Figure 4-4-2, use the incremental value to specify the distance from Point R to the hole bottom or use the absolute value to specify the coordinate values of the hole bottom. The feed rate is shown in Figure 4-4-1. Use the speed specified by F in Activity 3. Choose fast feeding in Activity 5 based on different hole machining methods or use the speed specified by F code. |
| | R | As shown in Figure 4-4-2, use the incremental value to specify the distance from the initial point plane to Point R, or use the absolute value to specify the coordinate values of Point R. The feed rate is shown in Figure 4-4-1. Fast feeding is used in Activities 2 and 6. |
| | Q | Specify G73, the depth per cutting in G83, or G76, translation distance in G87 (incremental value) |
| | P | Specify the pause time at the bottom of the hole. The fixed cycle code can have a parameter P_, which is used to specify the pause time when the tool has reached the Z plane. Unit: ms. The minimum value of the parameter is set by Data Parameter P281, while the maximum value is set by P282. |
| | F | Specify the cutting feed rate |
| | K | Specify the number of repetitions in the parameter value of K_, and K is valid only in the specified program segment. It can be omitted with the default of one repetition. The maximum frequency of drilling is 99999. When a negative value is specified, it is executed according to its absolute value. When it is zero, the drilling operation is not performed. Only change the mode |

**Restrictions:**

- The fixed cycle G is a modal code that remains valid until the G code that cancels the fixed cycle is specified.
- The G code that cancels the fixed cycle includes G80 and the G code in Group 01.
- Once the machining data is specified in the fixed cycle, it will remain valid until the fixed cycle is canceled. Therefore, all the necessary hole machining data should be specified at the beginning of the fixed cycle, and only the data to be changed will be specified in the subsequent fixed cycle.

**Note 1**: The cutting speed in the F command will be maintained even if the fixed cycle is canceled.

**Note 2**: When the cycle is fixed, the Z-axis (cutting axis) scaling function is invalid.

**Note 3:** In the single-segment mode, the fixed cycle generally adopts a three-stage machining mode: positioning → R plane → initial plane.

**Note 4:** In the fixed cycle, when the system bit parameter NO: 36#1 is 1, in case of reset or emergency stop, the hole machining data and hole position data will be eliminated. Examples of maintained and eliminated data above mentioned are shown below.

Table 4-4-2

| Sequence | Designation of data | Description |
|---|---|---|
| ① | G00X_M3； | |
| ② | G81X_Y_Z_R_F_； | The required values for Z, R and F should be assigned at the beginning. |
| ③ | Y_； | Due to sharing the same hole machining method and hole machining data specified in Hole ②,<br>G81 and Z-R-F- can be omitted. The hole moves by Y,<br>and is processed once with the G81 method |
| ④ | G82X_P_； | Only move in the X-axis direction relative to the position of Hole ③. Machine the hole with the G82 method, and use Z, R and F specified in ② and P specified in ④ as hole machining data to |

| | | |
|---|---|---|
| | | machine the hole. |
| ⑤ | G80X_ Y_ | No hole machining. Cancel all of the hole machining data. |
| ⑥ | G85X_Z_R_P_； | Since all the data is canceled in ⑤, Z and R need to be specified again.<br>F is the same as that specified in ②, so it can be omitted.<br>P is not needed in this program segment, and thus just save it |
| ⑦ | X_Z_； | Hole machining with the Z value different from that in ⑥.<br>And the hole only moves in the X-axis direction. |
| ⑧ | G89X_Y_； | Use Z specified in ⑦, R and P specified in ⑥ and F specified in ② as the hole machining data to machine the hole with the G89 method. |
| ⑨ | G01X_Y_； | Eliminate the hole machining method and relevant data. |

**A. Absolute value/incremental value codes of the fixed cycle (G90/G91)**

The change in the movement distance along the drilling axis relative to G90 and G91 is as shown in Figure 4-4-2 (usually programmed with G90. If programmed with G91, then Z and R are processed according to the plus-minus sign of the command).

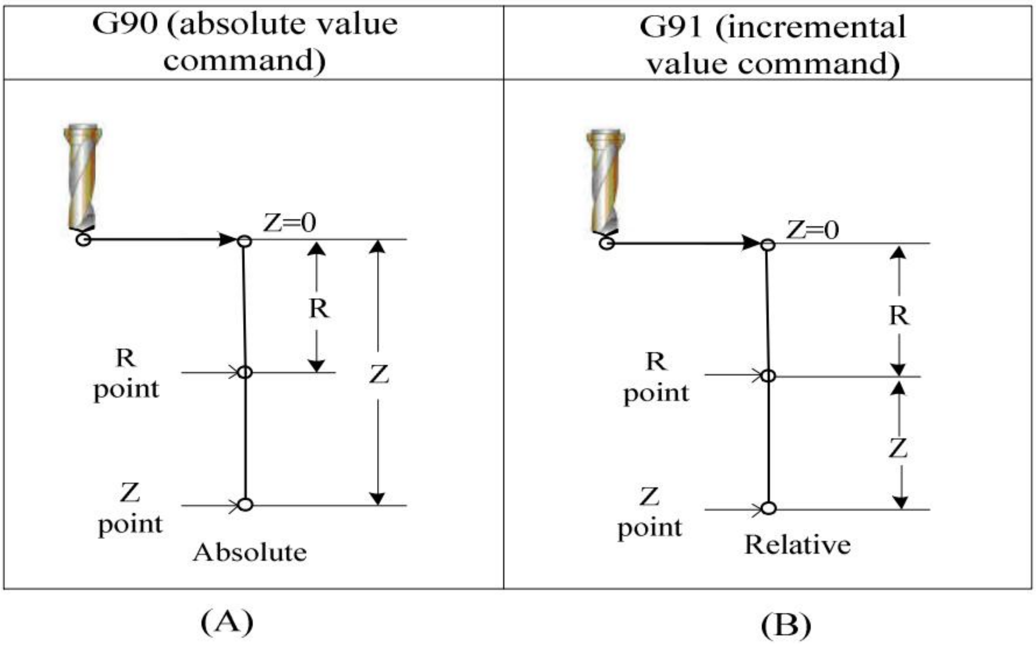



Fig **4-4-2**

**B. Code for the fixed cycle to return to the plane (G98/G99)**

When the tool reaches the bottom of the hole, it can return to the R point plane or the initial position plane. The tool can return to the initial point plane or the R point plane depending on G98 and G99.

In general, G99 is used for the first drilling and G98 is used for the last drilling. Even if the hole is machined using G99, the initial plane will remain unchanged. The activities included in codes G98 and G99 are shown in the figure below.

The system will choose G98 by default.

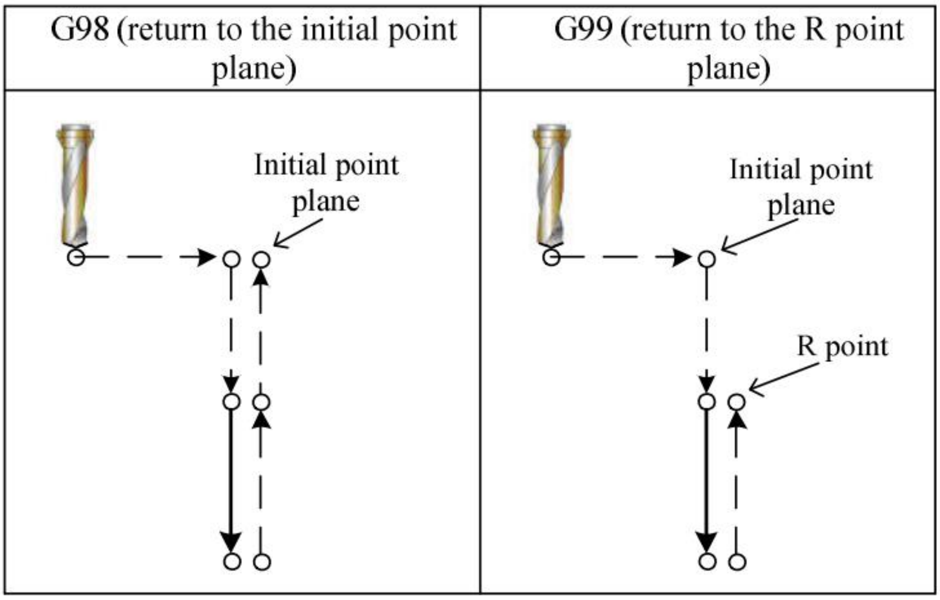



Fig **4-4-3**

Each fixed cycle diagram will use following symbols:

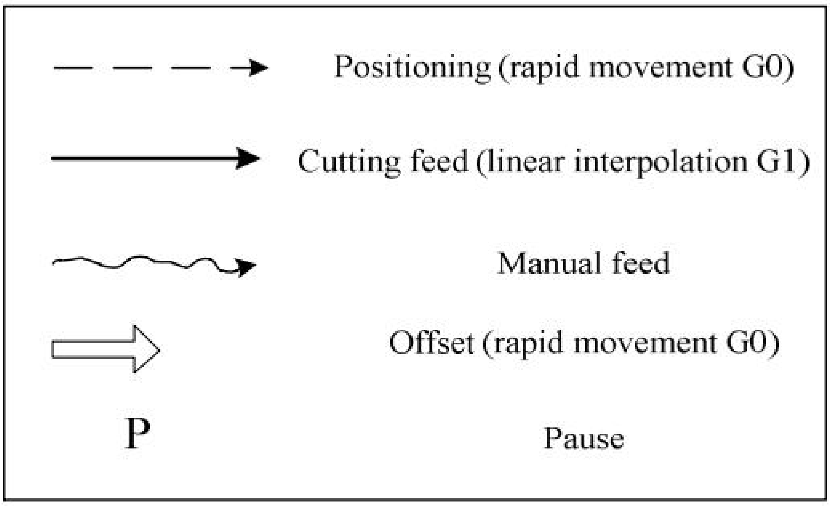



Fig **4-4-4**

### Fixed Cycle Comparison Table (G73 - G89)

Table **4-4-3**

| G code | Drilling (-Z direction) | Bottom activity | Retracting (+Z direction) | Purpose |
|---|---|---|---|---|
| G22 | Cutting feed | | Rapid movement | Groove rough milling inside circle (CCW) |
| G23 | Cutting feed | | Rapid movement | Groove rough milling inside circle (CW) |
| G24 | Cutting feed | | Rapid movement | Finish milling cycle inside full circle (CCW) |
| G25 | Cutting feed | | Rapid movement | Finish milling cycle inside full circle (CW) |
| G26 | Cutting feed | | Rapid movement | Finish milling cycle outside circle (CCW) |
| G32 | Cutting feed | | Rapid movement | Finish milling cycle outside circle (CW) |
| G33 | Cutting feed | | Rapid movement | Rectangular groove rough milling (CCW) |
| G34 | Cutting feed | | Rapid movement | Rectangular groove rough milling (CW) |
| G35 | Cutting feed | | Rapid movement | Finish milling cycle inside rectangular groove (CCW) |
| G36 | Cutting feed | | Rapid movement | Finish milling cycle inside rectangular groove (CW) |
| G37 | Cutting feed | | Rapid movement | Finish milling cycle outside rectangle (CCW) |
| G38 | Cutting feed | | Rapid movement | Finish milling cycle outside rectangle (CW) |
| G73 | Intermittent feed | | Rapid movement | High-speed deep hole machining |
| G74 | Cutting feed | Suspend the positive rotation of spindle | Rapid movement | Anti-tapping cycle |
| G76 | Cutting feed | Spindle orientation stops | Rapid movement | Precision boring |
| G80 | | | | Cancel the fixed cycle |
| G81 | Cutting feed | | Rapid movement | Drilling, spot drilling |
| G82 | Cutting feed | Stop cutting | Rapid movement | Drilling, boring step hole |
| G83 | Intermittent feed | | Rapid movement | Deep hole machining cycle |
| G84 | Cutting feed | Stop cutting ♦ negative rotation of spindle | Cutting feed | Tapping |
| G85 | Cutting feed | | Cutting feed | Boring |
| G86 | Cutting feed | Spindle stop | Rapid movement | Boring |
| G87 | Cutting feed | Positive rotation of spindle | Rapid movement | Boring |
| G88 | Cutting feed | Pause → Spindle stop | Manual → Positive rotation of spindle | Boring |
| G89 | Cutting feed | Pause | Cutting feed | Boring |

**Restrictions:**
The tool radius offset (D) will be ignored during fixed cycle positioning.

# 4.4.1 **Groove Rough Milling Inside Circle (G22/G23)**

**Format:**

**G22**

**G98/G99** **X_ Y_ Z_ R_ I_ L_ W_ Q_ V_ D_ F_ K_**

**G23**

**Function:** Starting from the center of the circle, multiple circular interpolations are performed in a spiral manner until a circular groove of the programmed size is formed by machining.

**Description:**

G22: groove rough milling inside circle (CCW);
G23: groove rough milling inside circle (CW)
X, Y: Starting position of the X, Y plane;
Z: Machining depth, which is an absolute position in case of G90, and a position relative to the R reference plane in case of G91;
R: R reference plane position, which is an absolute position in case of G90, and a position relative to the start point of this program segment in case of G91;
I: Radius of groove inside circle;
L: Width increment in XY plane of cutting
W: First cutting depth in the direction of Z axis, which is a distance from the R reference plane to the bottom, and should be greater than 0 (if the first cutting depth exceeds the groove bottom, the machining will be done directly at the groove bottom);
Q: Cutting depth of each cutting feed;
V: Distance from the unmachined surface at the time of fast entry;
D: Tool compensation number (the corresponding tool compensation value will be determined according to the given serial number);
K: Repetition times

**Cycle process:**

⑴ Quickly locate to the position determined by offset of the specified point (X, Y) to the negative direction of X axis by a tool radius D multiplied by spiral entry factor;
⑵ Quickly move down to the R point plane;
⑶ Cutting in a down spiral manner at the cutting speed for W distance depth → cutting feed to the circle center
⑷ Mill out a circular surface with its radius of I in a spiral manner from the center outwardly with a progressive increase of the L value;
⑸ The Z axis quickly returns to the R reference plane;
⑹ The X and Y axis are quickly positioned to the starting position
⑺ The Z axis rapidly drops with a distance of V from the unmachined surface;
⑻ Depth of Z-axis downward cutting (Q+V);
⑼ Repeat Activity (4) to (8) until the machining for a circular surface with total depth is done.
⑽ Return to the initial point plane or the R point plane according to the specified G98 or G99.

**Code track:**

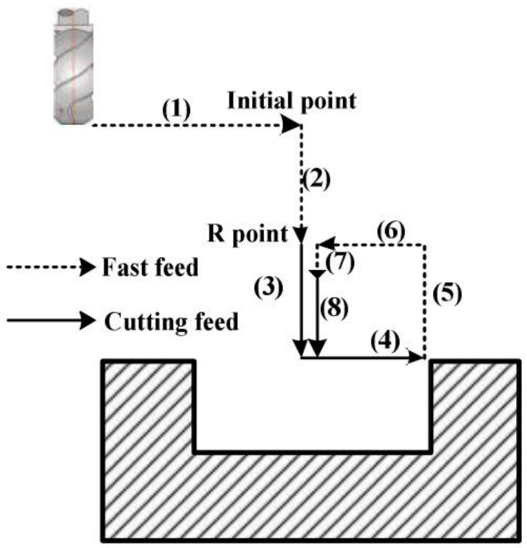



Fig. **4-4-1-1**

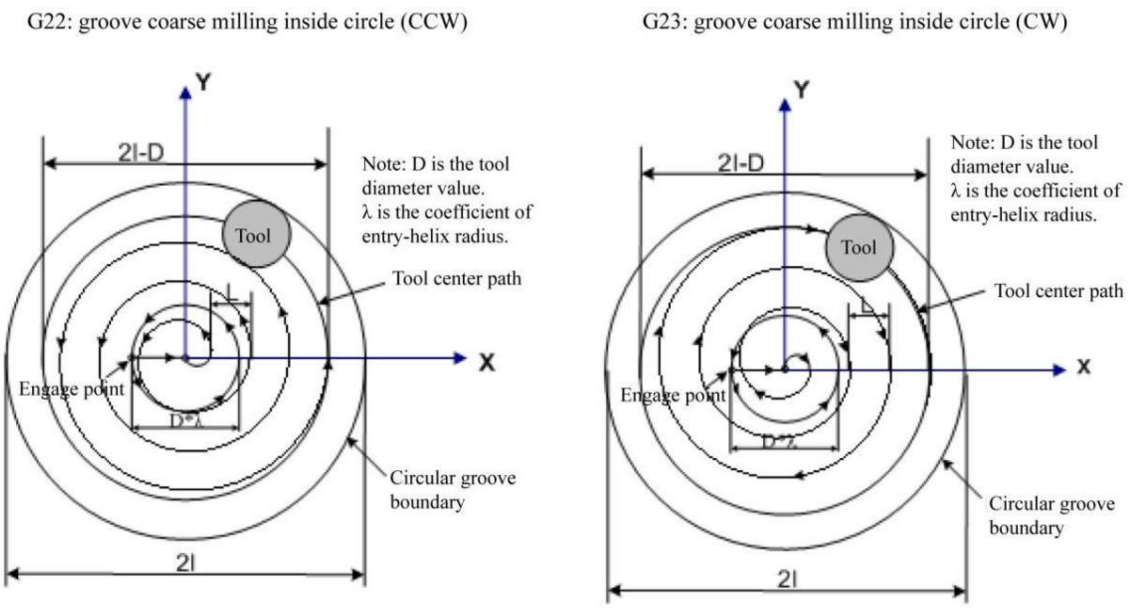



Fig. **4-4-1-2**

**Note:**

1. When this code is used, it is recommended to change NO: 12#1 to 1.

**Example:** Rough milling of a groove inside circle with the fixed cycle G22 is as shown below:

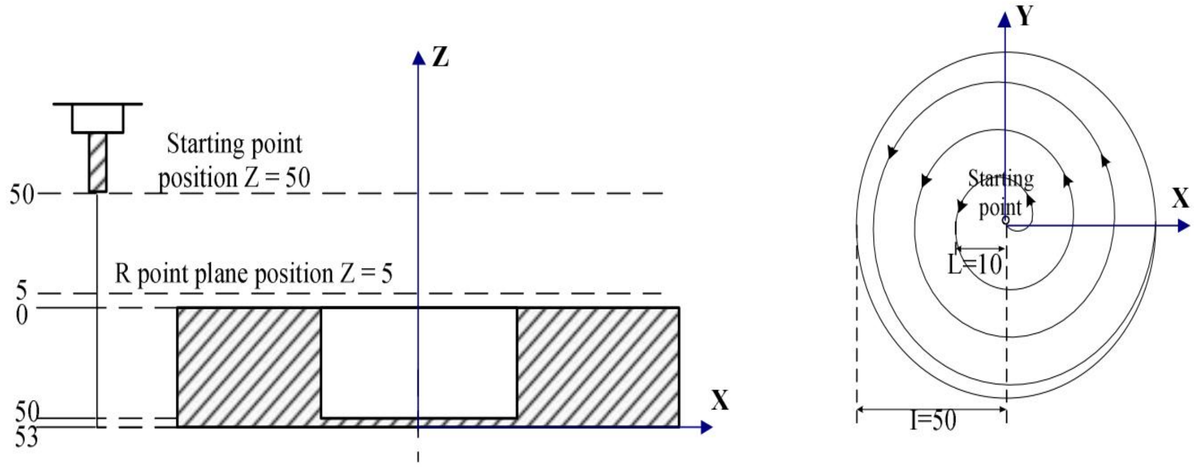



Fig. 4-6-1-3

G90 G00 X50 Y50 Z50; (G00 fast positioning)
G99 G22 X25 Y25 Z-50 R5 I50 L10 W20 (Perform groove rough milling inside circle)
Q10 V10 D1 F800;
G80 X50 Y50 Z50; (Cancel the fixed cycle and return from the R point plane)
M30;

**Restrictions:** When the G22/G23 command and the G code (G00 to G03, G60 are modal codes (when the position parameter NO: 48#0 is 1)) in Group 01 of the same program segment are used, the system will execute the final modal code.

**Tool radius compensation:** During positioning with this fixed cycle command, the tool radius offset will be ignored and during cutting feed, the tool radius compensation specified by the program will be called.

## 4.4.2 **Finish Milling Cycle Inside Full Circle (G24/G25)**

**Format:**

**G24**
**G98/G99** **X_ Y_ Z_ R_ I_ J_ D_ F_ K_**
**G25**

**Function:** The tool finishes a full circle inside circle with the specified radius value I and direction, and returns upon completion of finish milling.

**Description:** G24: finish milling cycle inside full circle (CCW).
G25: finish milling cycle inside full circle (CW).
X, Y: Starting position of the X, Y plane;
Z: Machining depth, which is an absolute position in case of G90, and a position relative to the R reference plane in case of G91;
R: R reference plane position, which is an absolute position in case of G90, and a position relative to the start point of this program segment in case of G91;
I: Finish-milling circle radius
J: Distance between finish-milling origin and finish-milling circle center;
D: Tool compensation number (the corresponding tool compensation value will be determined according to the given serial number);
K: Repetition times

**Cycle process:**

(1) Fast positioning on the XY plane;
(2) Quickly move down to the R point plane;
(3) Cutting feed to the starting point of hole bottom machining;
(4) Perform circular interpolation from the starting point with Transition Arc 1 as trajectory;
(5) Perform full circle interpolation with the inner circle of finish milling as trajectory;

(6) Perform circular interpolation with Transition Arc 4 as trajectory and return to the starting point;
(7) Return to the initial point plane or the R point plane according to the specified G98 or G99.

**Code track:**

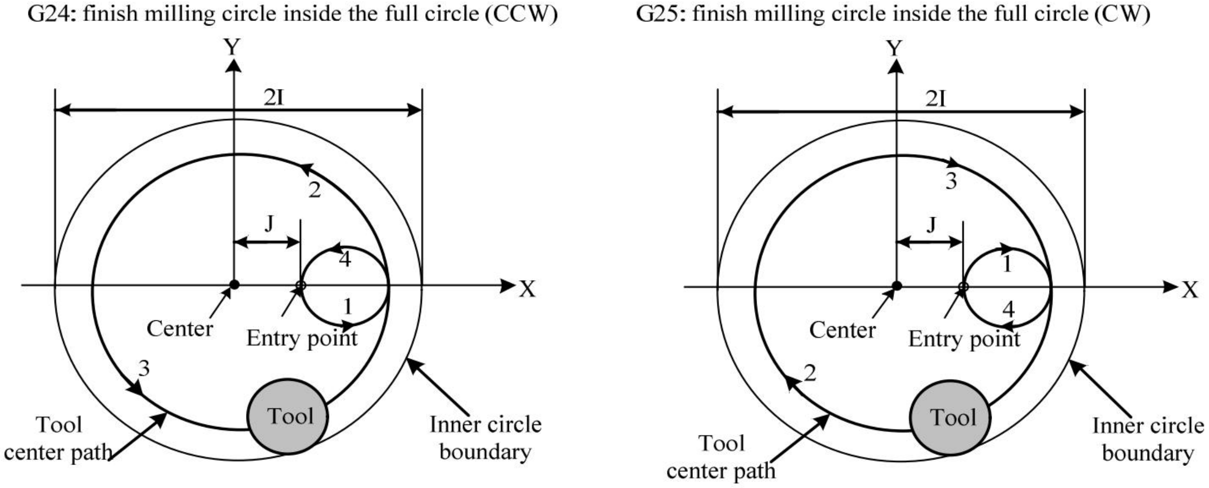



Fig. **4-4-2-1**

**Note:** When this code is used, it is recommended to change NO: 12#1 to 1.
**Example:** Use the fixed cycle G24 for finish milling of a roughly milled circular groove as shown below

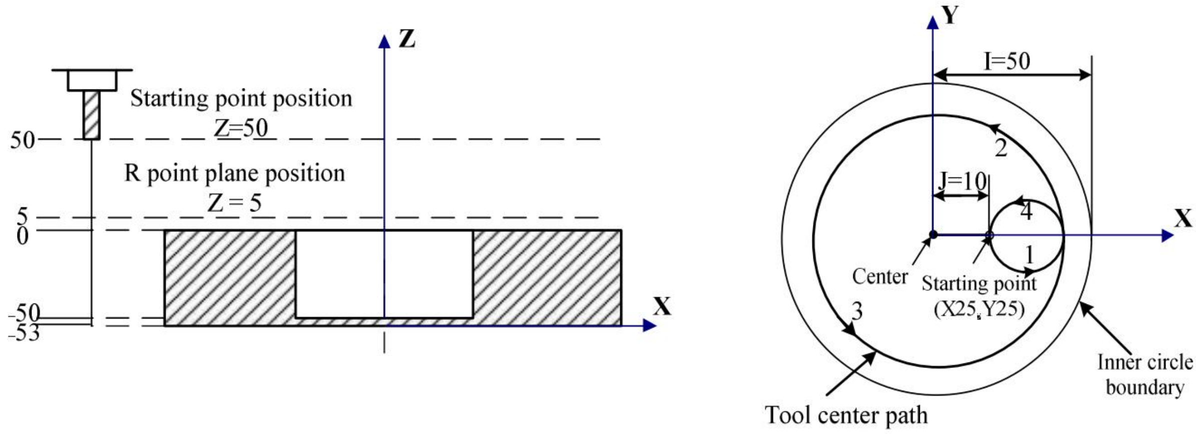



Fig. **4-4-2-2**

G90 G00 X50 Y50 Z50； （G00 fast positioning）
G99 G24 X25 Y25 Z-50 R5 I50 J10 D1F800； （Start the fixed cycle to move down to the bottom of the hole for finish milling cycle inside circle）
G80 X50 Y50 Z50； （Cancel the fixed cycle and return from the R point plane）
M30；

**Restrictions:** When the G24/G25 command and the G code (G00 to G03, G60 are modal codes (when the position parameter NO: 48#0 is 1)) in Group 01 of the same program segment are used, the system will execute the final modal code.
**Tool radius compensation:** During positioning with this fixed cycle command, the tool radius offset will be ignored and during cutting feed, the tool radius compensation specified by the program will be called.

# 4.4.3 **Finish Milling Cycle Outside Circle (G26/G32)**

**Format：**

**G98/G99 G26/G32 X_ Y_ Z_ R_ I_ J_ D_ F_ K_；**

**Function:** The tool finishes a full circle outside circle with the specified radius value I and direction, and returns upon completion of finish milling.

**Description:**

G26: finish milling cycle outside circle (CCW).
G32: finish milling cycle outside circle (CW)
X, Y: Starting position of the X, Y plane;
Z: Machining depth, which is an absolute position in case of G90, and a position relative to the R reference plane in case of G91;
R: R reference plane position, which is an absolute position in case of G90, and a position relative to the start point of this program segment in case of G91;
I: Finish-milling circle radius
J: Distance between finish-milling origin and finish-milling circle edge;
D: Tool compensation number (the corresponding tool compensation value will be determined according to the given serial number);
K: Repetition times

**Cycle process:**

(1) Fast positioning on the XY plane;
(2) Quickly move down to the R point plane;
(3) Cutting feed to the bottom of the hole;
(4) Perform circular interpolation from the starting point with Transition Arc 1 as trajectory;
(5) Perform full circle interpolation with Arc 2 and Arc 3 as trajectory;
(6) Perform circular interpolation with Transition Arc 4 as trajectory and return to the starting point;
(7) Return to the initial point plane or the R point plane according to the specified G98 or G99.

**Code track:**

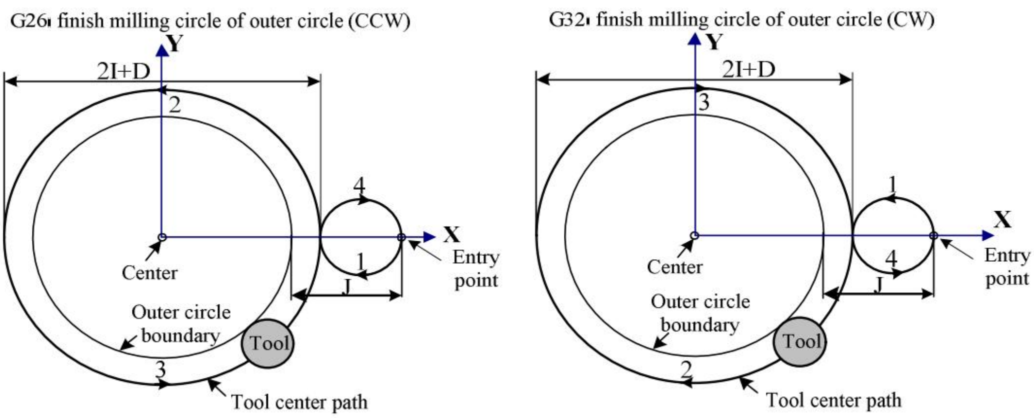



Fig. **4-4-3-1**

**Description:**

For finish milling outside circle, the interpolation direction of transition arc differs from that of finish milling arc, and the interpolation direction in the code description refers to the interpolation direction of finish milling arc.

**Example:** Use the fixed cycle G26 for finish milling of a roughly milled circular groove as shown below

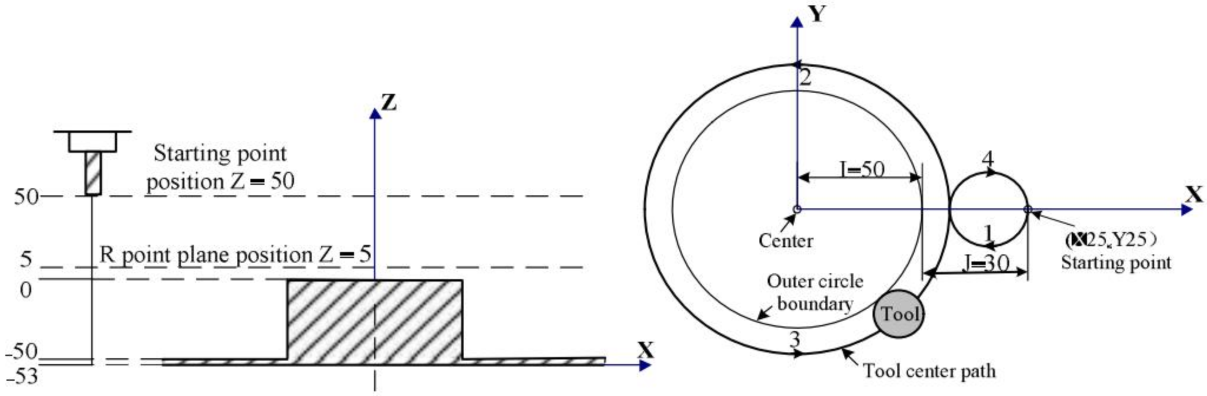



Fig. **4-4-3-2**

G90 G00 X50 Y50 Z50; (G00 fast positioning)

G99 G26 X25 Y25 Z-50 R5 I50 J30 D1 F800; (Start the fixed cycle to move down to the bottom of the hole for finish milling cycle outside circle)

G80 X50 Y50 Z50; (Cancel the fixed cycle and return from the R point plane)

M30;

**Restrictions:** When the G26/G32 command and the G codein Group 01 of the same program segment are used, the system will execute the final modal code.

**Tool radius compensation:** During positioning with this fixed cycle command, the tool radius offset will be ignored and during cutting feed, the tool radius compensation specified by the program will be called.

## 4.4.4 **Rectangular Groove Rough Milling (G33/G34)**

**Format:**

**G33**

**G98/G99 X_ Y_ Z_ R_ I_ J_ L_ W_ Q_ V_ U_ D_ F_ K_**

**G34**

**Function:** Starting from the rectangle center, the specified parameter data is used for linear cutting cycle until the programmed rectangular groove is machined.

**Description:**

G33: Rectangular groove rough milling (CCW);
G34: Rectangular groove rough milling (CW);
X, Y: Starting position of the X, Y plane;
Z: Machining depth, which is an absolute position in case of G90, and a position relative to the R reference plane in case of G91;
R: R reference plane position, which is an absolute position in case of G90, and a position relative to the start point of this program segment in case of G91;
I: Rectangular groove width in the X-axis direction;
J: Rectangular groove width in the Y-axis direction;
L: Cutting width increment in the specified plane;
W: First cutting depth in the direction of Z axis, which is a distance from the R reference plane to the bottom, and should be greater than 0 (if the first cutting depth exceeds the groove bottom, the machining will be done directly at the groove bottom);
Q: Cutting depth of each cutting feed;
V: Distance from the unmachined surface at the time of fast entry;
U: Corner arc radius, which indicates no corner arc transition if omitted;

D: Tool compensation number (the corresponding tool compensation value will be determined according to the given serial number);
K: Repetition times

**Cycle process:**

(1) Fast positioning to the starting point of spiral entry on the XY plane;
(2) Quickly move down to the R point plane;
(3) Use the radius compensation value multiplied by the value of Data Parameter No. 269 as the diameter for spiral entry W distance;
(4) Feeding to the rectangle center;
(5) Mill out a rectangular surface from the center outwardly with progressive increase of the L value;
(6) The Z axis quickly returns to the R reference plane;
(7) Fast positioning to the starting point of spiral entry on the XY plane;
(8) The Z axis rapidly drops with a distance of V from the unmachined surface;
(9) Depth of Z-axis downward cutting (Q+V);
(10) Repeat Activity (4) to (9) until the machining for a rectangular surface with total depth is done.
(11) Return to the initial point plane or the R point plane according to the specified G98 or G99.

**Code track:**

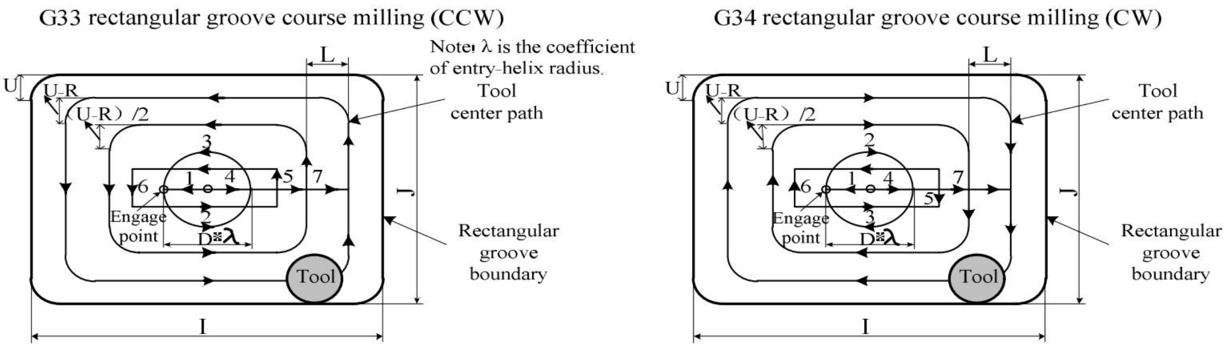



Fig. **4-4-4-1**

**Note:** When this code is used, it is recommended to change NO: 12#1 revised as 1

**Example:** Rough milling of a groove inside rectangle with the fixed cycle G33 is as shown below:

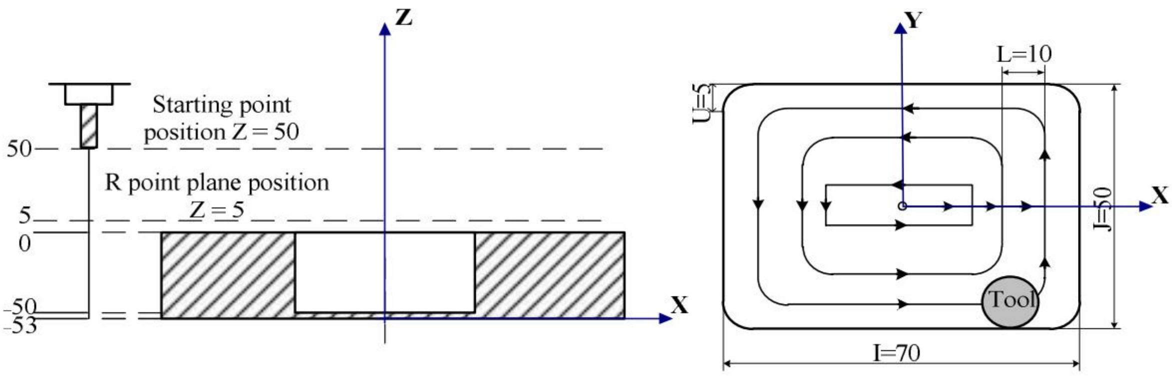



Fig. **4-4-4-2**

G90 G00 X50 Y50 Z50; (G00 fast positioning)
G99 G33 X25 Y25 Z-50 R5 I70 J50 L10 W20 Q10 V10 U5 D1 F800; (Perform groove rough milling inside rectangle)
G80 X50 Y50 Z50; (Cancel the fixed cycle and return from the R point plane)
M30;

**Restrictions:** When the G33/G34 command and the G code (G00 to G03, G60 are modal codes (when the position parameter NO: 48#0 is 1)) in Group 01 of the same program segment are used, the system will execute the final modal code.

**Tool radius compensation:** During positioning with this fixed cycle command, the tool radius offset will be ignored and during cutting feed, the tool radius compensation specified by the program will be called.

## 4.4.5 **Finish Milling Cycle Inside Rectangular Groove (G35/G36)**

**Format:**

**G35**
**G98/G99** **X_ Y_ Z_ R_ I_ J_ L_ U_ D_ F_ K_；**
**G36**

**Function:** The tool performs finish milling inside rectangle with the specified width and direction and returns upon completion of finish milling.

**Description:**

G35: Finish milling cycle inside rectangular groove (CCW).
G36: Finish milling cycle inside rectangular groove (CW).
X, Y: Starting position of the X, Y plane;
Z: Machining depth, which is an absolute position in case of G90, and a position relative to the R reference plane in case of G91;
R: R reference plane position, which is an absolute position in case of G90, and a position relative to the start point of this program segment in case of G91;
I: Rectangle width in the X-axis direction;
J: Rectangle width in the Y-axis direction;
L: Distance between finish-milling origin and rectangular side X-axis positive direction;
U: Corner arc radius, which indicates no corner arc transition if omitted; When 0<|U|<tool radius, it will give an alarm;
D: Tool compensation number (the corresponding tool compensation value will be determined according to the given serial number);
K: Repetition times

**Cycle process:**

(1) Fast positioning to the starting position on the XY plane;
(2) Quickly move down to the R point plane;
(3) Cutting feed to the bottom of the hole;
(4) Perform circular interpolation from the starting point with Transition Arc 1 as trajectory;
(5) Perform linear and circular interpolation with 2-3-4-5-6 as trajectory;
(6) Perform circular interpolation with Transition Arc 7 as trajectory and return to the starting point;
(7) Return to the initial point plane or the R point plane according to the specified G98 or G99.

**Code track:**

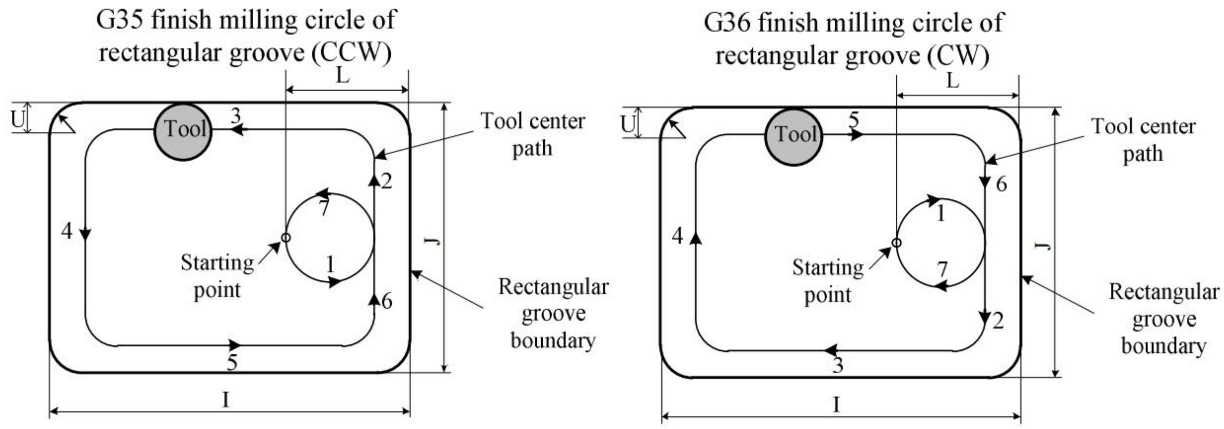



Fig. **4-4-5-1**

**Note:** When this code is used, it is recommended to change NO: 12#1 revised as 1

**Example:** Use the fixed cycle G35 for finish milling of a roughly milled groove as shown below

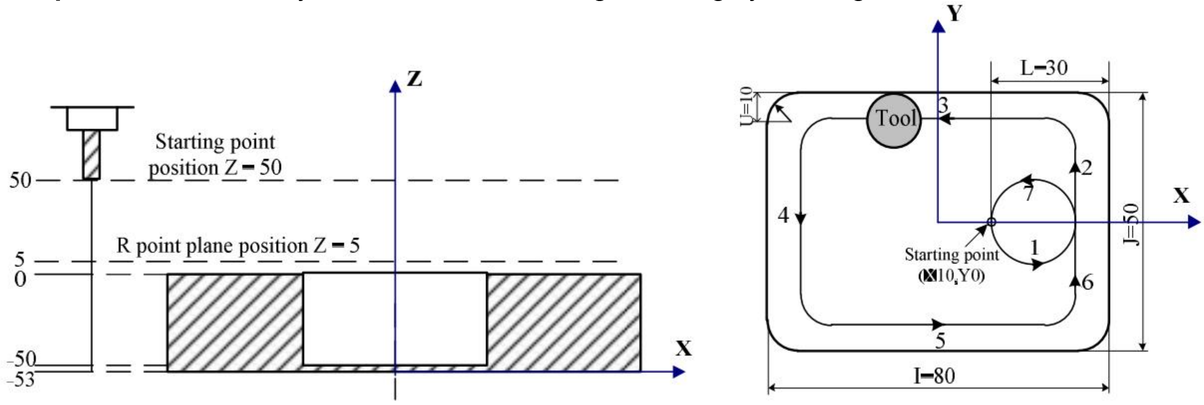



Fig. **4-4-5-2**

G90 G00 X50 Y50 Z50; (G00 fast positioning)

G99 G35 X10 Y0 Z-50 R5 I80 J50 L30 U10 D1 F800; (start the fixed cycle to move down to the hole bottom for milling inside rectangular groove)

G80 X50 Y50 Z50; (Cancel the fixed cycle and return from the R point plane)

M30;

**Restrictions:** When the G35/G36 command and the G code (G00 to G03, G60 are modal codes (when the position parameter NO: 48#0 is 1)) in Group 01 of the same program segment are used, the system will execute the final modal code.

**Tool radius compensation:** During positioning with this fixed cycle command, the tool radius offset will be ignored and during cutting feed, the tool radius compensation specified by the program will be called.

## 4.4.6 **Finish Milling Cycle Outside Rectangle (G37/G38)**

**Format:**

**G98/G99** **G37** / **G38** **X_ Y_ Z_ R_ I_ J_ L_ U_ D_ F_ K_**

**Function:** The tool performs finish milling outside rectangle with the specified width and direction and returns upon completion of finish milling.

**Description:**

G37: Finish milling cycle outside rectangle (CCW).

G38: Finish milling cycle outside rectangle (CW).

X, Y: Starting position of the X, Y plane;

Z: Machining depth, which is an absolute position in case of G90, and a position relative to the R reference plane in case of G91;

R: R reference plane position, which is an absolute position in case of G90, and a position relative to the start point of this program segment in case of G91;

I: Rectangle width in the X-axis direction;

J: Rectangle width in the Y-axis direction;

L: Distance between finish-milling origin and rectangular side X-axis direction;

U: Corner arc radius, which indicates no corner arc transition if omitted;

D: Tool compensation number (the corresponding tool compensation value will be determined according to the given serial number);

K: Repetition times

**Cycle process:**

(1) Fast positioning to the starting position on the XY plane;

(2) Quickly move down to the R point plane;

⑶ Cutting feed to the bottom of the hole;
⑷ Perform circular interpolation from the starting point with Transition Arc 1 as trajectory;
⑸ Perform linear and circular interpolation with 2-3-4-5-6 as trajectory;
⑹ Perform circular interpolation with Transition Arc 7 as trajectory and return to the starting point;
⑺ Return to the initial point plane or the R point plane according to the specified G98 or G99.

**Code track:**

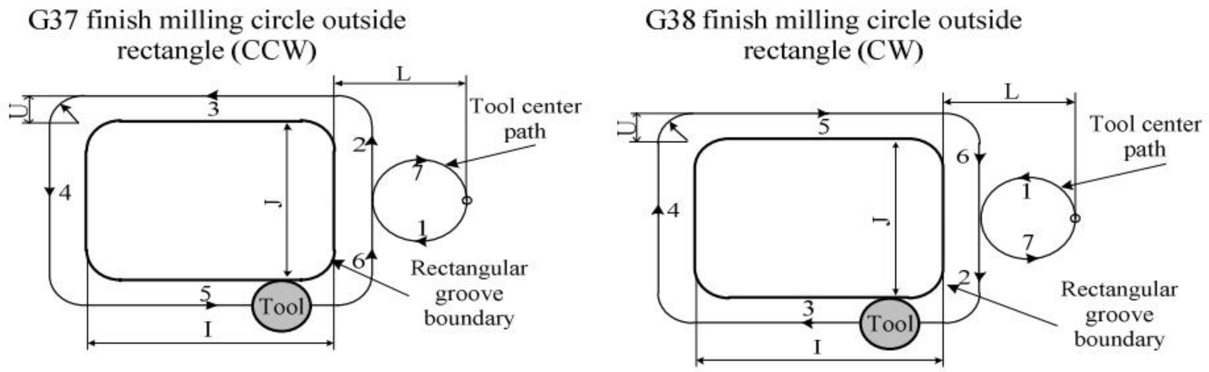



Fig. **4-4-6-1**

**Description:** For finish milling outside rectangle, the interpolation direction of transition arc differs from that of finish milling arc, and the interpolation direction in the code description refers to the interpolation direction of finish milling arc.

**Example:** Perform finish milling outside rectangle using the fixed cycle G37.

G90 G00 X50 Y50 Z50; (G00 fast positioning)
G99 G37 X25 Y25 Z-50 R5 I80 J50 L30 (Perform finish milling outside rectangle at the bottom of the hole in fixed cycle)
U10 D1 F800;
G80 X50 Y50 Z50; (Cancel the fixed cycle and return from the R point plane)
M30;

**Restrictions:** When the G37/G38 command and the G code (G00 to G03, G60 are modal codes (when the position parameter NO: 48#0 is 1)) in Group 01 of the same program segment are used, the system will execute the final modal code.

**Tool radius compensation:** During positioning with this fixed cycle command, the tool radius offset will be ignored and during cutting feed, the tool radius compensation specified by the program will be called.

## 4.4.7 **High-Speed Deep Hole Machining Cycle (G73)**

**Format: G73 X_Y_Z_R_Q_F_K_**

**Function:** This cycle is designed to perform high-speed deep drilling, which performs intermittent cutting feed to the bottom of the hole, and quickly retracts from the hole while feeding and removes the chips. The activity diagram is shown in Figure 4-4-1-1.

**Description:**

X_Y: Hole positioning data;
Z: Incremental programming defines the distance from Point R to the bottom of the hole; absolute programming defines the absolute coordinate values of the bottom of the hole;
R: Incremental programming defines the distance from the initial point plane to Point R; absolute programming defines the absolute coordinate values of Point R;
Q: Cutting depth of each cutting feed;
F: Cutting feed rate;

K: Repetition times

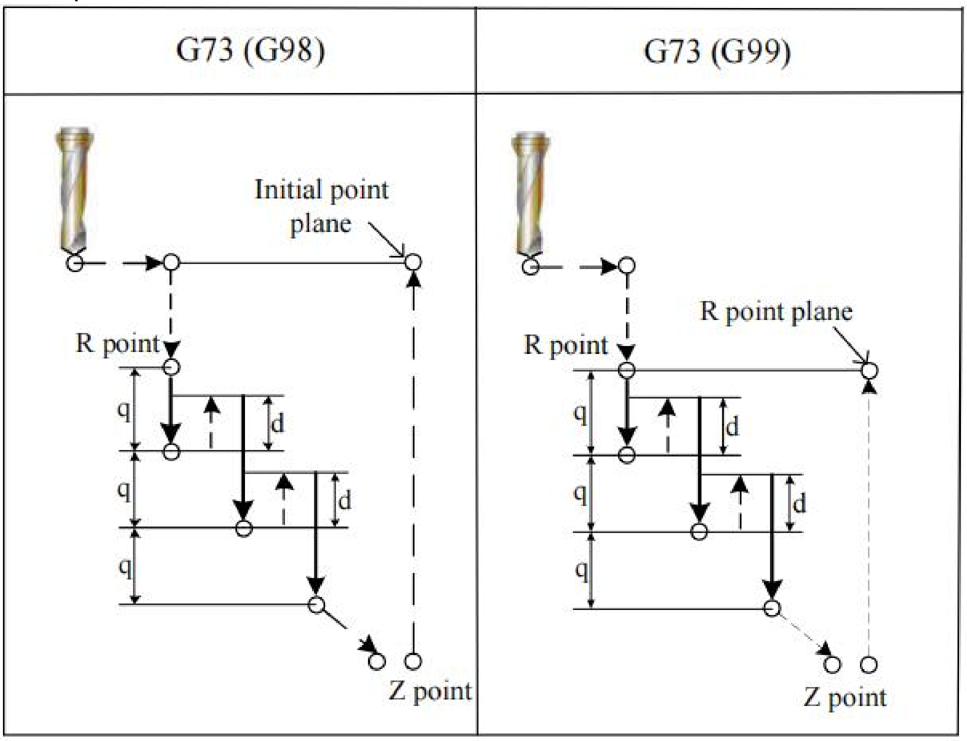



Fig. **4-4-7-1**

**Z, R:** In case of lack of both of Parameters Z and R when drilling is conducted to form the first hole, the system will not perform the Z-axis activities.

**Q:** When the code parameter Q is specified, the intermittent feed as shown in the figure above will be performed. At the moment, the system will retract using the retraction amount d (as shown in Fig. 4-4-1-1.) set in Data Parameter P270, and the tool will intermittently perform the fast moving retraction with the distance d for each feed.

When **G73** and the M code are specified by the same program segment, the M code is executed during positioning of the first hole, and then the system deals with the next drilling activity.

When repetition times K is specified, the M code is executed only for the first hole, and not for subsequent holes.

**Note 1:** In the current version, M00, M01, M02, M06, M30, M98, and M99 are post-program executed, and the above M code will be executed after the current segment is executed.

**Note 2:** When the bit parameter NO: 43# 1=0, in case of deep drilling (G73, G83) without specified cutting depth, no alarm will be triggered, and in this case, the code parameter Q is not specified or Q is specified as 0, and the system will perform hole positioning in the X and Y planes, but not perform drilling. When the bit parameter NO: 43#1=1, in case of deep drilling (G73, G83) without specified cutting depth, an alarm will be triggered. That is, when the code parameter Q is not specified or Q is specified as 0, the system will provide the alarm: "0045: Address Q is not found or Q value is 0 (G73/G83)". If Q is specified as a negative value, the system will perform intermittent feed with its absolute value.

**Note 3:** Tool length compensation: When the tool length compensation G43, G44 or G49 shares the same program segment with the fixed cycle command, add or cancel the offset when positioning to Point R; in the fixed cycle mode, when the tool compensation G43, G44 or G49 is alone in a program segment, the system can add or cancel the offset in real time.

**Restrictions:**

When the G73 command and the G code (G00 to G03, G60 are modal codes (when the position parameter NO: 48#0 is 1)) in Group 01 of the same program segment are used, the system will execute the final modal code.

**Tool radius compensation:** In this fixed cycle command, the tool radius compensation will be ignored because the command function does not require tool radius compensation.

**Example:**

M3 S1500; the spindle starts rotating.
G90 G99 G73 X0 Y0 Z-15 R-10 Q5 F120; positioning, Hole 1 drilling, and then return to Point R
Y-50; Positioning, Hole 2 drilling, and then return to Point R

Y-80; Positioning, Hole 3 drilling, and then return to Point R
Y-10; Positioning, Hole 4 drilling, and then return to Point R
Y-10; Positioning, Hole 5 drilling, and then return to Point R
G98 Y75; Positioning, Hole 6 drilling, and then return to the initial position plane
G80;
G28 G91 X0 Y0 Z0; return to reference point
M5; the spindle stops rotating
M30;

**Note:** In the case of Holes 2 - 6 machining in the above examples, although Q is omitted, the chips will also be removed.

## 4.4.8 **Drilling Cycle and Point Drilling Cycle (G81)**

**Format:G81 X_ Y_ Z_ R_ F_ K_**

**Function:** This cycle is used for normal drilling cutting feed, which is carried out to the bottom of the hole, and then the tool will quickly retract from the bottom.

**Description:**

X_Y_: Hole positioning data;
Z_: Incremental programming defines the distance from Point R to the bottom of the hole; absolute programming defines the absolute coordinate values of the bottom of the hole;
R_: Incremental programming defines the distance from the initial point plane to Point R; absolute programming defines the absolute coordinate values of Point R;
F_: Cutting feed rate;
K_: Repetition times (if necessary).

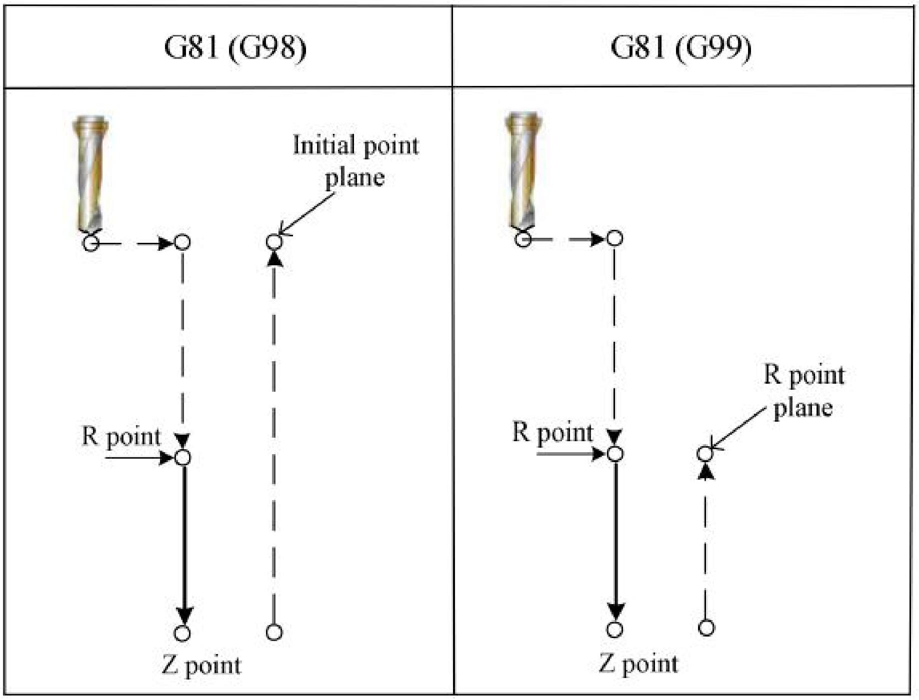



Fig. **4-4-8-1**

**Z, R:** In case of lack of any of Parameters Z and R when drilling is conducted to form the first hole, the system will only change its mode and not perform the Z-axis activities.
After positioning along the X and Y axis, move quickly to Point R, perform drilling from Point R to Point Z, and then quickly retract the tool.
Rotate the spindle with the auxiliary function M code before G81 is specified.

When G81 and the M code are specified by the same program segment, the M code is executed during positioning of the first hole, and then the system deals with the next drilling activity.

When repetition times K is specified, the M code is executed only for the first hole, and not for subsequent holes.

**Note:** In the current version, M00, M01, M02, M06, M30, M98, and M99 are post-program executed, and the above M code will be executed after the current segment is executed.

**Tool length compensation:** When the tool length compensation G43, G44 or G49 shares the same program segment with the fixed cycle command, add or cancel the offset when positioning to Point R; in the fixed cycle mode, when the tool compensation G43, G44 or G49 is alone in a program segment, the system can add or cancel the offset in real time.

**Example:**

M3 S2000; the spindle starts rotating
G90 G99 G81 X300 Y-250 Z-150 R-10 F120; positioning, Hole 1 drilling, and then return to Point R
Y-550; Positioning, Hole 2 drilling, and then return to Point R
Y-750; Positioning, Hole 3 drilling, and then return to Point R
Y1000; Positioning, Hole 4 drilling, and then return to Point R
Y-550; Positioning, Hole 5 drilling, and then return to Point R
G98 Y-750; Positioning, Hole 6 drilling, and then return to the initial position plane
G80;
G28 G91 X0 Y0 Z0; Return to reference point
M5; The spindle stops rotating
M30;

**Restrictions:** When the G81 command and the G code (G00 to G03, G60 are modal codes (when the position parameter NO: 48#0 is 1)) in Group 01 of the same program segment are used, the system will execute the final modal code.

**Tool radius compensation:** In this fixed cycle command, the tool radius compensation will be ignored because the command function does not require tool radius compensation.

## 4.4.9 **Drilling Cycle and Boring Cycle (G82)**

**Format: G82 X_ Y_ Z_ R_ P_ F_ K_；**

**Function:** This cycle is used for normal drilling; cutting feed to the bottom of the hole and pause execution, and then the tool will quickly retract from the bottom.

**Description:**

X_Y_: Hole positioning data;
Z_: Incremental programming defines the distance from Point R to the bottom of the hole; absolute programming defines the absolute coordinate values of the bottom of the hole;
R_: Incremental programming defines the distance from the initial point plane to Point R; absolute programming defines the absolute coordinate values of Point R;
F_: Cutting feed rate;
P_: Pause time in the bottom of the hole;
K_: Repetition times

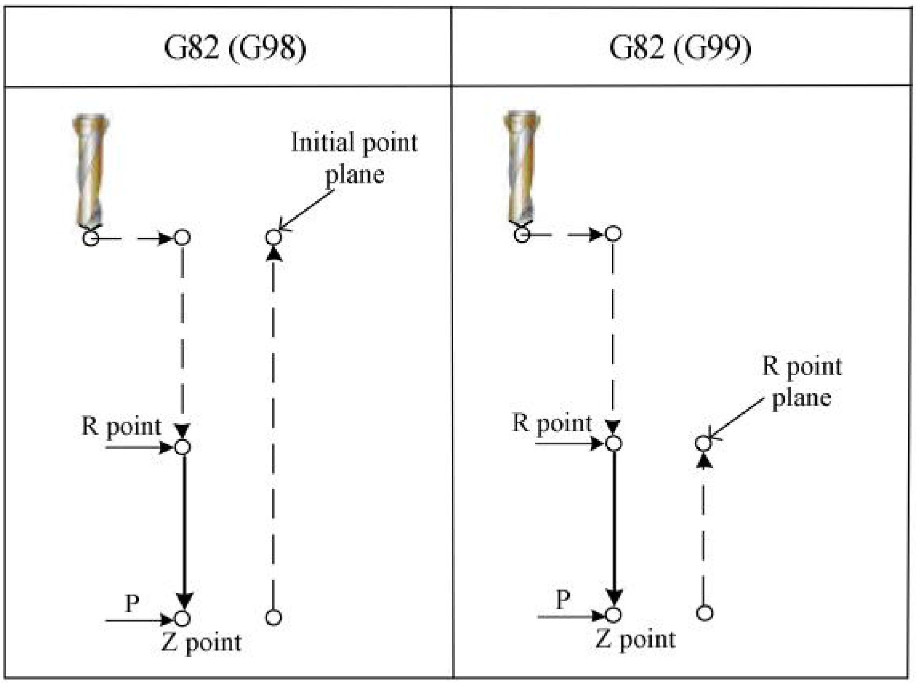



Fig. **4-4-9-1**

After positioning along the X and Y axis, move quickly to Point R, and perform drilling from Point
R to Point Z When it reaches the bottom of the hole, the pause will be executed and the tool will retract quickly.
Rotate the spindle with the auxiliary function M code before G82 is specified.
When G82 and the M code are specified by the same program segment, the M code is executed during positioning of the first hole, and then the system deals with the next drilling activity.
When repetition times K is specified, the M code is executed only for the first hole, and not for subsequent holes.

**Note:** In the current version, M00, M01, M02, M06, M30, M98, and M99 are post-program executed, and the above M code will be executed after the current segment is executed.

Tool length compensation: When the tool length compensation G43, G44 or G49 shares the same program segment with the fixed cycle command, add or cancel the offset when positioning to Point R; in the fixed cycle mode, when the tool compensation G43, G44 or G49 is alone in a program segment, the system can add or cancel the offset in real time.
P is a modal code; the minimum value of the parameter is set by Data Parameter **P296**, while the maximum value is set by **P297**. When the P value is less than that set by **P296**, it will run at the minimum value; when the P value is greater than that set by **P297**, it will run at the maximum value.

**Example:**

M3 S2000; the spindle starts to rotate
G90 G99 G82 X300 Y-250 Z-150 R-100 P1000 F120; Positioning, Hole 1 drilling, pause for 1 second at the bottom and then return to Point R
Y-550; Positioning, Hole 2 drilling, pause for 1 second at the hole bottom and then return to Point R
Y-750; Positioning, Hole 3 drilling, pause for 1 second at the bottom and then return to Point R
X1000.; Positioning, Hole 4 drilling, pause for 1 second at the bottom and then return to Point R
Y-550; Positioning, Hole 5 drilling, pause for 1 second at the bottom and then return to Point R
G98 Y-750; Positioning, Hole 6 drilling, pause for 1 second at the bottom and then return to the initial position plane

G80; Cancel the fixed cycle
G28 G91 X0 Y0 Z0; Return to reference point
M5; the spindle stops rotating
M30;

**Restrictions:** When the G82 command and the G code (G00 to G03, G60 are modal codes (when the position parameter NO: 48#0 is 1)) in Group 01 of the same program segment are used, the system will execute the final modal code.

**Tool radius compensation:** In this fixed cycle command, the tool radius compensation will be ignored because the command function does not require tool radius compensation.

## 4.4.10 **Chip removal drilling cycle (G83)**

**Format: G83 X_ Y_ Z_ R_ Q_ F_ K_**

**Function:** This cycle is used for deep drilling, which performs intermittent cutting feed to the bottom of the hole, and chips are removed from the hole during drilling.

**Description:**

X_Y_: Hole positioning data;
Z_: Incremental programming defines the distance from Point R to the bottom of the hole; absolute programming defines the absolute coordinate values of the bottom of the hole;
R_: Incremental programming defines the distance from the initial point plane to Point R; absolute programming defines the absolute coordinate values of Point R;
Q_: Cutting depth of each cutting feed;
F_: Cutting feed rate;
K_: Repetition times

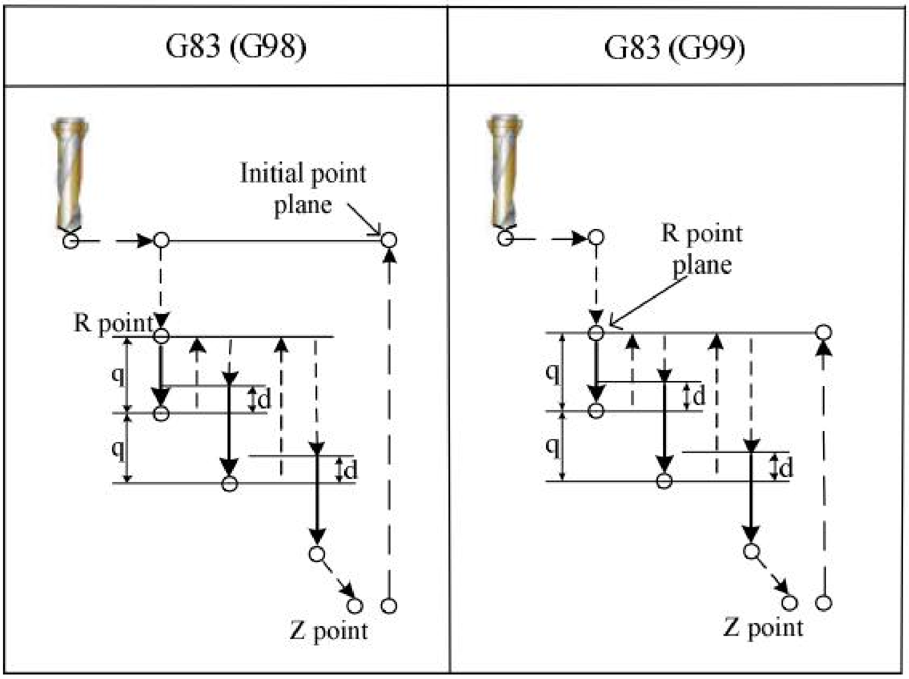



Fig. **4-4-10-1**

**Q**: It is cutting depth for each cutting feed, which must be expressed in incremental value. In the second and subsequent cutting feeds, quickly move to the point with the distance of d before the end of the previous drilling, and then perform the cutting feed again, and the value of d is set by Parameter **P295**. As shown in the figure **4-4-4-1.**
A positive value must be specified in Q, and the negative sign will be ignored, and the system will still deal with it as a positive value.
Q is specified in the program segment in which drilling is performed, and Q will not be stored as
modal data if specified in the program segment in which drilling is not performed.

Rotate the spindle (M code) with the auxiliary function before G83 is specified.
When G83 and the M code are specified by the same program segment, the M code is executed during positioning of the first hole, and then the system deals with the next drilling activity.
When repetition times K is specified, the M code is executed only for the first hole, and not for subsequent holes.

**Note 1:** In the current version, M00, M01, M02, M06, M30, M98, and M99 are post program executed, and the above M code will be executed after the current segment is executed.

**Note 2:** When the bit parameter NO: 43# 1=0, in case of deep drilling (G73, G83) without specified cutting depth, no alarm will be triggered, and in this case, the code parameter Q is not specified or Q is specified as 0, and the system will perform hole positioning in the X and Y planes, but not perform drilling. When the bit parameter NO: 43#1=1, in case of deep drilling (G73, G83) without specified cutting depth, an alarm will be triggered. That is, when the code parameter Q is not specified or Q is specified as 0, the system will provide the alarm: "0045: Address Q is not found or Q value is 0 (G73/G83)". If Q is specified as a negative value, the system will perform intermittent feed with its absolute value.

**Tool length compensation:** When the tool length compensation G43, G44 or G49 shares the same program segment with the fixed cycle command, add or cancel the offset when positioning to Point R; in the fixed cycle mode, when the tool compensation G43, G44 or G49 is alone in a program segment, the system can add or cancel the offset in real time.

**Example:**

M3 S2000; The spindle starts rotating
G90 G99 G83 X300 Y-250 Z-150 R-100 Q15 F120; Positioning, Hole 1 drilling, and then return to Point R
Y-550; Positioning, Hole 2 drilling, and then return to Point R
Y-750; Positioning, Hole 3 drilling, and then return to Point R
X1000; Positioning, Hole 4 drilling, and then return to Point R
Y-550; Positioning, Hole 5 drilling, and then return to Point R
G98 Y-750; Positioning, Hole 6 drilling, and then return to the initial position plane
G80;
G28 G91 X0 Y0 Z0; Return to reference point
M5; The spindle stops rotating
M30;

**Restrictions:** When the G83 command and the G code (G00 to G03, G60 are modal codes (when the position parameter NO: 48#0 is 1)) in Group 01 of the same program segment are used, the system will execute the final modal code, i.e. G60.

**Tool radius compensation:** In this fixed cycle command, the tool radius compensation will be ignored because the command function does not require tool radius compensation.

## 4.4.11 **Tapping Cycle (G74 Or G84)**

**Format: G74/G84 X_ Y_ Z_ R_ P_ F_**

**Function:** This cycle performs tapping. In the tapping cycle, a pause will be performed when the tapping shaft reaches the bottom of the hole, and then the spindle reversely rotates back to the tapping shaft. (G74 is a left tapping cycle and G84 is a right tapping cycle)

**Description:**

X_Y_: Hole positioning data;
Z_: Incremental programming defines the distance from Point R to the bottom of the hole; absolute programming defines the absolute coordinate values of the bottom of the hole;
R_: Incremental programming defines the distance from the initial point plane to Point R; absolute programming defines the absolute coordinate values of Point R;
P_: Pause time in the bottom of the hole;
F_: Tapping feed rate;
K_: Repetition times; (specified when needed)

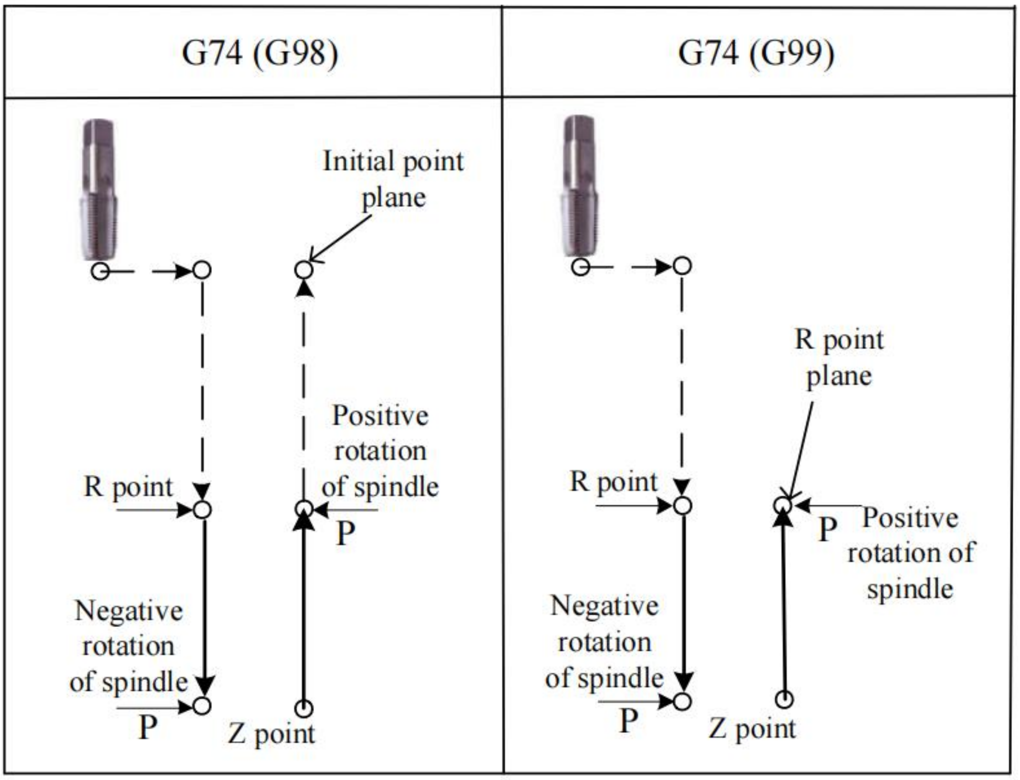
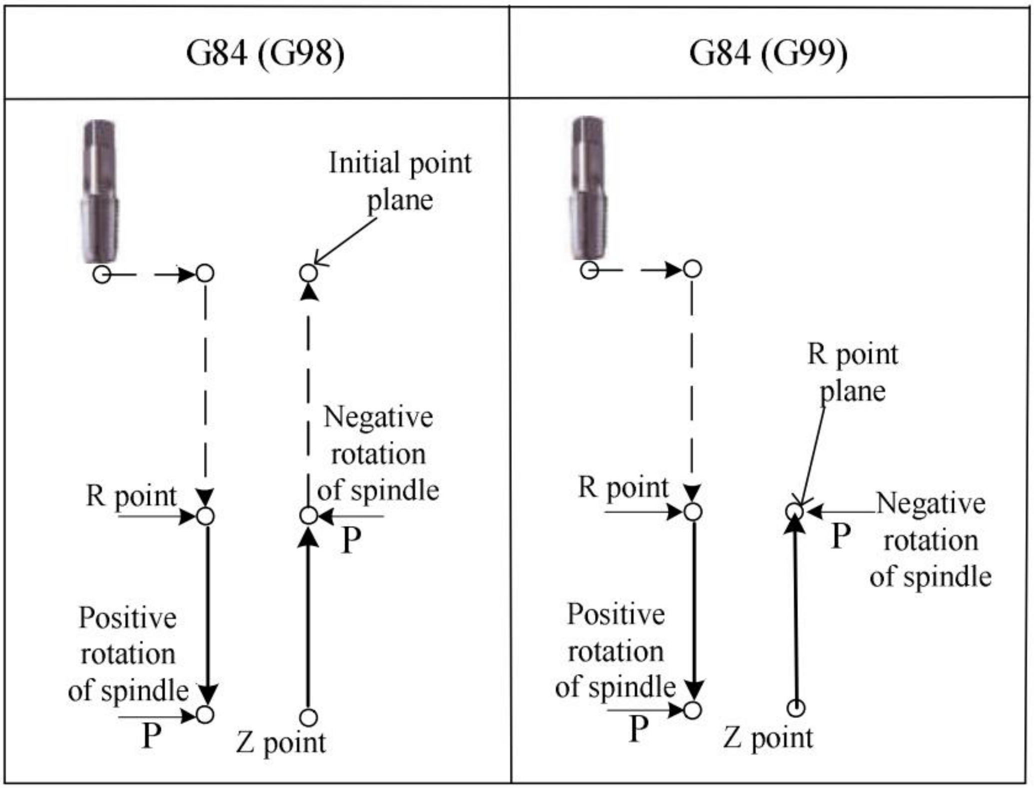



Fig. **4-4-11-1**




In case of the G74 command, the spindle rotates clockwise (while in case of the G84 command, the spindle rotates counterclockwise) to perform tapping. A pause will be executed when it reaches the bottom of the hole. And the spindle will rotate reversely and retract to the tapping shaft at the specified feed rate. Threads will be formed in this process.

**Example:**

G94 Feed mode per minute;

M29 S1000; The spindle stops rotating and its rotation speed is specified

G43 / G44 H10; Call the tool length compensation
G90 G99 G74 / G84 X100 Y110 Z -50 R5 P3000 F100; Positioning, Hole 1 tapping, then return to Point R
Y150; Positioning, Hole 2 tapping, then return to Point R
G91 X50 K5; Use X100, Y150 as the reference point, along the X axis Perform 5 times of tapping with an increment of 50mm
G98 Y-750; Positioning, Hole 8 tapping, then return to the initial point
G80; Cancel the tapping cycle
G28 G91 X0 Y0 Z0; Return to reference point
M30; Program ends

**Tool length compensation:** When the tool length compensation G43, G44 or G49 shares the same program segment with the fixed cycle command, add or cancel the offset when positioning to Point R; in the fixed cycle mode, when the tool compensation G43, G44 or G49 is alone in a program segment, the system can add or cancel the offset in real time.

**Thread lead:** In the feed mode per minute, the relationship between the thread lead and the feed rate and the spindle speed:
Feed rate F = screw tap thread pitch × spindle speed S
For example: For tapping of M12×1.5 threaded holes on a component, the following parameters can be used:
S500=500 r/min; F=1.5×500=750 mm/min;
In case of multiple thread, multiply by the number of heads to get the F value.
In the feed mode per rotation, the thread lead is equal to the feed rate.

**For example:**

| Feed mode per minute: | Feed mode per rotation: |
|---|---|
| Spindle speed 1000 r/min; | Spindle speed 1000 r/min; |
| Thread lead 1.0 mm; | Thread lead 1.0 mm; |
| Then Z-axis feed rate = 1000*1 = 1000 mm/min; | Then Z-axis feed rate = thread lead = 1mm/r; |
| G94 feed mode per minute | G95 feed mode per rotation |
| G00 X120 Y100; Positioning | G00 X120 Y100; Positioning |
| M29 S1000; Specify rigid mode specified | M29 S1000; Specify rigid mode |
| G84 Z-100 R-20 F1000; Right rigid tapping | G84 Z-100 R-20 F1; Right rigid tapping |
| G80 Cancel the tapping cycle | G80 Cancel the tapping cycle |
| G28 G91 X0 Y0 Z0 Return to reference point | G28 G91 X0 Y0 Z0 Return to reference point |
| M30 Program ends | M30 Program ends |

**Restrictions:**

**G code:** When the G74/G84 command and the G code (G00 to G03, G60 are modal codes (when the position parameter NO: 48#0 is 1)) in Group 01 of the same program segment are used, the system will execute the final modal code.

**M code:** Rotate the spindle with the auxiliary function M code before G74/G84 is specified. If the spindle rotates without command, the system automatically will adjust to clockwise rotation (G74)/counterclockwise rotation (G84) based on the current spindle command speed in the R plane.
When G74/G84 and the M code are specified by the same program segment, the M code is executed during positioning of the first hole, and then the system deals with the next tapping activity.
When repetition times K is specified, the M code is executed only for the first hole, and not for subsequent holes.

**Note:** In the current version, M00, M01, M02, M06, M30, M98, and M99 are post-program executed, and the above M code will be executed after the current segment is executed.

**S command:** If the specified spindle speed exceeds the maximum spindle speed during tapping (Data Parameter P257: spindle upper limit speed during tapping cycle), the system will send an alarm; the maximum spindle speed during rigid tapping is set by Data Parameter P294 - P296.

**F command:** If the specified F value exceeds the upper limit of the cutting feed rate (data parameter: the upper limit is set by P96), the upper limit shall prevail.

**P command:** P is a modal code; the minimum value of the parameter is set by Data Parameter **P296**, while the maximum value is set by **P297**. When the P value is less than that set by

**P296**, it will run at the minimum value; when the P value is greater than that set by **P297**, it will run at the maximum value.

**Shaft switch:** The fixed cycle must be canceled before switching the tapping shaft. If the tapping shaft is changed in the rigid mode, the system will send No. 206 alarm.

**Override:** During tapping, the feedrate override and spindle speed override are 100% by default, and the machine tool will not stop working when the hold button is pressed until the return activity is completed.

**Tool radius compensation:** In this fixed cycle command, the tool radius compensation will be ignored because the command function does not require tool radius compensation.

**Program restart:** In the tapping cycle, the program restart function is invalid.

## 4.4.12 **Precision Boring Cycle** G76

**Format: G76 X_Y_Z_Q_R_P_F_K_**

**Function:** This cycle is applied to precision boring of holes.

When the bottom of the hole is reached, the spindle will stop rotating and the cutting tool will leave the workpiece surface for return.

Retraction marks should be prevented to avoid impact on smooth finish of the machined surface and damages to the tool.

**Description:**

X_Y_: Hole positioning data;
Z_: Incremental programming defines the distance from Point R to the bottom of the hole; absolute programming defines the absolute coordinate values of the bottom of the hole;
R_: Incremental programming defines the distance from the initial point plane to Point R; absolute programming defines the absolute coordinate values of Point R;
Q_: Offset of the bottom of the hole;
P_: Pause time in the bottom of the hole;
F_: Cutting feed rate;
K_: Number of times of precision boring

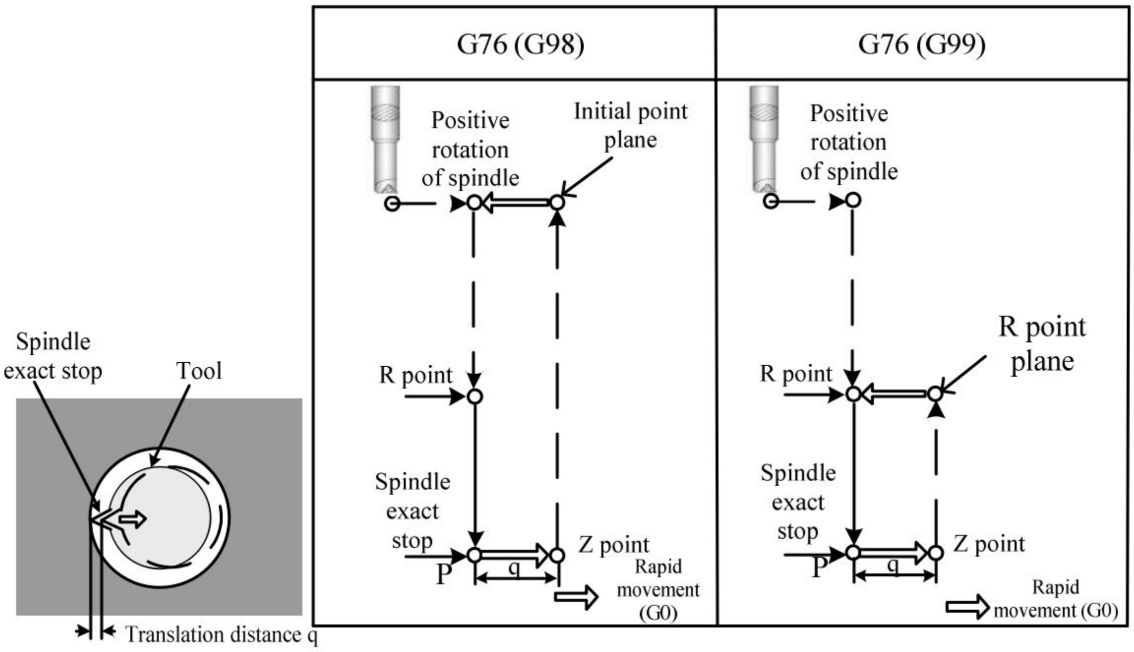



Fig. **4-4-13-1**

When the tool reaches the bottom of the hole, the spindle will stop at a fixed rotation position and the tool will retract in the opposite direction of the tool tip. This ensures that the machined surface is not damaged to achieve precise and efficient boring. The retraction distance is specified by parameter Q. The retraction direction and retraction axis are specified by Bit Parameters N0:42#4 and N0:42#5. And Q must be a positive value. Even if a negative value is used, the negative sign does not work. The offset of Q at the bottom of the hole is the modal value that is stored in the fixed cycle and must be specified carefully. The reason is that it is also used as the cutting depth of G73 and G83.

Rotate the spindle with the auxiliary function M code before G76 is specified.

When G76 and the M code are specified by the same program segment, the M code is executed during positioning of the first hole, and then the system deals with the next activity. When repetition times K is specified, the M code is executed only for the first hole, and not for subsequent holes.

**Note:** In the current version, M00, M01, M02, M06, M30, M98, and M99 are post-program executed, and the above M code will be executed after the current segment is executed.

Tool length compensation: When the tool length compensation G43, G44 or G49 shares the same program segment with the fixed cycle command, add or cancel the offset when positioning to Point R; in the fixed cycle mode, when the tool compensation G43, G44 or G49 is alone in a program segment, the system can add or cancel the offset in real time.

Shaft switch: The fixed cycle must be canceled before the drilling shaft is changed.

Boring: No boring is performed in program segments without X, Y, Z or other axis.

**Example:**

M3 S500; the spindle starts rotating

G90 G99 G76 X300 Y-250 Z-150 R-100 Q5 P1000 F120; Positioning, Hole 1 boring, then return to Point R, orientation in the bottom of the hole, movement for 5 mm and pause at the bottom for 1s

Y-550; Positioning, Hole 2 boring, and then return to Point R

Y-750; Positioning, Hole 3 boring, and then return to Point R

X1000; Positioning, Hole 4 boring, and then return to Point R

Y-550; Positioning, Hole 5 boring, and then return to Point R

G98 Y-750; Positioning, Hole 6 boring, and then return to the initial position plane

G80 G28 G91 X0 Y0 Z0; Return to reference point

M5; The spindle stops rotating

**Restrictions:** When the G76 command and the G codein Group 01 of the same program segment are used, the system will execute the final modal code.

Tool radius compensation: In this fixed cycle command, the tool radius compensation will be ignored because the command function does not require tool radius compensation.

**Note:** In this command, the feed shaft and the feed direction are fixed, and the rotation of the G68 coordinate system has no impact on the feed direction.

## 4.4.13 **Boring Cycle (G85)**

**Format: G85 X_ Y_ Z_ R_ F_ K_**

**Function:** This cycle is used for boring.

**Description:**

X_Y_: Hole positioning data;

Z_: Incremental programming defines the distance from Point R to the bottom of the hole; absolute programming defines the absolute coordinate values of the bottom of the hole;

R_: Incremental programming defines the distance from the initial point plane to Point R; absolute programming defines the absolute coordinate values of Point R;

F_: Cutting feed rate;

K_: Repetition times

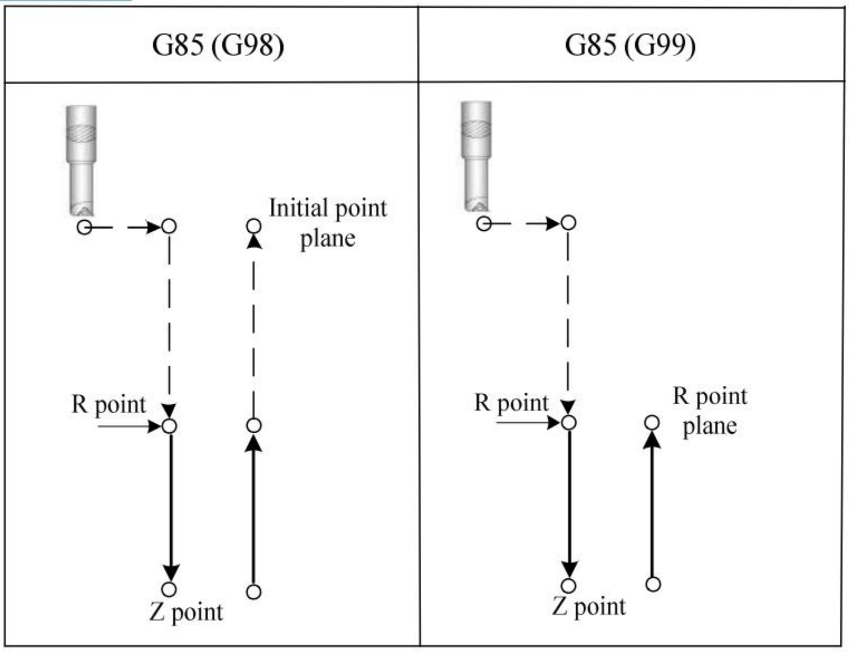



Fig. **4-4-14-1**

After positioning along the X and Y axis, move quickly to Point R, then perform boring from Point R to Point Z. When the bottom of the hole is reached, perform cutting feed and then return to Point R.

Rotate the spindle with the auxiliary function M code before G85 is specified.

When G85 and the M code are specified by the same program segment, the M code is executed during positioning of the first hole, and then the system deals with the next activity.

When repetition times K is specified, the M code is executed only for the first hole, and not for subsequent holes.

**Note:** In the current version, M00, M01, M02, M06, M30, M98, and M99 are post-program executed, and the above M code will be executed after the current segment is executed.

Tool length compensation: When the tool length compensation G43, G44 or G49 shares the same program segment with the fixed cycle command, add or cancel the offset when positioning to Point R; in the fixed cycle mode, when the tool compensation G43, G44 or G49 is alone in a program segment, the system can add or cancel the offset in real time.

Shaft switch: The fixed cycle must be canceled before the drilling shaft is changed.

Boring: No boring is performed in program segments without X, Y, Z or other axis.

**Example:**

M3 S100; The spindle starts rotating

G90 G99 G85 X300 Y-250 Z-150 R-120 F120; positioning, Hole 1 boring, and then return to Point R

Y-550; Positioning, Hole 2 boring, and then return to Point R

Y-750; Positioning, Hole 3 boring, and then return to Point R

X1000; Positioning, Hole 4 boring, and then return to Point R

Y-550; Positioning, Hole 5 boring, and then return to Point R

G98 Y-750; Positioning, Hole 6 boring, and then return to the initial position plane

G80;

G28 G91 X0 Y0 Z0; Return to reference point

M5; The spindle stops rotating

M30;

**Restrictions:** When the G85 command and the G code in Group 01 of the same program segment are used, the system will execute the final modal code.

**Tool radius compensation:** In this fixed cycle command, the tool radius compensation will be ignored because the command function does not require tool radius compensation.

# 4.4.14 Boring Cycle (G86)

**Format: G86 X_ Y_ Z_ R_ F_ K_;**
**Function:** This cycle code is used for boring cycle. (No pause is required at the bottom of the hole)
**Description:**

X_Y_: Hole positioning data;
Z_: Incremental programming defines the distance from Point R to the bottom of the hole; absolute programming defines the absolute coordinate values of the bottom of the hole;
R_: Incremental programming defines the distance from the initial point plane to Point R; absolute programming defines the absolute coordinate values of Point R;
F_: Cutting feed rate;
K_: Repetition times

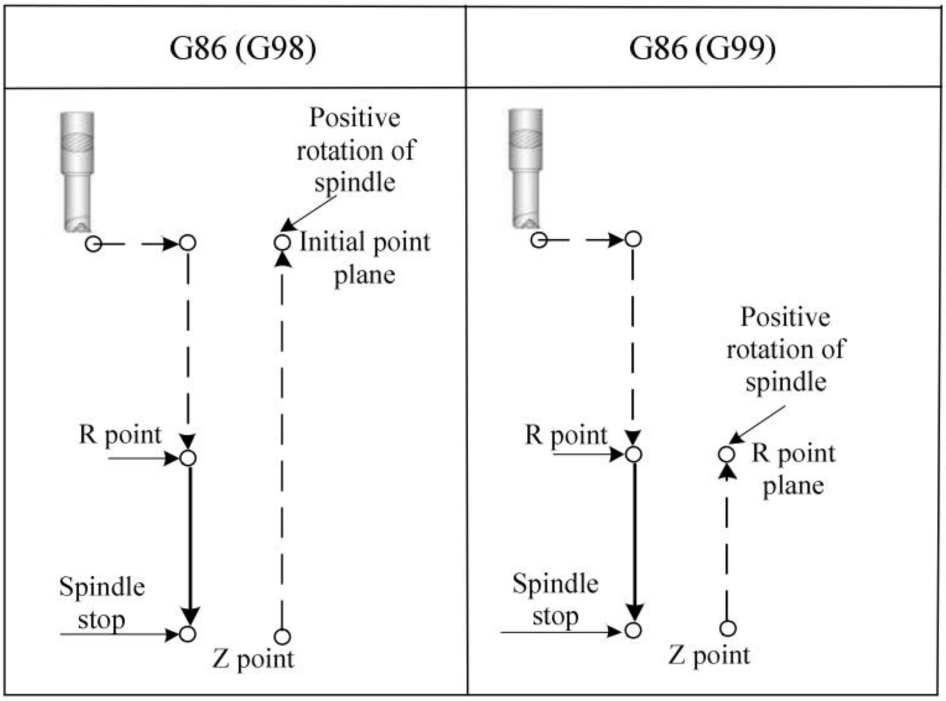



Fig. **4-4-15-1**

After positioning along the X and Y axis, move quickly to Point R, and then perform boring from Point R to Point Z. When the spindle stops at the bottom of the hole, the tool retracts with rapid movement.
Rotate the spindle with the auxiliary function M code before G86 is specified.
When G86 and the M code are specified by the same program segment, the M code is executed during positioning of the first hole, and then the system deals with the next activity.
When repetition times K is specified, the M code is executed only for the first hole, and not for subsequent holes.

**Note:** In the current version, M00, M01, M02, M06, M30, M98, and M99 are post-program executed, and the above M code will be executed after the current segment is executed.

Tool length compensation: When the tool length compensation G43, G44 or G49 shares the same program segment with the fixed cycle command, add or cancel the offset when positioning to Point R; in the fixed cycle mode, when the tool compensation G43, G44 or G49 is alone in a program segment, the system can add or cancel the offset in real time.
Shaft switch: The fixed cycle must be canceled before the drilling shaft is changed.
Boring: No boring is performed in program segments without X, Y, Z or other axis.

**Example:**

M3 S2000; The spindle starts rotating
G90 G99 G86 X300 Y-250 Z-150 R-100 F120; Positioning, Hole 1 boring, and then return to Point R
Y-550; Positioning, Hole 2 boring, and then return to Point R
Y-750; Positioning, Hole 3 boring, and then return to Point R

X1000; Positioning, Hole 4 boring, and then return to Point R
Y-550; Positioning, Hole 5 boring, and then return to Point R
G98 Y-750; Positioning, Hole 6 boring, and then return to the initial position plane
G80;
G28 G91 X0 Y0 Z0; Return to reference point
M5; The spindle stops rotating
M30;

**Restrictions:** When the G86 command and the G code (G00 to G03, G60 are modal codes (when the position parameter NO: 48#0 is 1)) in Group 01 of the same program segment are used, the system will execute the final modal code.

**Tool radius compensation:** In this fixed cycle command, the tool radius compensation will be ignored because the command function does not require tool radius compensation.

## 4.4.15 **Boring Cycle and Back Boring Cycle (G87)**

**Format: G87 X_Y_Z_R_Q_P_ F_;**

**Function:** This cycle performs precision boring

**Description:**

X_Y_: Hole positioning data;
Z_: Incremental programming defines the distance from the R point to Pint Z; absolute programming defines the absolute coordinate values of Point Z;
R_: Incremental programming defines the distance from the initial point plane to Point R; absolute programming defines the absolute coordinate values of Point R; (the bottom of the hole)
Q_: Offset of the bottom of the hole;
P_: Pause time in the bottom of the hole;
F_: Cutting feed rate;

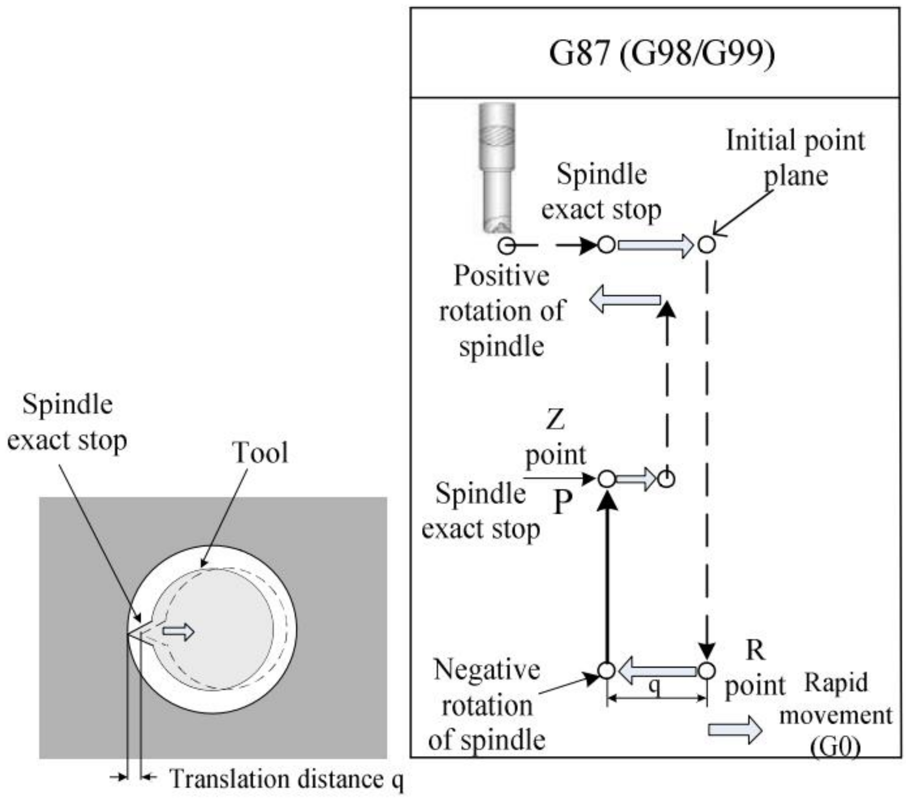



Fig. **4-4-16-1**

After positioning along the X and Y axis, the spindle stops the tool when oriented, moves in the opposite direction of the tool tip, and moves at the feed speed at Point R at the bottom of the hole. Then the tool moves in the direction of the tool tip while the spindle rotates reversely along the positive boring on the Z axis to Point Z. When oriented again at Point Z, the spindle stops at a fixed rotational position and the tool retracts in the opposite direction of the tool tip and returns to the initial plane. The tool shifts from the spindle and rotates forward in the direction of the tool tip and performs the machining in the next program segment.

The retraction distance is specified by Parameter Q. And Q must be a positive value. Even if a negative value is used, the negative sign does not work. The offset of Q at the bottom of the hole is a modal value that is stored in the fixed cycle and must be specified carefully, because it is also used as the cutting depth for G73 and G83.

Rotate the spindle with the auxiliary function M code before G87 is specified.

When G87 and the M code are specified by the same program segment, the M code is executed during positioning of the first hole, and then the system deals with the next drilling activity.

When repetition times K is specified, the M code is executed only for the first hole, and not for subsequent holes.

**Note:** In the current version, M00, M01, M02, M06, M30, M98, and M99 are post-program executed, and the above M code will be executed after the current segment is executed.

Tool length compensation: When the tool length compensation G43, G44 or G49 shares the same program segment with the fixed cycle command, add or cancel the offset when positioning to Point R; in the fixed cycle mode, when the tool compensation G43, G44 or G49 is alone in a program segment, the system can add or cancel the offset in real time.

The fixed cycle can only be performed on the G17 plane.

Boring: No boring is performed in program segments without X, Y, Z or other auxiliary axis.

**Hint:** When programming the back boring cycle, remember that the Z value and the R value should be specified. In general, the Z position here is above the R position. Otherwise, the system will give an alarm.

**Example:**

M3 S500; the spindle starts rotating

G90 G99 G87 X300 Y-250 Z-120 R-150 Q5 P1000 F120; (Positioning, Hole 1 boring, oriented at the initial position and then offset by 5mm and pause at Point Z for 1 second)

Y-550; Positioning, Hole 2 boring, and then return to the initial position plane

Y-750; Positioning, Hole 3 boring, and then return to the initial position plane

X1000; Positioning, Hole 4 boring, and then return to the initial position plane

Y-550; Positioning, Hole 5 boring, and then return to the initial position plane

G98 Y-750; Positioning, Hole 6 boring, and then return to the initial position plane

G80 G28 G91 X0 Y0 Z0; Return to reference point

M5; The spindle stops rotating

M30;

**Restrictions:** When the G87 command and the G code (G00 to G03, G60 are modal codes (when the position parameter NO: 48#0 is 1)) in Group 01 of the same program segment are used, the system will execute the final modal code.

**Tool radius compensation:** In this fixed cycle command, the tool radius compensation will be ignored because the command function does not require tool radius compensation.

**Note:** In this command, the feed shaft and the feed direction are fixed, and the rotation of the G68 coordinate system has no impact on the feed direction.

## 4.4.16 **Boring Cycle (G88)**

**Format: G88 X_Y_Z_R_ P_F_**

**Function:** This cycle is used for boring.

**Description:**

X_Y_: Hole positioning data;

Z_: Incremental programming defines the distance from Point R to the bottom of the hole; absolute programming defines the absolute coordinate values of the bottom of the hole;

R_: Incremental programming defines the distance from the initial point plane to Point R; absolute programming defines the absolute coordinate values of Point R;

P_: Pause time in the bottom of the hole;

F_: Cutting feed rate.

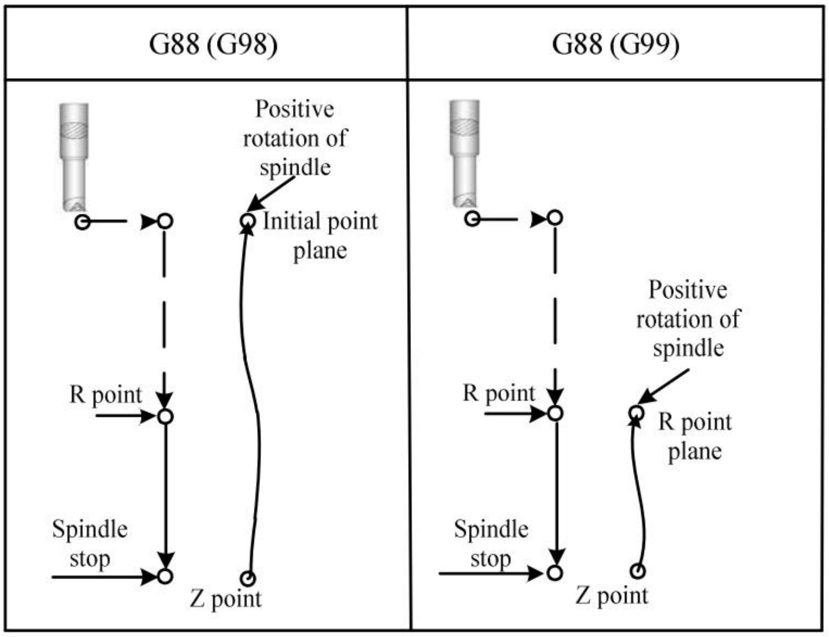



Fig. **4-4-17-1**

After positioning along the X and Y axis, quickly move to Point R, and then perform boring from Point R to Point Z. When the boring is completed, the pause is executed, and then the spindle stops, and the tool is manually returned from Point Z point at the bottom to Point R (in case of G99) or the initial point (in case of G98), and the spindle rotates forward.

Rotate the spindle with the auxiliary function M code before G88 is specified.

When G88 and the M code are specified by the same program segment, the M code is executed during positioning of the first hole, and then the system deals with the next drilling activity.

When repetition times K is specified, the M code is executed only for the first hole, and not for subsequent holes.

**Note:** In the current version, M00, M01, M02, M06, M30, M98, and M99 are post-program executed, and the above M code will be executed after the current segment is executed.

P is a modal code; the minimum value of the parameter is set by Data Parameter P296, while the maximum value is set by P297. When the P value is less than that set by P296, it will run at the minimum value; when the P value is greater than that set by P297, it will run at the maximum value.

Tool length compensation: When the tool length compensation G43, G44 or G49 shares the same program segment with the fixed cycle command, add or cancel the offset when positioning to Point R; in the fixed cycle mode, when the tool compensation G43, G44 or G49 is alone in a program segment, the system can add or cancel the offset in real time.

Shaft switch: The fixed cycle must be canceled before switching the boring shaft.

Boring: No boring is performed in program segments without X, Y, Z or other auxiliary axis.

**Example:**

M3 S2000; Spindle starts rotating
G90 G99 G88 X300 Y-250 Z-150 R-100 P1000 F120; Positioning, Hole 1 boring, and then return to Point R
Y-550; Positioning, Hole 2 boring, and then return to Point R
Y-750; Positioning, Hole 3 boring, and then return to Point R
X1000; Positioning, Hole 4 boring, and then return to Point R
Y-550; Positioning, Hole 5 boring, and then return to Point R
G98 Y-750; Positioning, Hole 6 boring, and then return to the initial position plane
G80 G28 G91 X0 Y0 Z0; Return to reference point
M5; The spindle stops rotating

**Restrictions:** When the G88 command and the G code in Group 01 of the same program segment are used, the system will execute the final modal code.

**Tool radius compensation:** In this fixed cycle command, the tool radius compensation will be ignored because the command function does not require tool radius compensation.

## 4.4.17 **Boring Cycle (G89)**

**Format: G89 X_ Y_ Z_ R_ P_ F_ K_**
**Function:** This cycle is used for boring.
**Description:**

X_Y_: Hole positioning data;
Z_: Incremental programming defines the distance from Point R to the bottom of the hole; absolute programming defines the absolute coordinate values of the bottom of the hole;
R_: Incremental programming defines the distance from the initial point plane to Point R; absolute programming defines the absolute coordinate values of Point R;
P_: Pause time in the bottom of the hole;
F_: Cutting feed rate;
K_: Repetition times

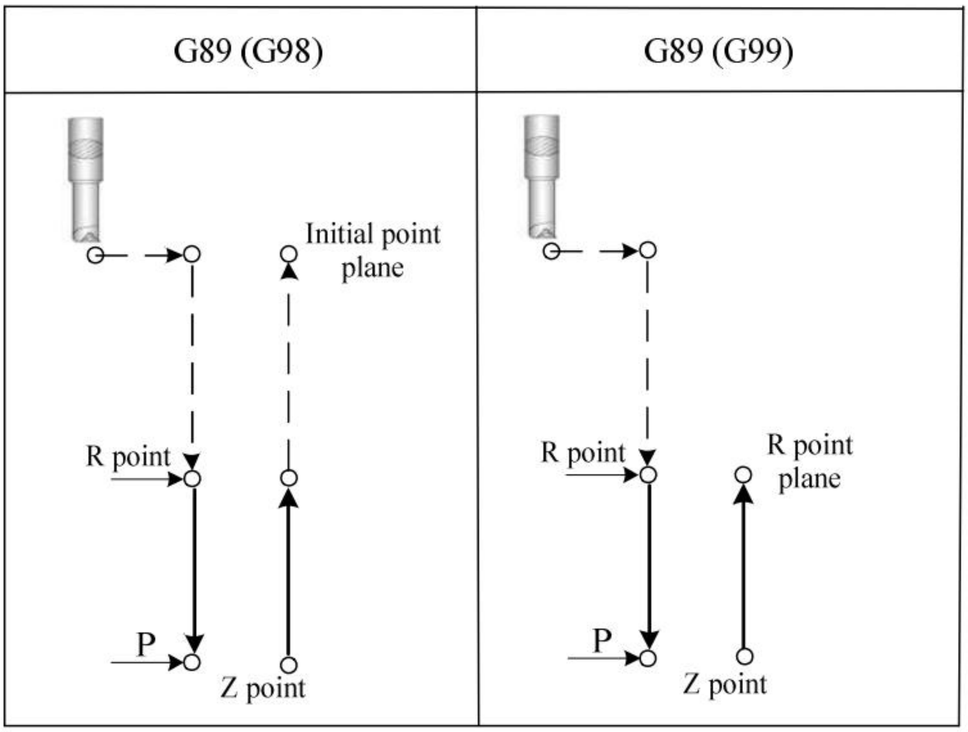



Fig. **4-4-18-1**

This cycle is almost identical to G85, except that this cycle performs a pause at the bottom of the hole.
Rotate the spindle with the auxiliary function M code before G89 is specified.
When G89 and the M code are specified by the same program segment, the M code is executed during positioning of the first hole, and then the system deals with the next drilling activity.
When repetition times K is specified, the M code is executed only for the first hole, and not for subsequent holes.

**Note:** In the current version, M00, M01, M02, M06, M30, M98, and M99 are post-program executed, and the above M code will be executed after the current segment is executed.

P is a modal code; the minimum value of the parameter is set by Data Parameter P337, while the maximum value is set by P338. When the P value is less than that set by P337, it will run at the minimum value; when the P value is greater than that set by P338, it will run at the maximum value.
Tool length compensation: When the tool length compensation G43, G44 or G49 shares the same program segment with the fixed cycle command, add or cancel the offset when

positioning to Point R; in the fixed cycle mode, when the tool compensation G43, G44 or G49 is alone in a program segment, the system can add or cancel the offset in real time.
Shaft switch: The fixed cycle must be canceled before switching the boring shaft.
Boring: No boring is performed in program segments without X, Y, Z or other auxiliary axis.

**Example:**

M3 S100; The spindle starts rotating
G90 G99 G89 X300 Y-250 Z-150 R-120 P1000 F120; positioning, Hole 1 boring, then return to Point R and pause at the bottom of the hole for 1 second
Y-550; Positioning, Hole 2 boring, and then return to Point R
Y-750; Positioning, Hole 3 boring, and then return to Point R
X1000; Positioning, Hole 4 boring, and then return to Point R
Y-550; Positioning, Hole 5 boring, and then return to Point R
G98 Y-750; Positioning, Hole 6 boring, and then return to the initial position plane
G80;
G28 G91 X0 Y0 Z0; Return to reference point
M5; The spindle stops rotating
M30;

**Restrictions:** When the G89 command and the G codein Group 01 of the same program segment are used, the system will execute the final modal code.

**Tool radius compensation:** In this fixed cycle command, the tool radius compensation will be ignored because the command function does not require tool radius compensation.

## 4.4.18 **Left Rigid Tapping (G74)**

**Format: G74 X_Y_Z_R_P_F_K_**

**Function:** In rigid mode, the spindle motor works as a servo motor, and this code can realize left high-speed high-precision tapping.

**Description:**

X_Y_: Hole positioning data
Z_: Incremental programming defines the distance from Point R to the bottom of the hole; absolute programming defines the absolute coordinate values of the bottom of the hole;
R_: Incremental programming defines the distance from the initial point plane to Point R; absolute programming defines the absolute coordinate values of Point R;
P_: Pause time at the bottom of the hole or pause time at Point R during backing.
F_: Cutting feed rate.
K_: Repetition times (It should be specified when needed)

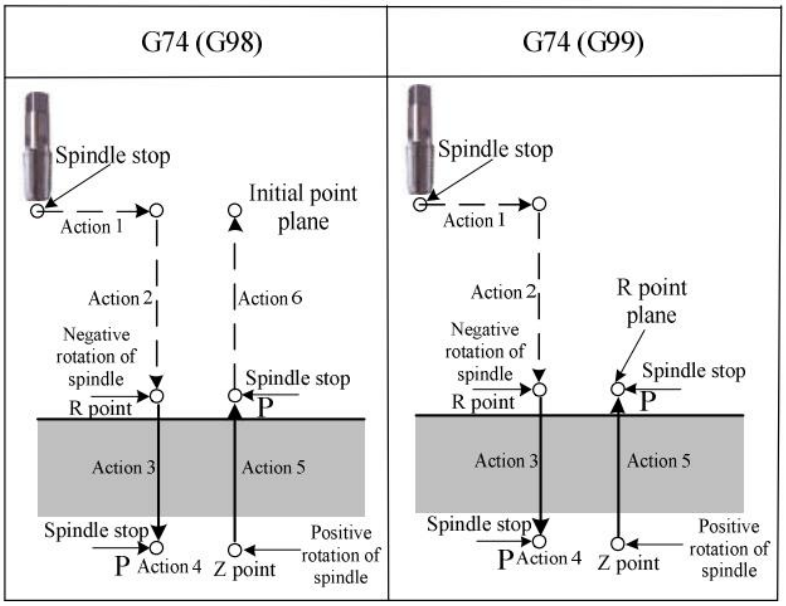



Fig. **4-4-19-1**

After positioning along the X and Y axis, quickly move from the Z axis to Point R, and execute G74 to make the spindle rotate reversely. The tapping is performed from Point R to Point Z. When the tapping is completed, the spindle stops and a pause is executed. And then the spindle rotates reversely and the tool returns to Point R. When the spindle stops, perform a quick move to the initial position. The feedrate override and the spindle override are considered to be 100% when the tapping is being performed.

**Rigid mode:** In the position mode (position parameter NO: 46#1 set to 1, K parameter NO: 7#7 set to 1), the rigid mode can be specified when M29 S***** is specified before the tapping code.

**Tool length compensation:** When the tool length compensation G43, G44 or G49 shares the same program segment with the fixed cycle command, add or cancel the offset when positioning to Point R; in the fixed cycle mode, when the tool compensation G43, G44 or G49 is alone in a program segment, the system can add or cancel the offset in real time.

**Thread lead:** In the feed mode per minute, the relationship between the thread lead and the feed rate and the spindle speed:

Feed rate F = screw tap thread pitch × spindle speed S

For example: For tapping of M12×1.5 threaded holes on a component, the following parameters can be used:

S500=500 r/min; F=1.5×500=750 mm/min;

In case of multiple thread, multiply by the number of heads to get the F value.

In the feed mode per rotation, the thread lead is equal to the feed rate.

**For example:**

| Feed mode per minute: | Feed mode per rotation: |
|---|---|
| Spindle speed 1000 r/min; | Spindle speed 1000 r/min; |
| Thread lead 1.0 mm; | Thread lead 1.0 mm; |
| Then Z-axis feed rate = 1000*1 = 1000 mm/min; | Then Z-axis feed rate = thread lead 1 = 1mm/r; |
| G94 feed mode per minute | G95 Feed mode per rotation |
| G00 X120 Y100; Positioning | G00 X120 Y100; Positioning |
| M29 S1000; Specify rigid mode | M29 S1000; Specify rigid mode |
| G74 Z-100 R-20 F1000; left rigid tapping | G74 Z-100 R-20 F1; Left rigid tapping |
| G80 Cancel the tapping cycle | G80 Cancel the tapping cycle |
| G28 G91 X0 Y0 Z0 Return to reference point | G28 G91 X0 Y0 Z0 Return to reference point |
| M30 Program ends | M30 Program ends |

## 4.4.19 **Right Rigid Tapping (G84)**

**Format: G84 X_Y_Z_R_P_F_K_**

**Function:** In rigid mode, the spindle motor works as a servo motor and can realize high-speed high-precision tapping. It can be guaranteed that the starting position of tapping is consistent without changing Point R. That is, the tapping is repeatedly performed in one position, and the thread will not be buckled in a mess or rotted.

**Description:**

X_Y_: Hole positioning data;
Z_: Incremental programming defines the distance from Point R to the bottom of the hole; absolute programming defines the absolute coordinate values of the bottom of the hole;
R_: Incremental programming defines the distance from the initial point plane to Point R; absolute programming defines the absolute coordinate values of Point R;
P_: Pause time at the bottom of the hole or pause time at Point R during backing;
F_: Cutting feed rate;
K_: Repetition times (It should be specified when needed)

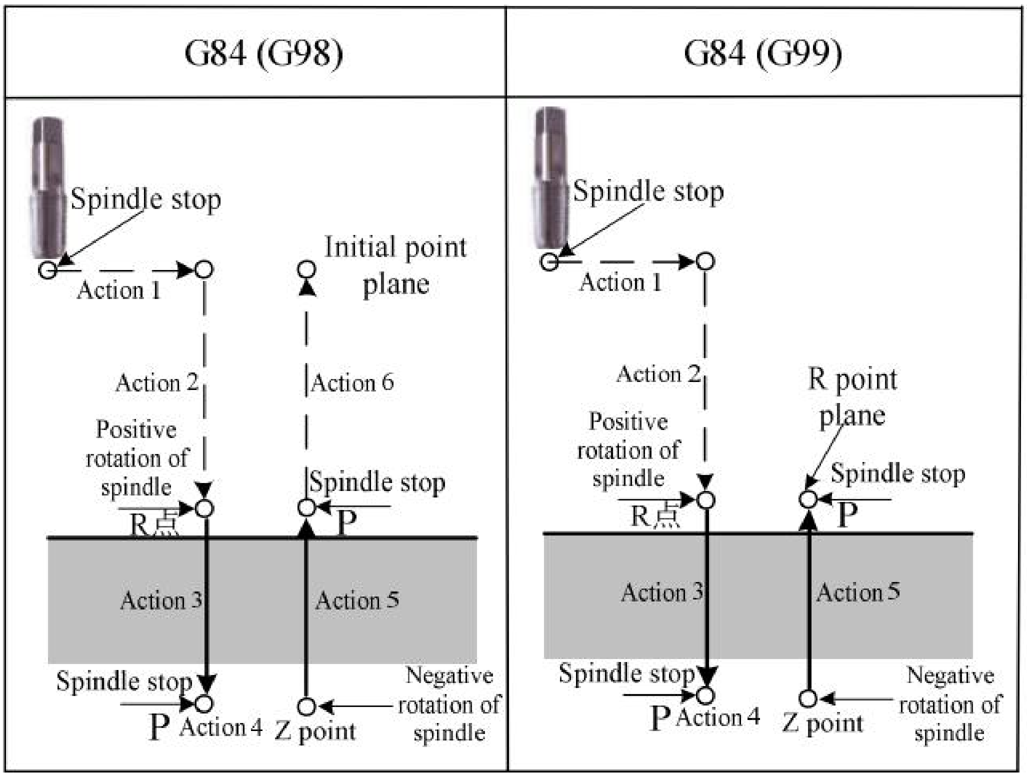



Fig. **4-4-20-1**

After positioning along the X and Y axis, quickly move from the Z axis to Point R, and execute G84 to make the spindle rotate reversely. The tapping is performed from Point R to Point Z. When the tapping is completed, the spindle stops and a pause is executed. And then the spindle rotates reversely and the tool returns to Point R, and the spindle stops. Then perform a quick move to the initial position. The feedrate override and the spindle override are considered to be 100% when the tapping is being performed.

**Tool length compensation:** When the tool length compensation G43, G44 or G49 shares the same program segment with the fixed cycle command, add or cancel the offset when positioning to Point R; in the fixed cycle mode, when the tool compensation G43, G44 or G49 is alone in a program segment, the system can add or cancel the offset in real time.

**Thread lead:** In the feed mode per minute, the relationship between the thread lead and the feed rate and the spindle speed:

Feed rate F = screw tap thread pitch × spindle speed S

For example: For tapping of M12×1.5 threaded holes on a component, the following parameters can be used:

S500=500 r/min; F=1.5×500=750 mm/min;

In case of multiple thread, multiply by the number of heads to get the F value.

In the feed mode per rotation, the thread lead is equal to the feed rate.

**For example:** Feed mode per minute: Feed mode per rotation:

| Feed mode per minute | Feed mode per rotation |
|---|---|
| Spindle speed 1000 r/min; | Spindle speed 1000 r/min; |
| Thread lead 1.0 mm; | Thread lead 1.0 mm; |
| Then Z-axis feed rate = 1000*1 = 1000 mm/min; | Then Z-axis feed rate = thread lead 1 = 1mm/r; |
| G94 feed mode per minute | G95 feed mode per rotation |
| G00 X120 Y100; Positioning | G00 X120 Y100; Positioning |
| M29 S1000; Specify rigid mode | M29 S1000; Specify rigid mode |
| G84 Z-100 R-20 F1000; Right rigid tapping | G84 Z-100 R-20 F1; Right rigid tapping |
| G80 Cancel the tapping cycle | G80 Cancel the tapping cycle |
| G28 G91 X0 Y0 Z0 Return to reference point | G28 G91 X0 Y0 Z0 Return to reference point |
| M30 Program ends | M30 Program ends |

**Restrictions:**

**G code:** When the G84 command and the G code in Group 01 of the same program segment are used, the system will execute the final modal code.

**M code:** Rotate the spindle with the auxiliary function M code before G84 is specified. If the spindle rotates without command, the system will automatically adjust to anti-clockwise rotation based on the current spindle command speed in the R plane.

When G84 and the M code are specified by the same program segment, the M code is executed during positioning of the first hole, and then the system deals with the next drilling activity.

When repetition times K is specified, the M code is executed only for the first hole, and not for subsequent holes.

**Note:** In the current version, M00, M01, M02, M06, M30, M98, and M99 are post-program executed, and the above M code will be executed after the current segment is executed.

**S command:** If the specified spindle speed exceeds the maximum spindle speed during tapping (Data Parameter P257: spindle upper limit speed during tapping cycle), the system will send an alarm; the maximum spindle speed during rigid tapping is set by Data Parameter P294 - P296.

**F command:** If the specified F value exceeds the upper limit of the cutting feed rate (data parameter: the upper limit is set by P96), the upper limit shall prevail.

**P command: P** is a modal code; the minimum value of the parameter is set by Data Parameter **P296**, while the maximum value is set by **P297**. When the P value is less than that set by **P296**, it will run at the minimum value; when the P value is greater than that set by **P297**, it will run at the maximum value.

**Shaft switch:** The fixed cycle must be canceled before switching the tapping shaft. If the tapping shaft is changed in the rigid mode, the system will send No. 206 alarm.

**Override:** During rigid tapping, the feedrate override and spindle speed override are 100% by default, and the machine tool will not stop working when the feed hold button and reset button are pressed until the return activity is completed.

**Tool radius compensation:** In this fixed cycle command, the tool radius compensation will be ignored because the command function does not require tool radius compensation.

**Program restart:** In the tapping cycle, the program restart function is invalid.

**Note:** In the process of soft tapping, rigid tapping or deep-hole rigid tapping, it is necessary to cancel the constant surface cutting speed with G97 first, or otherwise there will be incorrect thread or broken tap.

## 4.4.20 **Deep-Hole Tapping (Chip Removal) Cycle**

**Format: G84（or G74） X_Y_Z_R_P_Q_F_K_**

**Function:** In deep-hole tapping, several advances of the tool are performed until the bottom of the hole is reached.

**Description:**

X_Y_: Hole positioning data;

Z_: Incremental programming defines the distance from Point R to the bottom of the hole; absolute programming defines the absolute coordinate values of the bottom of the hole;
R_: Incremental programming defines the distance from the initial point plane to Point R; absolute programming defines the absolute coordinate values of Point R;
P_: Pause time at the bottom of the hole or pause time at Point R during backing;
Q_: Cutting depth of each cutting feed;
F_: Cutting feed rate;
V_: Backing distance d, which will be set by Data Parameter **P300** if not specified;
K_: Repetition times (specified when needed).

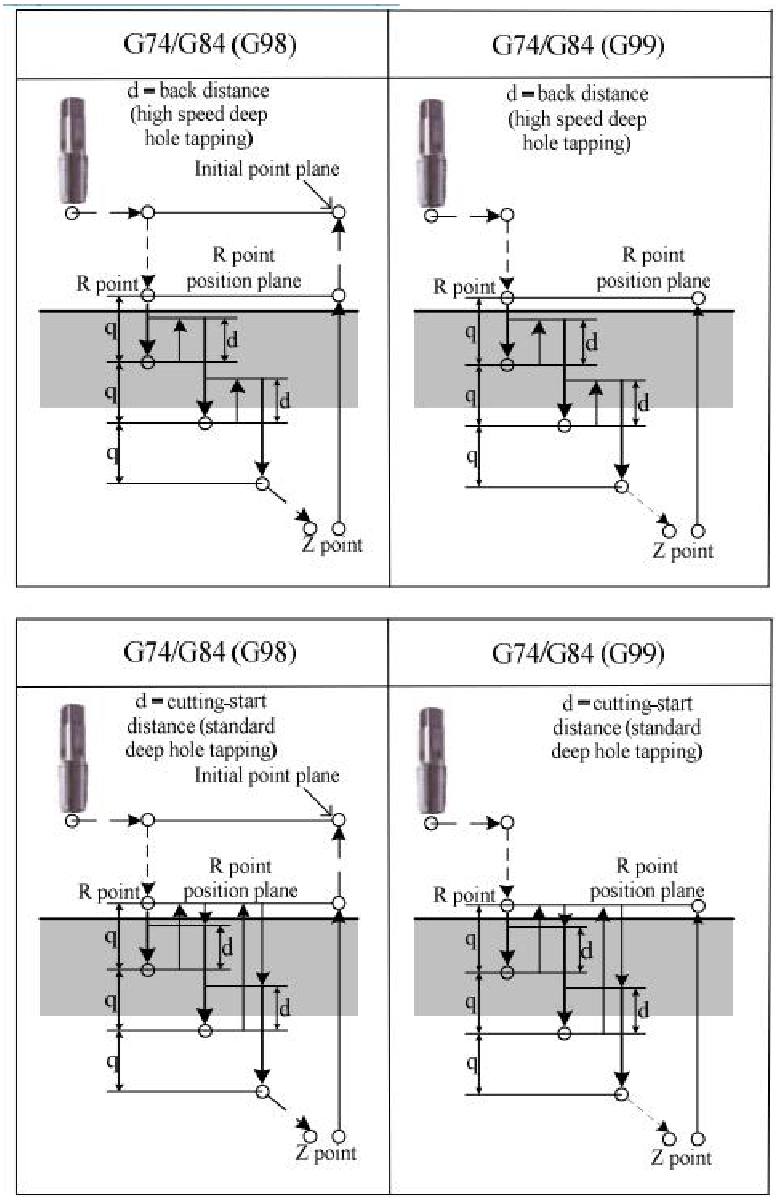



Fig. **4-4-21-1**

There are two ways of deep hole rigid tapping cycle: high-speed deep hole tapping cycle and standard deep hole tapping cycle.

When parameter - [Rigid Tapping] P002=1, it is a high-speed deep hole tapping cycle: after positioning along the X and Y axes, perform a quick movement to point R, perform cutting from point R with a feed depth Q (the depth of each cutting feed), and then retract the tool by a distance d (set by parameter - [Rigid Tapping] P008). When reaching point Z, the spindle stops and then rotates in the opposite direction to retreat.

When the parameter - [rigid tapping] P002=0, it is the standard deep hole tapping cycle: after positioning along the X and Y axes, perform a quick movement to the R point, perform cutting from the R point with a feed depth Q (the depth of each cutting feed), and then return to the R point. From the R point to a position that is d (determined by the parameter - [rigid tapping] P008) away from the end point of the previous cutting, perform cutting again at the value of cutting speed F; When reaching point Z, the spindle stops and then rotates backwards in the opposite direction.

**Limitations:**

**F:** If the specified F value exceeds the upper limit of the cutting feed rate, an alarm will be triggered.

**S:** If the speed is higher than the maximum speed of the specified level, an alarm will be triggered.

**Cancel:** Group 01 G code (G00 to G03) and G84 (or G74) cannot be specified in the same program segment, otherwise G84 (or G74) will be canceled.

**Tool offset:** The tool radius offset is ignored during the fixed cycle positioning process.

**Program restart:** Program restart is invalid during rigid tapping.

## 4.4.21 Fixed cycle cancellation G80

**Format: G80**

**Function:** Cancel fixed loop.

**explain:**

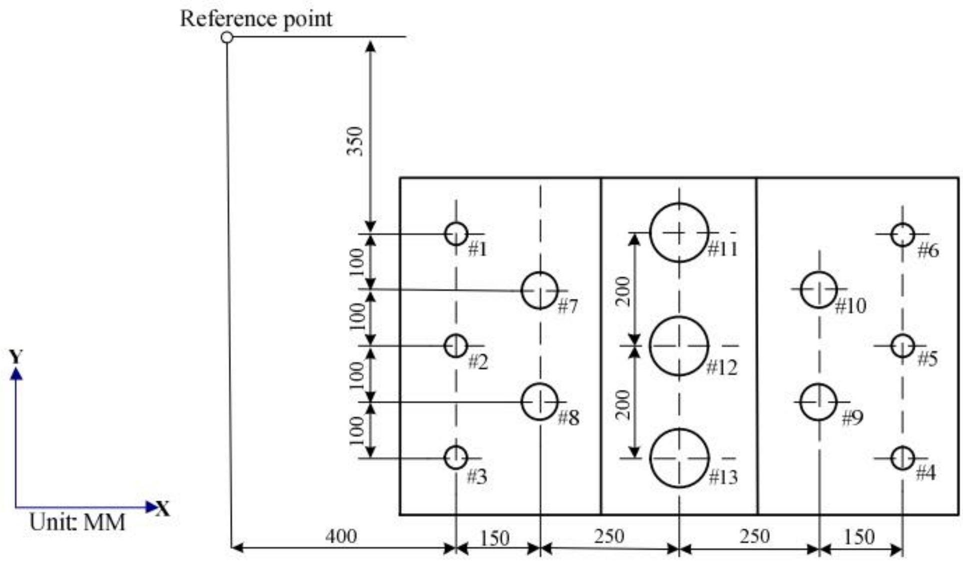
Cancel all fixed loops and perform normal operations. R and Z points are also cancelled. Other drilling and boring data are also cancelled and cleared.

**Example:**

| | |
|---|---|
| M3 S100； | The spindle begins to rotate |
| G90 G99 G88 X300 Y-250 Z-150 R-120 F120； | Positioning, boring 1 hole, and then returning to point R |
| Y-550； | Positioning, boring 2 holes, and then returning to point R |
| Y-750； | Positioning, boring 3 holes, and then returning to point R |
| X1000； | Positioning, boring 4 holes, and then returning to point R |
| Y-550； | Positioning, boring 5 holes, and then returning to point R |
| G98 Y-750； | Positioning, boring 6 holes, and then returning to the initial position plane |
| G80； | |
| G28 G91 X0 Y0 Z0； | Return to the reference point and cancel the fixed loop |
| M5； | The spindle stops rotating |

**Example:**

The following uses tool length compensation to comprehensively illustrate the use of fixed cycles.




**# 1～ 6...Drill a diameter ofΦ10 holes**

**# 7～10...Drill a diameter of Φ20 holes**

**#11～13..Drill a diameter of Φ95 holes**

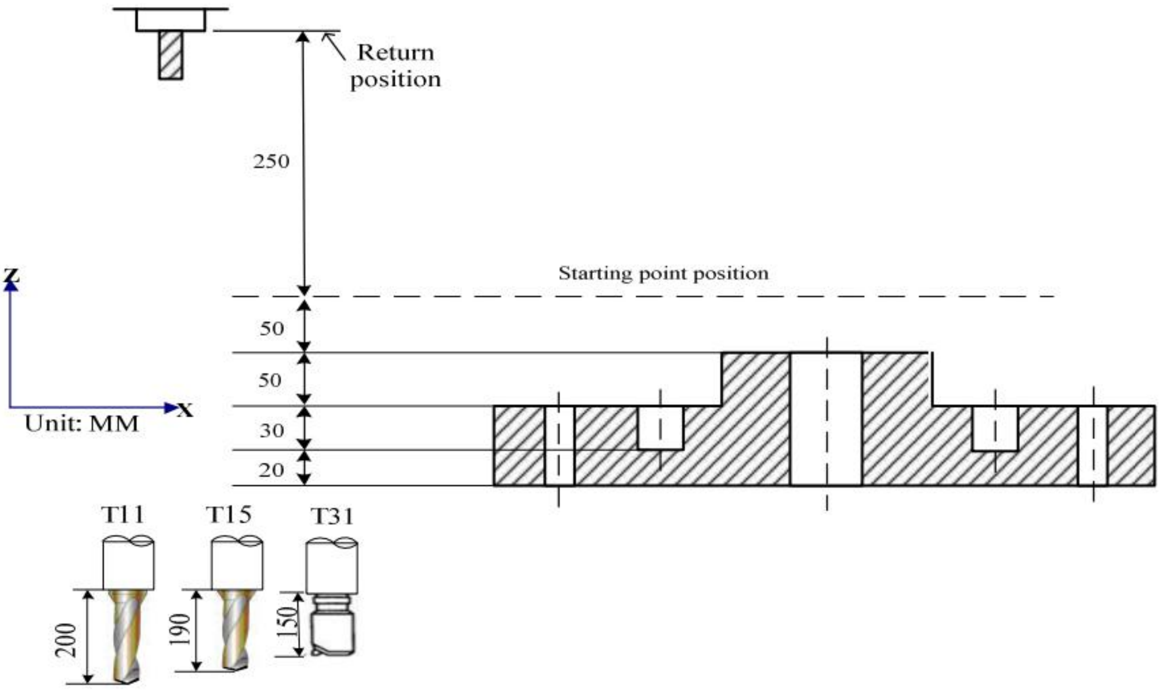



Fig. **4-4-22-1**

The offset values for offset 11, offset 15, and offset 31 are set to 200, 190, and 150, respectively. The program is as follows:

| | |
|---|---|
| N001 G92 X0 Y0 Z0 ； | The coordinate system is set at the reference point. |
| N002 G90 G00 Z250 T11 M6 ； | Change the tool. |

| | |
|---|---|
| N003 G43 Z0 H11 ； | Perform planar tool length compensation at the initial point. |
| N004 S300 M3 ； | The spindle is started. |

| | |
|---|---|
| N005 G99 G81 X400 Y-350 ；<br>Z-153 R-97 F120 ； | Process hole # 1 after positioning. |
| N006 Y-550 ； | Process hole # 2 after positioning and return to the R-point plane. |
| N007 G98 Y-750 ； | Process hole # 3 after positioning and return to the initial point plane. |
| N008 G99 X1200 ； | Process hole # 4 after positioning and return to the R-point plane. |
| N009 Y-550 ； | Process hole # 5 after positioning and return to the R point plane. |
| N010 G98 Y-350 ； | Process # 6 holes after positioning and return to the initial point plane. |
| N011 G00 X0 Y0 M5 ； | Return to reference point, spindle stopped. |
| N012 G49 Z250 T15 M6 ； | Cancel tool length compensation and replace the tool. |
| N013 G43 Z0 H15 ； | Initial point plane, tool length compensation. |
| N014 S200 M3 ； | The spindle is started. |
| N015 G99 G82 X550 Y-450 ；<br>Z-130 R-97 P30 F70 ； | After positioning, process hole # 7 and return to the R point plane. |
| N016 G98 Y-650 ； | After positioning, process hole # 8 and return to the R point plane. |
| N017 G99 X1050 ； | Process hole # 9 after positioning and return to the R point plane. |
| N018 G98 Y-450 ； | After positioning, process # 10 holes and return to the initial point plane. |
| N019 G00 X0 Y0 M5 ； | Return to reference point, spindle stopped. |
| N020 G49 Z250 T31 M6 ； | Cancel tool length compensation and replace the tool. |
| N021 G43 Z0 H31 ； | Initial point plane tool length compensation. |
| N022 S100 M3 ； | The spindle is started. |
| N023 G85 G99 X800 Y-350 ；<br>Z-153 R47 F50 ； | Process hole # 11 after positioning and return to the R-point plane. |
| N024 G91 Y-200 ；<br>Y-200 ； | Process holes # 12 and # 13 after positioning, and return to the R-point plane. |
| N025 G00 G90 X0 Y0 M5 ； | Return to reference point, spindle stopped. |
| N026 G49 Z0 ； | Cancel tool length compensation. |
| N027 M30 ； | Program stopped. |

# 4.5 **Tool compensation (G code)**

## 4.5.1 **Tool Length Compensation (G43, G44, G49)**

**Function:**

**G43:** specify the forward compensation of tool length.
**G44:** specify the reverse compensation of tool length.
**G49:** cancel the tool length compensation.

**Format:**

The system supports two tool length offset methods (A/B), and uses the position parameter **N0: 39#0** to set the tool length offset mode.

**Method A:**

$$\left.\begin{array}{l}G43\\G44\end{array}\right\}Z\_\ H\_\ ;$$

**Method B:**

G17 G43 Z_H；

G17 G44 Z_H；

G18 G43 Y_H；

G18 G44 Y_H；

G19 G43 X_H；

G19 G44 X_H；

Cancel the tool length offset mode: G49 or H0.

**Description:**

The above code is used to move the final position of specified axis command by an offset. The difference between the tool length value assumed during programming (usually set as the first tool) and the tool length value used in actual machining is preset in the offset memory, so there is no need to change the program, and it is possible to machine parts using tools of different lengths by only changing tool length compensation value.
G43 and G44 are used to specify different offset directions, and H code to specify the offset number.

1. Offset direction
   G43: Positive offset (most commonly used)
   G44: Negative offset
   Whether it is an absolute value code or an incremental value code, in case of G43, the coordinate value of final position of the specified axis movement command in the program is added to the offset specified by the H code (set in the offset memory); in case of G44, minus the offset specified by the H code, and then use the calculated result as the coordinate value of final position.
   G43 and G44 are modal G codes and valid before encountering other G codes in the same group.
2. Specification of depth offset
   A length offset number will be specified by the H code, and the depth offset corresponding to this offset number is added to or subtracted from the Z-axis movement code value in the program to form a new Z-axis movement code. The offset number can be specified from H00 to H255 as needed.
   The depth offset can be set within the following range:

**Table 4-5-1-1**

| | (input in mm) |
|---|---|
| Compensation quantity H | -999.999 mm～+999.999mm |

Offset Number 00, that is, the depth offset corresponding to H00 is 0 and cannot be set in the system.

**Note:** When the depth offset is changed due to change in the offset number, the old depth offset is directly replaced with the new one, instead of the new depth offset added to the old one.

For example:

H01........................ ................Depth offset 20
H02......................... ...............Depth offset 30
G90 G43 Z100 H01; ......... Move from Z to 120
G90 G43 Z100 H02; ......... Move from Z to 130

3. Effective order of the offset number

Once the length offset mode is established, the current offset number takes effect immediately, and when the offset number changes, the new offset value will immediately replace the old one. For example:

O×××××；
H01；
G43 Z10； (1) Offset Number H01 becomes effective
G44 Z20 H02； (2) Offset Number H02 becomes effective
H03； (3) Offset Number H03 becomes effective
G49； (4) Cancel the tool offset at the end of this segment
M30；

4. Cancel tool length compensation

Cancel tool compensation with G49 or H00. After the command G49, the system immediately cancels the tool length compensation; after the command H00, the compensation axis address or compensation command must be programmed, or otherwise the tool length compensation cannot be canceled.

**Note:** 1. Tool length offset method B: After two or more axes are executed, G49 is used to cancel the offset of all axes while H00 is only used to cancel the offset of the axis perpendicular to the specified plane.

2. It is recommended to add Z-axis movement code for establishment and cancellation of length compensation. Otherwise, the length compensation will be established or canceled with the current point. Therefore, when using G49, please make sure that the Z-axis is at a safe height to prevent collision or damage to workpiece.

5. Specific examples of tool length compensation

a) Tool length compensation (Holes 1, 2 and 3 machining)

b) H01=offset-4

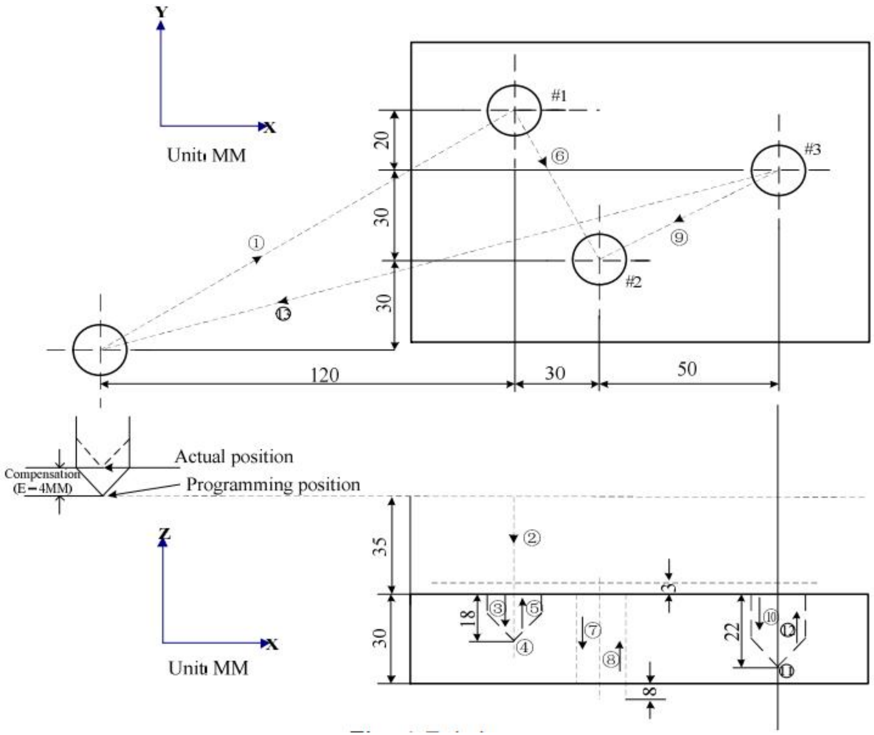



Fig. **4-5-1-1**

N1 G91 G00 X120 Y80 ; ....... (1)
N2 G43 Z-32 H01 ; .................. (2)
N3 G01 Z-21 F200 ; .................. ...(3)
N4 G04 P2000 ; ....................... . (4)
N5 G00 Z21 ; ...................... ......…(5)
N6 X30 Y-50 ; ............... ...............(6)
N7 G01 Z-41 F200 ; ........... ......…..(7)
N8 G00 Z41 ; ...................... ......…..(8)
N9 X50 Y30 ; ........................ …... (9)
N10 G01 Z-25 F100 ; ........... ......…(10)
N11 G04 P2000 ; ..................... ....…(11)
N12 G00 Z57 H00 ; ................. ... ....(12)
N13 X-200 Y-60 ; .............. ... . .........(13)
N14 M30 ;

## 4.5.2 **Tool Radius Compensation (G40/G41/G42)**

**Code format:**

**G41 D_ X_Y_;**
**G42 D_ X_Y_;**
**G40 X_Y_;**

**Function:**

G41: specify the left-side compensation in tool movement direction.
G42: specify the right-side compensation in tool movement direction.
G40: cancel the tool radius compensation.

**Description:**

1. Tool radius compensation function

As shown in the figure below, Workpiece A is cut with a tool of radius R; the tool center path is B as shown in the figure, and Path B is at a distance of R from A. The distance the tool offsets from Workpiece A at the radius is called compensation.

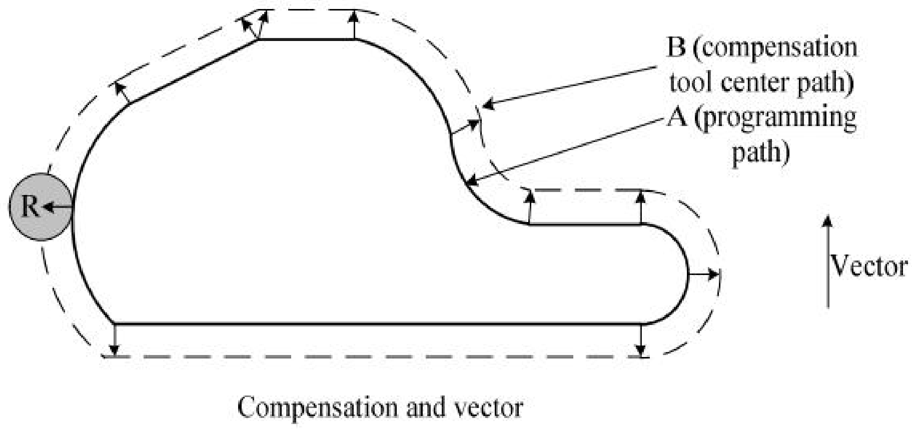



Fig. **4-5-2-1**

The programmer uses the tool radius compensation mode to program the machining process. During machining, the tool diameter is measured and recorded into the CNC memory, and the tool path becomes the compensation path B.

2. Compensation quantity (D value)

A radius offset number will be specified by the D code, and the depth offset corresponding to this offset number is added to or subtracted from the movement code value in the program to form a new movement code. The offset number can be specified from D00 to D255 as needed. Whether the compensation quantity is measured by diameter value or by radius value is set by Position Parameter **N0: 40#7.**

The LCD/MDI panel can be used to preset the depth offset corresponding to the offset number in the offset memory in advance.

The compensation quantity can be set within the following range:

**Table 4-5-2-1**

| | (input in mm) |
|---|---|
| Compensation quantity D | -999.9999ｍｍ～999.9999ｍｍ |

**Note: The compensation amount for D00 is set to 0 by default and cannot be set or modified by the user.**

The change of compensation plane must be carried out after canceling the compensation mode. Failure to cancel the compensation mode and change the compensation plane system will trigger an alarm.

3. Plane selection and vector

Compensation calculation is performed within the plane selected by G17, G18, and G19. This plane is called the compensation plane. For example, when selecting the XY plane, perform compensation and vector calculations using (X, Y) in the program. The coordinate values of axes that are not in the compensation plane are not affected by compensation.

When performing three-axis control simultaneously, only the tool path projected onto the compensation plane is compensated.

The change of compensation plane must be carried out after canceling the compensation mode.

**Table 4-5-2-2**

| G code | Compensation plane |
|---|---|
| G17 | X - Y plane |
| G18 | Z - X plane |
| G19 | Y - Z plane |

4、 G40, G41 and G42

Use G40, G41 and G42 to cancel and execute the tool radius compensation vector. They can be combined with G00 and G01 to define a mode which determines the value and direction of compensation vector.

**Table 4-5-2-3**

| G code | Compensation plane |
|---|---|
| G40 | Cancel tool radius |
| G41 | Tool radius left compensation |
| G42 | Tool radius right compensation |

5. G53, G28 and G30 in tool radius compensation mode

When G53, G28 or G30 is specified in the tool radius compensation mode, the offset vector of the tool radius offset axis is canceled when moving to the specified position (canceled when moving to the command position in case of G53 and canceled when moving to the reference point in case of G28 or G30), and the axes other than the tool radius offset axis are not canceled. When G53 is in the same segment as G41//G42, all axes will cancel the radius compensation when moving to the command position; when G28 or G30 is in the same segment as G41//G42, all axes will do so when moving to the reference point. The canceled tool radius compensation vector will be restored in the next compensation-plane program segment for recovery.

**Note:** In the compensation mode, it is possible to use Position Parameter NO: 40#2 to determine whether the compensation is temporarily canceled when G28 or G30 is specified to move to the intermediate point.

**Cancel tool radius compensation (G40)**

In the G00 or G01 state, use the following code, G40 X__ Y__;

The linear movement from the old vector of the starting point toward the end point. In the G00 mode, each axis moves quickly to the end point. Use this code to make the system state switch from tool compensation state to cancelled tool compensation.

If it is only G40, the tool will not move when there is no command for X__ Y__**.**

**Tool radius compensation - left (G41)**

**1) In case of G00, G01**

G41 X__ Y__ D__; At the end point of the program segment, the code forms a new vector perpendicular to the direction of (X, Y), and the tool moves from the tip of the old vector at the starting point to the tip of the new vector.

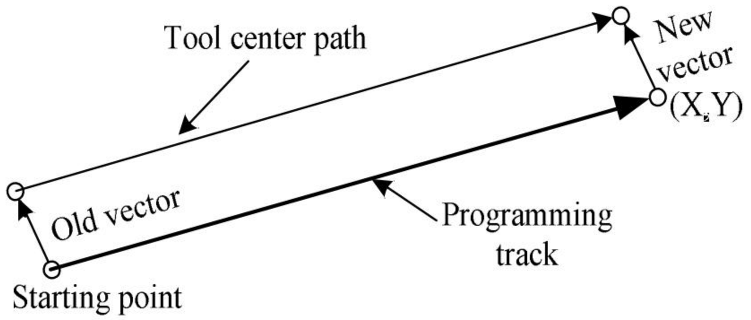



Fig. **4-5-2-2**

When the old vector is zero, the code is used to cause the tool state to switch from the cancelled tool offset to the tool radius compensation. At this moment, the offset value is specified by the D code.

**2）In case of G02, G03**

**G41……;**

**……**

**……**

**G02 /G03 X__ Y__ R__ ；**

The above program can from a new vector, which is located on the line connecting the center and end point of the arc. In view of the arc's forward direction pointing to the left (or right), the tool center moves along the arc from the old vector tip of the arc toward the new vector tip, however, provided that the old vector has been correctly formed.

The offset vector is centrifugal or points to the arc center from the starting or end point.

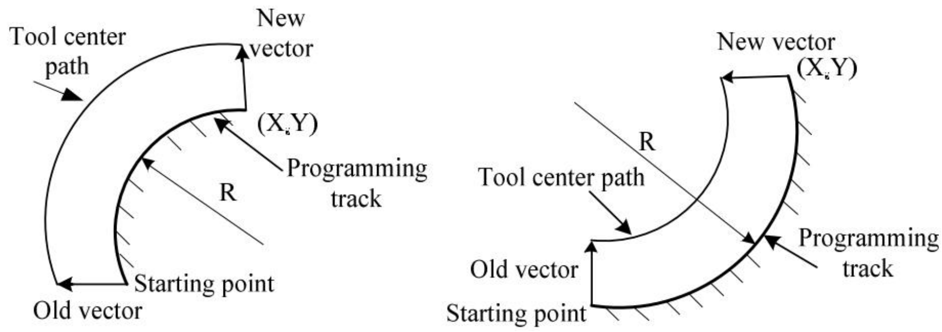



Fig. **4-5-2-3**

**Tool radius compensation - right (G42)**

G42 is the opposite of G41, and the tool will offset on the right side of the workpiece along the tool advance direction. That is, the vector direction defined by G42 is exactly opposite to that defined by G41. The offset method is identical to G41 except that the vector direction is opposite.

**1) In case of G00, G01**

**G42 X__ Y__ D__ ；**

**G42 X__ Y__ ；**

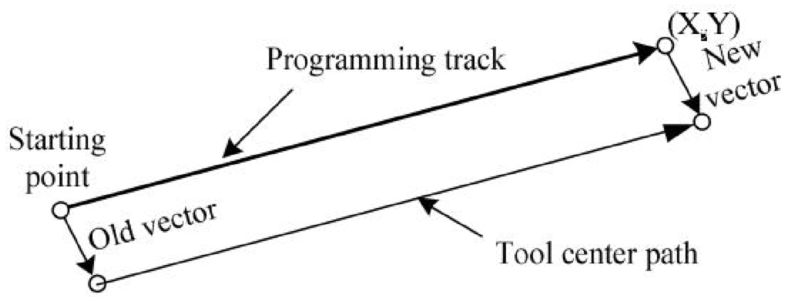



Fig. **4-5-2-4**

**2) In case of G02, G03**

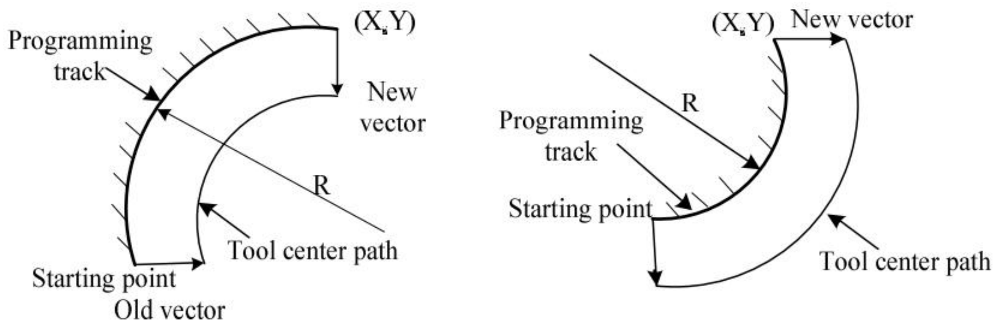



Fig. **4-5-2-5**

6. General considerations about offset:
   (A) Designation of offset number
   G41, G42 and G40 are modal codes, and the offset number is specified by the D code. It can be specified anywhere before the state shifts from the cancelled offset to the tool radius compensation.
   (B) Shift from the cancelled offset state to the tool radius compensation state
   The movement code when entering into the tool radius compensation state from the cancelled offset state must be positioning (G00) or linear interpolation (G01), and circular interpolation (G02, G03) cannot be used.
   (C) Conversion of left and right tool radius compensation
   When the offset direction changes from left to right, or from right to left, it will usually go through the offset cancelled state. However, positioning (G00) or linear interpolation (G01) can be directly converted without going through the offset cancelled state. The tool path at this time is as shown below:

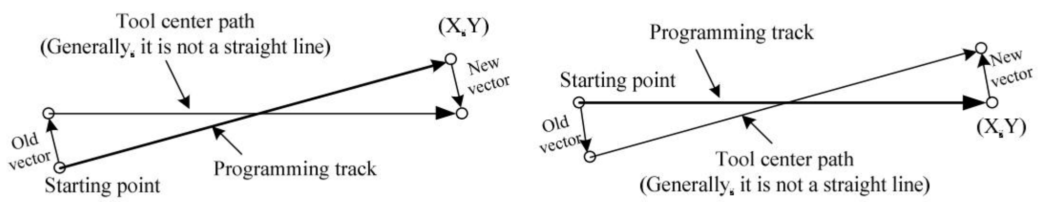



Fig. **4-5-2-6**

| | |
|---|---|
| G1G41 D__X__ Y__; | G42 D__X__ Y__; |
| …… | …… |
| G1G42 D__X__ Y__; | G41 D__ X__ Y__; |

(D) Change in offset
The change in offset generally occurs in the offset cancelled state during tool replacement, but can also occur in the offset state for positioning (G00) and linear interpolation, as shown in the following figure.

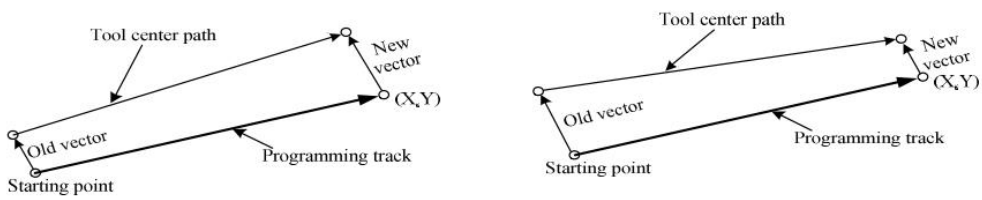



Fig. **4-5-2-7(**Change in offset**)**

(E) Positive and negative offset and tool center path

When the offset is set to a negative value, the workpiece machined is equivalent to the case where G41 and G42 on the program sheet are converted. Therefore, cutting or machining along the outer side of workpiece becomes cutting or machining along the inner side of workpiece.

As general programming shown below, assume the offset is a positive value:

When the tool path is programmed as shown in Figure (A), if the offset is set to a negative value, the tool movement path is as shown in Figure (B); similarly, when the tool path is programmed as shown in Figure (B), if the offset is set negative, the tool movement path is as shown in Figure (A).

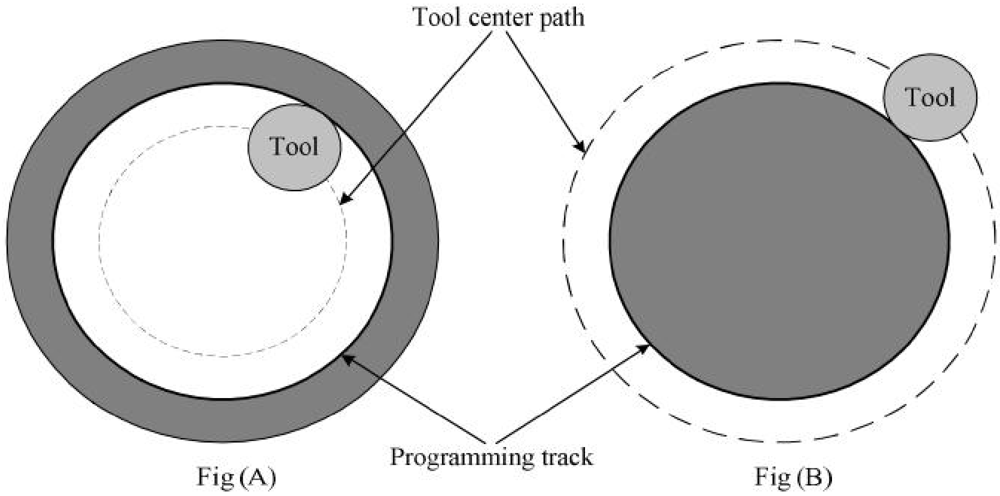



Fig. **4-5-2-8**

Graphs with sharp corners are commonly used (patterns with sharp circular interpolation). However, when the offset is set to a negative value, the inner circle of the part cannot be machined. In case of cutting an inner sharp corner, insert an arc of appropriate radius there, and then perform cutting after smooth transition.

Selection of left or right compensation depends on whether the compensation direction is on the left or right side of the tool movement direction relative to the workpiece (the workpiece is considered motionless). G41 or G42 puts the system into compensation mode while G40 causes the system to cancel the compensation mode.

An example of compensation program is as follows:

Program Segment (1) is called start part, and in this segment, G41 changes the mode from compensation cancelled to compensation. At the end point of this segment, the tool center is compensated with the tool radius perpendicular to the next program path (from P1 to P2). The tool compensation quantity is specified by D07, that is, the compensation number is set to 7, and G41 represents the tool path left compensation. After the compensation starts, when the workpiece shape is programmed as P1 → P2 ... P9 → P10 → P11, the tool path compensation is automatically performed.

Example of tool path compensation program.

G92 X0 Y0 Z0;

(1) N1 G90 G17 G0 G41 D7 X250 Y550 ; (The compensation quantity must be preset with the compensation number)

(2) N2 G1 Y900 F150 ;

(3) N3 X450 ;

(4) N4 G3 X500 Y1150 R650 ;

(5) N5 G2 X900 R-250 ;

(6) N6 G3 X950 Y900 R650 ;

(7) N7 G1 X1150 ;

(8) N8 Y550 ；
(9) N9 X700 Y650 ；
(10) N10 X250 Y550 ；
(11) N11 G0 G40 X0 Y0 ；

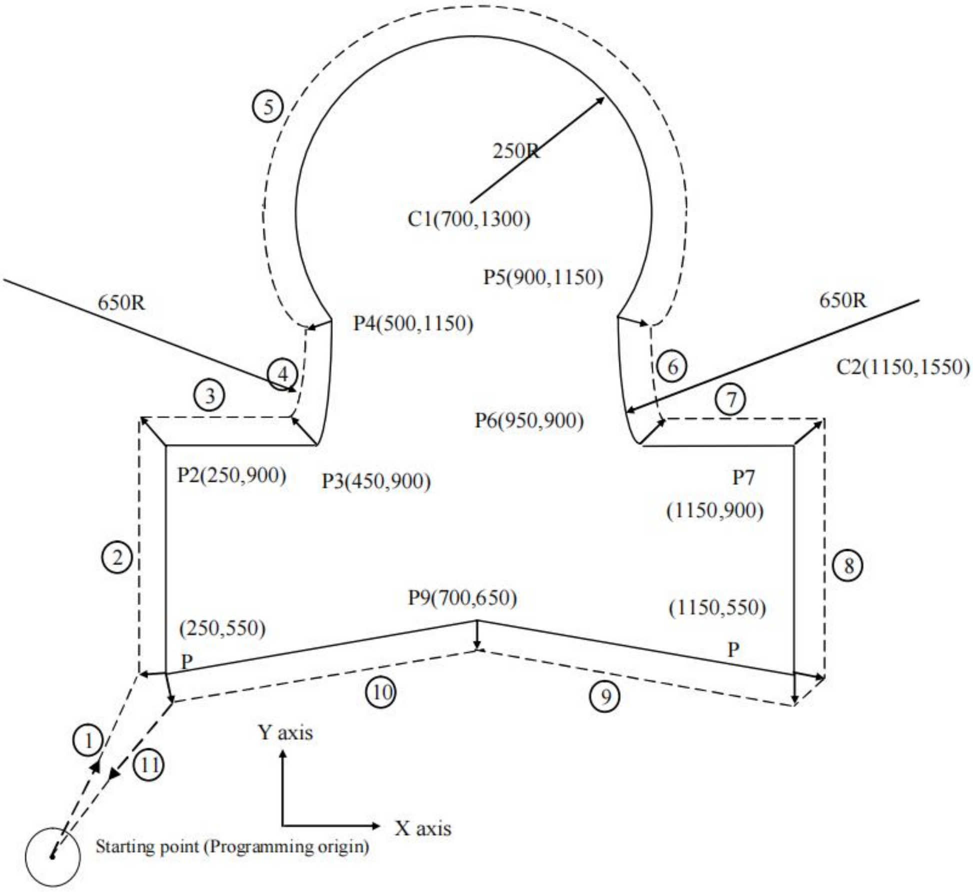



Fig. **4-5-2-9**

## 4.5.3 **Detailed Description of Tool Radius Compensation**

**Concept:** Inner side and outer side: When the angle of the tool paths established by two program segments exceeds 180°, the path is called the inner side, and when the angle is between 0° and 180°, it is called the outer side.

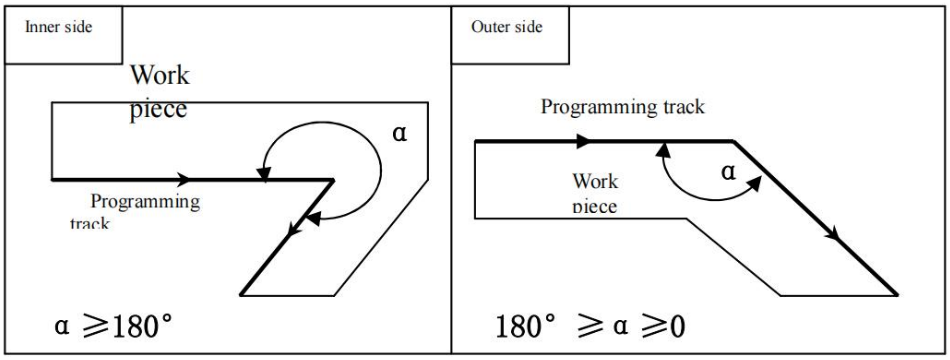



Fig. **4-5-3-1**

**Meaning of symbols:**

Use the following symbols in the following figures:

——S indicates that a single program segment is executed once at this position;

——SS indicates that a single program segment is executed twice in this position;

——SSS indicates that a single program segment is executed three times in this position;

——L indicates that the tool moves in a straight line;

——C indicates that the tool moves along the arc;

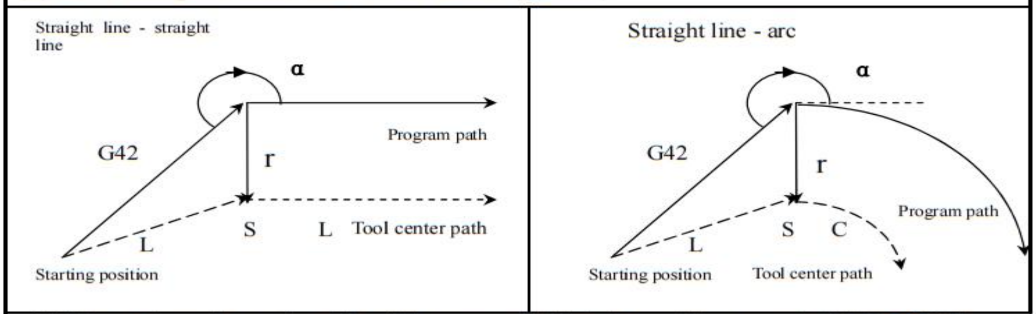
——r represents the tool radius compensation value;

——Point of intersection is a position where two programming tracks intersect after they are offset by r;

——O represents the tool center.

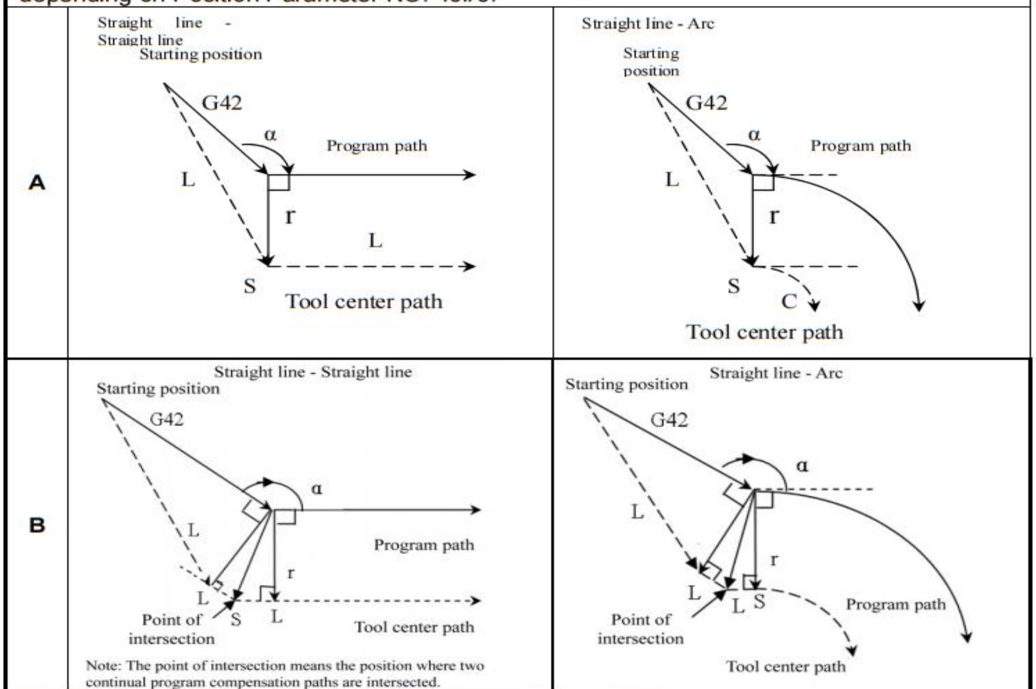
1. **Tool movement at the start of cutting:** when the offset cancelled mode is changed to the offset mode, the tool moves as shown below (at the start of cutting):

**(a) Move along the inner side of the corner (α≥180°)**




**(b) Move along the outer side of the corner which is an obtuse angle (180°>α≥90°)**

The tool path has 2 types (A and B) when the compensation is started or cancelled, depending on Position Parameter NO: 40#0:

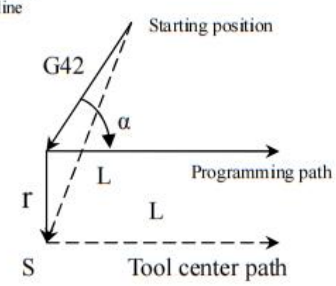



**(C) Move along the outer side of the acute angle (α<90°)**

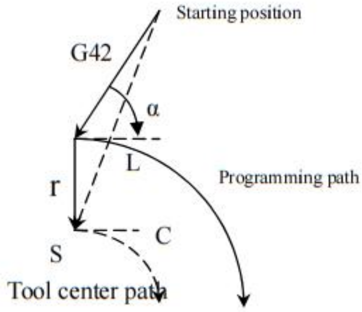
The tool path has 2 types (A and B) when the compensation is started or cancelled, depending on Position Parameter NO: 40#.0:

**A**

Straight line - Straight line

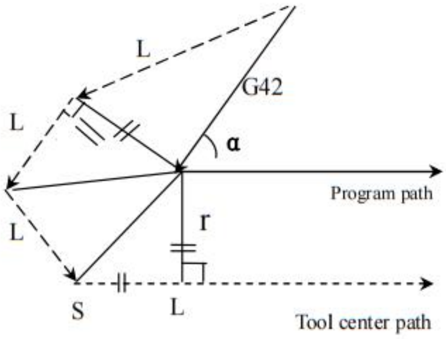



Straight line - Arc

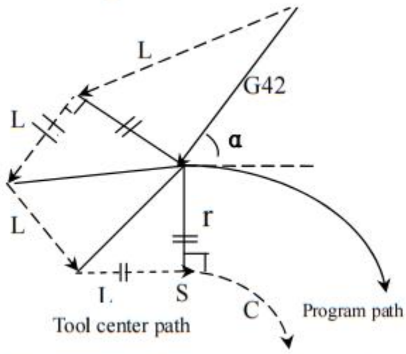



**B**

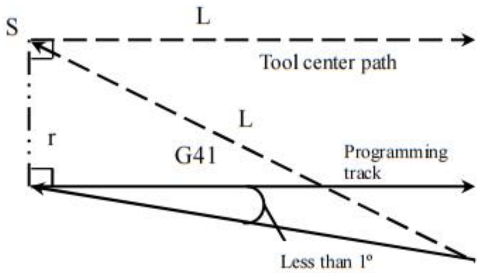
Straight line - straight line

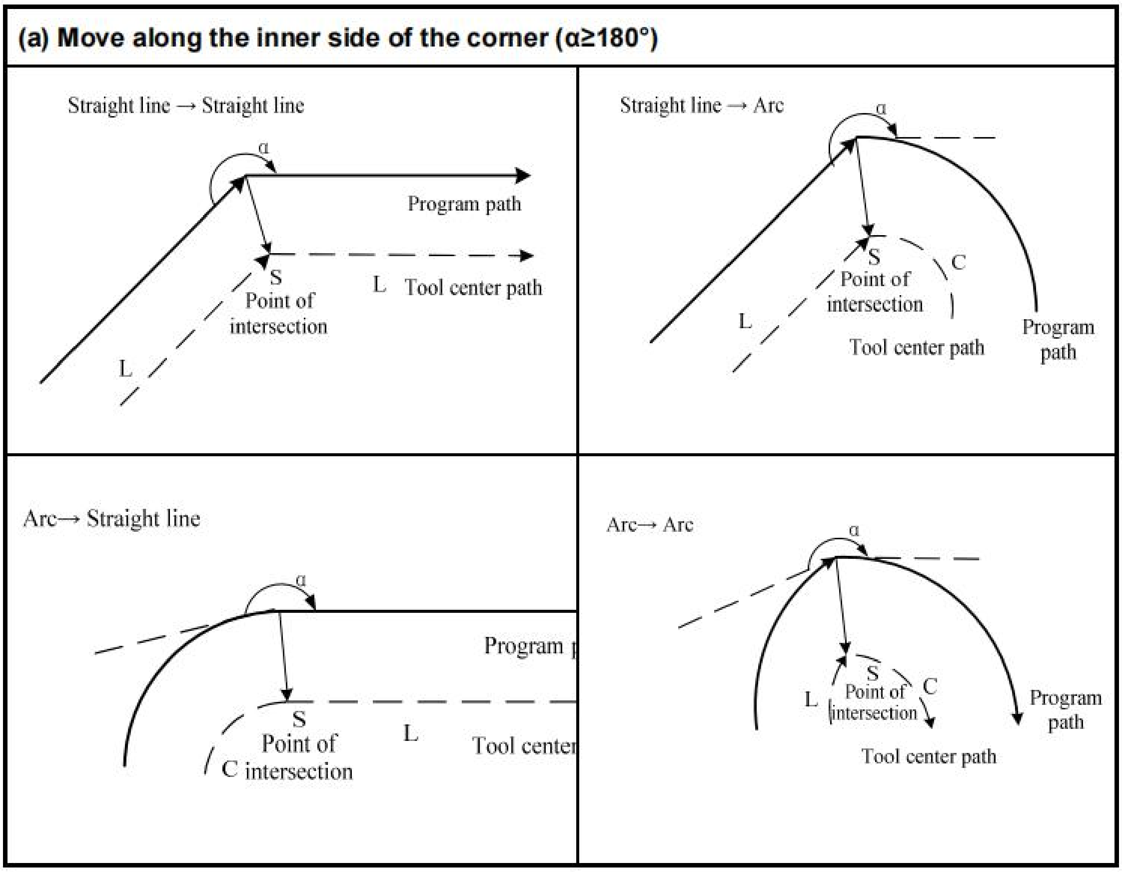



Straight line - arc




**(d) Move along the outer side of the corner which is an acute angle of less than 1°, straight line → straight line. (α<1°)**






### 3. Special circumstances

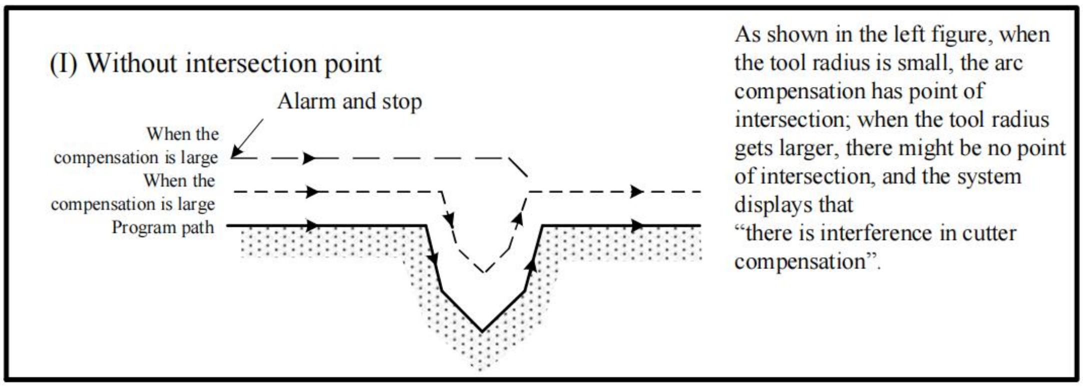



Fig. **4-5-3-2**

### 4. Tool movement in offset cancelled mode

In the compensation mode, when a program segment that meets any of the following conditions is executed, the system will enter into the compensation cancelled mode, and the activity of this segment is called compensation cancellation.

a) G40

b) The tool radius compensation number is 0.

The arc code (G03 and G02) cannot be used to perform compensation cancellation. If a command arc is generated, an alarm will be generated and the tool will stop.

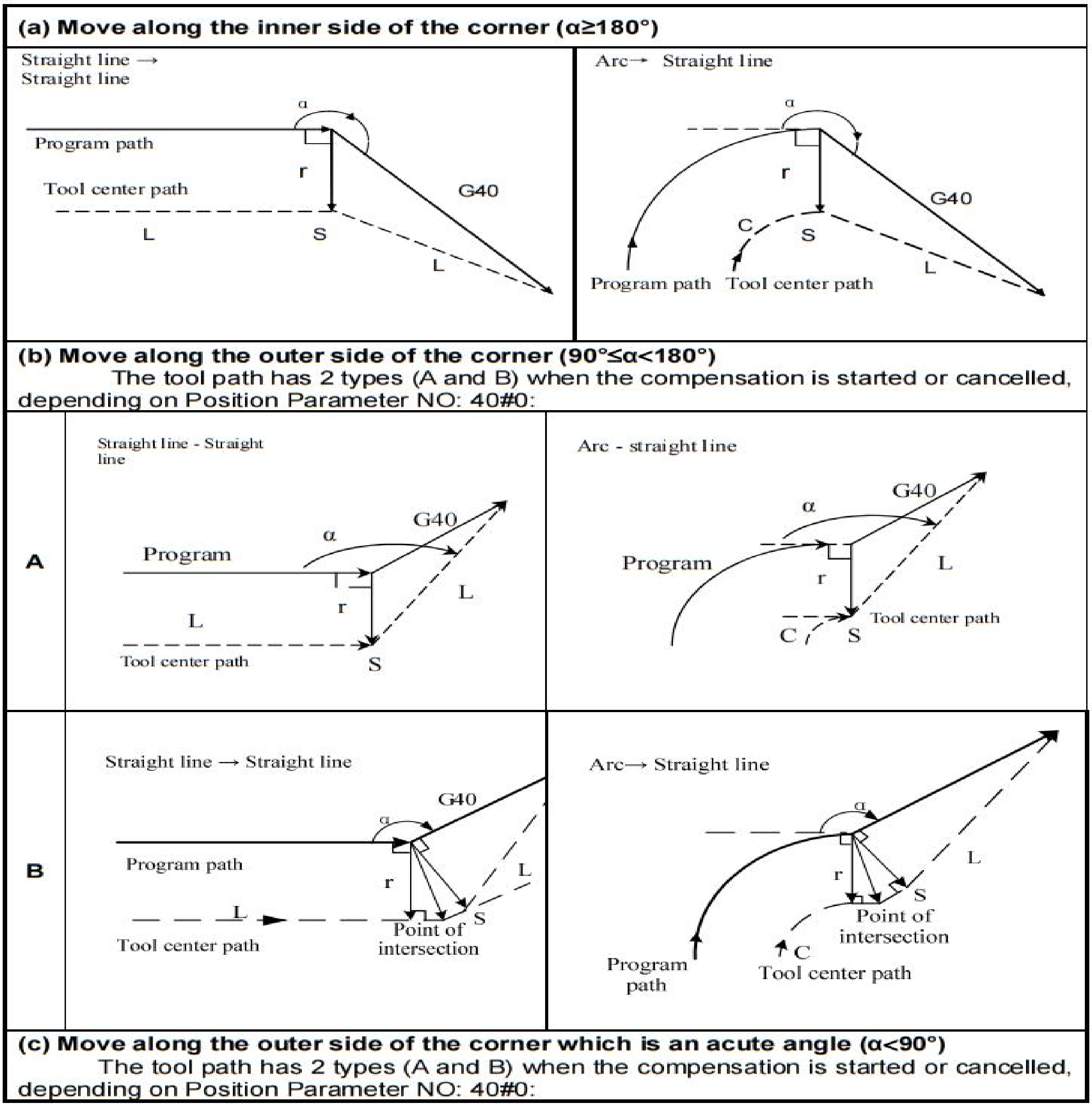
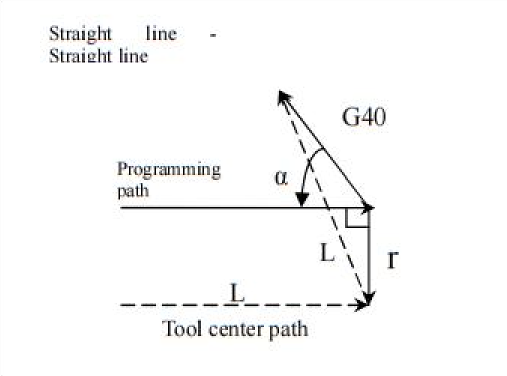
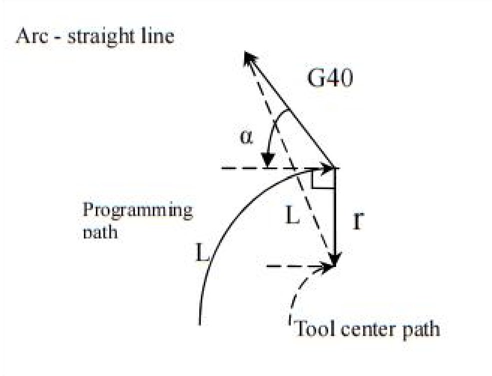
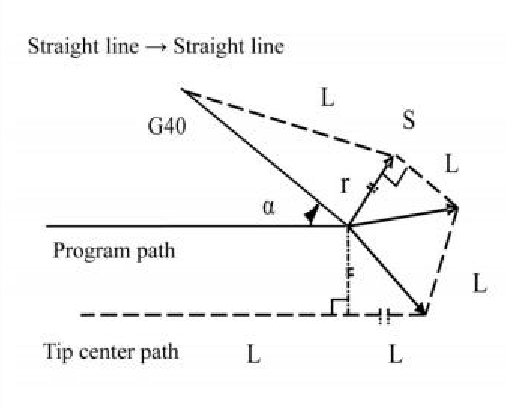
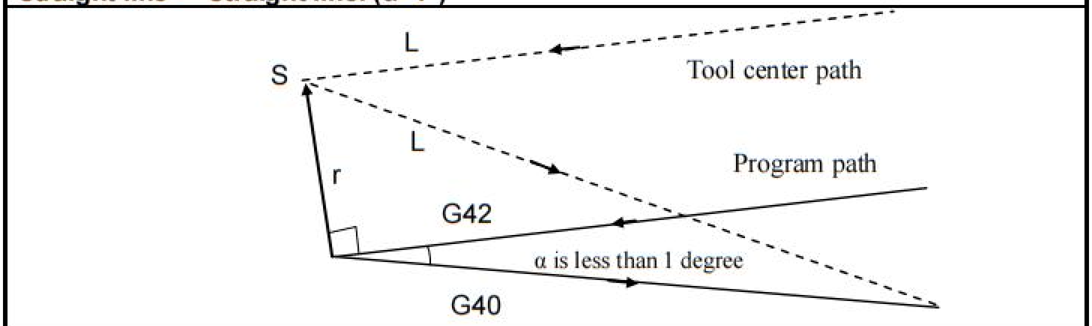

(a) Move along the inner side of the corner (α≥180°)
Straight line → Straight line
α
Program path
r
Tool center path
G40
L
S
L
Arc→ Straight line
α
r
G40
C
S
L
Program path
Tool center path
(b) Move along the outer side of the corner (90°≤α<180°)
The tool path has 2 types (A and B) when the compensation is started or cancelled, depending on Position Parameter NO: 40#0:
A
Straight line - Straight line
G40
α
Program
L
r
L
Tool center path
S
Arc - straight line
G40
α
Program
L
r
Tool center path
C
S
B
Straight line → Straight line
G40
α
Program path
r
L
L
S
Point of intersection
Tool center path
Arc→ Straight line
α
L
r
S
Point of intersection
C
Program path
Tool center path
(c) Move along the outer side of the corner which is an acute angle (α<90°)
The tool path has 2 types (A and B) when the compensation is started or cancelled, depending on Position Parameter NO: 40#0:

Fig. **4-5-3-3**

| | | |
|---|---|---|
| **A** |  |  |
| **B** |  | v |

**(d) Move along the outer side of the corner which is an acute angle of less than 1°, straight line → straight line. (α<1°)**

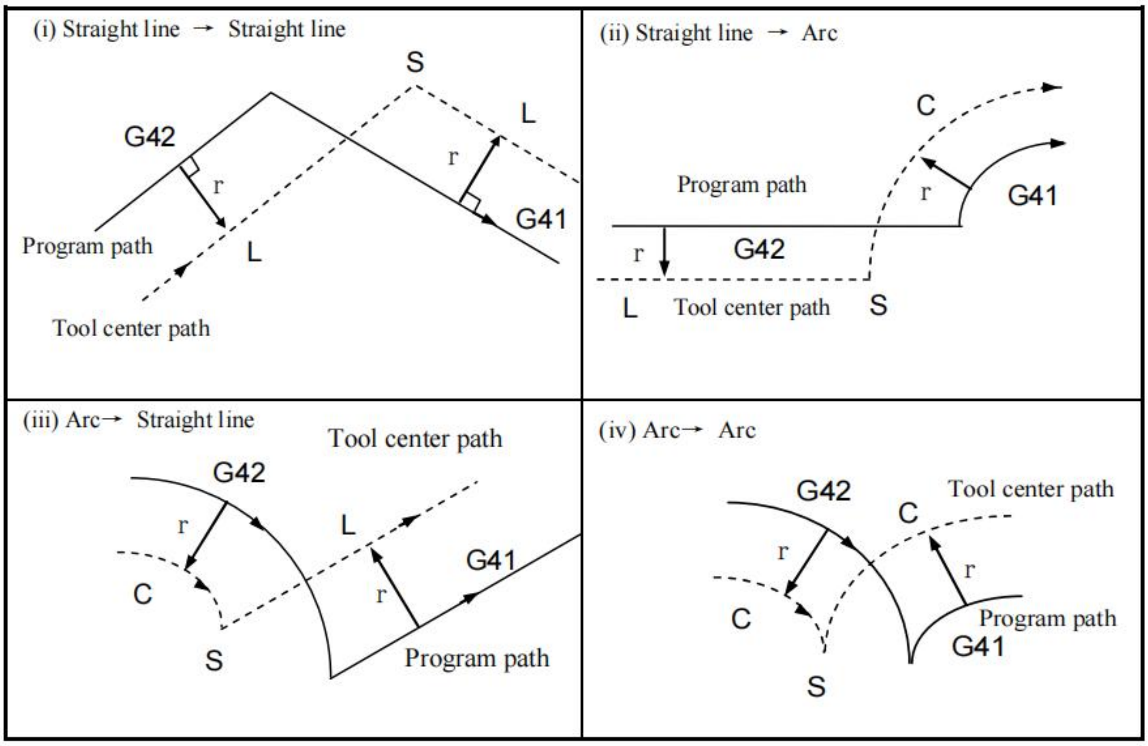



**5. Change of compensation direction in compensation mode**

The tool radius compensation G code (G41 and G42) determines the compensation direction. Compensation symbols are shown as follows:

| Compensation symbols<br>G code | + | - |
|---|---|---|
| G41 | Left compensation | Right compensation |
| G42 | Right compensation | Left compensation |

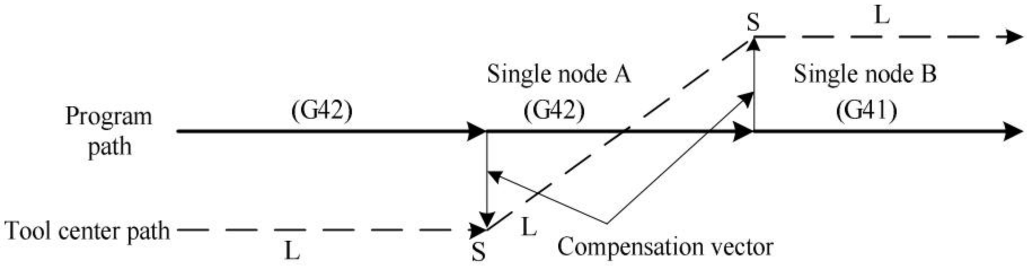
In special cases, the compensation direction can be changed in the compensation mode. However, it cannot be changed in the start program segment and subsequent segments. For change of compensation direction, there is no difference between inner side and outer side. The following compensations are assumed to be positive.



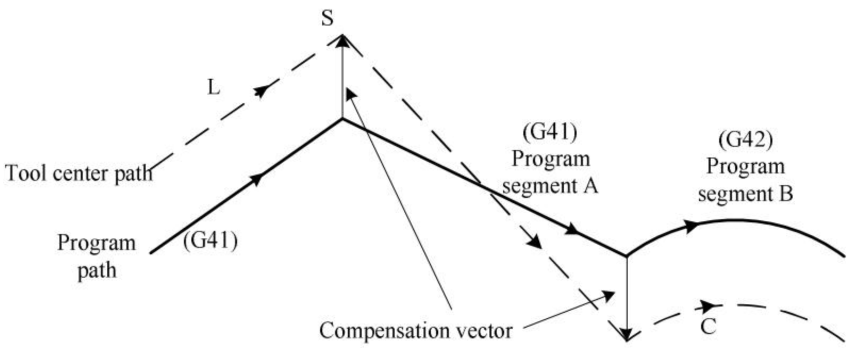

(v) If the compensation is performed normally, but there is no intersection

When G41 and G42 are used to change the offset direction from Program Segment A to Program Segment B, if the intersection of compensation paths is not required, a vector perpendicular to Program Segment B will be made at the starting point of Program Segment B.

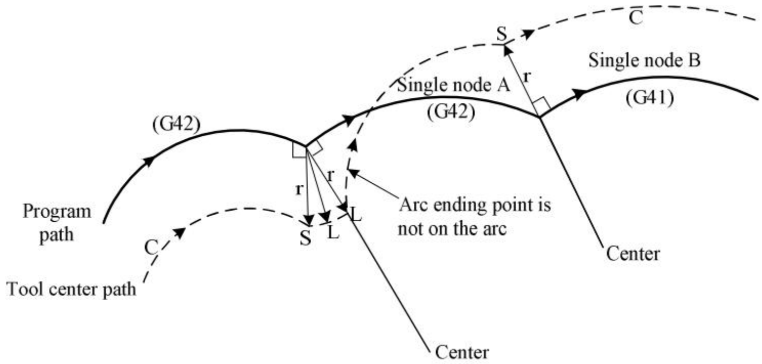
(1) Straight line - straight line




(2) Straight line - arc



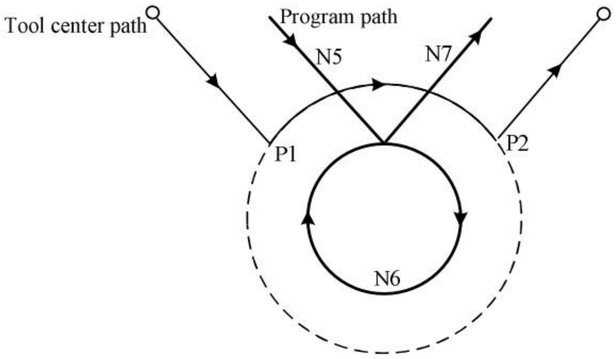
(3) Arc - arc




(vi) When the tool radius compensation makes the tool center path longer than one circle, the following conditions will usually not occur. However, when G41 and G42 are changed, the following conditions may occur:

Arc - arc (straight line - arc) The system will give an alarm when the tool compensation direction is changed. When the tool number is D0, the alarm prompts that the arc code cannot cancel the tool compensation!

Straight line - straight line can change the tool compensation direction.




(G42)
N5 G01 G91 X500 Y-700;
N6 G41 G02 J-500;
N7 G42 G01 X500 Y700;

At this point, the tool center path is not an arc, but a section of arc from P1 to P2. In some circumstances, there might be an alarm due to interference checking.

If a full circle movement is desired for the tool, the command must be segmented.

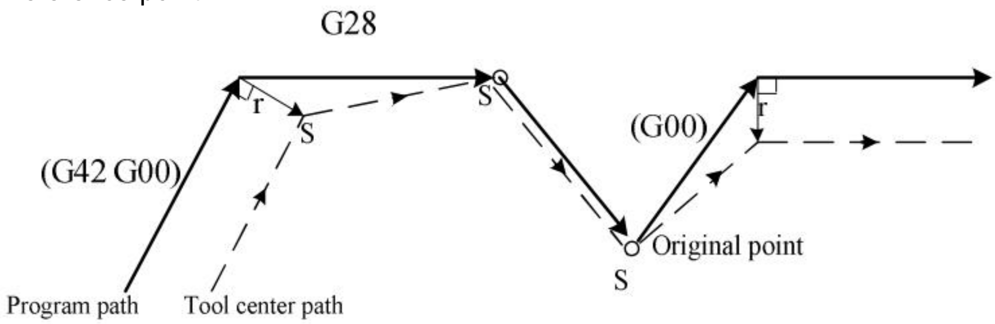
**6. Temporary compensation cancellation**

In the compensation mode, Position Parameter NO: 40#2 can be used to determine whether the compensation will be temporarily cancelled at the intermediate point when G28 or G30 is specified.

Please see the detailed description of compensation cancellation and compensation start for detailed method of this operation.

**a) Automatic return to reference point (G28)**

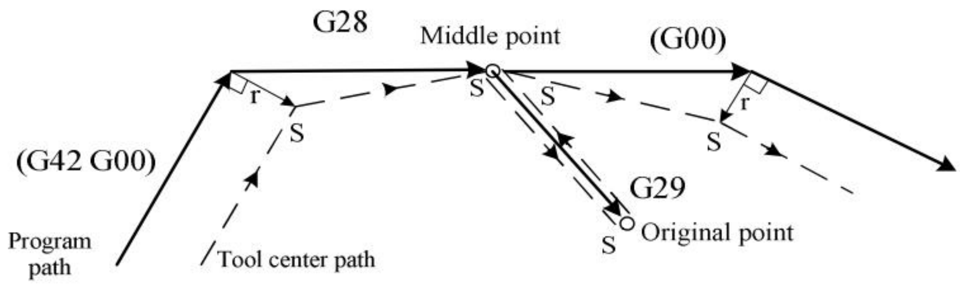
In the compensation mode, if G28 is specified, the compensation will be cancelled at the intermediate point, and the compensation mode will be automatically restored upon return to the reference point.



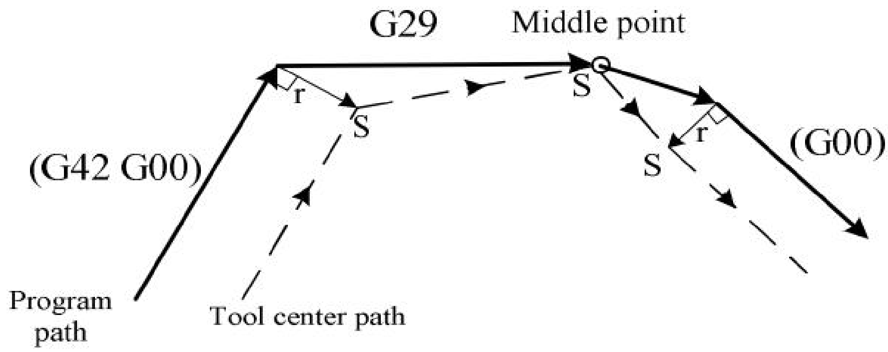

**b) Automatic return from reference origin (G29)**

In the compensation mode, if G29 is specified, the compensation will be cancelled at the intermediate point, and the compensation mode will be automatically restored during return to the point specified by G29.

In case of immediate command after G28:




In case of non-immediate command after G28:

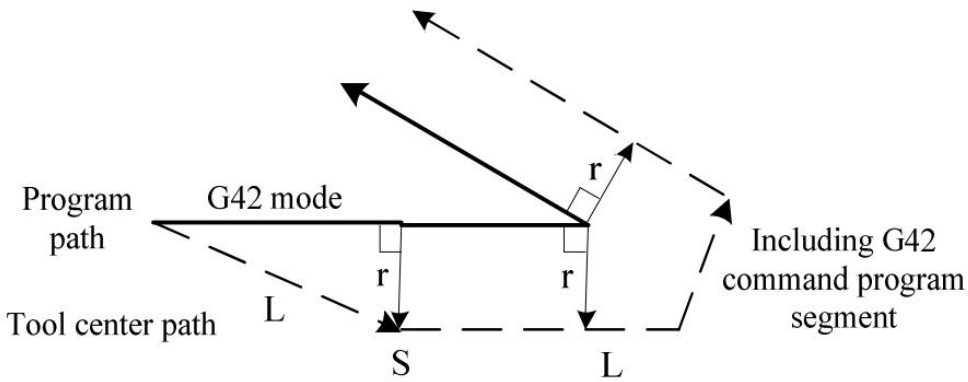



**7. Tool radius compensation in compensation mode (G code)**

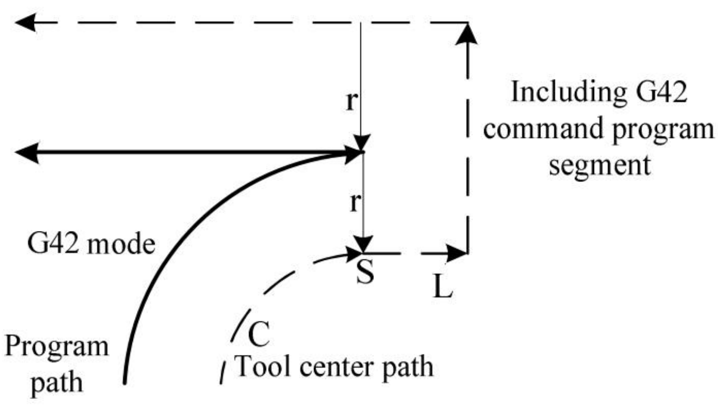
In the compensation mode, when the tool radius compensation G code (G41, G42) is specified, it will generate a vector at right angle to the front program segment with respect to the movement direction, regardless of the inner or outer side of machining. However, if such a G code is specified in the arc code, the correct arc cannot be obtained.

When changing the compensation direction with the tool radius compensation G code (G41, G42), please refer to (5).

Straight line - straight line




Arc - straight line



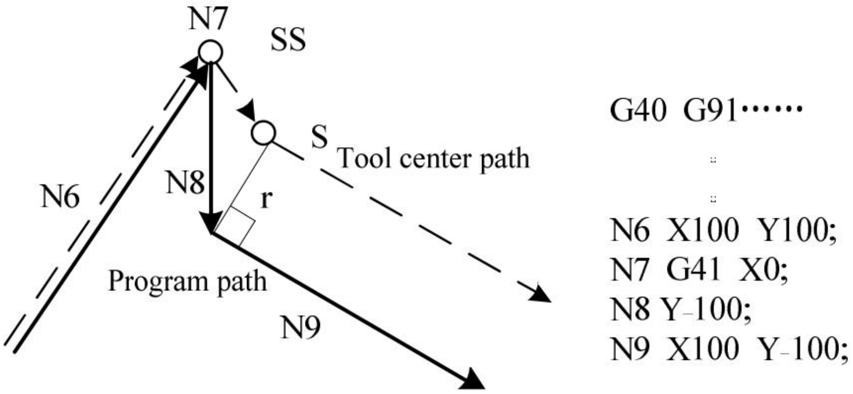
**8. Code segments where the tool does not move**

The followings are program segments where the tool does not move. In these segments, the tool will not move even if the tool radius compensation mode is effective.

(1) M05; …………………… M code output
(2) S21; ........................
S code output
(3) G04 X10; ............... Pause
(4) (G17) Z100;.................. No movement code in the compensation plane
(5) G90;...........................only G code
(6) G01 G91 X0;...............No movement

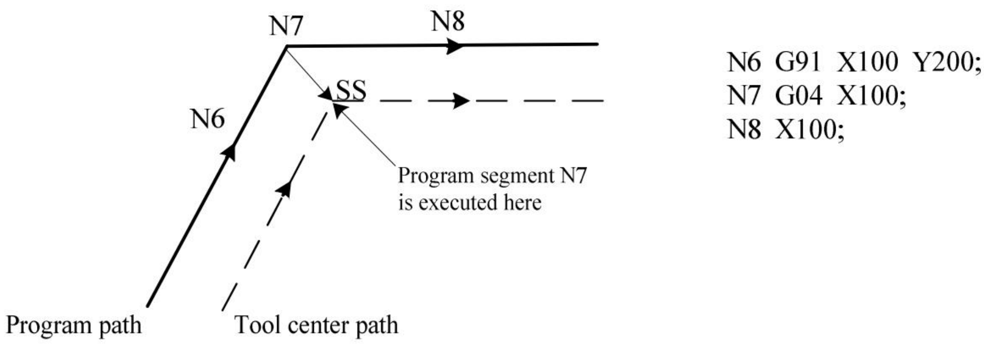
**a) Code at the beginning of compensation**

In case of no movement of the program segment for the start of cutting, the system will generate an activity to start cutting in the next segment of movement code.

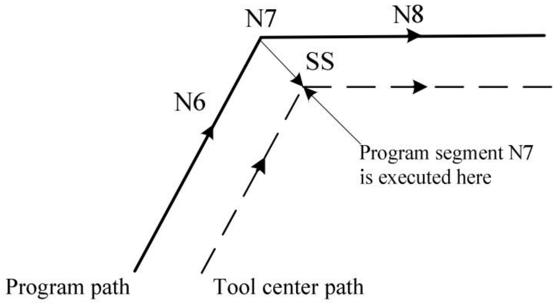



**b) When the compensation mode is specified**

When only one program segment without tool movement is specified in the compensation mode, the vector and tool center path remain unchanged. (Refer to Item (3) for compensation mode) This program segment will be executed at the stop point of a single program segment.




However, in case of no movement of the program segment, even if only one program segment is specified, the tool moves in the same manner as two or more segments without tool movement.

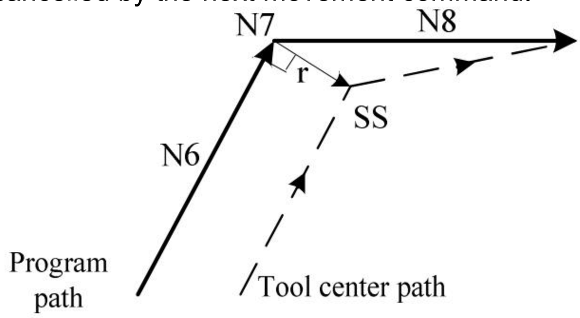



```
N6 G91 X100 Y200;
N7 X0;
N8 X100;
```

**Note:** The above program segment runs under the condition of G1, G42, and the track in case of G0 is not consistent with the figure.

**c) When specified together with compensation cancellation**

In case that the program segment, specified together with compensation cancellation, features no tool movement, it will generate a vector with its length as compensation and perpendicular to the movement direction of the front program segment, and this vector will be cancelled by the next movement command.



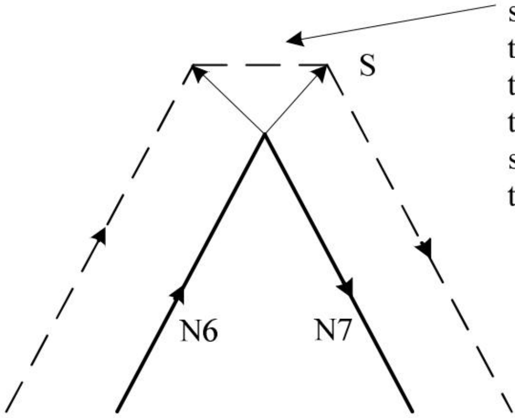

```
N6 G91 X100 Y100;
N7 G40;
N8 X100 Y0;
```

**9. Corner movement**

If more than two vectors are generated at the end of a program segment, the tool moves from one vector line to another vector. Such movement is called corner movement.

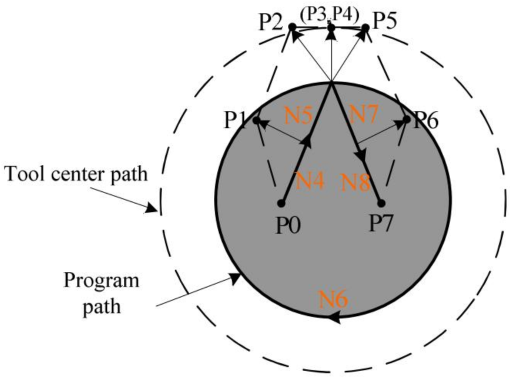
If ΔVX ≤ ΔV limit and ΔVY ≤ ΔV limit, the latter vector is ignored.

If these vectors are inconsistent, a movement along the corner is produced. This movement belongs to the front program segment.




These movements belong to the program segment N6. Therefore the feed rate equals to the one of the program segment N6. If the program segment N6 is G00 mode, the tool moves at fast feed rate; if the program segment N6 is G01, G02, G03 mode, the tool moves at cutting feed rate.

However, if the path of the next segment exceeds a semicircle, the above functions are not executed. The reasons include:




```
N4 G41 G91 X150 Y200;
N5 X150 Y200;
N6 G02 J-600;
N7 G01 X150 Y-200;
N8 G40 X150 Y-200;
```

If the vector is not ignored, the tool path is shown as follows:
P0 → P1 → P2 → P3 (arc) → P4 → P5 → P6 → P7
However, if the distance between P2 and P3 is ignored, P3 will be ignored. The tool path is shown as follows:
P0 → P1 → P2 → P4 → P6 → P7 The arc cutting of Program Segment N6 is ignored.

**10. Interference check**

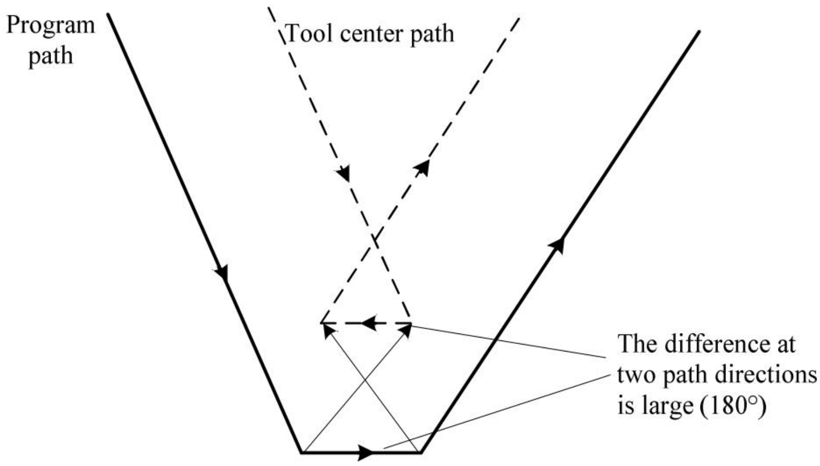
Excessive cutting of the tool is called “interference”. Interference can be used to pre-check excessive tool cutting. The system will give an alarm if an interference is detected in the syntax check after the program is loaded in. It is set by Position Parameter **NO: 41#6** whether an interference check is performed during radius compensation.

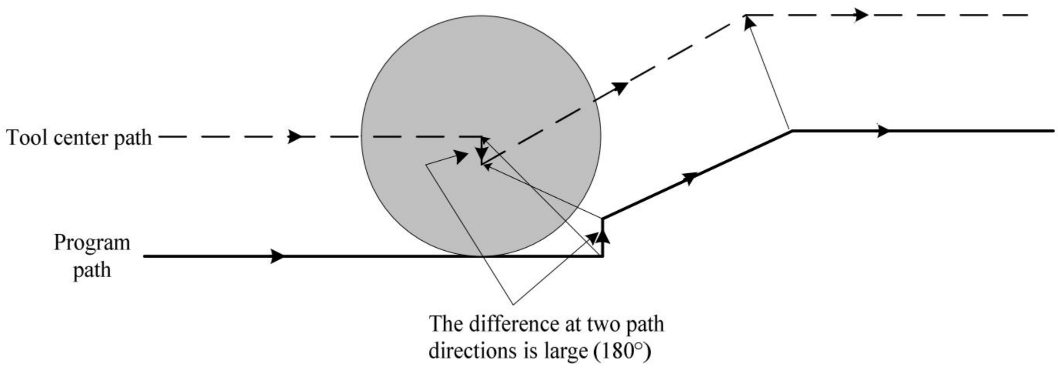
Basic conditions for interference:

**(1)** The movement distance of the program segment for tool radius compensation is smaller than the tool radius.

**(2)** The tool path direction is different from the program path direction. (The angle between the paths is between 90° and 270°).

**(3)** During arc machining, in addition to the above conditions, the angle between the start point and the end point of the tool center path is greatly different from the angle between the start point and the end point of the program path (180° or more).






**11. Manual operation**

Please see Part II Operation Instructions for manual operation related to the tool radius compensation.

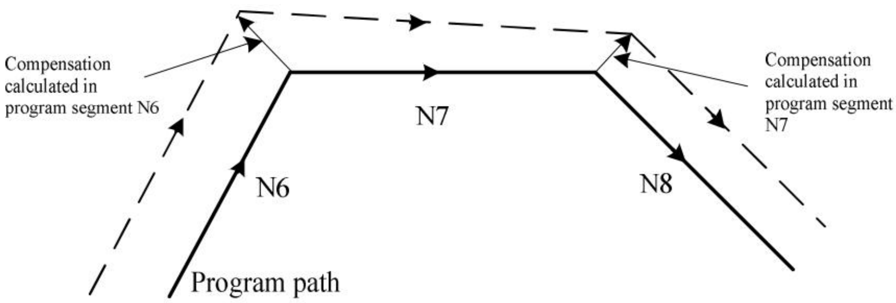
**12. General considerations about compensation**

**a) Specify the compensation quantity**

The compensation quantity is specified by the D code. Once specified, the D code remains valid until another D code is specified, or the compensation is cancelled. The D code is not only used to specify the tool radius compensation quantity, but also used to specify the tool offset value.

**b) Change the compensation quantity**

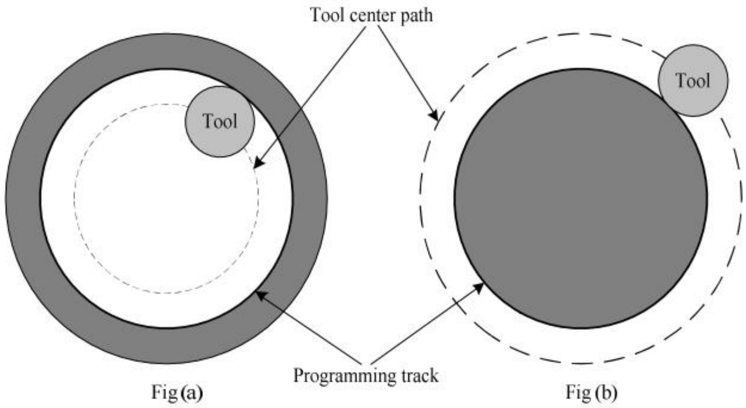
Normally, when the tool is replaced, the compensation quantity must be changed in the compensation cancelled mode. If changed in the compensation mode, the new compensation quantity will be calculated at the end point of the program segment.




**c) Positive or negative compensation quantity and tool center path**

If the compensation quantity is a negative value (-), G41 and G42 in the program will be interchanged. If the tool center moves along the outer side of the workpiece, it will move along the inner side and vice versa, as shown in the following example.

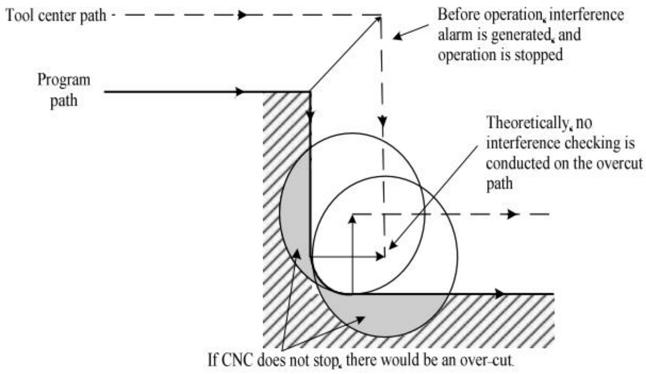
In general, the compensation quantity is positive (+) during programming. When the tool path is programmed as shown in Figure (a), if the compensation quantity is negative (-), the tool center will move as shown in Figure (b) and vice versa. Therefore, the same program can be cut into a male or female form, and the gap between them can be adjusted by selection of compensation quantity.




**d) Overcuting with tool radius compensation**

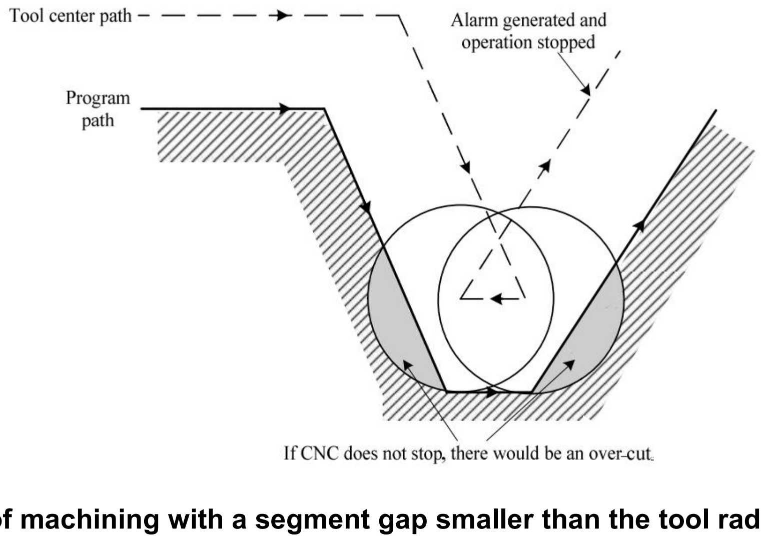
**(1) In case of inner machining with an arc smaller than the tool radius**

When the corner radius is smaller than the tool radius, since inner compensation of the tool will cause excessive cutting, an interference alarm will be generated before the program is executed and the system will stop working.



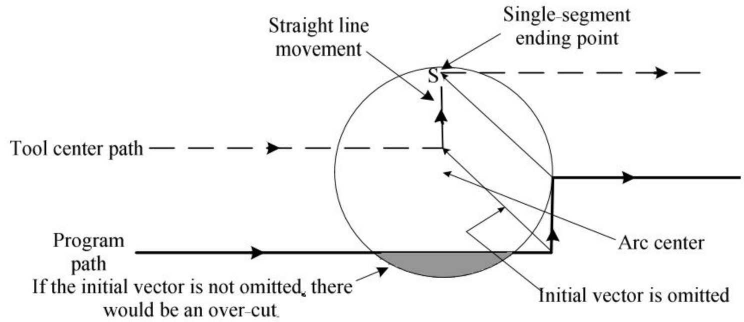

**(2) In case of machining with a groove smaller than the tool radius**

When a groove smaller than the tool radius is used for machining, excessive cutting will occur because the tool radius compensation forces the tool center path to move in the opposite of the program path.




**(3) In case of machining with a segment gap smaller than the tool radius**

If there is a segment gap smaller than the tool radius in the program, when arc machining is used to specify the machining in this segment gap, the tool center path of normal compensation will be in the opposite to the program direction. At this moment, the initial vector is ignored and the tool moves straight to the second vector. Single-segment execution stops here. In case of machining not in the single-segment mode, the automatic operation will continue. If the segment gap is a straight line, no alarm will be generated and correct cutting will be performed. However, there will be uncut portions.

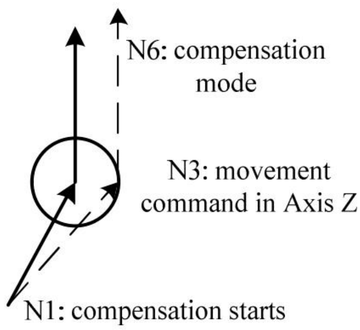



Tool radius compensation starts and moves in the Z axis
Generally, at the beginning of machining, when the tool radius compensation becomes valid, the tool will move along the Z axis at a certain distance from the workpiece. In the above case, if you want to divide the movement along the Z axis into rapid feed and cutting feed, please refer to the following two programs:
In case Program Segment N3 (movement code in the Z axis)
Divided as follows:

```
N1 G91 G00 G41 X500 Y500 D01;
N3 Z-250;
N5 G01 Z-50 F1;
N6 Y100 F2;
```

```
N1 G91 G0 G41 X500 Y500 D1;
N3 G01 Z-300 F1;
N6 Y100 F2;
```

When N3 is executed,
N6 also enters the buffer zone. Use the relationship between them. The correct compensation is shown in right figure.




## 4.5.4 **Corner Offset Circular Interpolation (G39)**

**Format:**

**G39 I_ J_**
**G39 I_ K_**
**G39 J_ K_**

**Function:** During the tool radius compensation, G39 can be used to specify the corner offset circular interpolation. The radius of corner compensation is equal to the compensation value. Position Parameter **NO: 41#5** is used to determine whether the corner arc is valid in radius compensation.

**Description:**

1. When G39 is specified, a corner circular interpolation with its radius equal to the compensation value can be performed.
2. G41 or G42 before this code determines whether the arc is clockwise or counterclockwise, and G39 is a non-modal G code.
3. In case of programming with G39, an arc is formed at the corner, so the vector at the end point of the arc is perpendicular to the start point of the next program segment. As shown in the figure:

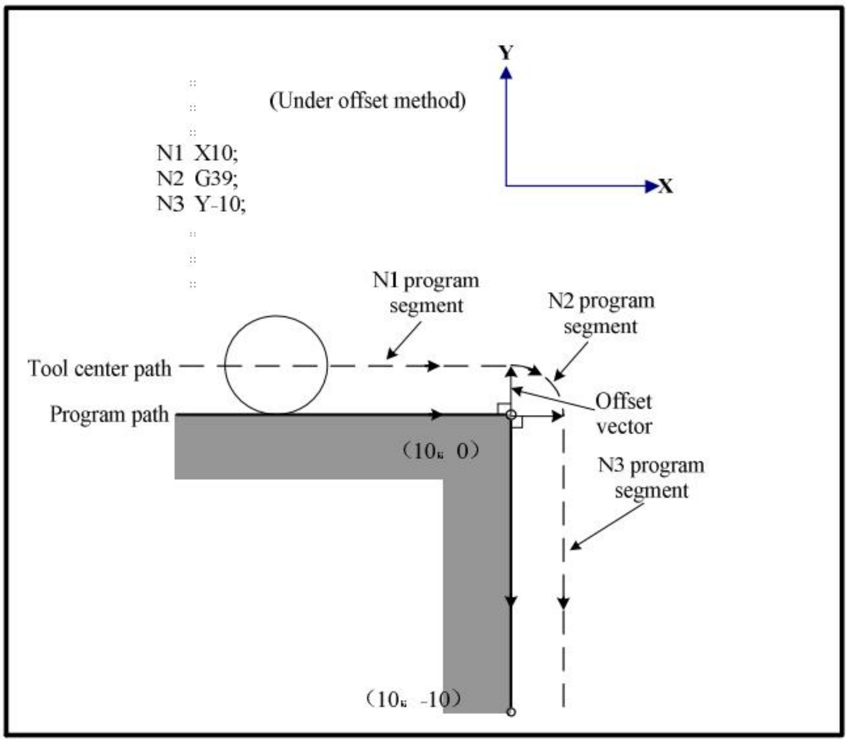



Fig. **4-5-4-2**

## 4.5.5 **Input of Tool Compensation Value and Compensation Number With Program (G10)**

**Format: G10 L10 P_ R_;** geometric compensation value of H code
**G10 L12 P_ R_;** geometric compensation value of D code
**G10 L11 P_ R_;** wear compensation value of H code
**G10 L13 P_ R_;** wear compensation value of D code
**P:** Tool compensation number.
**R:** Tool compensation value in absolute value code (G90) mode.
Tool compensation value in incremental value code (G91) mode, which is added to the value of the specified tool compensation number (the sum is the tool compensation value).

**Description:** Effective input range of tool compensation values:
Geometric compensation: Input in mm -999.999 mm~+999.999 mm;

**Note:** The maximum value of wear compensation is limited by Data Parameter P267

# 4.6 **Feed (G Code)**

## 4.6.1 Feed Mode (G64/G61/G63)

**Format:**
Exact stop mode **G61**

Tapping mode **G63**
Cutting mode **G64**

**Function:**

Exact stop mode G61: once specified, this function remains valid until G62, G63 or G64 is specified. The tool decelerates in the end point of the program segment to perform in-position check and then executes the next segment.

Tapping mode G63: once specified, this function remains valid until G61, G62 or G64 is specified. The tool executes the next program segment without decelerating at the end point of the current segment. When G63 is specified, the feedrate override and feed hold are invalid.

Cutting mode G64: once specified, this function remains valid until G61, G62 or G63 is specified. The tool executes the next program segment without decelerating at the end point of the current segment.

**Description:**

1. No parameter format.
2. G64 is the default feed mode of the system. The tool does not decelerate at the end point of the program segment and directly executes the next segment.
3. The in-position check in the exact stop mode is designed to check if the servo motor is placed in within the specified range.
4. In exact stop mode, cutting mode and tapping mode, the tool moves along different paths. See Figure 4-8-1-1 below for details.

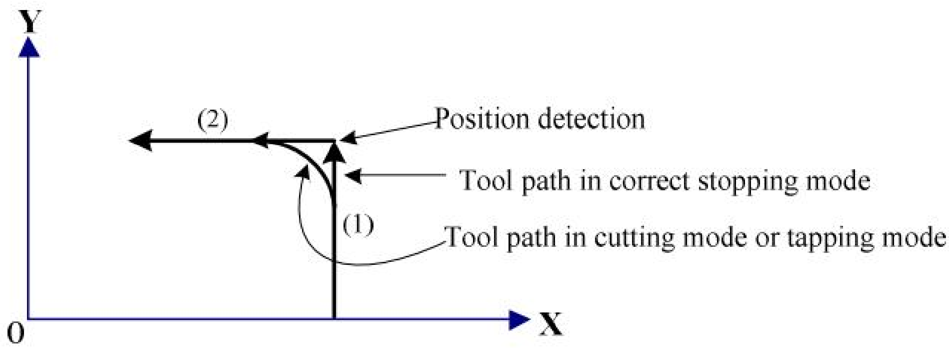



Fig. **4-6-1-1** Tool path from Program Segment 1 to Program Segment 2

## 4.6.2 **Automatic Corner Override (G62)**

**Format: G62**

**Function:** Automatic corner override mode G62: once specified, this function remains valid until G61, G63 or G64 is specified. During tool radius compensation, when the tool moves along the inner corner, the cutting feed rate is multiplied to suppress the cutting output per unit time, so that good surface accuracy can be achieved.

**Description:**

1. During tool radius compensation, the tool automatically decelerates to reduce the load on the tool and outputs smooth surfaces when moving in the inner corner and inner arc area.
2. It is set by position parameters NO: **16#7** whether the automatic corner override function is valid. It is set by position parameters NO: **15#2** to control the automatic corner deceleration function (0: angle control, 1: speed difference control).
3. When G62 is specified and the tool radius compensation function is applied and the inner corner is machined, the feedrate is automatically adjusted at both ends of the corner.

There are four types of inner corner as shown in Figure 4-8-2-1. In the figure: 2° ≤ θ ≤ θp ≤ 178°. θp is set by Data Parameter **P144**.

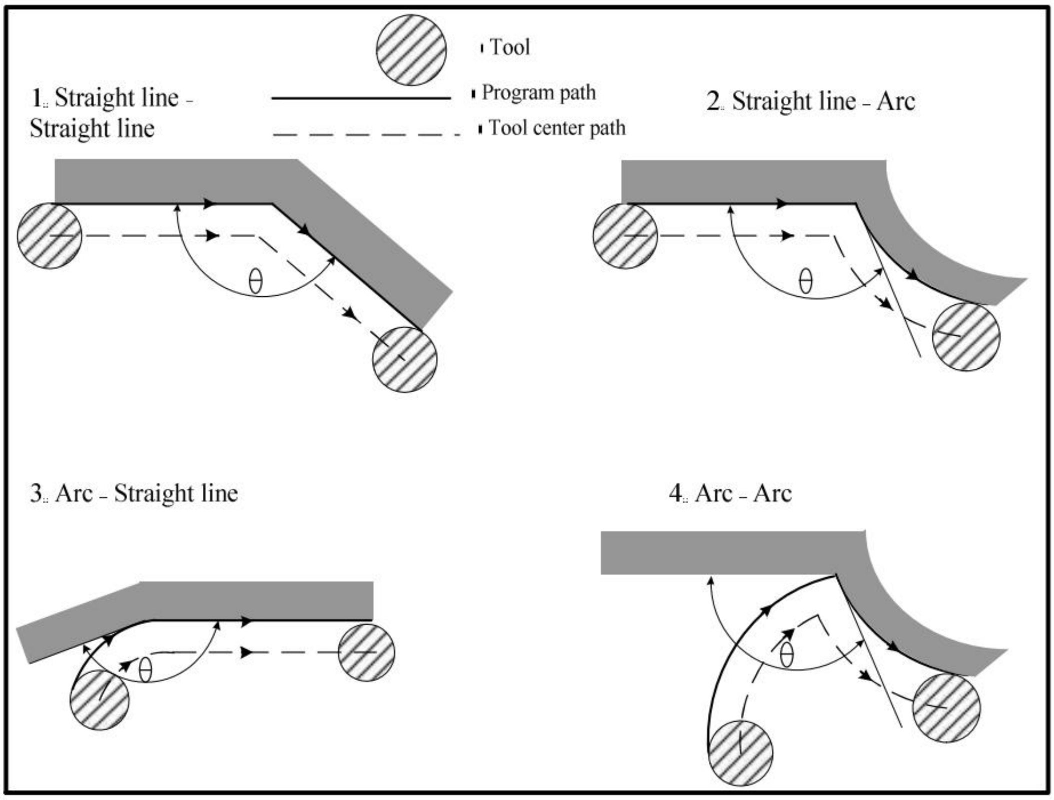



Fig. **4-6-2-1**

4. When the corner is determined as inner corner, the feedrate override is performed before and after the inner corner. The distance at which the feedrate override is performed is Ls 114 and Le, which is the distance from the point on the tool center path to the corner. As shown in Figure 4-6-2-2, where Ls+Le≤2mm.

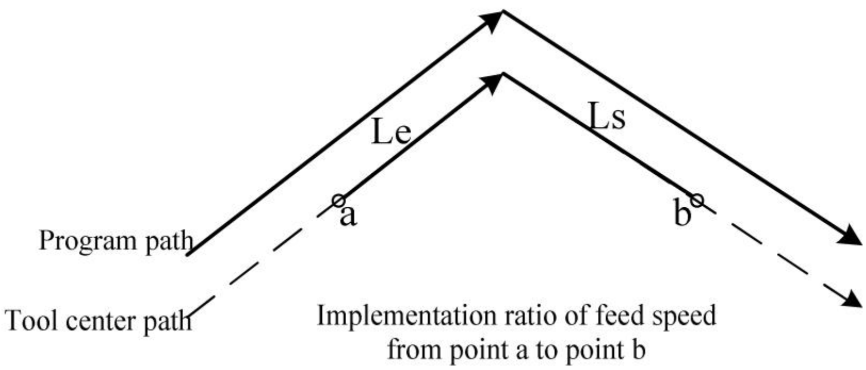



Fig. **4-6-2-2 Straight line to straight line**

5. When the programming track includes two arcs, if the start point and the end point are in the same quadrant or in adjacent quadrants, the feedrate is multiplied, and Data Parameter P145 is used to control the minimum feedrate of automatic corner deceleration, as shown in Figure .as shown in Figure 4-6-2-3。

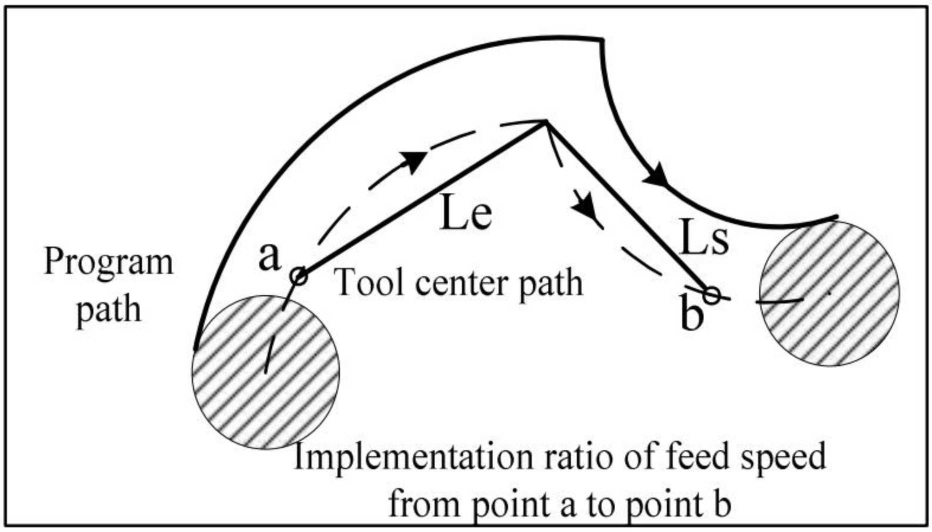



Fig. **4-6-2-3** Arc to arc

6. Given that a program includes straight line to arc and also arc to straight line, as shown in Figure 4-6-2-4, the feedrate is multiplied from Point a to Point b and from Point c to Point d.

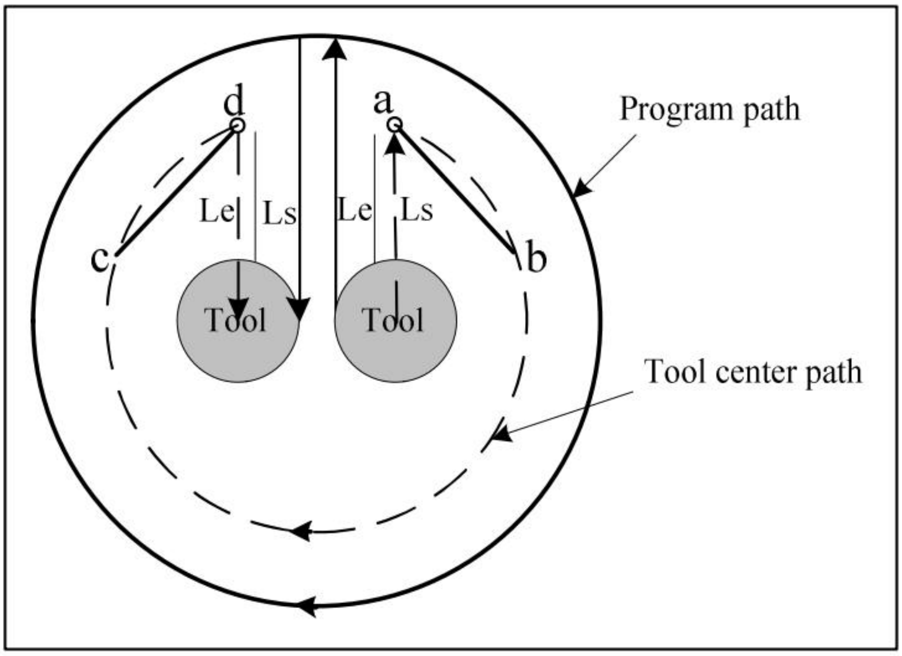



Fig. **4-6-2-4** Straight line to arc, arc to straight line

**Restrictions:**

1. During acceleration/deceleration before interpolation, the inner corner is invalid.
2. If there is a program segment for start of cutting before the corner or a program segment including G41 or G42 after the corner, the inner corner override is invalid.
3. If the offset is zero, the inner corner is not performed.

# 4.7 **Macro Function (G Code)**

## 4.7.1 **User Macro Program**

A certain function realized by a set of codes is pre-stored in the memory like subprograms, and a code is used to represent these functions. These functions can be realized just by writing the representative codes in the program. This set of codes is called user macro program itself, and the representative code is called "user macro code". Sometimes, user macro program itself is also known as macro program, and user macro code as macro program calling code.

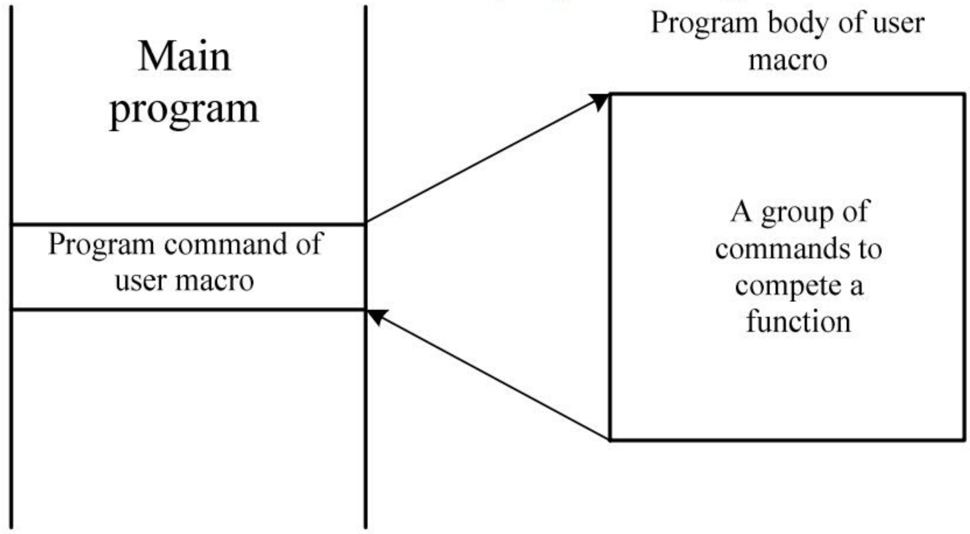



Fig.**4-7-1-1**

Variables can be used in the user macro program itself. Operations can be performed among variables which can be assigned with macro code.

## 4.7.2 **Macro Variables**

In a user macro program, general CNC commands can be used, as well as variables, operation and jump codes.
A user macro program starts with program number and ends with M99.

```
O0066;                    Program No.
G65 H01 ……;               Operation command
G90 G00 X#101 ……;         Use variable CNC
                          command
……
……
……
G65 H82 ……;               Transferring command
……
……
M99;                      Program body of user
                          macro completed
```

Fig. **4-7-1-2** Composition of user macro program itself

**1. How to use variables**

Variables can be used to specify the parameter values in user macro program itself. The value of a variable can be assigned by the main program or set by LCD/MDI, or assigned with the result figured out when the user macro program itself is executed.
Several variables can be used, which are distinguished by variable number.
(1) Representation of variables
Use # followed by a variable number to represent the variable; the format is as follows:

#i (i = 1， 2， 3， 4 ……)

(Example) #5， #109， #1005

(2) Reference to variables

A variable can replace the value after a parameter value.

(Example) F#103 In case of #103=15, it is the same as the F15 command.

G#130 In case of #130=3, it is the same as G3.

**Note:** 1. Reference to variables is not applicable to the parameters O and N (program number and sequence number). Programming with O#100, N#120 is not allowed.

2. When exceeding the maximum code value specified by the parameter, the variable cannot be used. In case of #30=120, M#30 exceeds the maximum code value.

3. Display and setting of variable values: The value of a variable can be displayed on the LCD screen, and can also be set in the MDI mode.

**2. Classification of variables**

Variables can be classified into empty variables, local variables, common variables, and system variables, which have different purposes and properties.

(1) Empty variables #0: (The variable is always empty and no value can be assigned to the variable)

(2) Local variables #1 - #50: local variables can only be used to store data in macro programs; Position Parameter NO: 52#7 can be used for reset setting or determine whether to delete data after emergency stop. When a macro program is called, the local variables are assigned by independent variables.

(3) Common variables #100 - #199, #500 - #999: Position parameter NO: 52#6 can be used for reset settings or determine whether to eliminate the common variables #100 - #199 after emergency stop.

Common variables can be shared in the main program and all user macro programs called by the main program. That is, the variable #i used in a certain user macro program is the same as #i used in other macro programs. Therefore, the common variable #i as a result of operation in a certain macro program can be used in other macro programs.

If there are no rules about the purpose of common variables in the system, these variables can be freely used by users.

Table **4-7-1-1**

| Variable number | Variable type | Function |
|---|---|---|
| ＃100～＃199 | Common variable | Cleared when the power is cut off and all reset to "empty" when power is on. |
| ＃500～＃999 | | The data is saved in the file and will not be lost even in case of outage |

(4) System variables: System variables are used to read and write changes to various data during CNC runtime, As shown below:

1) Interface input signal #1000 --- #1015 (Read by bit the signal input by PLC to the system, i.e. G signal)

#1032 (Read by byte the signal input by PLC to the system, i.e. G signal)

2) Interface output signal #1100 --- #1115 (Write by bit the signal output from the system to PLC, i.e. F signal)

#1132 (Write by byte the signal output from the system to PLC, i.e. F signal)

3)Tool length compensation value #1500 --- #1755 (readable and writable)

4) Length wear compensation value #1800 --- #2055 (readable and writable)

5) Tool radius compensation value #2100 --- #2355 (readable and writable)

6) Radius wear compensation value #2400 --- #2655 (readable and writable)

7) Alarm #3000

8) User Data Table #3500 --- #3755 (read only, not writable)

9) Modal information #4000 --- #4030 (read only, not writable)

10) Position information #5001 --- #5030 (read only, not writable)

11) Workpiece zero offset #5201 --- #5235 (readable and writable)

12) Additional workpiece coordinate system #7001 --- #7250 (readable and writable)

**3. Detailed description of system variables**

1) Modal information

## 3. Detailed description of system variables

1）Modal information

**Table 4-7-1-2**

| Variable number | Function | Group number |
|---|---|---|
| #4000 | G10,G11 | Group 00 |
| #4001 | G00,G01,G02,G03 | Group 01 |
| #4002 | G17,G18,G19 | Group 02 |
| #4003 | G90,G91 | Group 03 |
| #4004 | G94,G95 | Group 04 |
| #4005 | G54,G55,G56,G57,G58,G59 | Group 05 |
| #4006 | G20,G21 | Group 06 |
| #4007 | G40,G41,G42 | Group 07 |
| #4008 | G43,G44,G49 | Group 08 |
| #4009 | G73,G74,G76,G80,G81,G82,G83,G84,G85,G86,G87,G88,G89 | Group 09 |
| #4010 | G98,G99 | Group 10 |
| #4011 | G15,G16 | Group 11 |
| #4012 | G50,G51 | Group 12 |
| #4013 | G68,G69 | Group 13 |
| #4014 | G61,G62,G63,G64 | Group 14 |
| #4015 | G96,G97 | Group 15 |
| #4016 | To be extended | Group 16 |
| #4017 | To be extended | Group 17 |
| #4018 | To be extended | Group 18 |
| #4019 | To be extended | Group 19 |
| #4020 | To be extended | Group 20 |
| #4021 | To be extended | Group 21 |
| #4022 | D | |
| #4023 | H | |
| #4024 | F | |
| #4025 | M | |
| #4026 | S | |
| #4027 | T | |
| #4028 | N | |
| #4029 | O | |
| #4030 | P (additional workpiece coordinate system selected currently) | |

**Note 1:** The **P** code is the currently selected additional workpiece coordinate system.
**Note 2:** When G#4002 is executed, the value obtained in #4002 is 17, 18, or 19.
**Note 3:** Modal information can be read only and cannot be written.

2) Current position information

**Talble 4-7-1-3**

| Variable number | Position information | Related coordinate system | Read operation during movement | Tool offset value |
|---|---|---|---|---|
| #5001 | X-axis program segment end position (ABSIO) | Workpiece coordinate | OK | Tool tip position (which is specified by the program) not considered |
| #5002 | Y-axis program segment end position (ABSIO) | | | |
| #5003 | Z-axis program segment end position (ABSIO) | | | |
| #5004 | 4th-axis program segment end position (ABSIO) | | | |
| #5006 | X-axis program segment end position (ABSMT) | Machine tool | No. | Consider the tool |

| | | | | |
|---|---|---|---|---|
| #5007 | Y-axis program segment end position (ABSMT) | coordinate system | | reference point position (machine tool coordinate) |
| #5008 | Z-axis program segment end position (ABSMT) | | | |
| #5009 | 4th-axis program segment end position (ABSMT) | | | |
| #5011 | X-axis program segment end position (ABSOT) | Workpiece coordinate | | |
| #5012 | Y-axis program segment end position (ABSOT) | | | |
| #5013 | Z-axis program segment end position (ABSOT) | | | |
| #5014 | 4th-axis program segment end position (ABSOT) | | | |
| #5016 | X-axis program segment end position (ABSKP) | | OK | |
| #5017 | Y-axis program segment end position (ABSKP) | | | |
| #5018 | Z axis program segment end position (ABSKP) | | | |
| #5019 | 4th-axis program segment end position (ABSKP) | | | |
| #5021 | X-axis tool length compensation value | | | |
| #5022 | Y-axis tool length compensation value | | No. | |
| #5023 | Z-axis tool length compensation value | | | |
| #5024 | 4th-axis tool length compensation value | | | |
| #5026 | X-axis servo position compensation | | | |
| #5027 | Y-axis servo position compensation | | | |
| #5028 | Z-axis servo position compensation | | | |
| #5029 | 4th-axis servo position compensation | | | |

**Note 1:** ABSIO: Coordinate values of the end point of the previous program segment in the workpiece coordinate system.
**Note 2:** ABSMT: Current position in the machine tool coordinate system.
**Note 3:** ABSOT: Current position in the workpiece coordinate system.
**Note 4:** ABSKP: In the workpiece coordinate system, the position where the skip signal is valid in Program Segment G31.

3）Workpiece zero offset and additional zero offset:

**Table 4-7-1-4**

| Variable number | Function |
|---|---|
| #5201<br>…<br>#5204 | 1st-axis external workpiece zero offset value<br>…<br>4th-axis external workpiece zero offset |
| #5206<br>…<br>#5209 | 1st-axis G54 workpiece zero offset value<br>…<br>4th-axis G54 workpiece zero offset |
| #5211<br>…<br>#5214 | 1st-axis G55 workpiece zero offset value<br>…<br>4th-axis G55 workpiece zero offset |
| #5216<br>… | 1st-axis G56 workpiece zero offset value<br>…<br>4th-axis G56 workpiece zero offset |

| | |
|---|---|
| #5219 | |
| #5221<br>…<br>#5224 | 1st-axis G57 workpiece zero offset value<br>…<br>4th-axis G57 workpiece zero offset |
| #5226<br>…<br>#5229 | 1st-axis G58 workpiece zero offset value<br>…<br>4th-axis G58 workpiece zero offset |
| #5231<br>…<br>#5234 | 1st-axis G59 workpiece zero offset value<br>…<br>4th-axis G59 workpiece zero offset |
| #7001<br>…<br>#7004 | 1st-axis G54 P1 workpiece zero offset<br>…<br>4th-axis G54 P1 workpiece zero offset |
| #7006<br>…<br>#7009 | 1st-axis G54 P2 workpiece zero offset<br>…<br>4th-axis G54 P2 workpiece zero offset |
| #7246<br>…<br>#7249 | 1st-axis G54 P50 workpiece zero offset<br>…<br>4th-axis G54 P50 workpiece zero offset |

**4） Local variables**

Correspondence between address and local variables:

**Table 4-7-1-5**

| Address of independent variables | Local variable number | Address of independent variables | Local variable number |
|---|---|---|---|
| A | #1 | Q | #17 |
| B | #2 | R | #18 |
| C | #3 | S | #19 |
| I | #4 | T | #20 |
| J | #5 | U | #21 |
| K | #6 | V | #22 |
| D | #7 | W | #23 |
| E | #8 | X | #24 |
| F | #9 | Y | #25 |
| M | #13 | Z | #26 |

**Note 1:** a variable is assigned in such a form of English letter followed by a value. Except for G, L, O, N, H and P, the rest of 20 English letters can be used to assign values to independent variables. Each letter is used for one assignment, from A-B-C-D… to X-Y-Z. Assignments do not have to be done in alphabetical order, and the addresses that are not assigned can be omitted.

**Note 2:** G65 must be specified before any independent variable is used.

**5. Considerations about user macro program itself**

1) Input with # Key

Press down # Key to input # following G, X, Y, Z, R, I, J, K, F, H, M, S, T, P or Q.

2) In the MDI state, operation or jump codes can also be specified.

3) H, P, Q, and R of operation or jump code are used as parameters of the G65 command before and after G65.

H02 G65 P#100 Q#101 R#102;　　Correct.

N100 G65 H01 P#100 Q10;　　Correct

4) The input range of a variable cannot exceed fifteen significant digits. The operation result cannot exceed an integer of nine digits, and the manual input range of the variable is eight significant digits.

5) The result of variable value calculation can be a decimal with an accuracy of 0.0001. Only H11 (or operation), H12 (and operation), H13 (non-operation) and H23 (remainder operation) will ignore the fractional part of a variable in the calculation process, and other operations will not round off the decimal point for operation.

Example:
#100 ＝ 35， #101 ＝ 10， #102 ＝ 5
#110 ＝ #100÷#101 (＝3.5)
#111 ＝ #110×#102 (＝17.5)
#120 ＝ #100×#102 (＝175)
#121 ＝ #120÷#101 (＝17.5)

6) The execution time of operation and jump codes varies depending on specific conditions, generally 10 ms on average.

7) When a variable value is undefined, this variable will become an “empty” variable. Variable #0 is always an empty variable. It can be read only and cannot be written.
a. Reference to variables
When an undefined variable is referenced, the address itself is also ignored.
For example:

1) #111 ＝ #110×#102 (＝17.5)

2) #120 ＝ #100×#102 (＝175)

3) #121 ＝ #120÷#101 (＝17.5)

Except for assignments with <empty>, <empty> is the same as 0 in the other cases.

# 4.7.3 **User Macro Program Call**

When G65 is specified, the user macro program specified by address P is called and the data is passed to the user macro program itself via independent variables.

**The format is as follows:**

G65 P □□□□□L□□□□ <specified independent variable>;
Number of calls
Program number of the called macro program itself

After G65, the program number of user macro program is specified by the address P, the number of calls is specified by L, and the data is passed to the macro program via independent variables.

When repeating is required, the number of repetitions from 1 to 9999 is specified after the address L, and when L is omitted, the default number is 1.

The independent variable can be used to assign its value to the corresponding local variable.

**Note 1:** An alarm is generated when the subprogram number specified by the address P cannot be retrieved (PS 078).

**Note 2:** The subprograms No. 90000~99999 are the system retained programs. When the user calls such subprograms, the system can execute the subprogram content, but the cursor will stay in the G65 program segment, and the program interface will always display the main program content. (The subprogram contents can be displayed by modifying Position Parameter N0: 27#4.)

**Note 3:** Macro programs cannot be called in DNC mode.

**Note 4:** Macro program calls can be nested up to five levels.

# 4.7.4 **User Macro Program - Function A**

**1. General form:**

G65 Hm P#i Q#j R#k;
m: 01 to 99 indicate the function of operation code or jump code.
#i: Variable name in which the operation result is stored.
#j: Variable Name 1 to perform the operation. It can also be a constant. Directly represented by
the constant without #.
#k: Variable Name 2 to perform the operation. It can also be a constant.
Meaning: #i = #j ○ #k
Operational symbol, specified by Hm

(Example) P#100 Q#101 R#102……#100 = #101 ○ #102 ;
P#100 Q#101 R15 ……#100 = #101 ○ 15 ;
P#100 Q-100 R#102……#100 = -100 ○ #102

The H code specified by G65 has no effect on the selection of depth offset.

| G code | H code | Function | Definitions |
|---|---|---|---|
| G65 | H01 | Assignment | #i = #j |
| G65 | H02 | Addition | #i = #j + #k |
| G65 | H03 | Subtraction | #i = #j - #k |
| G65 | H04 | Multiplication | #i = #j × #k |
| G65 | H05 | Division | #i = #j ÷ #k |
| G65 | H11 | Logical add (OR) | #i = #j OR #k |
| G65 | H12 | Logic multiplication (AND) | #i = #j AND #k |
| G65 | H13 | Exclusive OR | #i = #j XOR #k |
| G65 | H21 | Square root | #i = $\sqrt{\#j}$ |
| G65 | H22 | Absolute value | #i = \|#j\| |
| G65 | H23 | Take the remainder | #i=#j-trunc(#j ÷ #k)×#k |
| G65 | H26 | Compound multiplication and division | #i = (#i × #j) ÷ #k |
| G65 | H27 | Compound square root | #i = $\sqrt{\#j^2+\#k^2}$ |
| G65 | H31 | Sine | #i = #j×SIN(#k) |
| G65 | H32 | Cosine | #i = #j×COS(#k) |
| G65 | H33 | Tangent | #i = #j×TAN(#k) |
| G65 | H34 | Inverse tangent | #i = ATAN(#j/#k) |
| G65 | H80 | Unconditional transfer | 转向N |
| G65 | H81 | Conditional transfer 1 | IF #j = #k， GOTO N |
| G65 | H82 | Conditional transfer 2 | IF #j ≠ #k， GOTO N |
| G65 | H83 | Conditional transfer 3 | IF #j > #k， GOTO N |
| G65 | H84 | Conditional transfer 4 | IF #j < #k， GOTO N |
| G65 | H85 | Conditional transfer 5 | IF #j ≥ #k， GOTO N |
| G65 | H86 | Conditional transfer 6 | IF #j ≤ #k， GOTO N |
| G65 | H99 | Alarm | |

Fig. **4-7-4-1**

**2. Operation code:**

Operation commands

1) Assignment of macro variables: # I = # J

**G65 H01 P#I Q#J**

(Example) G65 H01 P# 101 Q1005; (#101 = 1005)

G65 H01 P#101 Q#110; (#101 = #110)

G65 H01 P#101 Q-#102; (#101 = -#102)

2) Decimal add operation: # I = # J+# K

**G65 H02 P#I Q#J R#K**

(Example) G65 H02 P#101 Q#102 R15; (#101 = #102+15)

3) Decimal subtract operation: # I = # J－# K

**G65 H03 P#I Q#J R# K**

(Example) G65 H03 P#101 Q#102 R#103; (#101 = #102－#103)

4) Decimal multiplication operation: # I = # J×# K

**G65 H04 P#I Q#J R#K**

(Example) G65 H04 P#101 Q#102 R#103; (#101 = #102×#103)

5) Decimal division operation: # I = # J÷# K

**G65 H05 P#I Q#J R#K**

(Example) G65 H05 P#101 Q#102 R#103; (#101 = #102÷#103)

6) Binary logic add(or) : # I = # J.OR. # K

**G65 H11 P#I Q#J R#K**

(Example) G65 H11 P#101 Q#102 R#103; (#101 = #102.OR. #103)

7) Binary logic multiply(and) : # I = # J.AND. # K

**G65 H12 P#I Q#J R#K**

(Example ) G65 H12 P# 201 Q#102 R#103; (#101 = #102.AND.#103)

8) Binary executive or: # I = # J.XOR. # K

**G65 H13 P#I Q#J R#K**

(Example) G65 H13 P#101 Q#102 R#103; (#101 = #102.XOR. #103)

9) Decimal square root: # I = #J

**G65 H21 P#I Q#J**

(Example) G65 H21 P#101 Q#102 ; ( #101 = #102 )

10) Decimal absolute value: # I = | # J |

**G65 H22 P#I Q#J**

(Example) G65 H22 P#101 Q#102 ; (#101 = | #102 |)

11) Decimal remainder: # I = # J—TRUNC(#J/#K)×# K,TRUNC: Omit decimal fraction

**G65 H23 P#I Q#J R#K**

(Example) G65 H23 P#101 Q#102 R#103; (#101 = #102- TRUNC (#102/#103)×#103

12) Decimal converting into binary: # I = BIN (# J)

**G65 H24 P#I Q#J**

(Example) G65 H24 P#101 Q#102 ; (#101 = BIN(#102) )

13) Binary converting into decimal: # I = BCD (# J)

**G65 H25 P#I Q#J**

(Example) G65 H25 P#101 Q#102 ; (#101 = BCD(#102) )

14) Decimal multiplication/division operation: # I =(# I×# J) ÷# K

**G65 H26 P#I Q#J R# k**

(Example) G65 H26 P#101 Q#102 R#103; (#101 =(# 101×# 102) ÷#103)

15) Compound square root: $\#I = \sqrt{\#J^2+\#K^2}$

**G65 H27 P#I Q#J R#K**

(Example) G65 H27 P#101 Q#102 R#103; (#101 = $\sqrt{\#102^2+\#103^2}$ )

16) Sine: # I = # J•SIN(# K) (Unit: ‰)

**G65 H31 P#I Q#J R#K**

(Example) G65 H31 P#101 Q#102 R#103; (#101 = #102•SIN(#103) )

17) Cosine: # I = # J•COS(# K) (Unit: ‰)

**G65 H32 P#I Q#J R# k**

(Example) G65 H32 P#1Q#102 R#103; (#101 =#102•COS(#103) )

18) Tangent: # I = # J•TAM(# K) (Unit: ‰)

**G65 H33 P#I Q#J R# K**

(Example) G65 H33 P#101 Q#102 R#103; (#101 = #102•TAM(#103) )

19) Cosine: # I = ATAN(# J /# K) (Unit: ‰)

**G65 H34 P#I Q#J R# k**

(Example) G65 H34 P#101 Q#102 R#103; (#101 =ATAN(#102/#103) )

**Jump commands**

1) Unconditional jump

**G65 H80 Pn; n: Block number**

(Example) G65 H80 P120; (jump to N120)

2) Conditional jump 1 #J.EQ.# K ( = )

**G65 H81 Pn Q#J R# K; n: Block number**

(Example) G65 H81 P1000 Q#101 R#102;

The program jumps N1000 when # 101= #102 and executes in order when #101 ≠#102.

3) Conditional jump 2 #J.NE.# K ( ≠ )

**G65 H82 Pn Q#J R# K; n: Block number**

(Example) G65 H82 P1000 Q#101 R#102;

The program jumps N1000 when # 101 ≠ #102 and executes in order when #101 = #102.

4) Conditional jump 3 #J.GT.# K ( ＞ )

**G65 H83 Pn Q#J R# K; n: Block number**

(Example) G65 H83 P1000 Q#101 R#102;

The program jumps N1000 when # 101 ＞ #202 and executes in order when #101 ≤ #102.

5) Conditional jump 4 #J.LT.# K ( ＜ ＝)

**G65 H84 Pn Q#J R# K; n: Block number**

(Example) G65 H84 P1000 Q#101 R#102;

The program jumps N1000 when # 101＜#102 and executes in order when #101≥#102.

6) Conditional jump 5 #J.GE.# K ( ≥ )

**G65 H85 Pn Q#J R# K; n: Block number**

(Example) G65 H85 P1000 Q#101 R#102;

The program jumps N1000 when # 101 ≤ #1 and executes in order when #101 ＜ #102.

7) Conditional jump 6 #J.LE.# K ( ≤ )

**G65 H86 Pn Q#J R# K; n: Block number**

(Example) G65 H86 P1000 Q#101 R#102;

**Note:** A variable can be used to specify the sequence number. For example: G65 H81 P#100 Q#101 R#102; when the condition is satisfied, the program will jump to the program segment with its sequence number specified by #100.

**4. Logical AND, logical OR and logical negation codes**

Example:
G65 H01 P#101 Q3;

G65 H01 P#102 Q5;

G65 H11 P#100 Q#101 Q#102;

In the binary system, 5 is represented as 101, 3 as 011, and the calculation result is #100=7;
G65 H12 P#100 Q#101 Q#102;
In the binary system, 5 is represented as 101, 3 as 011, and the calculation result is #100=1;

**5. Macro variable alarm**

Example:
G65 H99 P1; Macro Variable Alarm 3001
G65 H99 P124; Macro Variable Alarm 3124

**User macro program examples**

**1. Bolt hole cycle**

On the circumference where the circle center is the reference point (X0, Y0) and the radius is (R), the starting angle is (A), and N equant holes are machined.

。

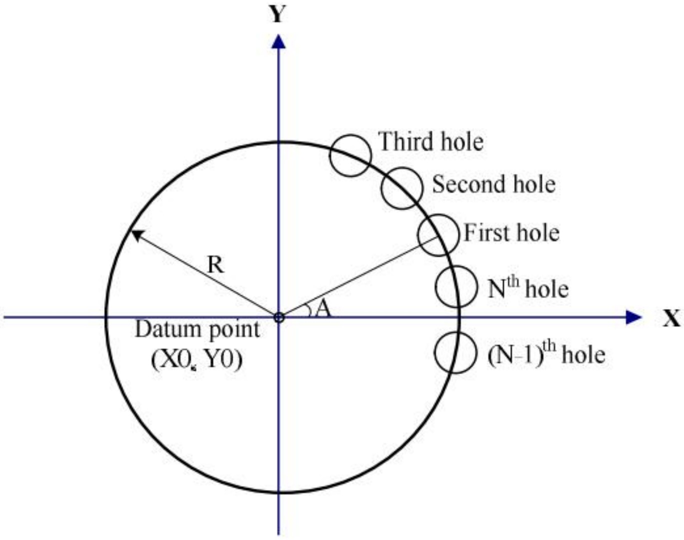



Fig. **4-7-5-1**

X0, Y0 are the coordinate values of the bolt hole cycle reference point.
R: Radius, A: Starting angle, N: The number of holes. The following variables are used in the above parameters.
#500: Coordinate value of reference point in X axis (X0)
#501: Coordinate value of reference point in Y axis (Y0)
#502: Radius (R)
#503: Starting angle (A)
#504: N holes
When N>0, rotate counterclockwise, and the number of holes is N.
When N<0, rotate clockwise, and the number of holes is N.
The following variables are used for operations in macro programs.
#100: indicates the count of Hole I machining (I)
#101: final value of the count (=|N|)(IE)
#102: angle of Hole I (θI)
#103: coordinate value of Hole I in X axis (Xi)
#104: coordinate value of Hole I in Y axis (Yi)

The user macro program itself can be written in the following forms:

| | |
|---|---|
| O9010； | |
| N100 G65 H01 P#100 Q0； | I＝0 |
| G65 H22 P#101 Q#504； | IE＝\|N\| |
| N200 G65 H04 P#102 Q#100 R360； | |
| G65 H05 P#102 Q#102 R#504； | θI＝A＋360°×I/N |
| G65 H02 P#102 Q#503 R#102； | |
| G65 H32 P#103 Q#502 R#102； | X I＝X I＋R·COS(θI) |
| G65 H02 P#103 Q#500 R#103； | |
| G65 H31 P#104 Q#502 R#102； | Y I＝Y I＋R·SIN(θI) |
| G65 H02 P#104 Q#501 R#104； | |
| G90 G00 X#103 Y#104； | |
| G**； | Specific G code for hole machining |

```
G65 H02 P#100 Q#100 R1；                 I＝I＋1
G65 H84 P200 Q#100 R#101；               When I < IE, go to N200 to machine IE holes.
M99；
```

**The followings are examples of programs to call the above user macro program itself:**

```
O0010；
G65 H01 P#500 Q100；     X0=100MM
G65 H01 P#501 Q-200；    Y0=-200MM
G65 H01 P#502 Q100；     R=100MM
G65 H01 P#503 Q20；      A=20°
G65 H01 P#504 Q12；                      N=12rotate counterclockwise
G92 X0 Y0 Z0；
M98 P9010；                              Call the user macro program
G80；
X0 Y0；
M30；
```

# Chapter V Auxiliary Function M Code

The M codes available to user of this machine tool are listed as follows:

**Table 5-1**

| | M code | Function |
|---|---|---|
| M code for program control | M30 | The program ends and returns to the program header, with the number of machined workpieces increased by 1. |
| | M02 | The program ends and returns to the program header, with the number of machined workpieces increased by 1. |
| | M98 | Call subprogram |
| | M99 | End and return of subprogram / repeat execution |
| Controlled by PLC with M code | M00 | Pause of program |
| | M01 | Selective halt of program |
| | M03 | The spindle rotates CW |
| | M04 | The spindle rotates CCW |
| | M05 | The spindle stops |
| | M06 | Replace the tool |
| | M07 | Blowing on |
| | M08 | Cooling on |
| | M09 | Cooling off/blowing off |
| | M10 | Unclamping of A axis |
| | M11 | Clamping of A axis |
| | M16 | Tool control - unclamping |
| | M17 | Tool control - clamping |
| | M18 | Cancel spindle orientation |
| | M19 | Spindle orientation |
| | M20 | Spindle neutral command |
| | M23 | ATC advances |
| | M24 | ATC retreat |
| | M25 | tool number calculation |
| | M26 | Return tool number calculation |
| | M28 | Cancel rigid tapping |
| | M29 | Rigid tapping |
| | M32 | Lubrication open |
| | M33 | Lubrication off |
| | M35 | Activate the chip conveyor |
| | M36 | Turn off the chip conveyor |
| | M45 | Circular buckle tool |
| | M46 | Disk exchange |
| | M47 | Disk Return |
| | M63 | The second main shaft rotates in the forward direction |
| | M64 | Second spindle reversal |
| | M65 | Second spindle stops |

When the motion code and auxiliary function are specified in the same program segment, the code is executed in one of the following two ways:

(1) Mobile code and auxiliary function code are executed simultaneously.

(2) After completing the execution of the mobile code, execute the auxiliary function code.

The selection of the order of the two depends on the settings of the machine tool factory. For detailed information, please refer to the machine tool factory's manual.

When a value is specified after address M, the code signal and gate signal are sent to the machine tool, which uses these signals to turn on or off the machine tool. This feature. Due to limitations in mechanical operations, certain M codes cannot be specified simultaneously. For restrictions on specifying multiple M codes for the same program segment in mechanical operations, please refer to the machine tool manufacturer's manual.

## 5.1 M Code Controlled By PLC

When the M code controlled by PLC shares the same segment with the movement code, the M code will be executed simultaneously with the movement code.

### 5.1.1 Spindle Rotation Cw and Ccw Commands (M03, M04)

**Code:** M03 (M04) Sx x x;
**Description:** According to applicable standards, the spindle rotation CCW is defined as forward rotation and the rotation CW as reverse rotation.
M03 means rotation CW while M04 means rotation CCW.
The Sx x x code refers to the speed of the spindle, or the gear position in case of gear control.
Unit: r/min
When controlled by frequency converter, Sx x x refers to the actual speed, for example: S1000 means that the spindle rotates at a speed of 1000 r/min.

### 5.1.2 Spindle Stop Code Command (M05)

**Code:** M05. When M05 is executed automatically, the spindle will stop rotating. However, the speed specified by the S code will be reserved. The deceleration mode in which the spindle stops rotating depends on the machine tool manufacturer’s settings. Usually, it is dynamic braking.

### 5.1.3 Cooling On and Off （M08、M09） Spindle Blowing On and Off (M07, M09)

**Code:** M08, make the cooling water pump on. M09, make the cooling water pump off. In the automatic mode, if the auxiliary function lock is used, the pump control code is not executed.
M07, make the spindle blowing on. M09, make the spindle blowing off.

### 5.1.4 A-Axis Unclamping and Clamping (M10, M11)

**Code:** M10, unclamping of A axis M11, clamping of A axis

### 5.1.5 Tool Control - Unclamping and Clamping （M16、M17）

**Code:** M16, tool control - unclamping M17, tool control - clamping

### 5.1.6 Spindle Orientation and Cancellation（M19、M18）

**Code:** M18, cancel the spindle orientation M19, spindle orientation for tool replacement positioning

### 5.1.7 Tool library zeroing instruction (M20)

Code: M20, return the tool magazine to zero, return to the No.1 tool position in the tool magazine.

### 5.1.8 Tool library forward and backward code instructions (M23, M24)

Code: M23, tool library forward code; M24， Tool library rollback code.

### 5.1.9 Code instructions for tool library retrieval and tool library return operations (M25, M26)

**Code:** M25, tool library tool retrieval operation code; M26， tool library return tool operation code.

### 5.1.10 Rigid tapping opening and closing (M29, M28)

**Code:** M29, rigid tapping.

### 5.1.11 Lubrication on/off (M32, M33)

**Code:** M32 (M33), controls the start and stop of the lubrication pump. If the auxiliary function lock is encountered in automatic mode, the water pump control code will not be executed.

### 5.1.12 Chip conveyor on/off (M35, M36)

**Code:** M35 (M36), controls the start and stop of the spiral chip conveyor.

### 5.1.13 Automatic tool change buckle, exchange, return (M45, M46, M47)

**Code:** M45 (M46, M47), controls the start and end of automatic tool change.

### 5.1.14 Second spindle forward rotation, reverse rotation, stop (M63, M64, M65)

**Code:** M63, M64, M65, second spindle rotates forward, reverse, and stops.

## 5.2 M Code For Program Control

The M code for program control is classified into main program control and macro program control. When the M code used for program control shares the same segment with the movement code, first executed is the movement code and then the M code.

**Note:** 1. M00, M01, M02, M06, M30, M98, M99 codes cannot be specified together with other M codes, or otherwise the system will give an alarm. When these M codes share the same segment with other non-M codes, first executed are the non-M codes and then the M codes.
2. Such M codes include those which cause CNC to send the M codes themselves to the machine tool and to perform internal operations, such as M codes that invalidate the read-ahead function of a program segment. In addition, the M codes, which only allow CNC to send the M codes themselves to the machine tool without performing internal operations, can be specified in the same program segment.

### 5.2.1 Program End and Return (M30, M02)

When running in automatic mode, the program stops running automatically when it reaches M30 (M02). If any program is executed afterwards, the spindle and cooling operation will be stopped, and the number of workpieces processed will be increased by 1. M30 can use the bit parameter N0: parameter - [Quick Debug] P054 to control whether to return the program header, while M02 can use parameter - [Quick Debug] P053 to control whether to return the program header. If M02 and M30 are at the end of the subroutine, return to the program that called the subroutine and continue executing the following program segments.

### 5.2.2 Program Halt (M00)

In the automatic operation mode, this mode will be temporarily halted when the program runs to M00, and at the moment, the previous modal information will be saved. When the loop start button is pressed, it will continue running. Its function is equivalent to pressing down the feed hold button.

### 5.2.3 Selective Halt of Program （M01）

In the automatic operation mode, this mode can be selectively halted when the program runs to M01. When the “Selection-stop” switch is placed on, M01 and M00 have the same effect; if the “Selection-stop” switch is off, M01 has no effect. Please refer to Part II Operation Instructions for details.

### 5.2.4 Code Command For Program Calling Subprogram（M98）

In a main program, M98 can be programmed to call a subprogram. Specific format:

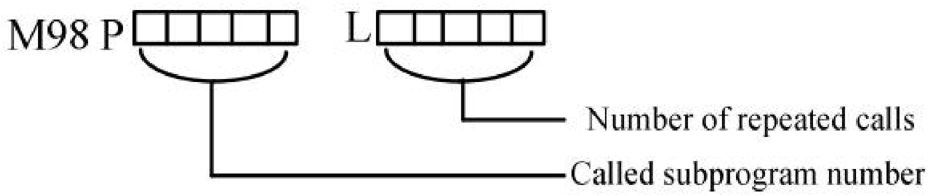



Fig. **5-2-4-1**

### 5.2.5 Program End and Return（M99）

1. In the automatic operation mode, if M99 is used at the end of a main program segment, it will return to the beginning of the program for automatic execution when the program runs to M99. After that, if there is any program that will not be executed, the number of machined workpieces will not increase accumulatively.
2. When M99 is used at the end of a subprogram, it will return to the main program when running to this segment, and the program following the subprogram segment is called to continue execution.
3. In the DNC mode, M99 is used as M30, and the cursor stays at the end of the program.

# Chapter VI Spindle Function S Code

Through the S codes and the values behind, the code signal is converted into an analog signal and sent to the machine tool for spindle control of the machine tool. S is a modal value.

## 6.1 **Spindle Analog Control**

### Spindle control (S5 bit simulation)

By specifying the speed through code S and its subsequent values, the code signal is converted into an analog signal and sent to the machine tool for spindle control.

**Code format: S_**

**Description:**

1. One S code can be specified in one program segment.
2. The address S and the values behind can directly specify the spindle speed (unit: r/min). For example: M3 S300 indicates that the spindle runs at 300 r/min.
3. When the movement code and S code share the same program segment, the movement code will be executed simultaneously with the S functional code.
4. The spindle speed is controlled via the S code and the values behind it.

## 6.2 **Constant Surface Cutting Speed Control G96/G97**

**Code format:**

Constant surface cutting speed control code G96 S_ Surface speed (mm/min or inch/min)
Constant surface cutting speed control cancellation code G97 S_ Spindle speed (r/min)
Controlled shaft code of constant surface cutting speed control G96 P_ P1 X-axis; P2 Y-axis; P3 Z-axis; P4 4th axis
Maximum spindle speed clamp G92 S_ S specified maximum spindle speed (r/min)

**Function:** Specify the surface speed (relative speed between the tool and the workpiece) after S to rotate the spindle and keep the surface cutting speed constant, regardless of the tool position.

**Description:**

1. G96 is the modal code. After the instruction G96, the program enters the constant speed control mode, and the value of S is the surface speed.
2. G96 code must specify the axis around which the constant speed control is adopted. . G97 code cancels G96 means.
3. In order to implement the constant surface cutting speed control, the workpiece coordinate system should be set so that the center coordinate of the rotation axis becomes zero.

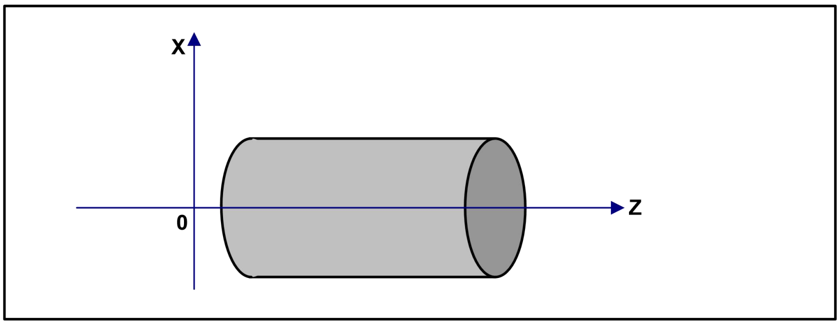



Fig.**6-3-1** Workpiece coordinate system of constant surface cutting speed control

4. When using constant surface cutting speed control, higher than G92 S_ setting, clamp on the highest spindle speed. When the power is switched on and the maximum spindle speed is not set, S in G96 code is treated as S=0 until M3 or M4 appears in the program.

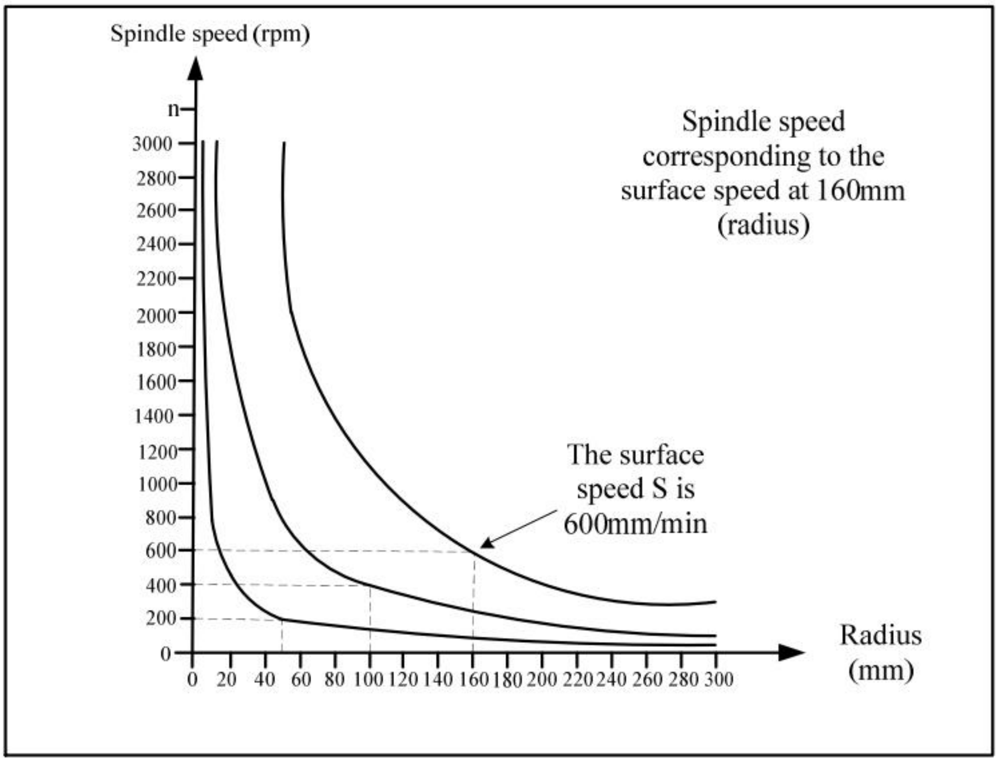
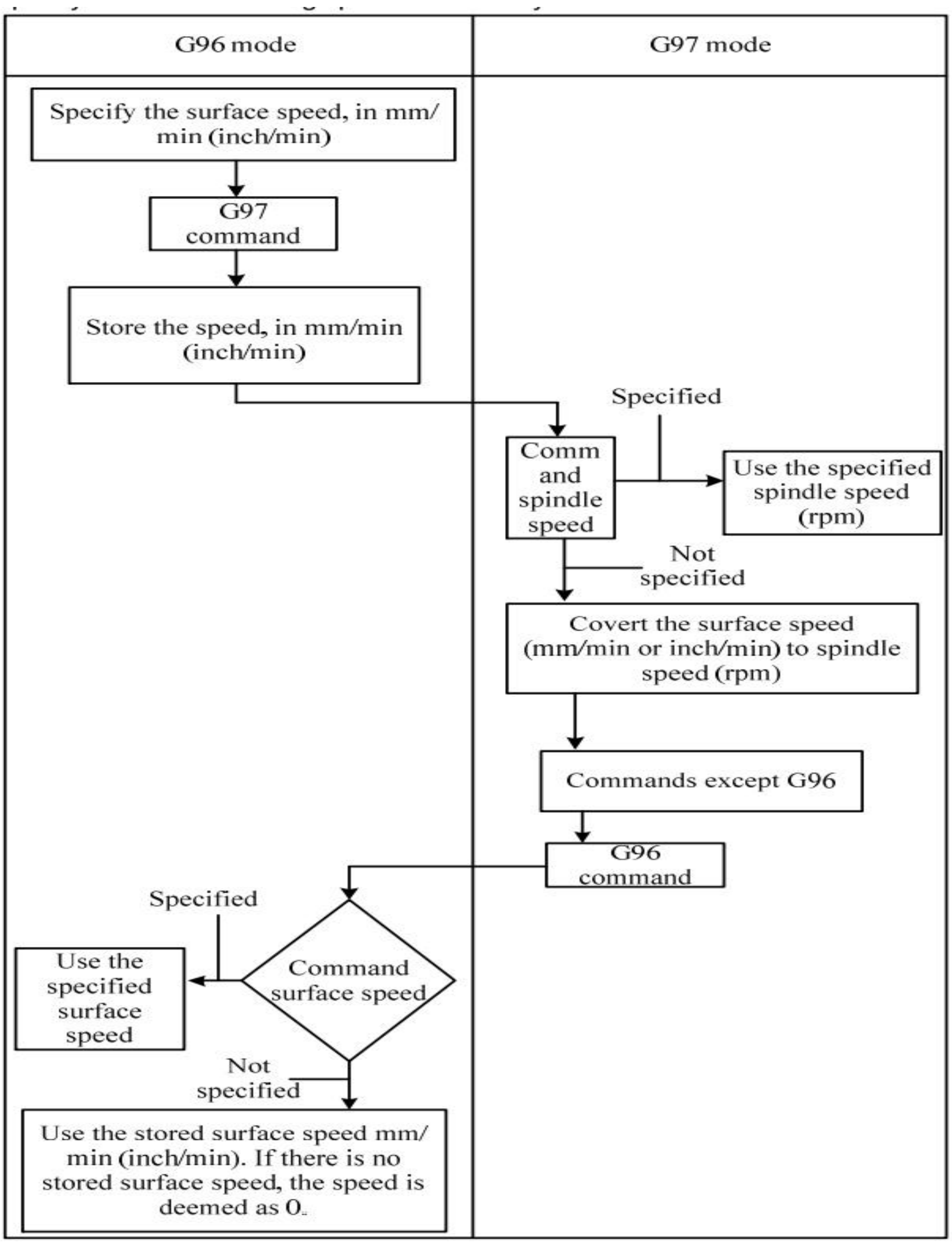



Fig.**6-3-2** Relationship between workpiece radius spindle speed and surface speed

5. Specify the surface cutting speed in G96 way:




6. G96 related parameter setting: Bit parameter No.37#2 is set to calculate the reference coordinate of G96 spindle rotating speed when G0 is positioned fast (0: end point, 1: current point); bit parameter No.37#3 set G96 spindle speed clamping (0: before spindle override, 1: after spindle override), and bit parameter No.61#0 set to determine whether constant cycle speed control is used.

**Restrictions:**

1. Because the response problem in the servo system is not considered when the spindle speed changes, and constant surface cutting speed control is also effective during thread cutting, G97 is used to cancel the constant surface cutting speed control before thread processing.
2. In the fast moving program segment specified by G00, the constant surface speed control is not calculated based on the instantaneous change in the tool position, but based on the end point of the segment; because the fast movement will not lead to cutting, no constant surface cutting speed is needed.
3. In the process of soft tapping, rigid tapping or deep-hole rigid tapping, it is necessary to cancel the constant surface cutting speed with G97 first, or otherwise there will be incorrect thread or broken tap.

# Chapter VII Feed Function F code

The feed function controls the feeding speed of the tool. The feed function and control mode are as follows:

## 7.1 **Fast Movement**

Use code (G00) for fast positioning. Fast feed speed is set by the data Parameter - [Feed axis parameter] P026-P028 The override adjustment key on the operation panel can be used for the following override adjustment:


Fig. **7-1-1** Fast feed override key

Where, F0 is set by the data Parameter - [Feed axis parameter] P052

The acceleration of fast positioning (G0) can be set by the data parameters **P110-P114**, and the acceleration / deceleration time constant can be set by the data Parameter - [Acceleration/Deceleration Parameters] P009-P011. According to the response characteristics of the machine tool and the motor, it can be set reasonably.

**Note:** In the G00 program section, even if the feed speed F code is specified, it is invalid. The system is positioned at G0 speed.

## 7.2 **Cutting Speed**

In linear interpolation (G01) and circular interpolation (G02, G03), the number following F code is used to command the feed speed of the tool. The unit is mm/min. The tool moves at the cutting feed rate programmed. Use the feed override key on the operation panel of the machine tool to implement the feed override of cutting (the range of the feed override adjustment is: 0％-150%).

The cutting feed rate in automatic mode when the power is turned on is set by the parameter - [feed axis parameter] P051.

The cutting speed can be specified in the following two ways:;

A)、 Feed per minute (G94): After F, specify the tool feed rate per minute.

B)、 Feed per revolution (G95): After F, specify the tool feed per revolution of the spindle.

**Note: When F specifies the cutting speed, the system displays it as an integer value. When the input value is a non integer, the system will display the value rounded to the nearest decimal point. The system still processes actual input values internally. When specifying the pitch, one decimal place can be displayed. Internally, the system processes actual input values.**

### 7.2.1 **Feed Per Minute (G94)**

**Code format: G94 F_**

**Function:** Tool feed per minute. Unit: mm/min or inch/min.

**Description:**

1. After G94 (feed per minute) is specified, the tool feed per minute is directly specified by the value following F.
2. G94 is modal code and, once specified, is valid until G95 is specified. When starting up, the feed mode per minute is in default, and the default cutting feed speed is set by the data parameter P87.

3. The **feed per minute** can be adjusted by the override adjustment key or the band switch on the panel, with the override from 0% to 150%.

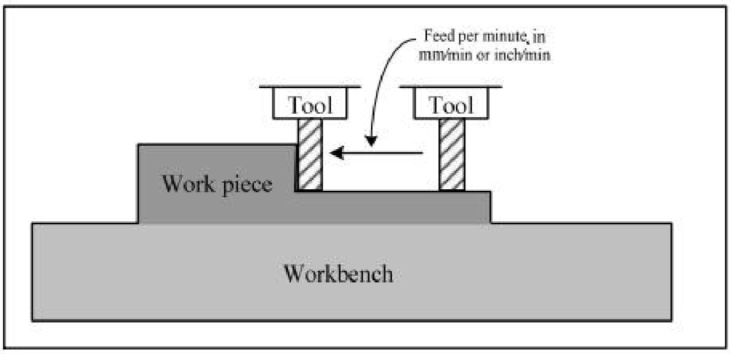



Fig.**7-2-1-1 Feed per minute**

## 7.2.2 **Feed Per Revolution(G95)**

**Code format: G95 F_**
**Function:** Tool feed per revolution. Unit: mm/r or inch/r.
**Description:**

1. Machine tools must be installed with spindle encoder to use this function.
2. After G95 (feed mode per revolution) is specified, the feed amount per revolution is directly specified by the value following F.
3. G95 is modal code and, once specified, is valid until G94 is specified. The feed speed per revolution in initialization is zero by default.
4. The feed per revolution can be adjusted by the override adjustment key or the band switch on the panel, with the override from 0% to 150%.

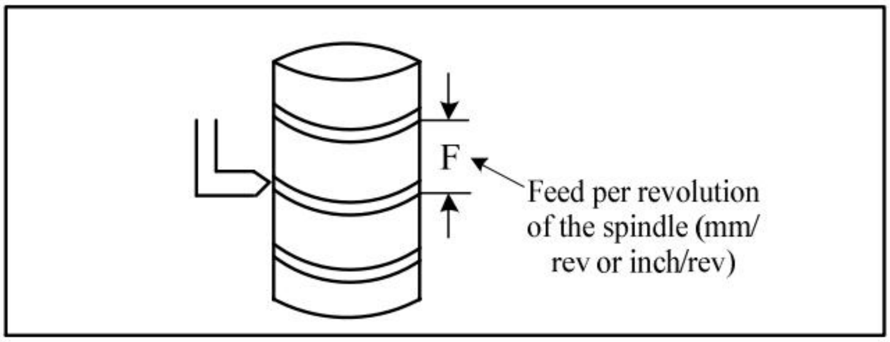



**图 7-2-2-1** Feed per revolution

**Note:** When the spindle speed is low, the feed speed may fluctuate. The lower the spindle speed is, the more frequently the feed amount fluctuates.

## 7.3 **Tangential Speed Control**

Generally, the cutting feed is to control the speed of the tangent direction of the contour track so as to reach the instructed speed value.

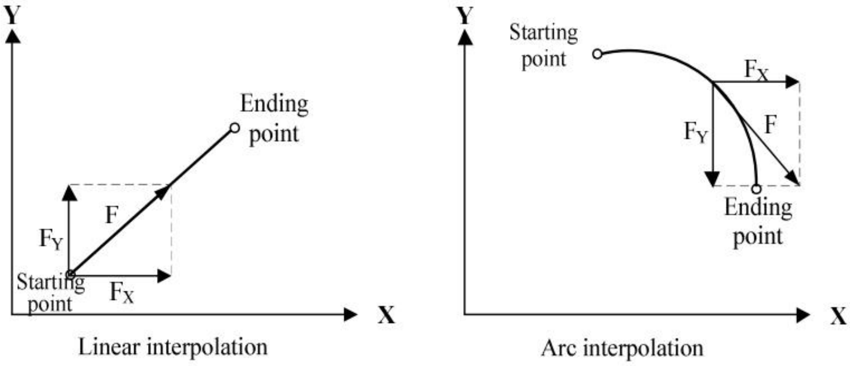



Fig. 7-3-1

$$F=\sqrt{F_X^2+F_Y^2+F_Z^2}$$

F: Speed in tangent direction
Fx: Speed in X axis direction
Fy: Speed in Y axis direction
Fz: Speed in Z axis direction

## 7.4 **Feed Speed Override Key**

The feed override under manual mode and automatic mode can be adjusted through the override adjustment key on the operation panel, with the range of 0-150% (10% for each gear, 21 gears in total). Under automatic mode, when the override adjustment key is set to zero, the system will stop feeding, showing the cutting override is zero; then the override adjustment key should be adjusted to keep the program running.

## 7.5 **Automatic Acceleration and Deceleration**

The system drives the motor to automatically accelerate and decelerate at the beginning and end of movement; so it can start and stop smoothly. It also automatically accelerates and decelerates when the movement speed changes; so the speed change can proceed smoothly. Therefore, there is no need to consider acceleration and deceleration when programming.

Fast feed: Forward acceleration and deceleration (0: linear; 1: S-type) Post acceleration and deceleration (0: linear; 1: exponential type).

Cutting feed: Forward acceleration and deceleration (0: linear; 1: S-type) Post acceleration and deceleration (0: linear; 1: exponential type).

Manual feed: Post acceleration and deceleration (0: linear; 1: exponential type).

(Use parameters to set the universal time constant of each axis)

## 7.6 **Acceleration and Deceleration Processing At Program Segment Corner**

**For example:** In the previous program segment, only Y moves, while in the next program segment, only X moves. When Y decelerates, X accelerates, and the path of the tool is as follows:

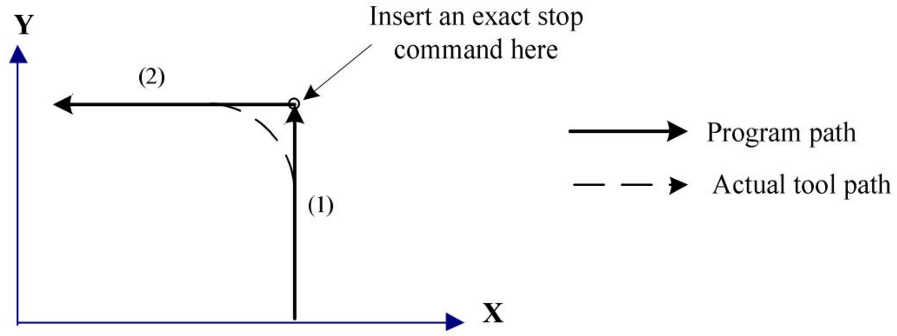



Fig.**7-6-1**

If the exact-stop code is added, the tool will move according to the program instructions as shown in the solid line above. Otherwise, the greater the cutting feed speed, or the longer the acceleration and deceleration time constant, the greater the radiant of the corner is. In arc command, the arc radius of the actual tool path is smaller than that given by the program. To reduce the corner error, the acceleration and deceleration time constant should be reduced as far as possible if the mechanical system allows.

# Chapter VIII Tool Functions

## 8.1 Tool Functions

Specify a value (up to 8 digits) after address T to select the tool on the machine.

In principle, two or more T codes cannot be instructed in the same program segment. If the same group of codes are set in the same segment, no alarm will appear. Please refer to the T code that appears later. For the number of digits that can be specified by address T and the machine tool action corresponding to T code, please refer to the operation manual of the machine tool plant.

When the movement code and T code are specified in the same program segment, the movement code and T code are implemented simultaneously.

When T code and tool change code M06 are in the same segment, T code will be implemented first, followed by the tool change code. When T code and tool change code M06 are in different segments, it is necessary to check whether the spindle tool number is consistent with code T tool during tool code change. If it is consistent, tool change will not be implemented.

**As shown in the following procedure example:**

```
O00010；
  N10 T2M6；        The tool on the spindle is T2
  N20 M6T3；        The tool on the spindle is T3
  N30 T4；          The tool on the spindle is T4
  N40 M6；          The tool on the spindle is T4
  N50 T5；          The tool on the spindle is T5
  N60 M30
%
```

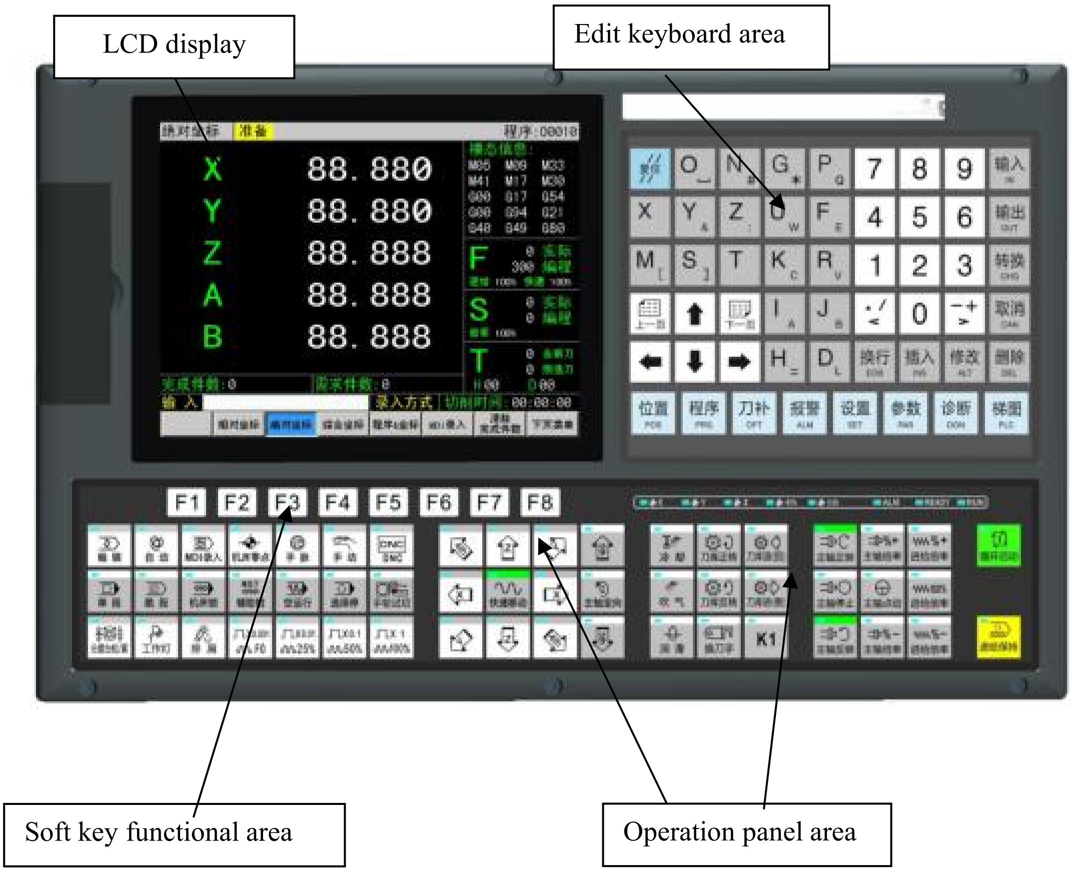
After the tool change, the tool on the spindle is T4.

# II Operation

# Chapter 1 Operation Panel

## 1.1 Panel division

The 980MFi CNC system has an integrated operation panel, which is divided into four main areas: LCD (liquid crystal display) area, editing keyboard area, soft key function area, and machine control area, as shown in the following figure:




## 1.2 Panel Function Description

### 1.2.1 LCD (Liquid Crystal Display) display area

The LCD of this system adopts a color 8-inch LCD display with a resolution of 800 × 600.

## 1.2.2 Edit keyboard area

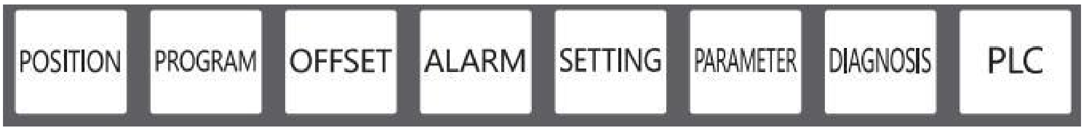

图 1-2-2-1

In the keyboard editing area, the functions of the keys are further divided into 12 zones, and the specific usage instructions for each zone are as follows:

| No | Name | Function |
|---|---|---|
| 1 | RESET key | CNC reset, feed, output stop etc. |
| 2 | Address key | Address input<br>Double address key, switching them by pressing it repetitively |
| 3 | Number key | Number input |
| 4 | Cancel key | Delete input characters (characters not stored in the buffer) and data; Cancel the last operation. |
| 5 | EOB key | Input numbers, addresses, or data into the buffer; Confirm the operation result. |
| 6 | Delete key | Delete input characters (characters not stored in the buffer). |
| 7 | F1--F8 key | Press any of the keys to enter the corresponding interface display. |
| 8 | PageUp<br>PageDown | Used for page conversion and program flipping in the same display mode. |
| 9 | ←↑→<br>↓ | Can move the cursor up, down, left, and right. |
| 10 | Alter key | Program character replacement. |
| 11 | Output key | Used for program output, parameter backup output, and other operations. |
| 12 | Input key | Used for program input, parameter input, parameter restoration input, and other operations. |

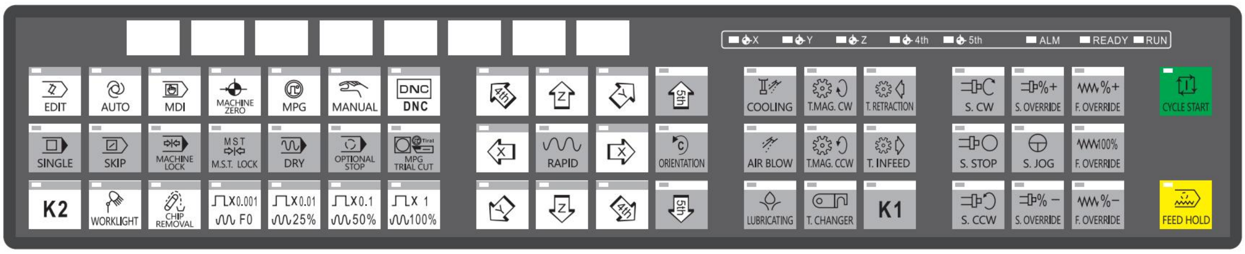
### 1.2.3 Introduction to screen operation keys

This system has a total of 8 operation page display keys arranged on the operation panel, as shown in the following figure:




| Name | Function Description | remarks |
|---|---|---|
| POSITION | Enter the location page | By using the corresponding soft key conversion, display the relative coordinates, absolute coordinates, total coordinates, process monitoring, and MDI display pages of the current point. |
| PRGGRAM | Enter the program page | By using the corresponding soft key conversion, the program and directory display pages are displayed, and the directory interface can view multiple page program names through the page turning key. |
| OFTSET | Enter the tool repair page | There are five interfaces in total, which can be converted and displayed through corresponding soft keys. The length and radius compensation of the tool can be set, as well as the pitch error compensation of each feed axis. By using the corresponding soft key conversion, display the bit parameter, number parameter, and macro variable 1 and macro variable 2 pages for viewing or modifying parameters and variables. |
| ALMRM | Enter the alarm page | There are five interfaces in total, which can be converted and displayed through corresponding soft keys. The length and radius compensation of the tool can be set, as well as the pitch error compensation of each feed axis. |
| SETTING | Enter the settings page | There are five interfaces in total, which can be used to convert display settings, parameter switches, coordinate settings, data and graphic display page settings through corresponding soft keys. |
| PARAMETER | Enter the parameter page | By using the corresponding soft key conversion, display the graphic parameters, and set the center, size, and scale of the displayed graphic for the graphic parameters. |
| DIAGNOSIS | Enter the diagnostic page | View the signal status of the I/O ports on each side of the system through the corresponding soft key conversion. |
| PLC | Enter the ladder diagram page | PLC page includes three subpages: PLC state, PLCmonitor, PLC data, Program list. |

### 1.2.4 Machine tool control area




| Button name | Select mode | Function | Remarks and operating instructions |
|---|---|---|---|
| Edit mode key | Edit Method Selection Key | To enter Edit mode | Switch to editing mode during automatic operation, and only switch to it after the system finishes running the current |

| | | | |
|---|---|---|---|
| | | | segment and stops. |
| Auto mode key | Automatic mode selection key | To enter Auto mode | When selecting automatic mode, the system selects the internal memory program. |
| MDI mode key | Input (MDI) mode selection key | To enter MDI mode | Automatically switch to MDI input mode, and only switch to it after the system finishes running the current segment and stops. |
| Machine zero return mode key | Machine tool zeroing mode selection key | To enter Machine zero return mode | When switching to zero return mode during automatic operation, the system immediately decelerates and stops. |
| Manual mode key | Manual mode selection key | To enter Manual mode | When switching to manual mode during automatic operation, the system immediately decelerates and stops. |
| Step/MPG mode key | Hand pulse mode selection key | To enter Step or MPG mode (one mode by parameter) | When switching to manual pulse mode during automatic operation, the system immediately decelerates and stops |
| MPG trial-cut selection key | To enter MPG trial-cut mode | To enter MPG trial-cut mode | Auto, MDI, Edit, Machine zero return, Step, MPG, Manual, MPG trial-cut mode 。 |
| DNC | DNC mode selection key | Enter DNC operation mode | Switch to DNC mode during automatic operation, and only switch to it after the current segment stops running. |
| S.CCW<br>S.STOP<br>S.CW | Spindle control keys | For spindle CCW<br>For spindle stop<br>For spindle CW | MPG, Manual, MPG trial-cut mode |
| Spindle override keys | Spindle magnification knob | spindle speed adjustment (spindle analog control active) | Auto, Edit, MDI, Machine zero return, Manual, Step, MPG, Program zero return mode |
| Tool magazine counterclockwise<br>tool magazine clockwise<br>tool magazine advances<br>tool magazine retreat | Tool magazine action key | tool magazine action on/off | MPG |
| Spindle clamp/loose blade | Manual loosening/clamping switch | Manual loosening/tightening tool switch | Manual |

| Changing tool Hands | Manual tool change | Complete manual tool change | MPG |
|---|---|---|---|
| Block Skip switch | Program segment selection switch | For skipping of block headed with"/"sign, if its switch is set for ON, the Block Skip indicator lights up | Auto, MDI mode |
| Single Block switch | Single segment switch | For switching of block/blocks execution, Single block indicator lights up if Single mode is active | Auto, MDI mode |
| Dry Run key | Empty operation switch | If dry run is active, the Dry run indicator lights up. Dry run for program/MDI codes | Auto, MDI mode |
| M.S.T. Lock key | Auxiliary functions switch | If the miscellaneous function is locked, its indicator lights up and M, S, T function output is inactive. | Auto, MDI mode |
| Machine Lock key | Machine tool lock switch | If the machine is locked, its indicator lights up, and X, Z axis output is inactive. | Auto, MDI, Edit, Machine zero return, Manual, Step, MPG, Program zero return mode |
| Work light | Machine tool work light switch | Machine tool work light on/off | Auto, Edit, MDI, Machine zero return, Manual, Step, MPG, Program zero return mode |
| Lubricating key | Lubrication key | For lubricating ON/OFF | Auto, Edit, MDI, Machine zero return, Manual, Step, MPG, Program zero return mode |
| Cooling key | Coolant switch key | For cooling ON/OFF | Auto, Edit, MDI, Machine zero return, Manual, Step, MPG, Program zero return mode |
| chip removal | Chip removal switch key | Chip removal on/off | Auto, Edit, MDI, Machine zero return, Manual, Step, MPG, Program zero return mode |

| | | | |
|---|---|---|---|
| Feedrate Override keys | Feed rate knob | Adjusting feedrate | Auto, MDI, Edit , Machine zero return, MPG, Step, Manual, Program zero return mode |
| Rapid traverse key | Fast feed key | For rapid traverse /feedrate switching | Auto, Edit, MDI, Machine zero return, Manual, Step, MPG, Program zero return mode |
| F0 （0．001、0．01 、0．1、1） | Rapid override keys | Adjusting rapid traverse | Auto, MDI, Machine zero return, Manual, Program zero return mode |
| +X/-X/+Y/-Y/+Z/-Z/+4/-4/+5/-5 | Manual feed button | Positive/negative movement of each axis in Manual, Step mode | Machine zero return, Step, Manual, Program zero return mode |
| Optional stop | Select stop/key | Execute M01 to pause when the optional stop is enabled | Auto, MDI mode |
| Feed hold key | Feed hold key | Dwell commanded by program, MDI code | Auto, MDI mode |
| Cycle Start key | Cycle start button | Cycle start commanded by program, MDI code | Auto, MDI mode |

# Chapter 2 System Power On, Power Off, and Safe Operation

## 2.1 System power on

Before powering on the CNC system, it should be confirmed that:

1. The machine tool is in normal condition.
2. The power supply voltage meets the requirements.
3. The wiring is correct and secure.

After the system self-test is normal and initialization is complete, the current position (relative coordinates) page is displayed. As shown in Figure 2-2-1

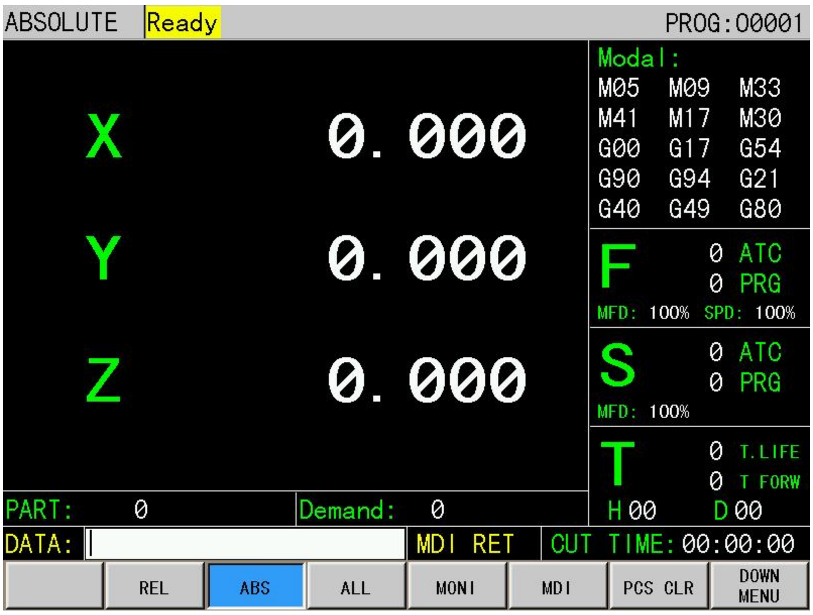



图2-1-1

## 2.2 turn off a machine

**Before shutting down, it should be confirmed that:**

1. The X, Y, and Z axes of CNC are in a stopped state;
2. Auxiliary functions (such as spindle, water pump, etc.) are turned off;
3. Cut off the CNC power first, and then cut off the machine tool power.

**When cutting off the power supply, the following checks should be performed:**

1. Check that the LED on the operation panel indicates that the cycle start should be in the stop state;
2. Check that all movable parts of the CNC machine are in a stopped state;
3. Button to shut down.

**Cut off power in emergency situations**

During the operation of the machine tool, the power supply can be immediately cut off in case of emergency to prevent accidents. However, it must be noted that there may be a deviation between the system coordinates and the actual position after cutting off the power, and operations such as resetting and tool alignment must be performed.

**Note: For the operation of cutting off the power supply of the machine tool, please refer to the machine tool user manual provided by the machine tool manufacturer.**

## 2.3 safe operation

### 2.3.1 Reset

After pressing the button 

, the system is in a reset state:

**1**、 All axis movements stop;

**2**、 The M function has stopped.

**3**、 Modify the corresponding parameters and set whether to retain the G codes of each group after resetting.

**4**、 Modify the corresponding parameters and set whether to clear the F, H, and D codes after resetting.

**5**、 Modify the corresponding parameters and set whether to delete the programming after resetting in MDI mode.

**6**、 Can be used for system abnormal output and coordinate axis abnormal actions.

### 2.3.2 ESP


In emergency situations, pressing the emergency stop button immediately stops all axis movements of the machine tool, such as the rotation of the spindle, and all coolant is also turned off. At the same time, the button remains locked in the stop position.

The button release method varies depending on the machine tool manufacturer, usually by pressing the button and rotating it clockwise to release.

Note 1: Pressing this button will cut off the motor enable, and whether the motor power supply is cut off depends on the circuit design of the machine tool manufacturer.

Note 2: The control unit is in the reset state.

Note 3: Troubleshooting is required before releasing the button.

Note 4: After the button is released, machine tools using incremental encoders should be manually operated to return to the reference point

The general emergency stop signal is a normally closed contact signal. When the contact is disconnected, the system enters an emergency stop state and causes the machine tool to stop urgently. The emergency stop signal circuit is connected as follows:

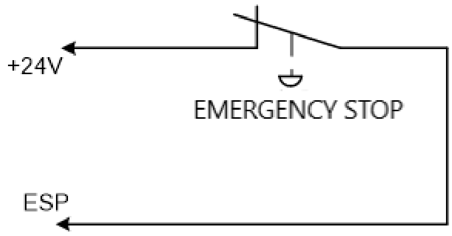
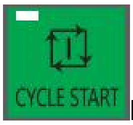



Figure **2-3-2-1**

## 2.4 **Cycle Start and Feed Hold**

The 

key and 

key in the control panel are used to start and stop the operation of the program in automatic mode, entry mode and DNC mode. Change PLC address **K5.1** to set whether to use external start and stop.

**Note 1**: switch among automatic, MDI and DNC modes. Before executing the current program segment, the cycle start is valid; press <Feed hold> to make feed hold invalid.

**Note 2:** switch automatic, MDI, DNC modes to edit mode. Before executing the current program segment, the cycle start is invalid; press <Feed hold> to make feed hold invalid.

**Note 3**: switch from automatic, MDI and DNC modes back to machine zeroing, one-step, manual and pulse modes. Press the feed hold button to make the feed hold function invalid.

**Note 4**: when the cycle start becomes valid, and automatic, MDI and DNC modes are switched or switch to the edit mode, press the feed hold button before executing the current program segment, and then make the feed hold function invalid.

## 2.5 Overstroke Protection

In order to avoid the damage of the machine caused by the overstroke of X, Y and Z axes, the machine must take overstroke protection measures.

### 2.5.1 **Hardware Overstroke Protection**

Install stroke limit switches at the maximum stroke of X axis, Y axis and Z axis of the machine respectively. When overstroke occurs, the running axis will slow down and stop when it touches the limit switch, and the system will prompt the overstroke alarm information.

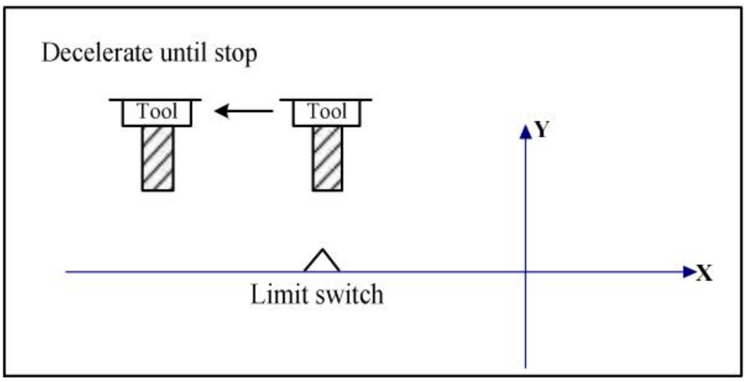



Figure **2-5-1-1**

**Detailed description:**
**Overstroke during automatic operation**
In the automatic operation mode, when the tool touches the limit switch in moving along a certain axis, all the axes will slow down and finally stop. At the same time, the overstroke alarm will be displayed and the program will stop at the overshoot segment.
**Overstroke during manual operation**
In the process of manual operation, as long as a certain axis of the machine touches the limit switch, the corresponding axis will immediately slow down and stop moving.

## 2.5.2 Software Overstroke Protection

The software travel range is set by - [Quick Parameters] P008~P013, with machine coordinate values as reference values. If the moving axis exceeds the soft limit parameter setting, an overtravel alarm will occur. When the soft limit is set to overtravel by parameter - [emergency stop limit] P007, an alarm will be triggered at overtravel (0: 5mm before, 1: after). After the overtravel alarm is triggered, move the axis in the opposite direction in<manual>mode and remove it from the overtravel range to clear the alarm.

## 2.6 **Stroke Inspection**

Use stored stroke inspection 1 and stroke inspection 2 to specify the areas where 2 tools are unaccessible to.

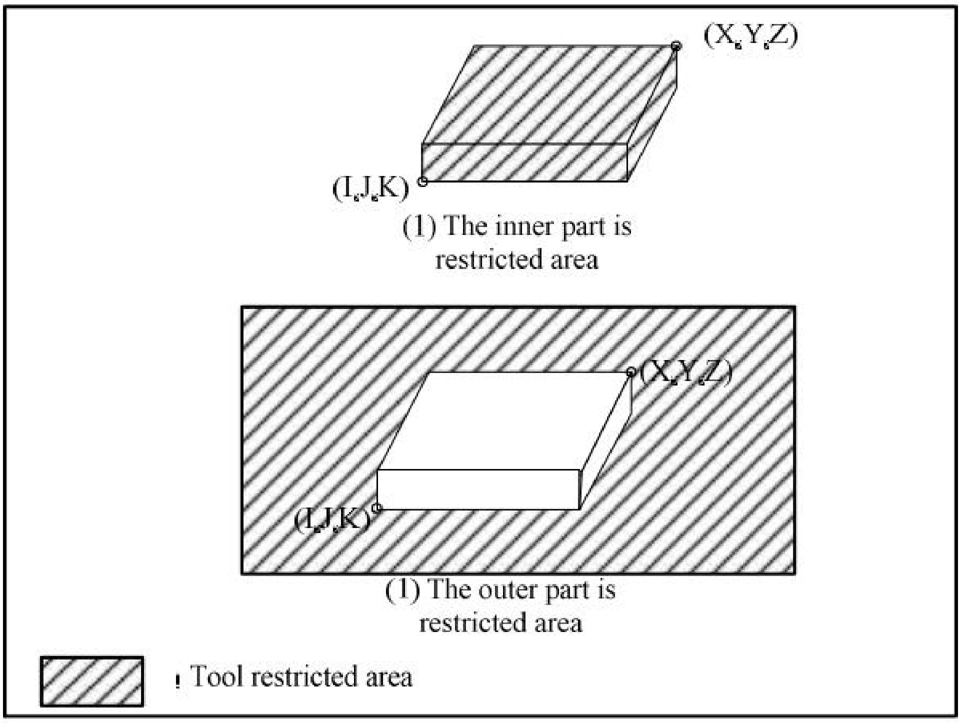



Figure **2-6-1** Stroke inspection

When the tool exceeds the stored stroke limit, an alarm is displayed and the machine slows down and stops.
When the tool enters the restricted area and brings an alarm, the tool can move in the opposite direction as the tool enters.
**Detailed description:**
1. Stored stroke inspection **1**: set the boundary by the data parameters **P66-P73** beyond which

lies the restricted area which is generally set as the maximum stroke of the machine by the machine manufacturer.

2. Stored stroke inspection **2**: set the boundary by the data parameters **P76-P83** or the program code within or beyond which can be set as the restricted area which is set by the bit parameter **NO:11#0** (0: inside of restricted area; 1: outside of restricted area).

1) When using parameters to set the restricted area: points A and B in the figure below must be set.

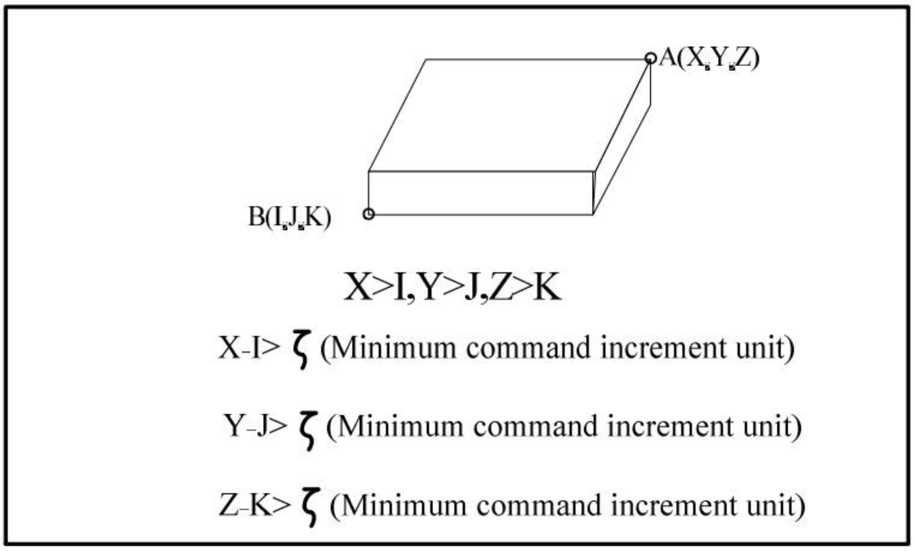



Figure **2-6-2** Restricted area creation or change with parameter

When the restricted area is set by the data parameters P76-P83, the data must give the distance in the machine coordinate system (output increment) in the smallest command increment unit.

2) When using program instructions: G12 prohibits the tool from entering the restricted area; G13 allows the tool to enter it.

Each G12 in the program must have a separate program segment instruction. The following commands are used to create or change the restricted area.

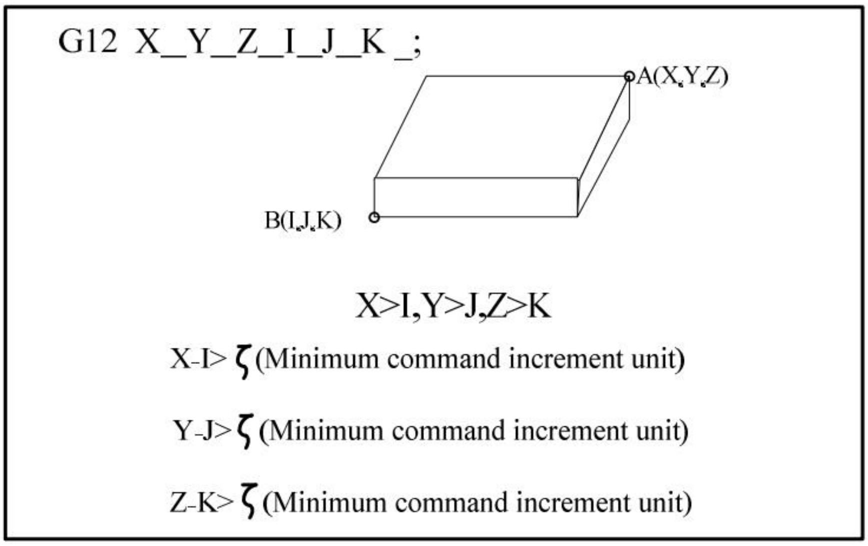



Figure **2-6-3** Restricted area creation or change with program

If set through G12 to specify the distance in the machine coordinate system in the minimum input increment unit (input increment).

The programmed data is converted into the numerical value of the minimum code unit in the smallest incremental unit, and this value is set in the parameter.

3) Checkpoints for restricted areas: Before programming the restricted areas, please confirm the location of the checkpoint (at the tip of the tool or the top of the tool cover). As shown in Figure

2-6-4, if the checkpoint is A (tool tip), the distance "a" should be set as the data for the storage function check; If the checkpoint is B (tool cover), the distance "b" should be set as the data for the storage function check. When the checkpoint is A (tool tip) and the length of the tool varies with different tools, the prohibited area should be set according to the longest tool to ensure safe operation.

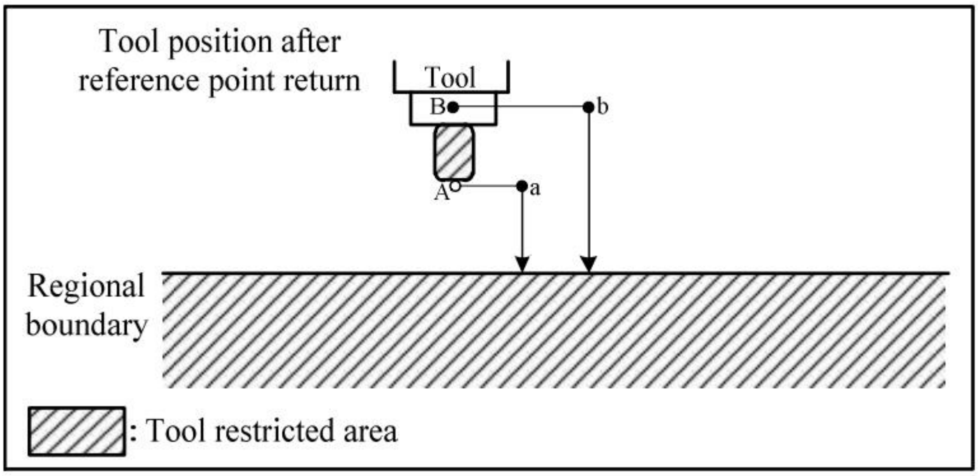



Figure **2-6-4** Restricted area setting

4） Overlap of restricted area for tools: Restricted area can be set in an overlapping manner. As shown in the figure

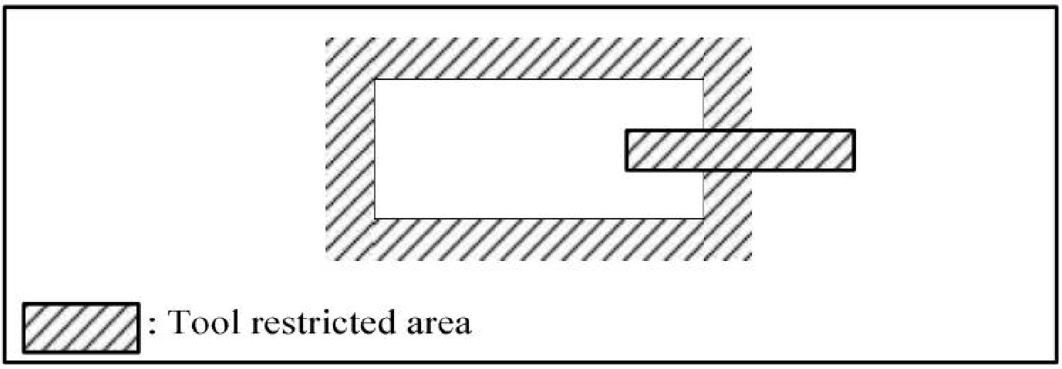



Figure **2-6-5** Restricted area overlapping setting

Unnecessary limits should be set outside the stroke of the machine.

5）The effective time of the restricted area: After power on and manual reference point return or automatic reference point return by code G28, the restricted area boundary will take effect. After power on, if the reference position is in the restricted area, an alarm will be immediately generated (only valid in G12 mode with storage travel limit 2).

6) Release alarm: If entering a restricted area and an alarm is triggered, the tool can only move in the opposite direction. To eliminate the alarm, move the tool in the opposite direction until it exits the restricted area and resets the system. After the alarm is eliminated, the tool can move forward or backward.

7) When switching from G13 to G12 in the restricted area, an alarm will be immediately triggered。

# Chapter 3 Interface Display and Data Modification and Setting

## 3.1 Position display

### 3.1.1 Five ways to display location pages

Press the key to enter the location page display, which has five methods: (relative coordinates), (absolute coordinates), (comprehensive coordinates), (program&coordinates), and (MDi input). You can view them through the corresponding soft keys or keep holding down the key to switch between viewing. The specific introductions of each interface are as follows:

1. Relative mode: Press the (Relative Coordinates) soft key to display the current position of the tool in the current coordinate system, hereinafter referred to as "Relative Coordinates"（**See figure 3-1-1-1**）

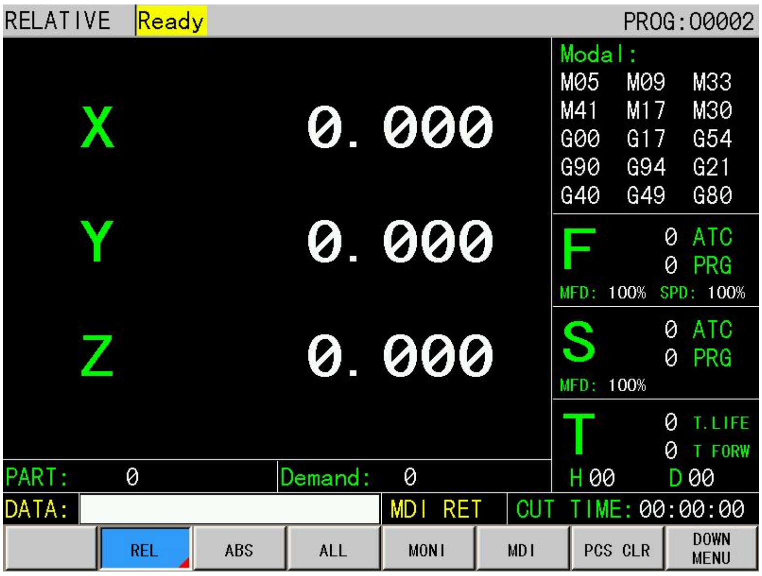



Figure **3-1-1-1**

In the figure, the left side represents the relative coordinate values in the coordinate system, and the first progress bar F on the right represents the feed rate, which can be adjusted through the feed rate knob. The progress bars for S and rapid magnification below can also be adjusted by selecting different magnification values.

Steps for resetting the relative coordinate system: Select the axis to be modified using the X, Y, and Z keys corresponding to the letter. The selected area will flash. Press the cancel button to reset the coordinates; Alternatively, by pressing the coordinates corresponding to the soft keys, each axis can be reset to zero. （**See figure 3-1-1-2**）

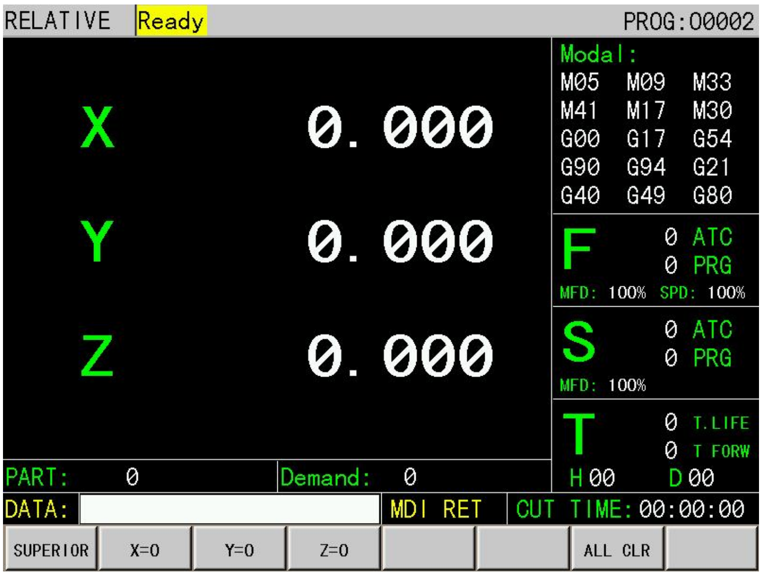



Figure **3-1-1-2**

**2)** Absolute mode: Press the (absolute coordinates) soft key to display the current position of the tool in the absolute coordinate system, hereinafter referred to as "absolute coordinates"

（**See figure 3-1-1-3**）

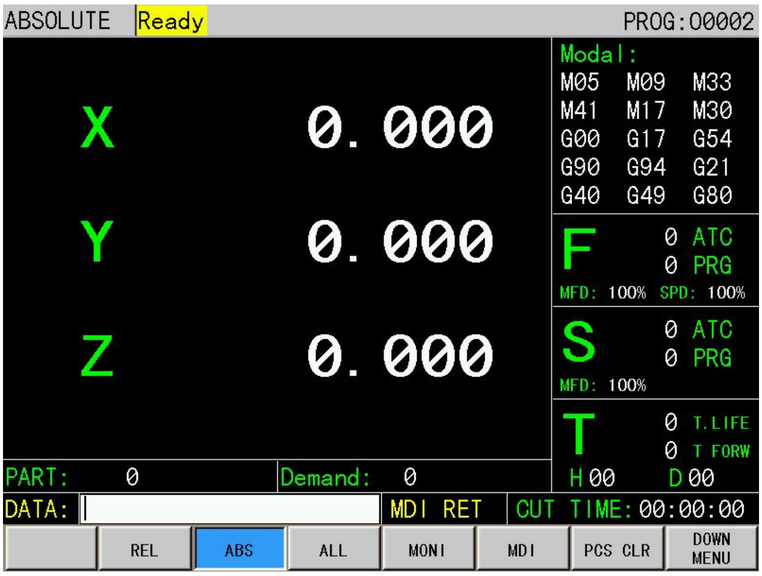



Figure **3-1-1-3**

### 3) Integrated interface

Press the (Comprehensive Coordinates) soft key to enter the (Comprehensive Coordinates) interface. In the comprehensive interface, the coordinate position values in the following coordinate system can be displayed simultaneously:

(A) Position in relative coordinate system;

(B) Position in absolute coordinate system;

(C) Position in the machine coordinate system;

(D) Residual displacement (displayed only in automatic, input, and DNC modes).

Display page as shown in the figure below（**See figure 3-1-1-4**）：


Figure 3-1-1-4

**4**）Program&Coordinate Interface

Press the (Program&Coordinates) soft key to enter the (Program&Coordinates) interface, where the absolute coordinates of the current position, remaining coordinates, modal information of the currently running program, and running program segments can be displayed simultaneously.（**See figure 3-1-1-5**）


Figure 3-1-1-5

**4）MDI method**

Press the (MDI Input) soft key to enter the (MDI Input) interface. In this interface, you can input and execute a single instruction or a single or multiple program segments. The program format is the same as that of the editing program. The MDI method is suitable for single instructions or short program segment operations（**See figure 3-1-1-6**）


Figure 3-1-1-6

## 3.1.2 Clear the number of completed items, set the required total number of items, and clear the cutting time

On the<Position>display interface, press the 【 Clear Completed Parts 】 soft key to reset the number of workpieces to zero

On the<Position>display interface, press the page menu - 【 Set Required Quantity 】 soft key to execute the workpiece quantity setting

On the<Position>display interface, press the page menu - 【 Clear Cutting Time 】 soft key to reset the cutting time to zero

## 3.1.3 Steps for resetting machine coordinates

Location interface → Next menu → Press the (Reset mechanical coordinates) soft key, prompt to enter the second level password 111111. After the password is correct, the operation box will be displayed（**See figure 3-1-1-7**）

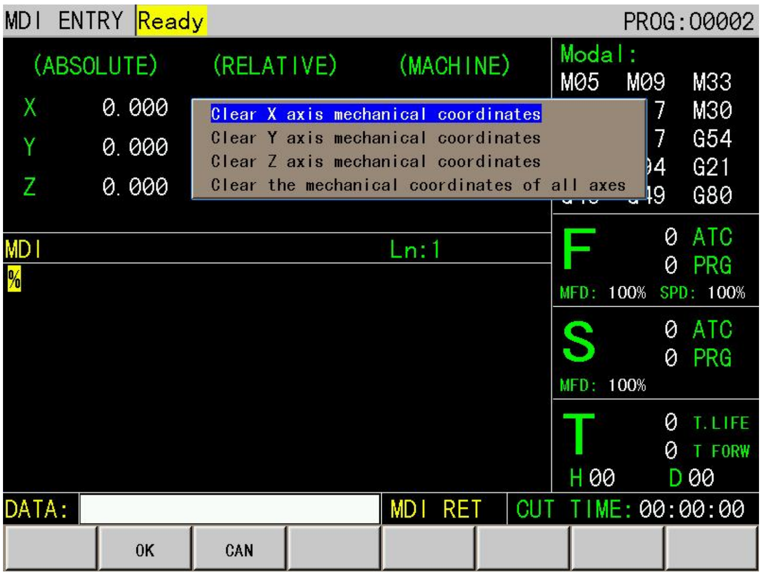



Figure 3-1-1-7

## 3.2 Program display

Press the keys 

on the panel to enter the program page display. The program display page has six methods: [New Program], [Program Catalog], [Save Program], [Save as Program], [USB Drive], and [Copy Selection]. Each interface can be viewed and modified through the corresponding soft keys. Specifically, as follows,**See figure 3-2-1**。 Specifically, as follows:

## 3.2.1Program display

Press the program key to enter the program display interface, where the program on the page where the executing program segment is located in the memory is displayed （**See figure 3-2-1-1**）。

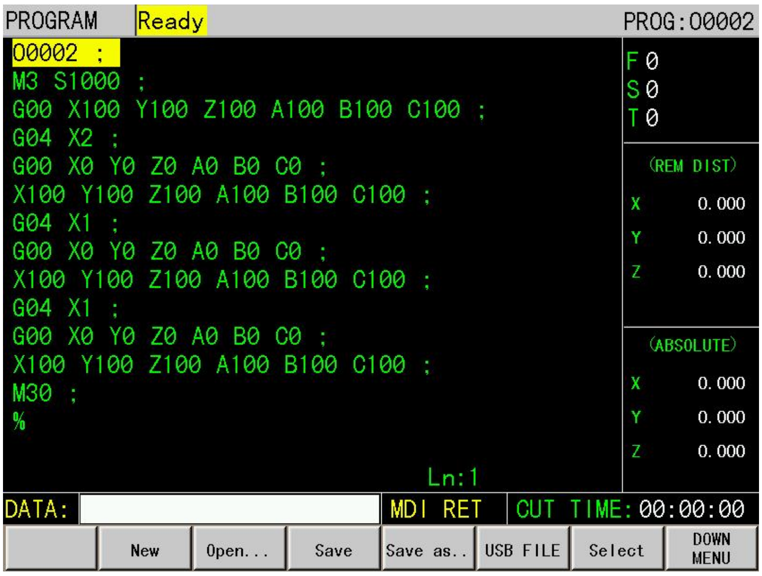



Figure 3-2-1-1

After pressing the soft key (New Program) again, the interface enters the editing and modification page of the program（**See figure 3-2--1-2**）：

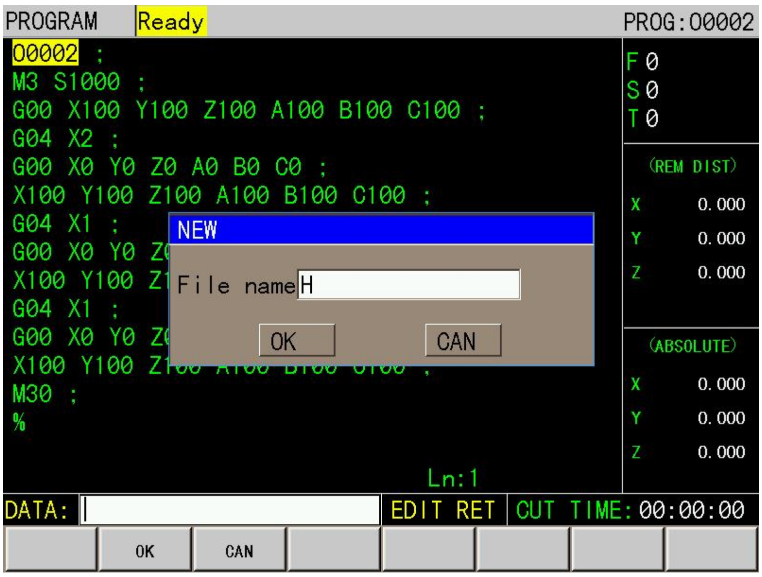



Figure 3-2-1-2

Pop up a dialog box, please enter the file name. The file name can have a combination of Chinese characters, letters, or numbers, or it can be used as a separate program name. The Chinese file name needs to be entered according to the software's corresponding [Chinese Input], open Chinese Pinyin, and enter the corresponding letter to display Chinese.

This function allows for the insertion, modification, deletion, copying, pasting, and replacement of characters during program editing. In addition, it also includes the function of deleting complete programs and automatically inserting sequence numbers.（**See figure 3-2-1-3**）

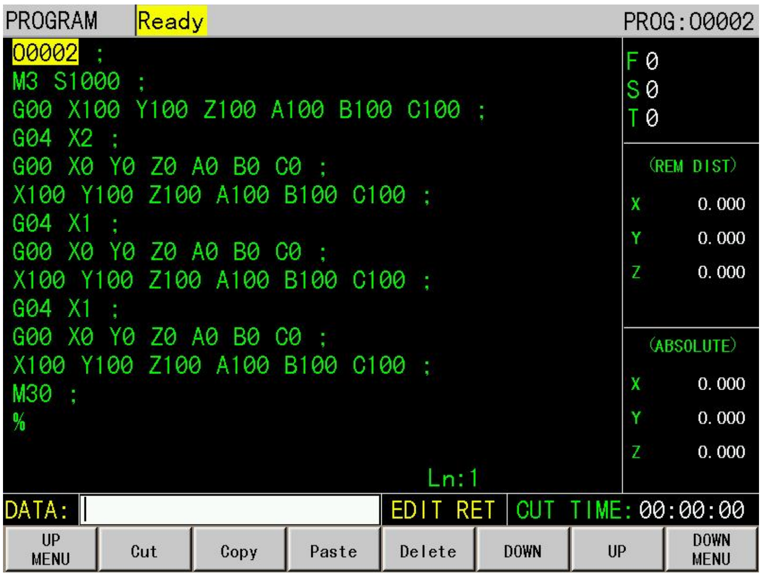



Figure 3-2-1-3

## 3.2.2 Program (directory) display

Press the (Program Catalog) soft key to enter the directory display interface, and the displayed content is as follows (see Figure 3-2-2-1):

(a) Stored program number: The number of stored programs (including subroutines). Remaining: The number of programs that can still be stored.

(b) Used storage capacity: The storage capacity occupied by the stored program (displayed in characters).

Remaining: Program storage capacity that can still be used.

(c) You can delete all programs or delete individual programs.


Figure 3-2-2-1

**Description: Display all program numbers in the memory through the page key.**

## 3.2.3 Program (USB) display

Press the 【 USB 】 soft key to enter the U display interface, and the displayed content is as follows (see Figure 3-2-3-1):

(a) The system disk program can be copied to a USB drive. Move the cursor to the program you want to copy, and the program number will be displayed in yellow. Press the [Copy] key corresponding to the output or soft key

(b) The USB program can be copied to the system disk. Move the cursor to the program you want to copy, and the program number will be displayed in yellow. Press the [Copy] key corresponding to the output or soft key

(c) You can operate to copy all programs or delete individual programs.

Figure 3-2-3-1

# 3.3tool compensation display, modification, and settings

## 3.3.1 tool repair display

1. Page entry

Press OFFSET the key to enter the bias and setting interface, where there are four interfaces: [bias], [coordinate system], [common variables], and [system variables]. You can view or modify them through the corresponding soft keys, or press OFFSET to switch between different interfaces. The specific content is shown in the following figure（**See figure** 3-3-1-1）：

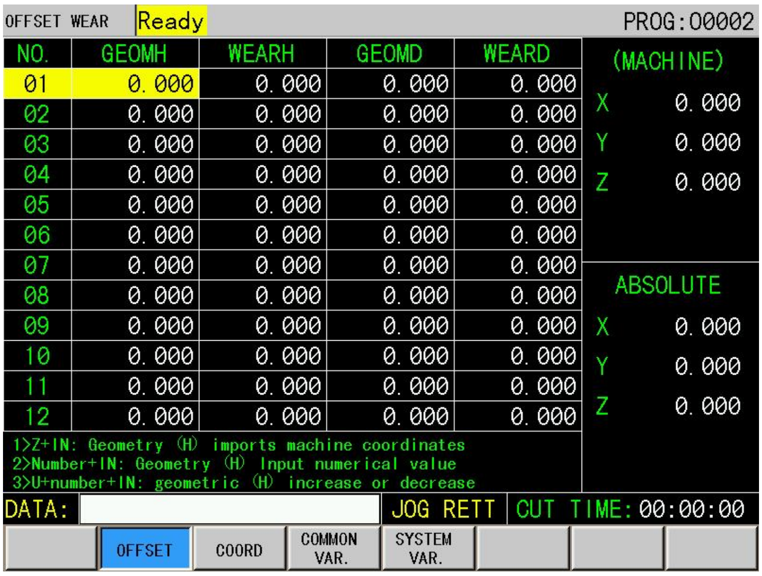



Figure 3-3-1-1

1) Press the function key OFFSET 。**See figure** 3-3-1-1

## 3.3.2 [Coordinate System] Interface Operation Instructions

Press the soft key button 【 Coordinate System 】 to enter the workpiece system interface（**See figure** 3-3-2-1）：


Figure 3-3-2-1

Perform the following operations:

(a) Enter the<Enter>/<Edit>operation mode;

(b) Press the up and down keys to move the cursor to the item to be changed;

The base offset is the offset of all coordinate systems. Users can input numbers directly or press INPUT the corresponding axis letter keys X and Z to import machine coordinates into the coordinate system. The operation of G54~G59 is similar.

## 3-3-3 Press the (Common Variables) soft key on the common page to enter the common variable interface.See figure 3-3-3-1:

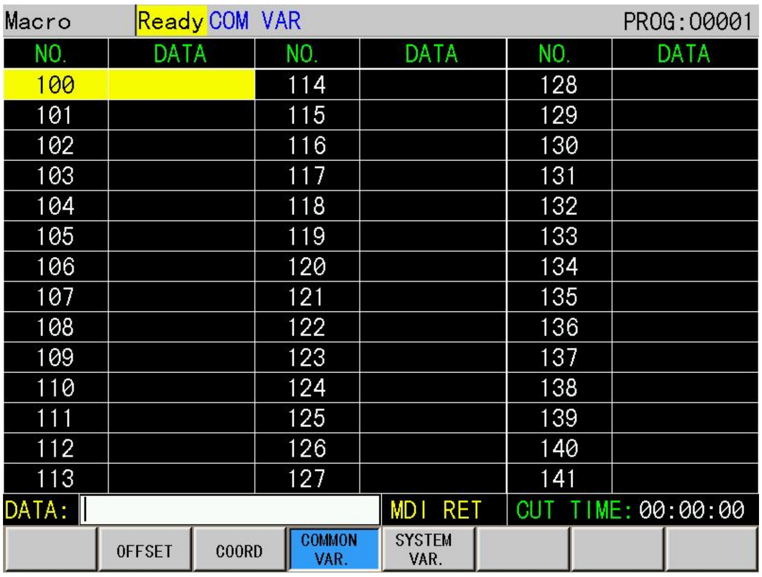
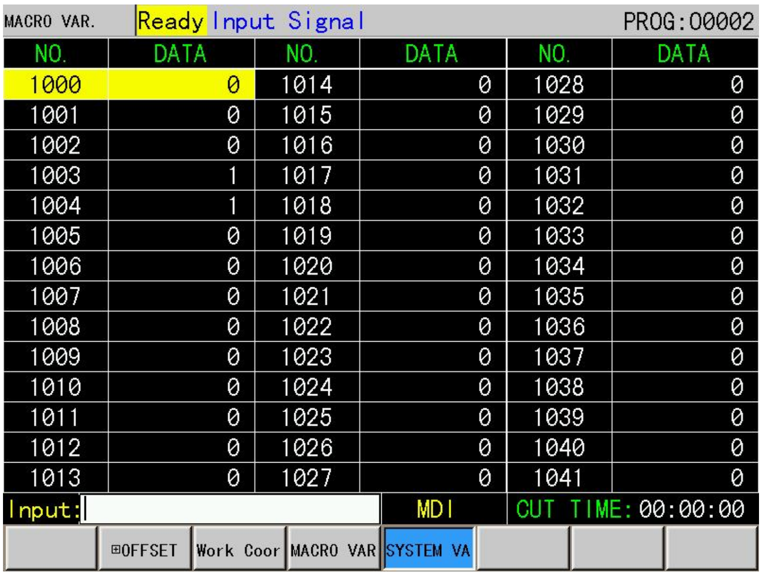



Figure 3-3-3-1

### 3-3-4 Press the (System Variables) soft key on the System Variables page to enter the System Variables interface（See figure 3-3-4-1）:



| MACRO VAR. | Ready Input Signal | | | | PROG:O0002 |
|---|---|---|---|---|---|
| NO. | DATA | NO. | DATA | NO. | DATA |
| 1000 | 0 | 1014 | 0 | 1028 | 0 |
| 1001 | 0 | 1015 | 0 | 1029 | 0 |
| 1002 | 0 | 1016 | 0 | 1030 | 0 |
| 1003 | 1 | 1017 | 0 | 1031 | 0 |
| 1004 | 1 | 1018 | 0 | 1032 | 0 |
| 1005 | 0 | 1019 | 0 | 1033 | 0 |
| 1006 | 0 | 1020 | 0 | 1034 | 0 |
| 1007 | 0 | 1021 | 0 | 1035 | 0 |
| 1008 | 0 | 1022 | 0 | 1036 | 0 |
| 1009 | 0 | 1023 | 0 | 1037 | 0 |
| 1010 | 0 | 1024 | 0 | 1038 | 0 |
| 1011 | 0 | 1025 | 0 | 1039 | 0 |
| 1012 | 0 | 1026 | 0 | 1040 | 0 |
| 1013 | 0 | 1027 | 0 | 1041 | 0 |
| Input: | | MDI | | CUT TIME:00:00:00 | |



Figure 3-3-4-1

## 3.4 Parameter switch, modification and setting

## 3.4.1 Set Display

Press the  key to enter the bias information display page, which has five interfaces: [Switch], [Password], [Data Restoration], [Data Backup], and [Graphics]. You can view or modify them through the corresponding soft keys, as follows:

**Press the 【 Switch 】 soft key on the switch interface to enter the bias interface（See figure 3-4-1-1）:**

1) Select the MDI mode key.

2) Press the soft key 【 Switch 】 to display the parameter switch and program switch screen.

3) Use the up and down arrow keys to locate the parameter or program and move it to the item that needs to be changed.

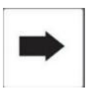
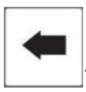
4) Press  the button to turn on the switch, press  the button to turn off the switch. When the parameter switch is set to "off", it is prohibited to modify or set system parameters. When the program switch is set to "off", it is prohibited to edit the program.


Figure 3-4-1-1

## 3.4.2 Password modification and setting

Operation of the login interface（**See figure 3-4-2-1**）

In order to prevent malicious modification of machining programs, CNC parameters, etc., the system provides permission setting function, with password levels divided into 5 levels, from high to low: level 1 (system manufacturer level), level 2 (machine tool manufacturer level), level 3 (system debugging level), level 4 (end user level), and level 5 (machining operation level). The system defaults to the lowest level when powered on（**See figure 3-4-2-1**）。

Level 1 and Level 2: Allow modification of CNC status parameters, data parameters, tool compensation data, transmission of PLC ladder diagrams, etc.

Level 3: Allow modification of CNC status parameters, data parameters, tool compensation data, etc.

Level 4: Can modify blade compensation data and macro variables. Partial CNC status parameters, data parameters, and screw connections can be modified.

Level 5: No password level, can operate the machine tool control panel, cannot modify tool repair parameters, cannot modify CNC status parameters, data parameters, and screw repair data.

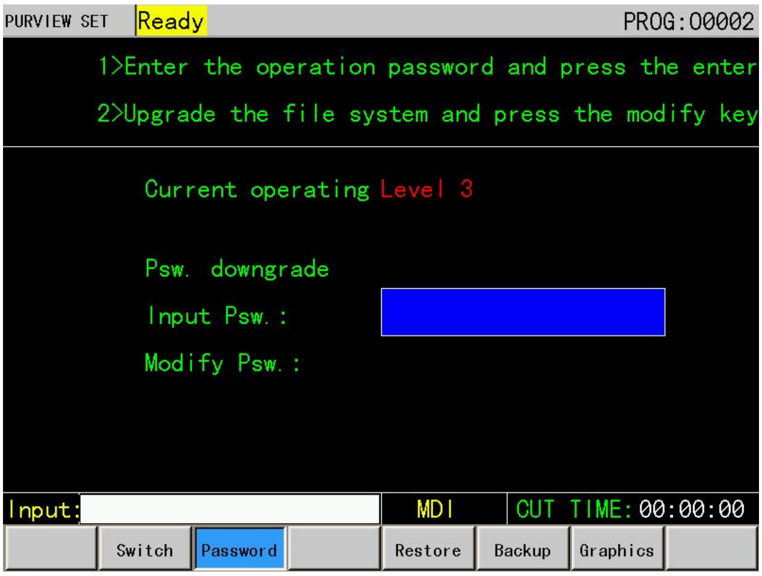



Figure 3-4-2-1

**1**）After entering the interface under<Input Method>, move the cursor to the item that needs to be changed.

**2**）Enter the password for the corresponding level, press INPUT the key, and if it is correct, the system will prompt "password correct"; otherwise, "password incorrect"

**3**）Modify the parameters and settings corresponding to the system password.

a、When changing the password, enter 0-6 digits or letters.

**4**）The steps to change password are as follows:

a) After entering the CNC settings page, follow the method described in "Enter Operation Level" to enter the level where you want to change the password;

b) Move the cursor to the "Change Operation Password" line;

c) Enter the new operation password and press the key INPUT;

d) CNC prompts' Please enter new password again ';

e) After entering the new operation password again, press the key INPUT. If the password entered twice is the same, CNC will prompt "Password has been changed, please save the new password properly", and the operation password has been successfully changed.

## 3.4.3 Data restoration

Press the (Data Restore) soft key to enter the data interface. User data (such as status parameters, data parameters, tool parameters, screw data, ladder diagrams, and various programs) can be restored (read)；（**See figure 3-4-3-1**）

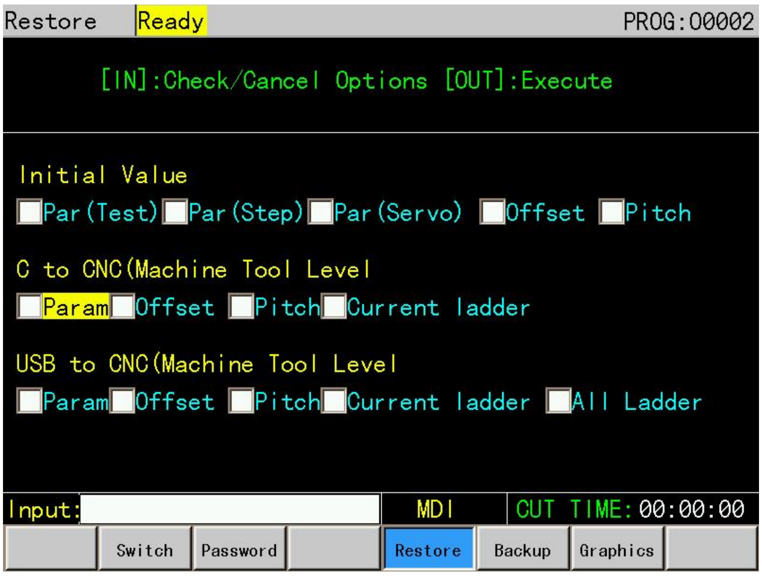



Figure 3-4-3-1

Operation method: 1. Press the 【 Password 】 soft key and set the corresponding level password in the password interface. Ladder diagrams and parameters require manufacturer permissions to operate, while system parameters, tool compensation, pitch compensation, and system macro variables require debugging level or higher permissions to operate.

2、Return to the （Data Restoration）page, move the cursor to the target position, press the key 

, display a check mark, and press the key 

 to complete the data recovery.

## 3.4.4 Data backup（See figure 3-4-4-1）

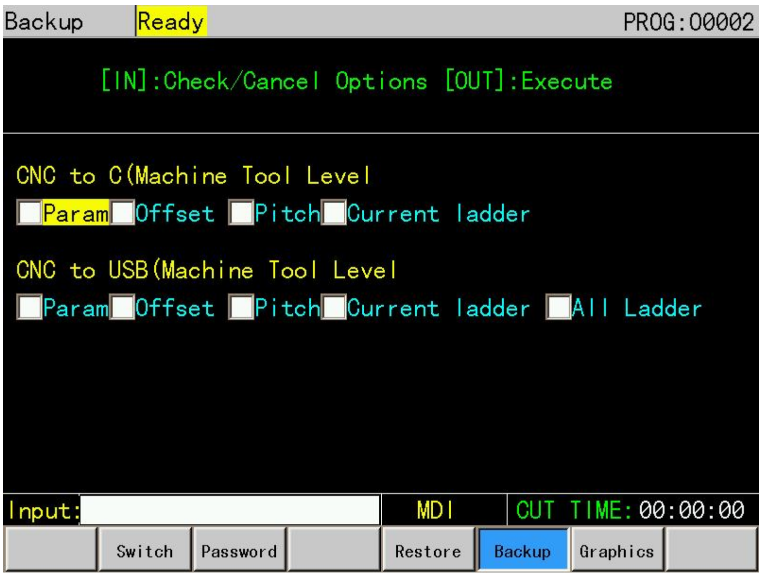



Figure 3-4-4-1

Note: The operation is the same as 3-4-3

## 3.4.5 Graphical

On the screen, the tool path of the program can be drawn. By observing the trajectory on the screen, the graphic display of the machining process can be checked, and the displayed graphics can be enlarged/reduced.（**See figure 3-4-5-1**）

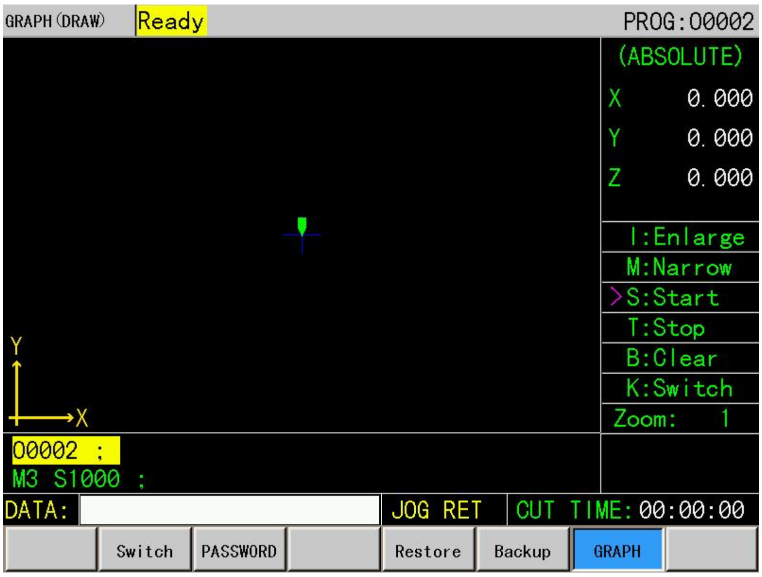
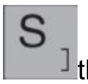



Figure 3-4-5-1

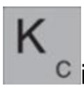
In the graphic page, the machining trajectory of the running program can be monitored.

**A**、Press 

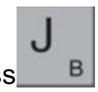
the button to enter the start drawing state, and the '*' sign will move to S: before starting drawing;

**B**、Press 

the button to enter the stop drawing state, and the '*' sign will move to T: before the drawing stops;

**C**、Each time the soft key 

is pressed, the graphic switches between the coordinate displays corresponding to 0-5.

**D**、Press 

the button to clear the drawn graphics.

# 3.5 PARAMETER DISPLAY

## 3.5.1 Parameter page

1、Page entry

Press PARAMETER the key to enter the settings information display interface, where there are parameter interfaces such as (Debug), (Spindle), (Servo axis), (Tool), (Chuck), (Zero), etc. You can view or modify them through the corresponding soft keys, as shown in the following figure（**See figure 3-5-1-1**）：

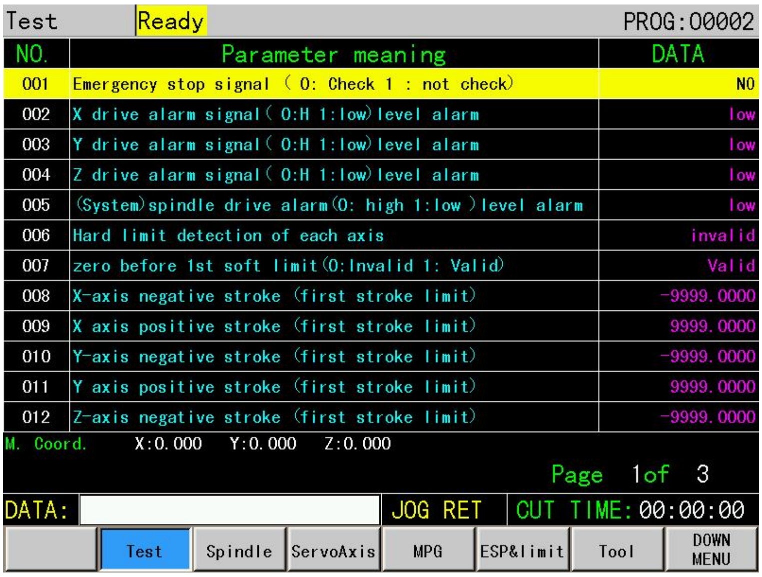



Figure 3-5-1-1

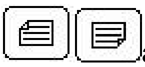
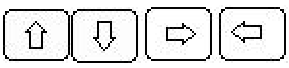
Steps for parameter display and setting:

1） Press SETTING the button to enter the 【 Switch 】 setting interface and turn on the parameter switch.

2） Press PARAMETER the function key to enter the parameter interface.

3） Use the following method to move the cursor to the parameter number to be written or displayed. The specific method is as follows:

a Enter the P letter key and number key parameter number, press INPUT the key。

b Use the up and down page keys  and directional  keys to move the cursor to the parameter number.

4） Use INPUT keys or ALTER buttons to modify parameters.

## 3.6 Diagnostic display

Press the DIAGNOSIS key to enter the bias information display page, where there are six interfaces: [CNC Diagnosis], [PLC Signal], [IO Monitoring], [User M Code], [Version Information], and [CNC Help]. You can view them through the corresponding soft keys, as follows:

### 3.6.1 Diagnostic data display

1、Press the 【 CNC Diagnosis 】 soft key in the<Diagnosis>interface to enter the CNC diagnosis interface.**See figure 3-6-1-1：**

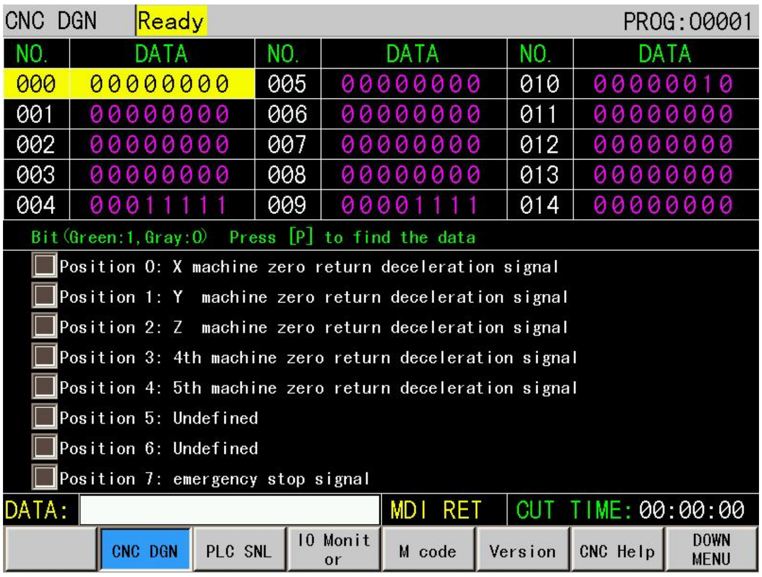



Figure 3-6-1-1

The status of input/output signals between CNC and machine tools, the status of signals transmitted between CNC and PLC, internal data of PLC and CNC and their internal status can be displayed through diagnosis. Press the DIAGNOSIS key to enter the CNC diagnostic page display, which includes keyboard diagnosis, status diagnosis, and auxiliary functional parameters. It can be viewed through PAGE-UP NEXT PAGE keys.

## 3.6.2 PLC signal display

1、Press the 【 PLC Signal 】 soft key in the<Diagnosis>interface of the CNC diagnostic interface to enter the PLC signal interface.**See figure 3-6-2-1**：

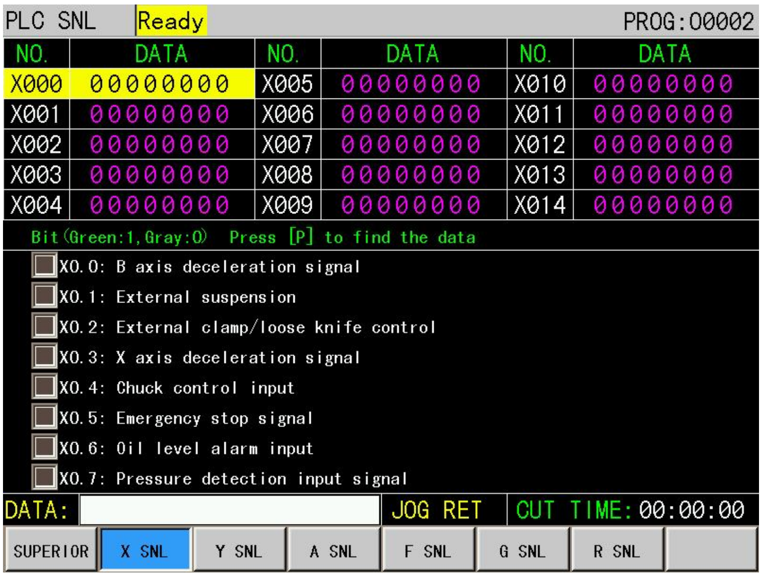



Figure 3-6-2-1

PLC status interface

On the PLC signal status interface, press the 'PLC Signal' soft key again, and the page will display address statuses such as X0000~X0063, Y0000~Y0047, A0000~A0031, F0000~F063, G0000~G063, R0000~R0511, etc. in sequence. By 

key, 

key search,See the signal status of each address in the PLC. The PLC address interface provides a detailed introduction to PLC addresses, symbols, and meanings. Those who are unfamiliar or unclear about PLC addresses can search and compare them in this interface.

## 3.6.3 IO monitoring display

1、Press the [IO Monitoring] soft key in the<Diagnosis>interface of the CNC diagnostic interface to enter the IO monitoring interface.As shown in Figure 3-6-3-1：

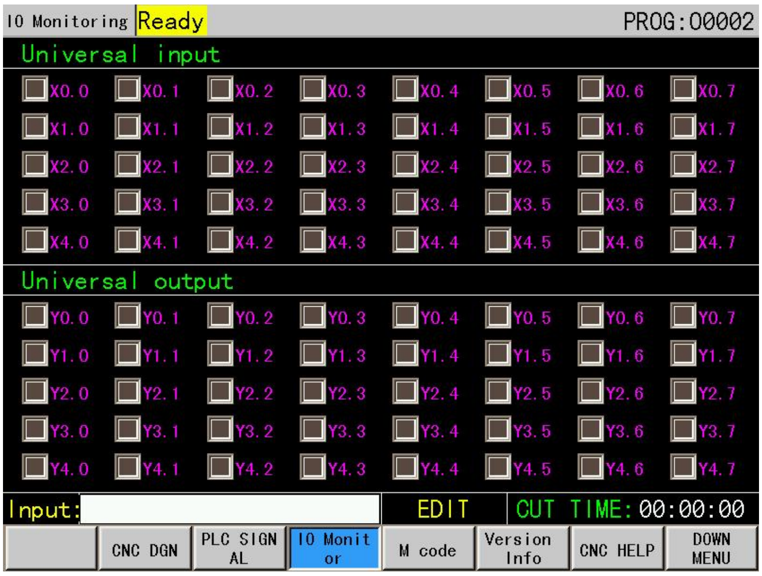



Figure 3-6-3-1

## 3.6.4 User M code display

1、Press the【 User M Code 】soft key in the<Diagnosis>interface of the CNC diagnostic interface to enter the User M Code interface.**See figure 3-6-4-1**:

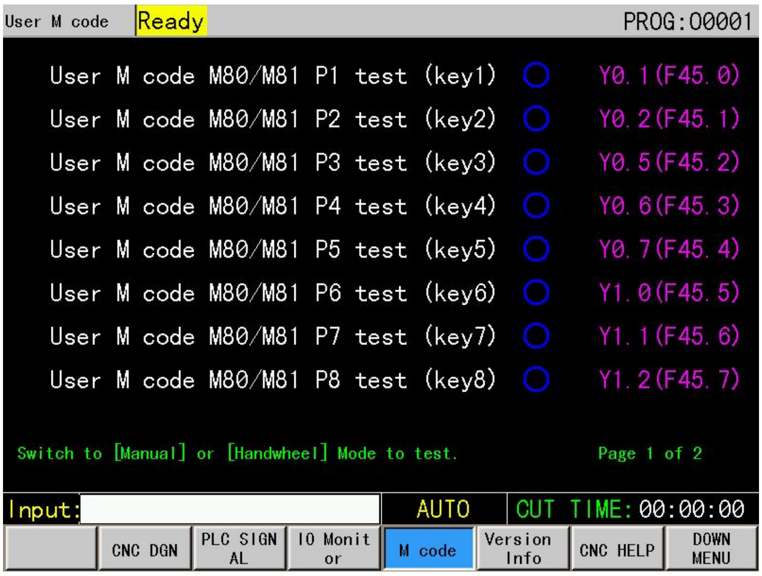



Figure 3-6-4-1

**User M code**

In the user M code interface, M80 and M81 can be opened by setting the valid parameters - [PLC parameters] P013 through parameters. In the handwheel or manual mode, press the number key corresponding to M code P. The operation method is to press the number key once to output, and then press the number key again to turn off the output.

## 3.6.5 Version information

1、Press the 【 Version Information 】 soft key in the<Diagnosis>interface of the CNC diagnostic interface to enter the version information interface. Display the current software, hardware, system number, PLC version information, etc. of CNC on the version information page**See figure 3-6-5-1**：


## 3.6.6 CNC assistance

1、In the<Diagnosis>interface, press the 【 CNC Help 】 soft key to enter the CNC Help interface. Press the 'CNC Help' soft key again, and four interfaces will appear:（Operation Table）, （Alarm Table）,（G Code）, and （Macro Command）

**ALM TABLE 3-6-6-2**


Figure 3-6-6-2

# G CODE 3-6-6-3

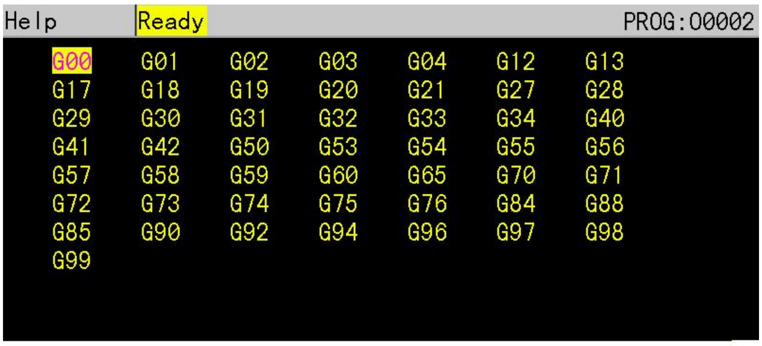



Figure 3-6-6-3

Note: Use the up and down arrow keys to move the cursor to the specified G-code. Press the 

key to open the G-code format; To exit the current G code format, please press the letter key P; If you want to know the format and usage of G-code, you can directly see the relevant information of each G-code after selecting it. This interface provides a detailed introduction to the format, function, and instructions of the instructions. Those who are unfamiliar or unclear with the instructions can search and compare them in this interface.

Macro instructions as shown in the figure **3-6-6-4**

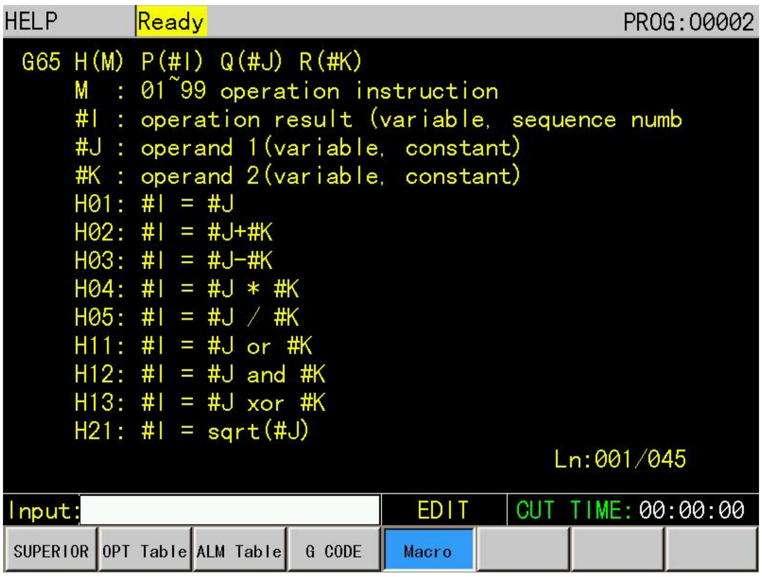



Figure 3-6-6-4

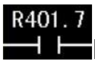
This interface introduces the format of macro instructions and various operation instructions, and provides the setting range of local variables, general variables, and the system. Those who are unfamiliar or unclear about macro instruction operations can search and compare in this interface.

## 3.7 Ladder diagram display and parameter modification

Press the PLC key to enter the bias information display page, where there are four interfaces: [ladder diagram], [PLC parameters], [PLC signal], and [PLC information], which can be viewed through the corresponding soft keys. The details are as follows:

### 3.7.1 ladder diagram

The monitoring page can view the current on/off status of contacts and coils, as well as the current values of timers and counters. When the contacts and coils are conducting, the background color is displayed in green. When they are not conducting, the background color is the same as the window background color.

If it means that contact R511.1 is conducting，

Indicates that the normally open contact is not conducting. **See figure** 3-7-1-1

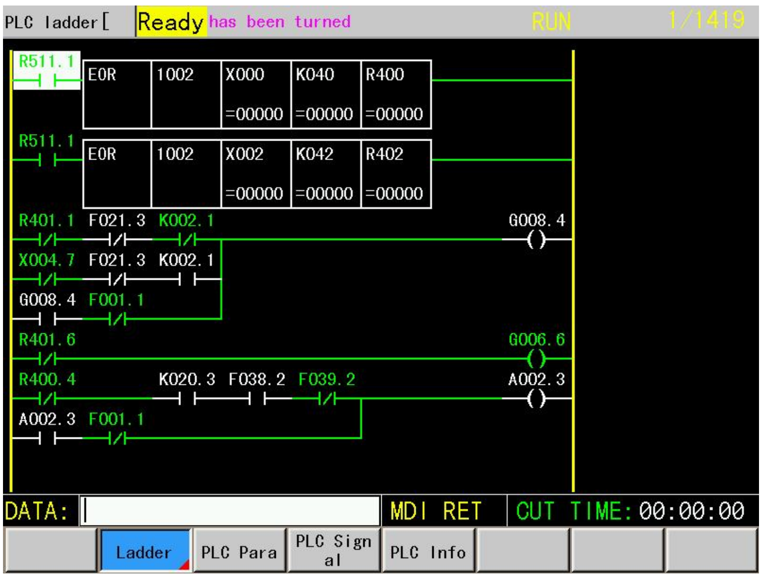



Figure 3-7-1-1

Ladder Chart - Press the 'Ladder Chart' soft key, and four operation bars will appear: up search, down search, search, and stop, as shown in the

figure 3-7-1-2

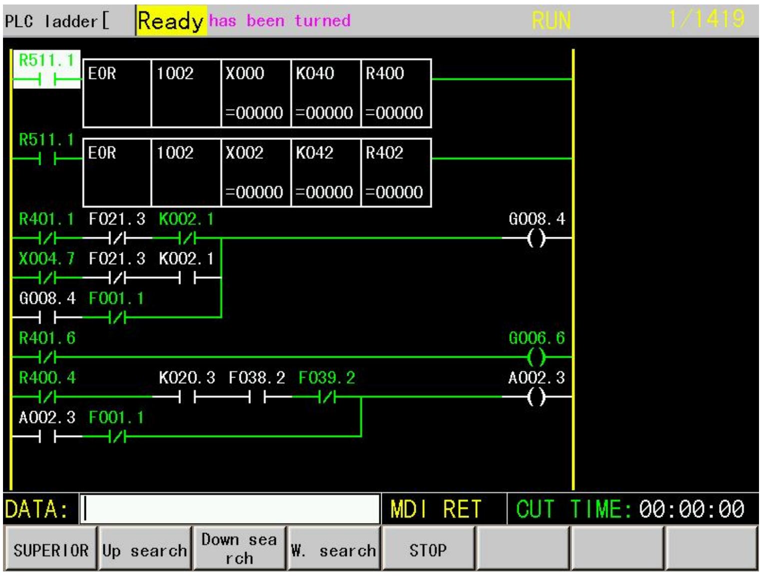



Figure 3-7-1-2

Operation: Search for signals, such as searching for X0.5 signal. Enter X0.5 and press the 【 Search Down 】 soft key to search. You can continuously press 【 Search Down 】. If no information is found, the system will prompt.

Steps for stopping the ladder diagram: First, in the settings interface - password interface, enter the level 2 password to stop the PLC and make PLC modifications.**See figure** 3-7-1-3

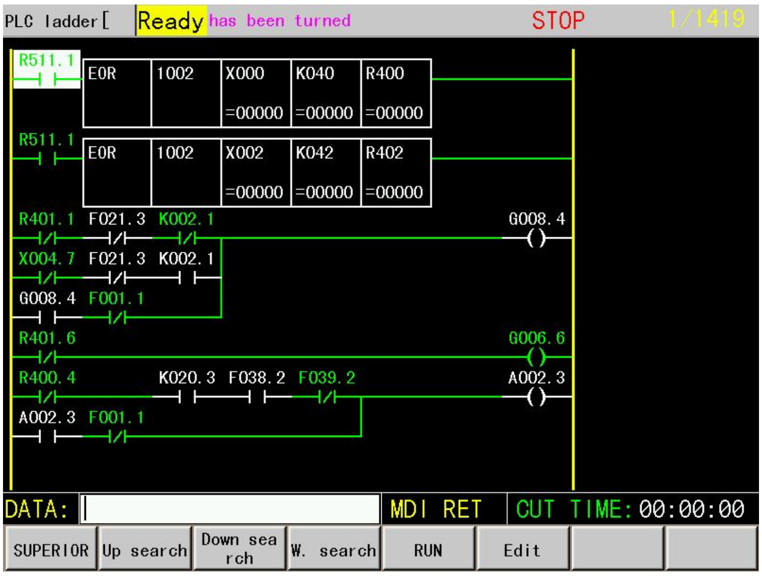



Figure 3-7-1-3

After entering the stop interface, press the 【 Edit 】 soft key to select the touch function,**See figure** 3-7-1-4, 3-7-1-5

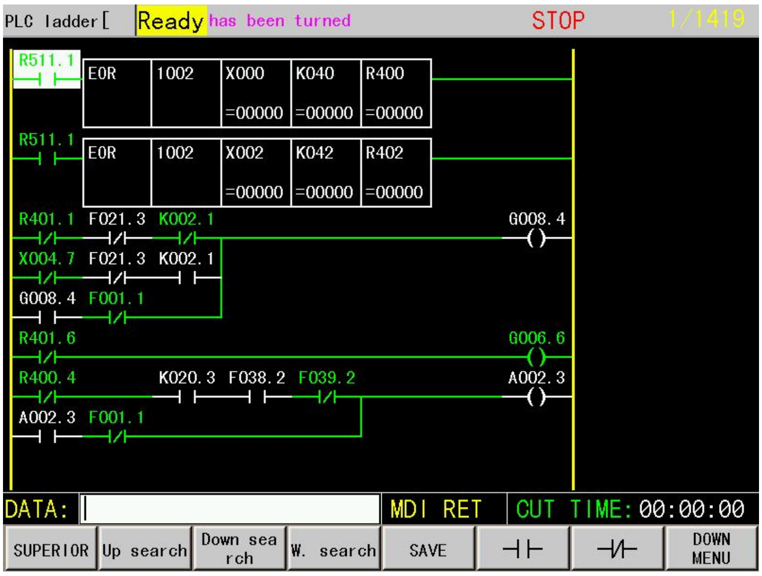



Figure 3-7-1-4

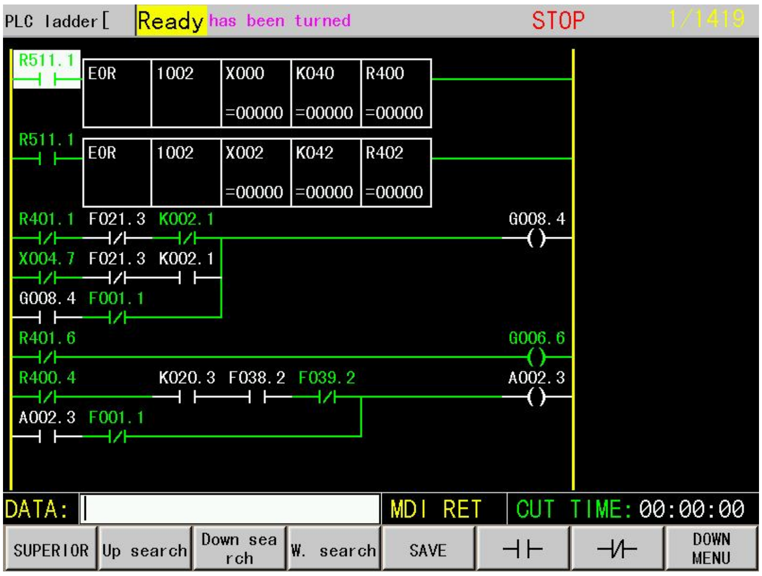



Figure 3-7-1-5

Note: After editing the ladder diagram, you must first press the 'Save' soft key, and then press the 'Run' software

## 3.7.2 PLC parameters

Press the 【 PLC Parameters 】 soft key in the<PLC>interface to enter the PLC parameter interface. Press the 'PLC Parameters' soft key again, and four interfaces will appear:' K Parameters', 'T Parameters',' D Parameters', and 'C Parameters'.

PLC data includes K, T, D, and C parameters, which can be set on this page.**See figure** 3-7-2-1

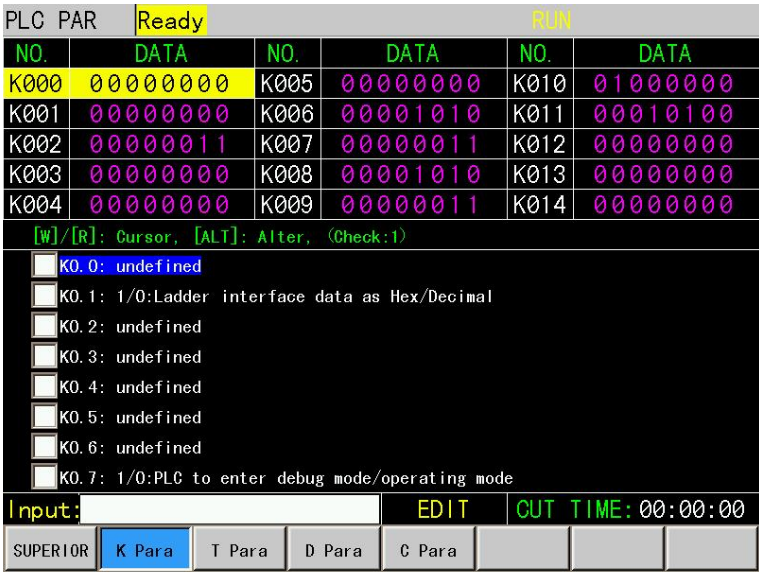


Figure 3-7-2-1

## 3.7.3 PLC signal

PLC signal interface**See figure** 3-7-3-1

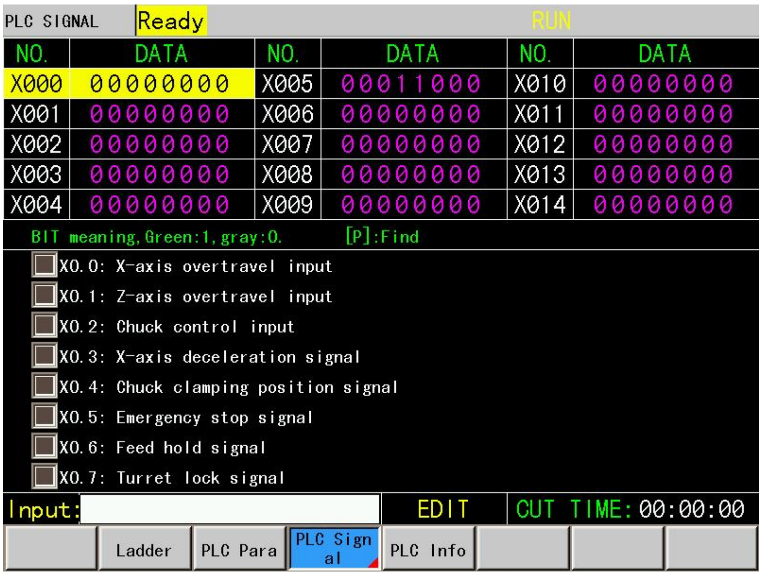


Figure 3-7-3-1

## 3.7.4 PLC information

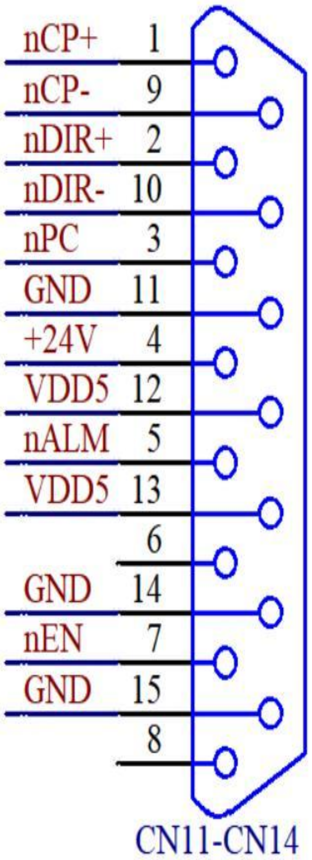
PLC Information Interface **See figure** 3-7-4-1


PLC INFO
Ready
RUN
RUN LADDER : Ladder01
LADDER VER: V1.0
LADDER REM: paidao
LADDER DES: beigeda01
MODIFIED: 2025-01-21 14:28
LADDER ROWS: 1419/1600
EXEC STEPS : 4448/4700
X(MT->PLC) X0-X63 K(Relay) K0-K63
Y(PLC->MT) Y0-Y47 R(Relay) R0-R511
F(NC->PLC) F0-F63 D(DATA TABLE) D0-D127
G(PLC->NC) G0-G63 C(Counter ) C0-C127
A(A Class alarm) A0-A31 T(Timer) T0-T127
DATA:
MDI RET
CUT TIME:00:00:00
Ladder
PLC Para
PLC Signal
PLC Info
Figure 3-7-4-1

# Definition and Connection of Interface III

# 1.1 X. Connection of Y-axis and Z-axis and A-axis interfaces

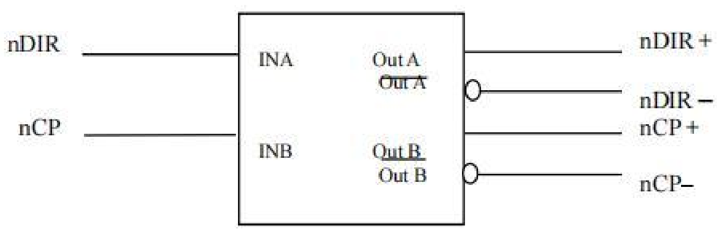
## 1.1.1 Driver interface definition




| Pin | signal | description | Wire Color |
|---|---|---|---|
| 1 | nCP+ | Command pulse signal+ | yellow |
| 2 | nDIR + | Command direction signal+ | blue |
| 3 | nPC | Zero point signal | black |
| 4 | +24V | +24V | Red/Red and White |
| 5 | nALM | Drive alarm signal | black and white |
| 7 | nEN | Axis enable signal | green black |
| 9 | nCP− | Command pulse signal- | Yellow black |
| 10 | nDIR − | Command direction signal- | blue-black |
| 11. 14. 15 | GND | 0V | green |
| 12. 13 | VDD5 | 5V | |

## 1.1.2 Command pulse signal and command direction signal

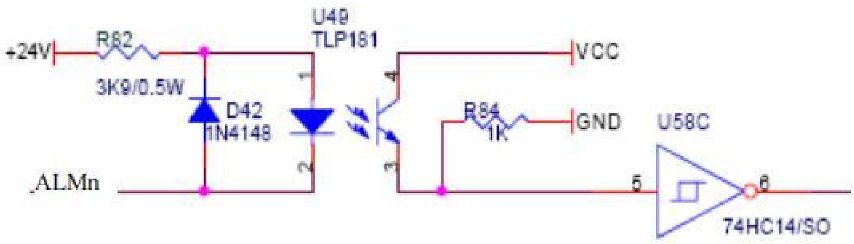
CPn+and CPn - are instruction pulse signals, and DIRn+and DIRn - are instruction direction signals. Both sets of signals are differential (AM26LS31) outputs, and the internal circuit is shown in the following figure




Internal circuit diagram of instruction pulse signal and instruction direction signal

## 1.1.3 Drive unit alarm signal nALM

Set the alarm level of the X, Y, Z, 4th, and 5th axis drive units to low or high based on the CNC status parameter №. 009 Bit0 to Bit4 or in the classification parameter [feed axis]. Internal circuit diagram




Internal circuit diagram of drive unit alarm signal

## 1.1.4 Axis enable signal ENn

When the CNC is working normally, the ENn signal output is valid (ENn signal is connected to 0V). When the driver alarms, the CNC turns off the ENn signal output (ENn signal is disconnected from 0V). The internal interface circuit is shown in the following figure

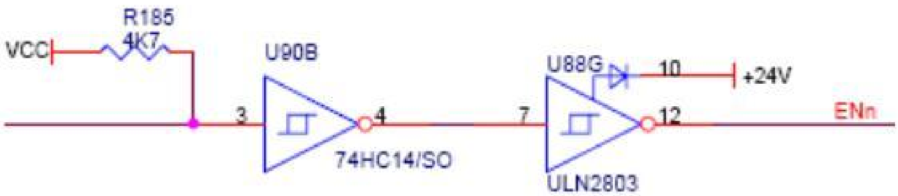



Figure1.1.4 Internal interface circuit diagram of axis enable signal

## 1.1.5 Zero point signal PCn

The zero point signal of the machine tool is obtained by using the one rotation signal of the motor encoder or the proximity switch signal. The internal connection circuit is shown in the following figure

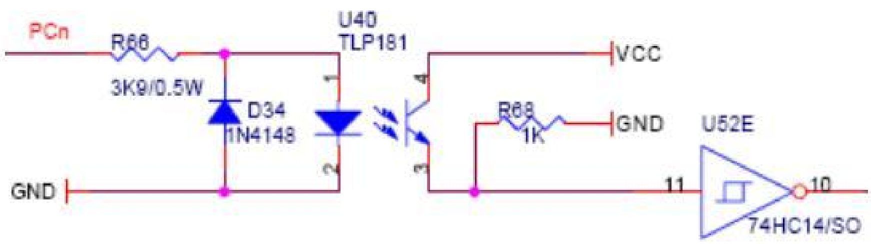



Figure1.1.5 Zero signal circuit diagram note: PCn signal adopts+24V level

a) The connection method for using an NPN Hall element as both a deceleration signal and a zero signal is shown in the following figure

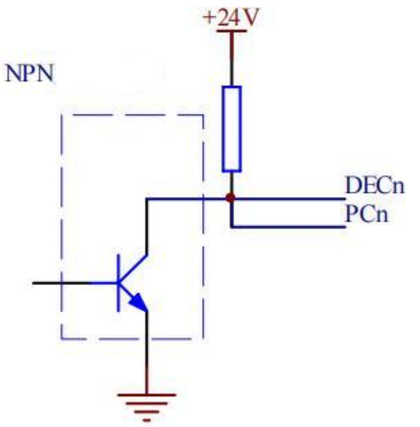



Figure1.1.5.2Connection diagram of NPN Hall element: Pull up resistor connected to 2K

b）The connection method for using a PNP Hall element for both deceleration signal and zero signal is shown in the following figure

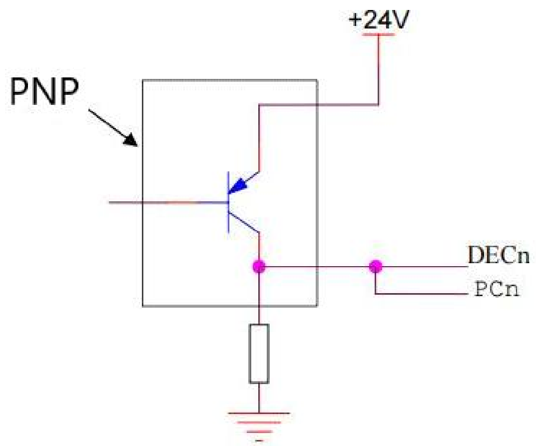



Figure1.1.5.3 Connection diagram using PNP type Hall element

The waveform of the system's PC signal is shown in the following figure:

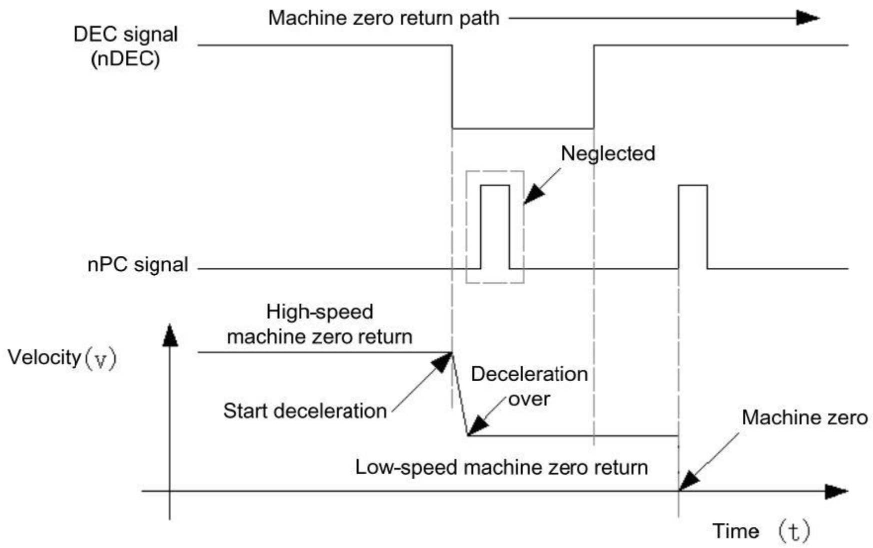
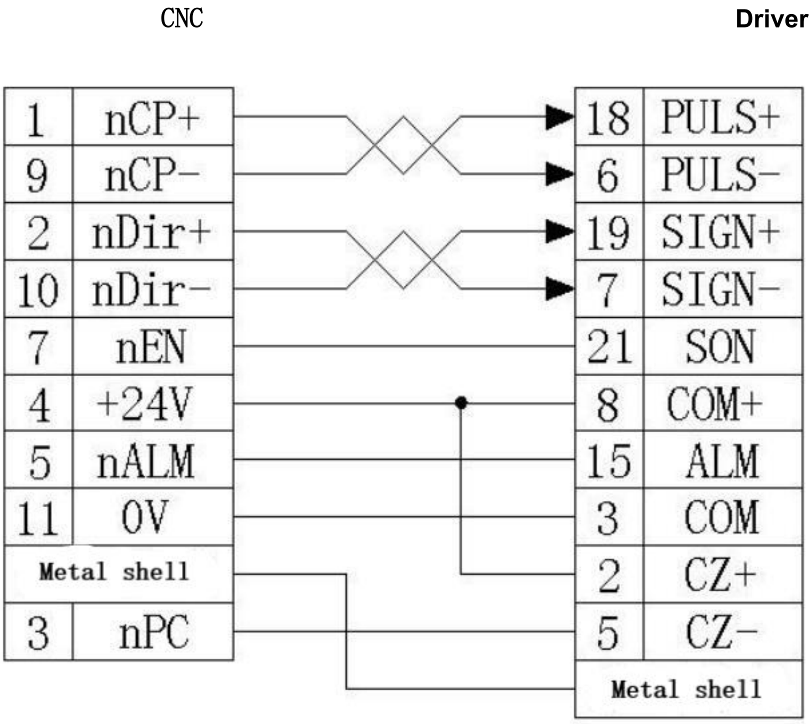



Figure1.1.5.4 PC signal waveform diagram

Note: When the machine tool returns to zero, CNC determines the position of the reference point by detecting the jump of the PC signal after the deceleration switch is disengaged, and both the rising and falling edges are valid.

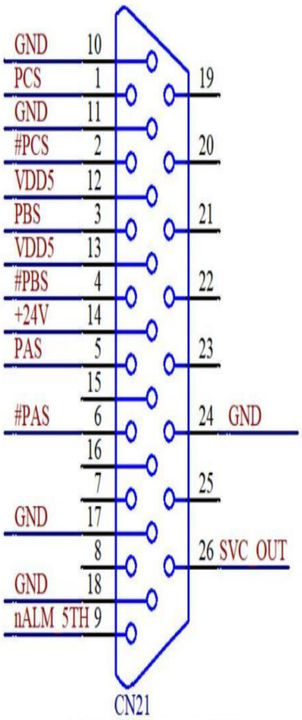
### 1.1.6 Connection with servo drive

The connection between the system and our company's driver unit is shown in the following figure:



# 1.2 Connection of spindle encoder interface

## 1.2.1 Definition of spindle encoder interface




| Pin | signal | description | Wire Color |
|---|---|---|---|
| 1 | PCS | Encoder Z phase pulse | |
| 2 | #PCS | Encoder Z- phase pulse | |
| 3 | PBS | Encoder B phase pulse | |
| 4 | #PBS | Encoder B- phase pulse | |
| 5 | PAS | Encoder A phase pulse | |
| 6 | #PAS | Encoder A- phase pulse | |
| 12. 13 | VDD5 | 5V | |
| 11 | GND | 0V | |
| 26 | SVC OUT | 0~10V Analog voltage | **Red** |
| 10. 24 | GND | 0V | **blue-black** |
| 9 | ALM | Spindle Alarm | **blue** |
| 17. 18 | GND | 0V | **Red-White** |

## 1.2.2 SVC signal description

The SVC end of the simulated spindle interface can output a voltage of 0~10V. The internal circuit of the signal is shown in the following figure:

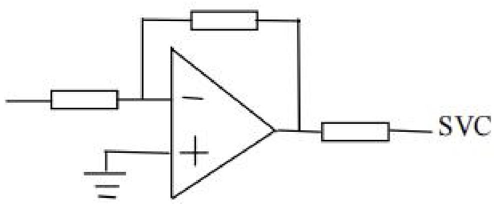



Figure1.2.2.1 SVC signal internal circuit diagram

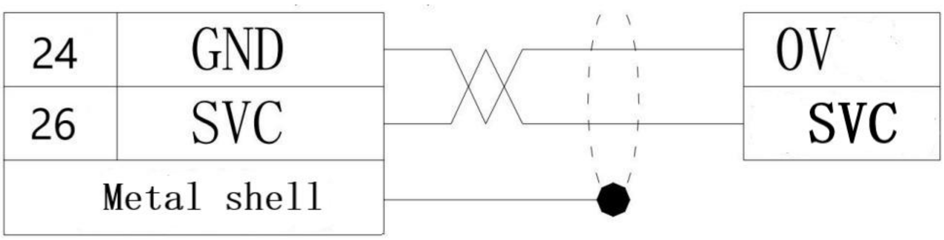
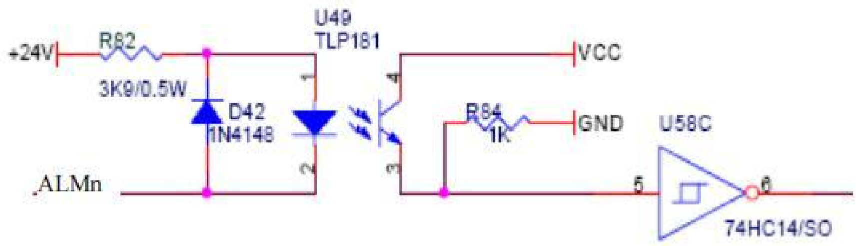



Figure1.2.2.2 Connection between the system and the frequency converter

## 1.2.3 ALM Description

The spindle alarm signal is used as an alarm signal for ordinary frequency converters or gear spindles




Internal structure diagram of spindle alarm note: spindle alarm input 0 volts is valid

## 1.2.4 signal description

*PCS/PCS,*PBS/PBS,*PAS/PAS are the encoder C, B, A phases differential input signals
respectively, which are received by 26LS32; *PAS/PAS,*PBS/PBS are orthogonal square wave with
phase shift 90°and their maximum signal frequency is less than 1MHz; the encoder pulses for
are set at will by parameter, the setting range is from 100 to 5000.
Its interior circuit is shown in Fig

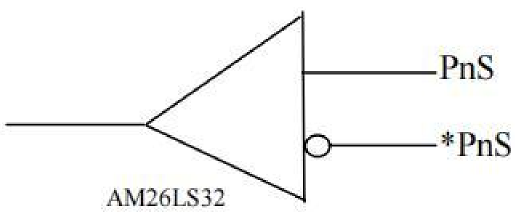



Figure1.2.4.1 Encoder signal circuit

## 1.2.5 Spindle encoder interface connection

The connection between the system and the spindle encoder is shown in the following figure, using twisted pair cables.

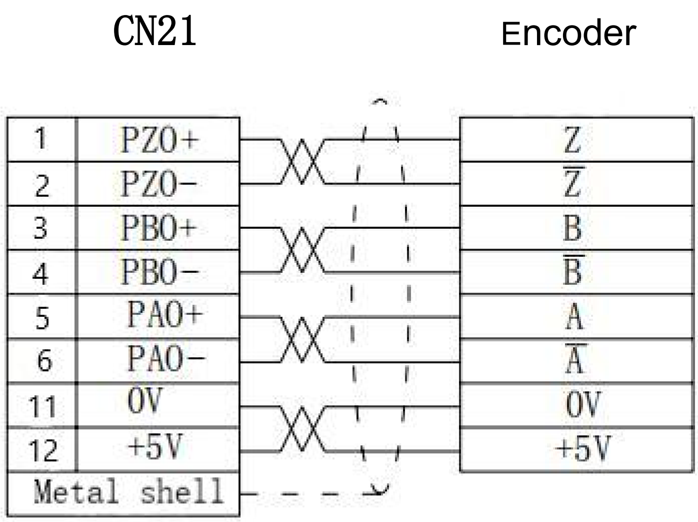
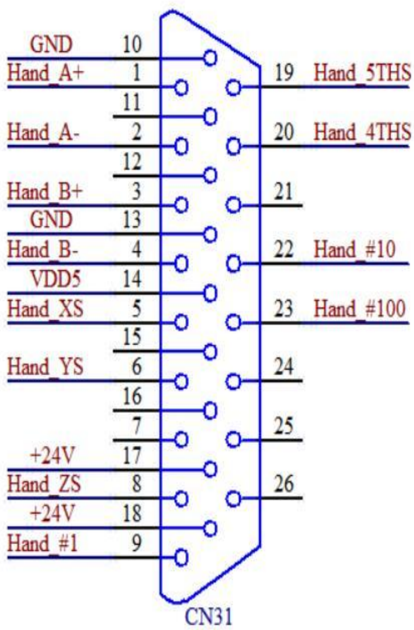



Figure1.2.5.1 Connection between system and encoder

# 1.3 Connection of hand pulse interface

## 1.3.1 Hand pulse interface definition




| Pin | signal | description |
|---|---|---|
| 1 | Hand A+ | MPG A+ phase signal |
| 2 | Hand A− | MPG A- phase signal |
| 3 | Hand B+ | MPG B+ phase signal |
| 4 | Hand B− | MPG B- phase signal |
| 5 | Hand XS | X MPG axis selection |
| 6 | Hand YS | Y MPG axis selection |
| 8 | Hand ZS | Z MPG axis selection |
| 9 | Hand #1 | Increment0.001 |
| 19 | Hand 5THS | 5TH_axis selection signal |
| 20 | Hand 4THS | 4TH_axis selection signal |
| 22 | Hand #10 | Increment0.01 |
| 23 | Hand #100 | Increment0.1 |
| 14 | VDD5 | 5V |
| 10. 13 | GND | 0V |
| 17. 18 | 24V | 24V axis selection common terminal |

# 1.3.2 signal description

HA+、HA-、HB+、HB- The input signals for hand pulse phases A and B. The internal connection circuit is shown in the following diagram

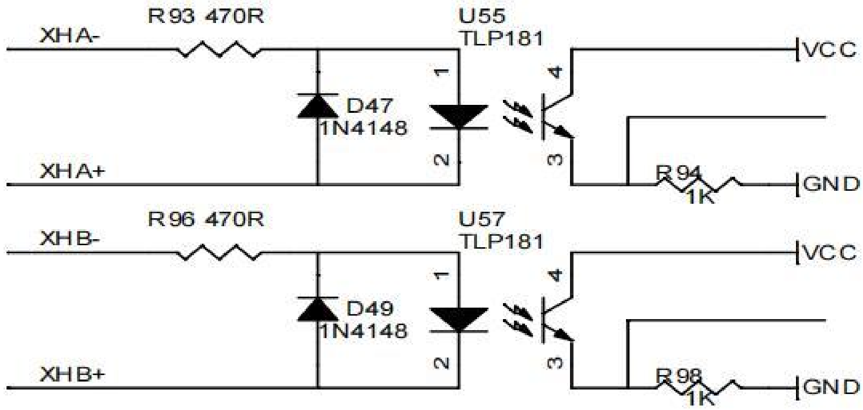
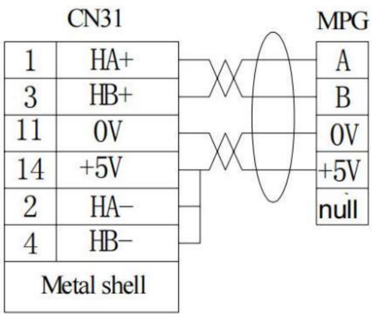
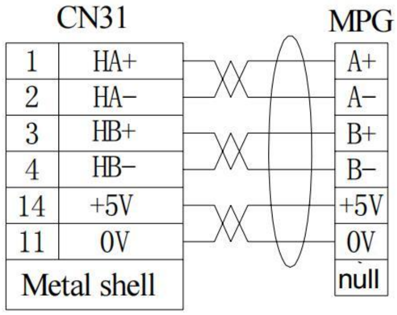


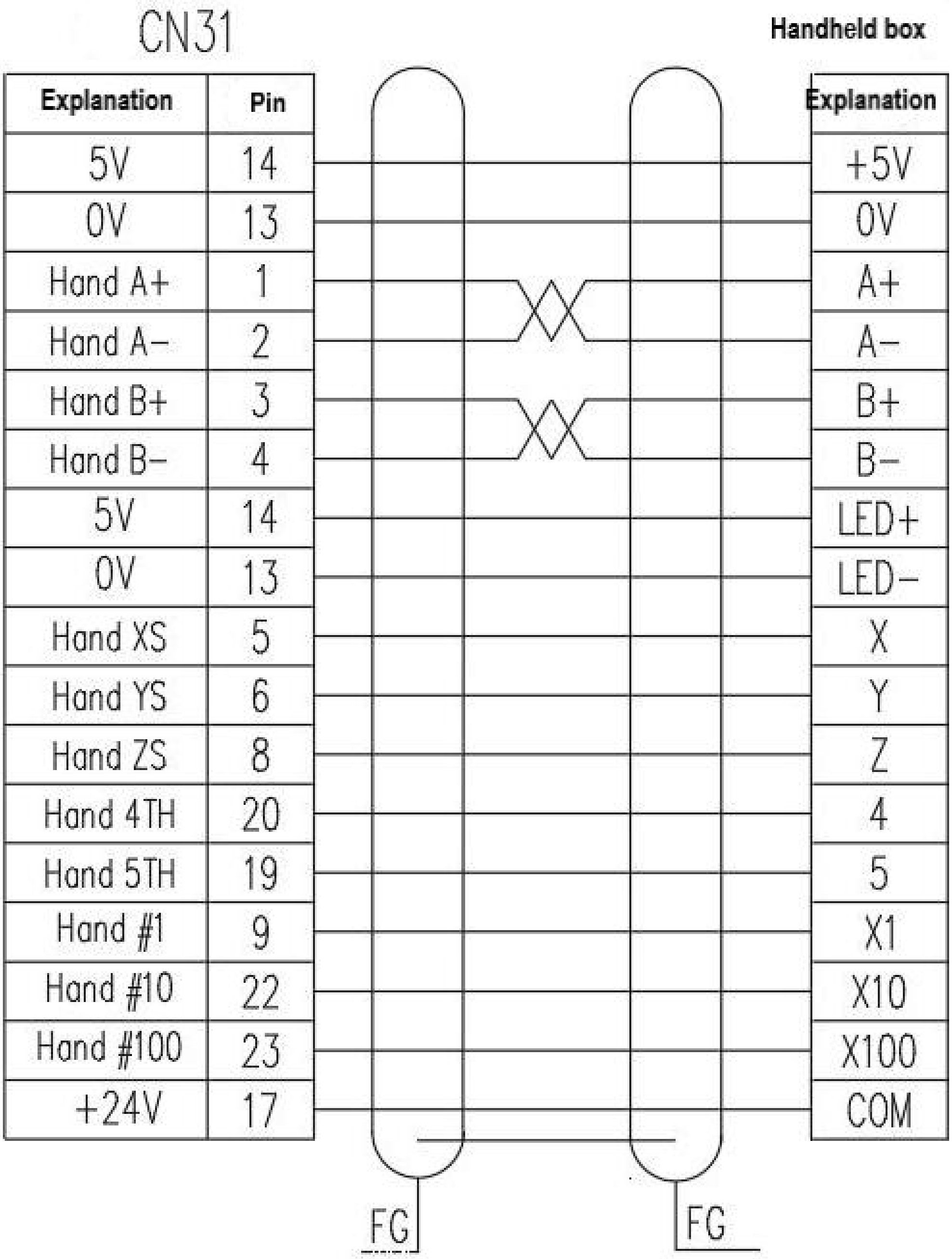

Figure1.3.2.1 MPG signal circuit

The connection between the system and the hand pulse is shown in the following figure



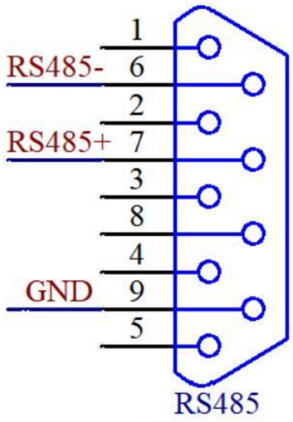




CN31 connects with the single-port MPG input difference MPG input




Note: The external handheld box handwheel axis selection and magnification wiring common terminal COM is 24 volts.

# 1.4 communication interface：




| Pin | signal | description |
|---|---|---|
| 6 | RS485- | signaling |
| 7 | RS485+ | signal reception |
| 9 | GND | 0V |

485 communication usage:

The system reads the motor position through 485 communication and driver connection. No other functions

## 1.5 I/O interface definition

**Note!**

**The meaning of the fixed address I/O function of the lathe CNC is defined by the PLC program (ladder diagram). When the lathe CNC is assembled with the machine tool, the I/O function is determined by the machine tool manufacturer's design. Please refer to the user manual of the machine tool manufacturer for details.**

**The I/O function without fixed address annotation in this section is described for standard PLC programs. Unless otherwise specified, the description also applies to the system. Please be aware!**

## 1.5.1 Input signal (CN61)

| Pin | Address | Explanation | Wire Color |
|---|---|---|---|
| 1 | X0.3 | Spindle tightening tool detection | orange |
| 2 | X0.2 | low oil pressure | |
| 3 | X0.1 | Z-axis deceleration signal back to reference point | Orange- Black |
| 4 | X0.0 | Spindle loosening detection | brown |
| 5 | X1.7 | X-axis deceleration signal back to reference point | dark -brown |
| 6 | X1.6 | X-axis limit switch | blue |
| 9 | X1.3 | Z-axis limit switch | blue-black |
| 12 | Y1.6 | Clamping output | Yellow |
| 13 | Y1.7 | Chuck loose output | Yellow- black |
| 19 | X1.5 | Y-axis deceleration signal back to reference point | green -black |
| 20 | X1.0 | Spindle external clamp/loose blade control | Red-White |
| 21 | X1.2 | Air pressure detection | black-white |
| 22 | X0.4 | Y-axis limit switch | black |
| 23 | +24V | 24V | red |
| 14～18 24.25 | 0V | 0V | green |
| 7 | X1.4 | External start signal ST | blue-black(4-core wire) |
| 8 | X1.1 | Emergency stop signal ESP | blue(4-core wire) |
| 10 | X0.5 | External pause signal SP | Red -white(4-core wire) |
| 11 | +24V | 24V | red(4-core wire) |

Note 1: The I/O function of lathe CNC is defined by the ladder diagram.
Note 2: When the input function is valid, the input signal is connected to+24V. When the input function is invalid, the signal is disconnected from+24V.
Note 3:+24V and 0V are equivalent to the same named terminals of the power box.

### Input signal

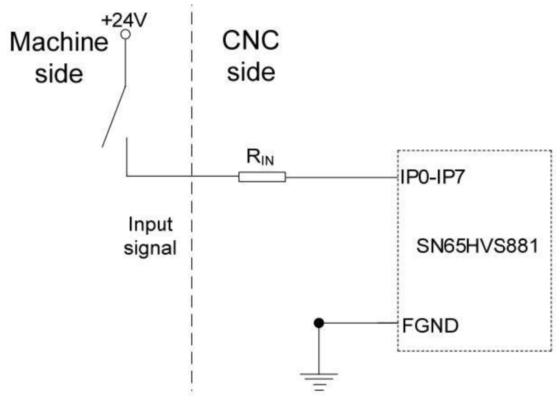
The input signal refers to the signal from the machine tool to the CNC, which is valid when connected to+24V; When the input signal is disconnected from+24V, the input is invalid. The contact points of the input signal on the machine tool side should meet the following conditions:

Contact capacity: DC30V, 16mA or above

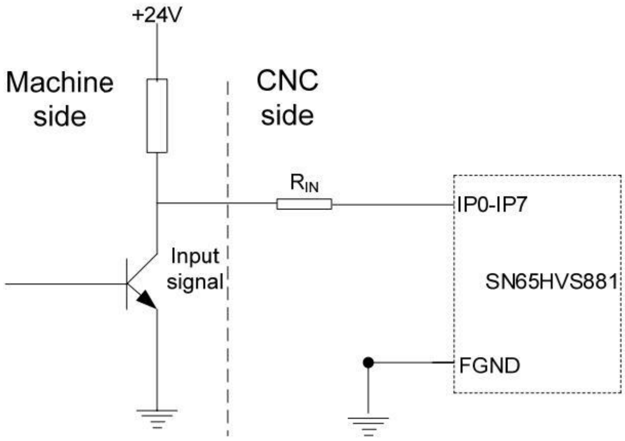
Leakage current between contacts during open circuit: below 1mA

Voltage drop between contacts during closed circuit: below 2V (current 8.5mA, including cable voltage drop)

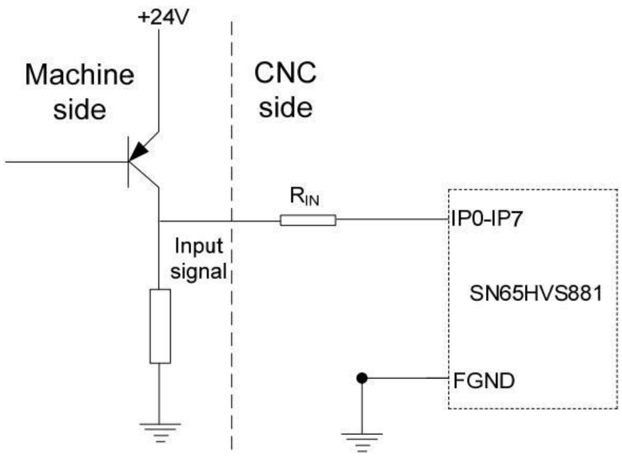
There are two ways to input external signals: one is to use a contact switch input, and the signals using this method come from the machine's side buttons, limit switches, and relay contacts, etc., connected as shown in the diagram




The other type is input by switch with no contacts (transistor) as shown in Fig:




Connection of NPN type




Connection of PNP type

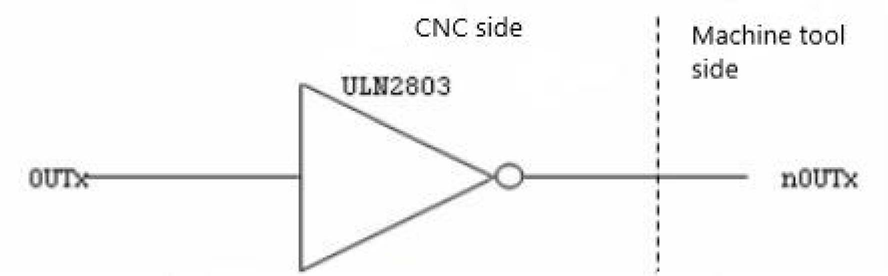
# 1.5.2 Output Signal (CN62)

| Pin | Address | Explanation | Wire Color |
|---|---|---|---|
| 1 | Y0.0 | cool | dark purple |
| 2 | Y0.2 | Spindle tool loose/tight tool output | purple |
| 3 | Y0.4 | Spindle reversal | Black- white |
| 4 | Y0.6 | lighting | black |
| 5 | Y1.0 | Gear spindle 1/spindle directional output | blue-black |
| 6 | Y1.1 | Gear spindle 2nd gear/position switching output/yellow light | blue |
| 7 | Y1.2 | Gear spindle 3/green light | dark- brown |
| 8 | Y1.3 | Gear spindle 4/red light | brown |
| 9 | Y1.4 | Inverted output of chip conveyor | Orange -Black |
| 10 | Y1.5 | Forward rotation output of chip conveyor | orange |
| 11 | X0.6 | Spindle switching completed input signal/tailstock input | Yellow- black |
| 12 | X0.7 | Spindle orientation completed input signal | Yellow |
| 14 | Y0.1 | Lubrication output | white |
| 15 | Y0.3 | Main spindle rotates forward | grey |
| 16 | Y0.5 | blowing | green -black |
| 17 | Y0.7 | Spindle brake output | Red -white |
| 18～24 | 0V | 0V | green |
| 13.25 | +24V | +24V | red |

Note 1: The I/O function of lathe CNC is defined by the ladder diagram.
Note 2: When the output function is valid, the output signal conducts with 0V. When the output function is invalid, the output signal is high impedance cutoff.
Note 3:+24V and 0V are equivalent to the same named terminals of the matching power box.

### Output signal

The output signal is used to drive the relays and indicator lights on the machine side. When the output signal is connected to 0V, the output function is effective; When disconnected from 0V, the output function is invalid. There are a total of 16 digital outputs in the I/O interface, all of which have the same structure, as shown in the figure

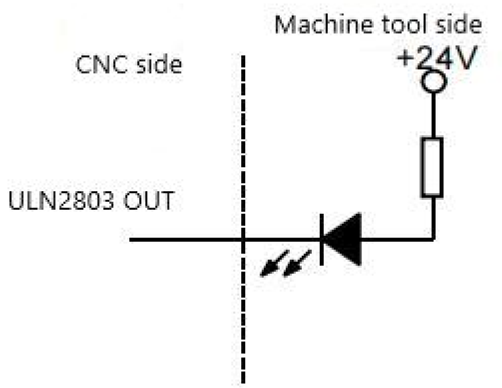



Circuit structure diagram of digital output module

The logic signal OUTx output by the motherboard is sent to the input terminal of the inverter (ULN2803) through the connector. nOUTx has two output states: 0V output or high resistance. Typical applications are as follows:

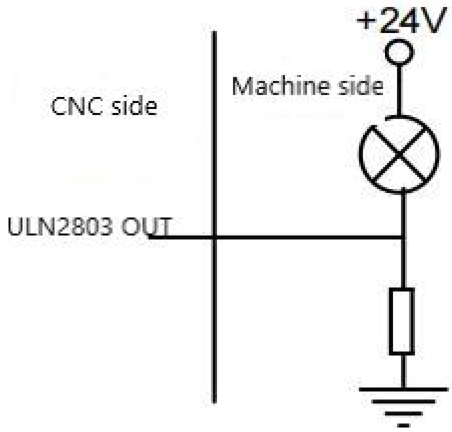
Drive LED

To drive a light-emitting diode with ULN2803 output, a resistor needs to be connected in series to limit the current flowing through the diode (usually around 10mA). As shown in the following figure



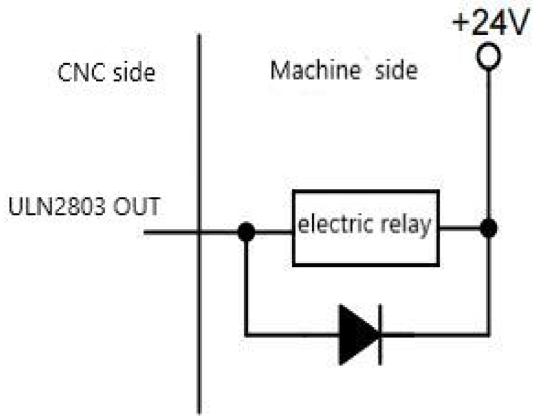

**Drive filament type indicator light**

To drive the filament type indicator light with ULN2803 output, an external preheating resistor is required to reduce the current surge during conduction. The resistance value of the preheating resistor should be set to prevent the indicator light from turning on, as shown in the following figure

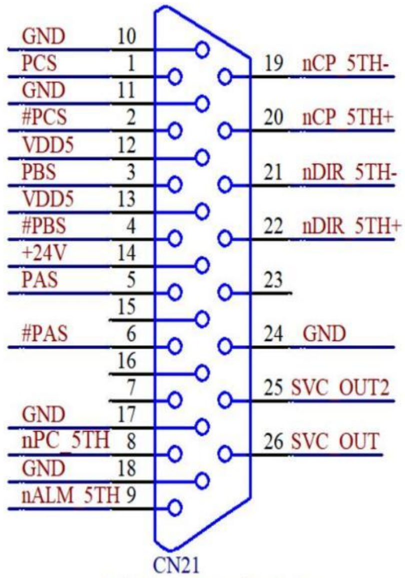



**Drive inductive loads (such as relays)**

When using ULN2803 output to drive inductive loads, it is necessary to connect a freewheeling diode near the coil to protect the output circuit and reduce interference. As shown in the following figure

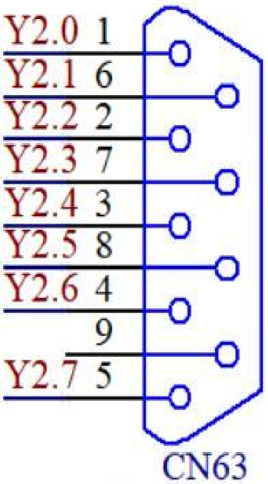


### 1.5.3 Fifth axis signal (5-axis system with additional port)



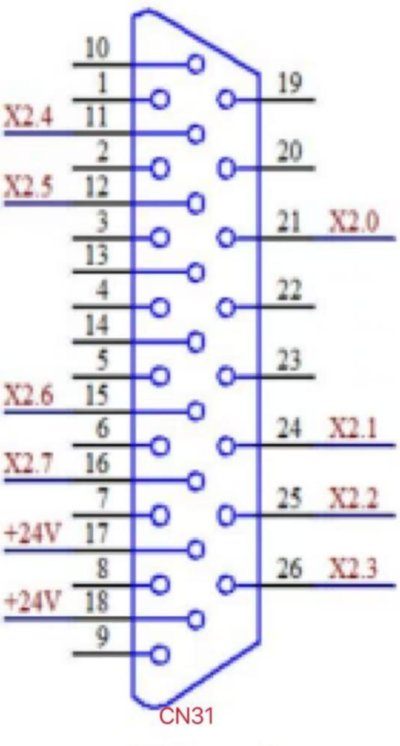

| Pin | signal | description | Wire Color |
|---|---|---|---|
| 19 | nCP 5TH- | Command pulse signal- | Yellow black |
| 20 | nCP 5TH+ | Command pulse signal+ | yellow |
| 21 | nDIR 5TH- | Command direction signal- | Blue-black |
| 22 | nDIR 5TH+ | Command direction signal+ | blue |
| 8 | PC 5TH | | |
| 25 | SVC OUT2 | Second spindle analog quantity | |
| 9 | nALM 5TH | Fifth axis alarm signal | |

### 1.5.4 Output signal 2 (5-axis system with additional port)




| Pin | signal | description |
|---|---|---|
| 1 | Y2.0 | tool magazine forward/tool sleeve vertical |
| 2 | Y2.2 | tool magazine rotates forward |
| 3 | Y2.4 | Mechanical arm output |
| 4 | Y2.6 | Fourth axis clamping output |
| 5 | Y2.7 | reserve |
| 6 | Y2.1 | tool magazine retreat/tool cover horizontal |
| 7 | Y2.3 | tool magazine reversal |
| 8 | Y2.5 | Release the output of the fourth axis |

## 1.5.5 Input signal 2 (5-axis system with additional port)




| Pin | signal | description |
|---|---|---|
| 11 | X2. 4 | tool arm buckle in place |
| 12 | X2. 5 | tool arm zero point |
| 15 | X2. 6 | tool arm brake |
| 16 | X2. 7 | Fourth axis release button input |
| 21 | X2. 0 | tool magazine return to zero signal |
| 24 | X2. 1 | tool counting signal |
| 25 | X2. 2 | The tool magazine advances in place/the tool sleeve is vertical |
| 26 | X2. 3 | The tool magazine retreats to the correct position/the tool cover is level |

## 1.6 Standard ladder diagram function

### 1.6.1 Limit and emergency stop

Related signals
ESP: X1.1 emergency stop signal, emergency stop alarm when disconnected from+24V
LMIX: X-axis travel limit detection input X1.6
LMIX: X-axis travel limit detection input X0.4
LMIZ: Z-axis travel limit detection input X1.3
External connection of machine tool
① The series connection between the emergency stop and travel switch is shown in Fig. 2-27:

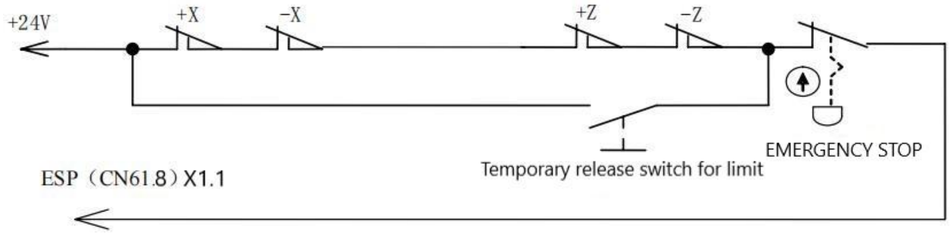



Fig.2-27 Series connection between emergency stop and travel switch

② The separate connection between the emergency stop and travel switch is shown in Fig. 2-34B

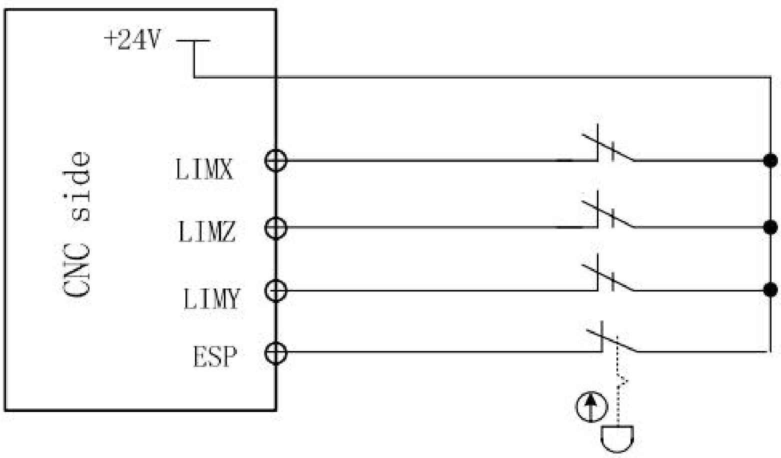



Fig. 2-28 Separate connection between the emergency stop and travel switch

**Control Logic**
① Travel limit and emergency stop are connected in series
When an overtravel occurs or the emergency stop button is pressed, the CNC will sound an "emergency stop" alarm. If it is an overtravel, press the overtravel release button without releasing it, press the reset button to cancel the alarm, and move in the opposite direction to release the overtravel. When an emergency stop alarm occurs, the CNC stops pulse output. In addition to the functions of CNC processing mentioned above, other functions can also be defined by the PLC program during emergency stop alarms. The function defined by the standard PLC program is to turn off the M03, M04, M08 signal output and output the M05 signal when an emergency stop alarm is triggered.
② Travel limit and emergency stop are independently connected
1. Each axis has only one overtravel contact, and the positive and negative overtravel alarms are determined by the direction of axis movement.
2. When an overtravel alarm occurs, it can be moved in the opposite direction. After moving out of the limit position, the alarm can be cleared by pressing the reset button.

Note: Before enabling the overtravel limit function, it is necessary to ensure that the machine tool trailer is between positive and negative travel, otherwise the alarm prompt will not match the actual situation.

**Control parameters**

Parameter - [Quick Debugging]

| | | |
|---|---|---|
| 001 | Did you check the emergency stop signal (0: check 1: not check) | check |

Parameter - [Limit Return to Zero]

| | | |
|---|---|---|
| 001 | Hard limit detection function for each axis (0: invalid 1: effective) | invalid |
| 002 | Hard limit detection signals for each axis (0: high level 1: low level) | low level |

| | | |
|---|---|---|
| 002 | Hard limit detection function for each axis (0: invalid 1: effective) | effective |
| 003 | X-axis hard limit X1.6 high and low level selection (0: high 1: low) | low |
| 004 | Y-axis hard limit X0.4 high and low level selection (0: high 1: low) | low |
| 005 | Z-axis hard limit X1.3 high and low level selection (0: high 1: low) | low |

### 1.6.2 External cycle start and feed hold

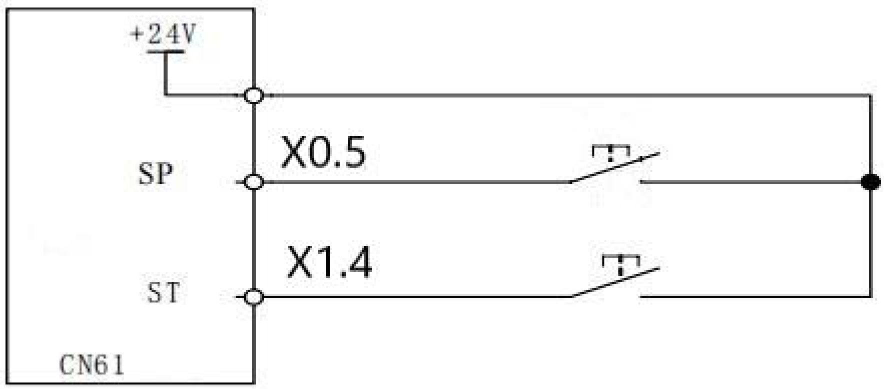
- Related signals (standard PLC program definition)

ST: The external automatic cycle start signal has the same function as the automatic cycle start button on the machine panel.

SP: The external feed hold signal has the same function as the feed hold key on the machine panel.

- External connection circuit

The external connections of SP and ST signals are shown in the following figure



- **Control parameters**

Parameters - [Quick Debugging]

| | | |
|---|---|---|
| 026 | External loop start signal (0: high level 1: low level) | high level |
| 027 | External input to hold signal (0: high level 1: low level) | high level |

## 1.6.3 Control of spindle counterclockwise and clockwise rotation

- **Address Definition**

Signal address when connecting the frequency converter:

Y0.3: Output signal of spindle counterclockwise rotation (M03)

Y0.4: Output signal for clockwise rotation of spindle (M04)

- **Control parameters**

Parameters - [PLC Parameters]

| | | |
|---|---|---|
| 001 | Lubrication and cooling of the spindle during reset (0: off 1: hold) | off |

Parameters - [Spindle Parameters]

| | | |
|---|---|---|
| 003 | First gear spindle maximum speed | 6000 |

- **control logic**

After CNC is powered on, the M05 output is effective.

When the output of M05 is valid, execute M03 or M04, and keep the output of M03 or M04 valid while turning off the output of M05.

When the output of M03 or M04 is valid, execute M05, turn off the output of M03 or M04, and keep the output of M05 valid;

When the output of M03 (or M04) is valid, the M04 (or M03) system will generate an alarm prompt.

> Note: During CNC emergency stop, turn off the M03 or M04 signal output and output the M05 signal at the same time.

## 1.6.4 Spindle orientation and position switching function

- **Address Definition**

Y1.0: Spindle orientation output signal (M19)

Y1.1: Output signal for spindle position switching (M28)

X0.6: Spindle orientation detection completion signal

X0.7: Spindle position detection completion signal

- **Control parameters**

Parameters - [Tool Library Parameters]

| | | |
|---|---|---|
| 006 | The delay filtering of the spindle orientation signal is turned off (unit: millimeter) | 50 |

Parameter - [Rigid Tapping]

| | | |
|---|---|---|
| 020 | Is the rigid tapping properly detected signal (0: not detected 1: detected) | detected |
| 021 | Does the rigid tapping output forward and reverse signals (0: no output 1: output) | no output |

- Directional control logic

① After executing the positioning function instruction M19, the PLC → Drive sends a positioning selection signal Y1.0 to determine the positioning position;
② Delay 40ms, PLC → Drive outputs spindle orientation signal SORI;
③ Drive begins to locate;
④ After the Drive positioning is completed, Drive → PLC outputs the spindle positioning completion signal COIN;
⑤ If the PLC does not receive the positioning completion signal within 6000ms after sending out the positioning selection signal, the system will issue an "spindle positioning time too long" alarm.
⑥ Before positioning, the spindle can be in a rotating or stopped state, and after positioning is completed, the spindle will be in a stopped state.

- Control logic for spindle position switching

① After executing the switching function instruction M28, the PLC → CNC sends a switching signal;
② CNC requires servo drive to switch to position control through Y1.1 signal;
③ After the servo drive switch is completed, the X0.7 signal is sent to the CNC indicating that the current servo has been switched to position control;
④ After receiving servo feedback information, CNC → PLC outputs a signal indicating the completion of spindle switching;
⑤ If the PLC does not receive the switch completion signal within the set time after sending the switch signal, the system will display an alarm of "First spindle position switch timeout, CS switch failure";
⑥ Before positioning, the spindle can be in a rotating or stopped state. After the switch is completed, the spindle will be in a position control state;
⑦ CNC executes function instruction M29, and the spindle will switch to speed control mode;
⑧ If the servo drive triggers an alarm or a switch timeout occurs, the CNC will switch the spindle back to speed control mode

## 1.6.5 Lubrication control

- Parameter - [Quick Debugging]

| | | |
|---|---|---|
| 32 | When automatic lubrication is effective, start up and output lubrication (0: No 1: Yes) | NO |
| 34 | Lubrication start output time (in milliseconds) T013 | 0 |
| 35 | Automatic lubrication interval time (in seconds) T014 | 7200 |
| 33 | Lubrication start output time (unlimited lubrication time) | invalid |

Lubrication start output time (unlimited lubrication time)

- **Functional Description**

There are two types of lubrication functions, non automatic lubrication and automatic lubrication. When T13=0 or T14=0, the automatic lubrication function is invalid.

**a) Non automatic lubrication**

When T13>0, lubrication output timing. When the panel key 

is valid or the M32 command is executed, the lubrication output is valid, and the indicator light signal output is valid. After the time set by T14, the lubrication output and indicator light output are cancelled; If the time set for T14 is not reached and M33 command is executed at this time, the lubrication output and indicator light output will be cancelled.

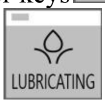
When T13=0, lubrication flip output. When the panel key 

is valid or the M32 command is executed, the lubrication output is valid, and at the same time, the indicator light signal output is valid; When the panel key 

is activated again or the M33 command is executed, the lubrication output and indicator light output are cancelled.

**b) Automatic lubrication**

When T13>0, When T14>0, the system starts timing the time set by T14 after power on, and then lubricates the output. After the time set by T13, the lubrication output stops and cycles in sequence. During automatic lubrication, if it is in the lubrication interval time, the panel keys 

and M32, M33 commands are valid; If it is in lubrication output time, the panel keys 

and M32, M33 commands are invalid.

> Note 1: When CNC emergency stop or M30 is executed, the output of M32 is cancelled and lubrication is turned off.
> Note 2: When resetting CNC, the Bit1 bit of K10 is used to set whether to cancel the output of M32.
> Note 3: M33 has no corresponding output signal. Execute M33 to cancel the output of M32 and turn off lubrication.

## 1.6.6 Main spindle elastic tool function

- **Address Definition**

X1.0: External clamp/loose blade control input

X0.0: Loose tool detection input

X0.3: Tight tool detection input

Y0.2: Loose/tight tool output

- **Control parameters**

Parameters - [Tool Library Parameters]

| | | |
|---|---|---|
| 010 | Whether to use a tool clamping device (0: No 1: Yes) | |

| | | |
|---|---|---|
| 011 | Does the elastic tool button maintain output (0: not maintained 1: maintained) | |
| 012 | Did you check the tightness tool in place signal when the tool magazine is not in use (0: No 1: Yes) | |
| 013 | Tightening tool in place X0.3 high and low level selection (0: high 1: low) | |
| 014 | Loosening tool in place X0.0 high and low level selection (0: high 1: low) | |
| 015 | Delay of spindle tool release completion (unit: milliseconds) | |
| 016 | Delay of spindle tool clamping completion (unit: milliseconds) | |
| 017 | Spindle tool release detection delay (unit: milliseconds) | |
| 018 | Delay in spindle tool clamping detection (unit: milliseconds) | |

- **Function Description**

Whether the elastic tool button maintains the output parameter description: When the parameter is changed to not maintain, manually press the elastic tool input signal, keep pressing it, and the loose tool signal will continue to output. When there is no signal input for the elastic tool signal, there will be no output for the loose tool signal.
When the parameter is changed to hold, press the input signal of the tension tool once, and the output signal of the tension tool will continue to be output. Press the input signal of the tension tool again, and the output signal of the tension tool will be turned off.

## 1.6.7 Rigid tapping debugging

- **correlated signal**

  Y1.1: Rigid tapping control signal
  X0.7: Spindle position detection completion signal

- **Control parameters**

Parameters - [Rigid Tapping Parameters]

| | | |
|---|---|---|
| 001 | The spindle control mode during tapping is (0: follow 1: servo) | servo |
| 002 | Has tapping become a high-speed deep hole tapping cycle (0: No 1: Yes) | No |
| 003 | Rigid tapping method (0: no M code decoding 1: with M code) | with M code |
| 004 | Rigid tapping (0: F value=speed * pitch 1: F value is the pitch value) | F value=speed * pitch |
| 005 | Rigid tapping retreat plane (0: R plane 1: initial Z plane) | R plane |
| 007 | Deep hole rigid tapping (0: retreat to reference R point 1: retreat according to P339) | retreat to reference R point |
| 008 | Retraction or clearance during deep hole tapping cycle | |
| 009 | Maximum spindle speed during tapping cycle | |
| 010 | Thread spindle instruction multiplication factor (CMR) (1st gear) | |
| 011 | Tapping spindle instruction frequency division coefficient (CMD) (1st gear) | |
| 020 | Is the rigid tapping properly detected signal (0: not detected 1: detected) | |

| | | |
|---|---|---|
| 021 | Does the rigid tapping output forward and reverse signals (0: no output 1: output) | output |

## ➢ Function Description

When the M29 program is executed, Y1.1 is output simultaneously. The electrical design of the machine tool can use this signal to control the spindle servo to switch to position mode. When the program reaches G84, the system simultaneously sends pulses to each Z of the spindle for rigid tapping processing.

When executing M28 to cancel rigid tapping, simultaneously close Y1.1

### ➢ Example of pulse ratio setting

Note: Tapping spindle instruction multiplication factor (CMR) (first gear), which is the ratio of rigid tapping pulses to molecules

Tap spindle instruction frequency division coefficient (CMD) (first gear), which is the denominator of the rigid tap pulse ratio

A> When the servo motor of the spindle is connected to the spindle of the machine tool in a 1:1 ratio, the encoder of the spindle servo motor is 1024 lines. If the spindle servo is

The frequency doubling coefficient of the servo is 4, so the number of pulses feedback for one rotation of the spindle servo is 1024 * 4=4096

The frequency division coefficient (CMD) of the tapping spindle command (first gear) is fixed at 1000, so the servo rigid tapping pulse ratio of the spindle in this configuration is: 4096:1000

Agreed share: 512:125

Thread spindle instruction multiplication factor (CMR) (first gear) set to 512

Tap spindle instruction frequency division coefficient (CMD) (first gear) set to 125

B> For example, when the servo motor of the spindle is connected 1:1 with the spindle of the machine tool, the encoder of the spindle servo motor is 2500 lines

C> If the multiplier coefficient of the spindle servo is 4, then the number of pulses fed back by one rotation of the spindle servo is 2500 * 4=10000

The frequency division coefficient (CMD) of the tapping spindle instruction (first gear) is fixed at 1000, so this configuration has a spindle servo rigid tapping

The pulse ratio is:

10000 ：1000

Agreed share: 10:1

Thread spindle instruction multiplication factor (CMR) (first gear) set to 10

The frequency division coefficient (CMD) of the tapping spindle command (first gear) is set to 1

The same applies to other configurations!

D> Method for verifying the correctness of setting the rigid tapping pulse ratio (gear ratio)

After setting the gear ratio, the following program can be run to verify the correctness of the rigid tapping pulse ratio (gear ratio) setting.

```
O0075;
G80 G90 G54 G0 X0 Y0 Z0; // Z-axis positioning to 0.000 (key is Z-axis);
M29 S1; // The specified spindle speed for rigid tapping is: 1 revolution;
G84 X0 Y0 Z-1.0 R0 F1.0; // Reference surface R0.000 Tapping depth: Z-1.000 Tapping pitch: 1.0
M28;//  Cancel rigid tapping
M30;// END
```

Explanation: When the above program runs, the spindle rotates forward once, and the Z-axis is precisely fed from the 0.000mm position to -1.000mm,

The spindle rotates in reverse once, and the Z-axis just retreats from the -1.000mm position to 0.000mm. Otherwise, it indicates that the parameter setting for the rigid tapping pulse ratio (gear ratio) is incorrect.

### 1.6.8 Machining center tool magazine

The standard ladder diagram supports four types of tool magazine control logic; Select which tool magazine to adapt to by setting the control bit corresponding to the parameter - [Tool Library Parameter] 001 tool library parameter。

| | | |
|---|---|---|
| 001 | Tool library type 1: CNC milling 2: bucket hat tool library 3: disc tool library 4: servo bucket hat tool library | CNC milling |

Parameter 001=1, CNC milling cutter
Parameter 001=2: Douli tool Library
Parameter 001=3: Disk tool magazine
Parameter 001=4: Servo bucket hat tool library
Note: Modifying this parameter requires a system power outage and restart

#### 1.6.8.1 Parameter 001=1, CNC milling cutter

➢ Parameters - [Tool Library Parameters]

| | | |
|---|---|---|
| 002 | Use of cutting tool library (0: prohibited 1: allowed) | prohibited |
| 003 | Is it forbidden to edit programs with program numbers 9000-9999? (0: No 1: Yes) | No |
| 010 | Whether to use a tool clamping device (0: No 1: Yes) | |
| 011 | Does the elastic tool button maintain output (0: not maintained 1: maintained) | |
| 012 | Did you check the tightness tool in place signal when the tool magazine is not in use (0: No 1: Yes) | |
| 013 | Tightening tool in place X0.3 high and low level selection (0: high 1: low) | |
| 014 | Loosening tool in place X0.0 high and low level selection (0: high 1: low) | |
| 015 | Delay of spindle tool release completion (unit: milliseconds) | |
| 016 | Delay of spindle tool clamping completion (unit: milliseconds) | |
| 017 | Spindle tool release detection delay (unit: milliseconds) | |
| 018 | Delay in spindle tool clamping detection (unit: milliseconds) | |

tool compensation - [common variable]

| | | |
|---|---|---|
| 500 | Total number of knives in the cutting magazine | |
| 501 | Does the spindle use directional function (1: Yes 0: No) | |
| 502 | Does the spindle have a tool function detected (1: Yes 0: No) | |
| 503 | The spindle does not use the directional function to delay the tool change time (unit: seconds) | |
| 504 | Return axis (0: Y-axis 1: X-axis) | |
| 505 | Return/retrieval speed | |

➢ Function Description

The initial tool number # 655 is required for the first use: one
Attention: The tool number coordinates are input into the variable as machine coordinates

Explanation of Return tool Variables
tool No.1:
Safe coordinates for returning knives in front of the tool magazine: X : #510
Y : #511
Z : #512
When the X-axis is used as the tool return axis, the X-axis loose tool coordinate is:# 513 (Y-axis tool return, Y-axis loose tool coordinate, Z-axis tool return in the same way)
tool No. 2:
Safe coordinates for returning knives in front of the tool magazine: X : #514
Y : #515
Z : #516
When the X-axis is used as the tool return axis, the X-axis loose tool coordinate is:# 517 (Y-axis tool return, Y-axis loose tool coordinate, Z-axis tool return in the same way)
tool No. 3:
Safe coordinates for returning knives in front of the tool magazine: X : #518
Y : #519
Z : #520
When the X-axis is used as the tool return axis, the X-axis loose tool coordinate is:# 521 (Y-axis tool return, Y-axis loose tool coordinate, Z-axis tool return in the same way)
tool No. 4:
Safe coordinates for returning knives in front of the tool magazine: X : #522
Y : #523
Z : #524
When the X-axis is used as the tool return axis, the X-axis loose tool coordinate is:# 525 (Y-axis tool return, Y-axis loose tool coordinate, Z-axis tool return in the same way)
tool No. 5:
Safe coordinates for returning knives in front of the tool magazine: X : #526
Y : #527
Z : #528
When the X-axis is used as the tool return axis, the X-axis loose tool coordinate is:# 529 (Y-axis tool return, Y-axis loose tool coordinate, Z-axis tool return in the same way)
tool No. 6:
Safe coordinates for returning knives in front of the tool magazine: X : #530
Y : #531
Z : #532
When the X-axis is used as the tool return axis, the X-axis loose tool coordinate is:# 533 (Y-axis tool return, Y-axis loose tool coordinate, Z-axis tool return in the same way)
tool No. 7:
Safe coordinates for returning knives in front of the tool magazine: X : #570

Y : #571
Z : #572
When the X-axis is used as the tool return axis, the X-axis loose tool coordinate is:# 573 (Y-axis tool return, Y-axis loose tool coordinate, Z-axis tool return in the same way)
tool No. 8:
Safe coordinates for returning knives in front of the tool magazine: X : #574
Y : #575
Z : #576
When the X-axis is used as the tool return axis, the X-axis loose tool coordinate is:# 577 (Y-axis tool return, Y-axis loose tool coordinate, Z-axis tool return in the same way)
tool No. 9:
Safe coordinates for returning knives in front of the tool magazine: X : #578
Y : #579
Z : #580
When the X-axis is used as the tool return axis, the X-axis loose tool coordinate is:# 581 (Y-axis tool return, Y-axis loose tool coordinate, Z-axis tool return in the same way)
tool No. 10:
Safe coordinates for returning knives in front of the tool magazine: X : #582
Y : #583
Z : #584
When the X-axis is used as the tool return axis, the X-axis loose tool coordinate is:# 585 (Y-axis tool return, Y-axis loose tool coordinate, Z-axis tool return in the same way)

**Explanation of tool Variables**

No.1 tool: Coordinate X above the tool magazine: # 550
Y : #551
tool coordinate Z:# 552
tool No. 2: Coordinate X above the tool magazine: # 553
Y : #554
tool coordinate Z:# 555
tool No. 3: Coordinate X above the tool magazine: # 556
Y : #557
tool coordinate Z:# 558
tool No. 4: Coordinate X above the tool magazine: # 559
Y : #560
tool coordinate Z:# 561
tool No. 5: Coordinate X above the tool magazine: # 562
Y : #563
tool coordinate Z:# 564
tool No. 6: Coordinate X above the tool magazine: # 565
Y : #566
tool coordinate Z:# 567
7th tool: Coordinate X above the tool magazine: # 590
Y : #591
tool coordinate Z:# 592

tool No. 8: Coordinate X above the tool magazine: # 593
Y : #594
tool coordinate Z:# 595
tool No. 9: Coordinate X above the tool magazine: # 596
Y : #597
tool coordinate Z:# 598
tool No. 10: Coordinate X above the tool magazine:# 599
Y : #600
tool coordinate Z: # 601

- **Example explanation (3 sets of tool banks): Use the X-axis as the tool return axis**

#500: Total number of knives in the 3000 row tool magazine
#501: Does the 1.000 spindle use orientation function (1: Yes 0: No)
#502: Does the spindle have a tool function detected for 0.000 (1: Yes 0: No)
#503: 2.000 Spindle not using directional function delayed tool change time (unit: seconds)
#504: 1.000 Return axis (0: Y-axis 1: X-axis)
#505: 1000.000 tool return/retrieval speed
#506-120000 Safety coordinate of X-axis in front of the tool magazine when returning the tool
#507: The safety coordinate of the Y-axis in front of the tool magazine when returning the tool at 0.000
#508:120.000 Safety coordinate of Z-axis above the tool magazine during tool retrieval
#510: 120.000 No.1 tool return X-axis coordinate
#511: -1000000 Y-axis coordinate of No.1 tool return
#512: 10.000 Z-axis coordinate of No.1 tool return
#513: 20.000 No.1 tool advancing tool compartment coordinates (Y or X, depending on the return axis)
#514: 120.000 No. 2 tool return X-axis coordinate
#515: 0.000 2nd tool return Y-axis coordinate
#516: 10.000 Z-axis coordinate of 2nd tool return
#517: 20.000 No.2 tool advancing tool compartment coordinates (Y or X, depending on the return axis)
#518: 120.000 No. 3 tool return X-axis coordinate
#519: 100000 Y-axis coordinate of 3rd tool return
#520: 10.000 Z-axis coordinate of No.3 tool return
#521: 20.000 3rd tool advance tool compartment coordinates (Y or X, depending on the return axis)
#550: 20.000 X axis coordinate of the 1st tool taking tool
#551: -1000000 Y-axis coordinate of tool # 1
#552: 10.000 Z-axis coordinate of No.1 tool puller
#553: 20.000 X-axis coordinate of the 2nd tool taking tool
#554: 0.000 Y-axis coordinate of tool # 2

#555: 10.000 Z-axis coordinate of No.2 tool puller
#556: 20.000, taking the X-axis coordinate of tool No.3
#557: 100.000 Y-axis coordinate of tool # 3
#558: 10.000 Z-axis coordinate of tool No.3

### 1.6.8.2 Parameter 001=2, Douli tool Library

| | | |
|---|---|---|
| 002 | Use of the bamboo hat tool library (0: prohibited 1: allowed) | allowed |
| 003 | Is it forbidden to edit programs with program numbers 9000-9999? (0: No 1: Yes) | No |
| 004 | Z-axis 2nd reference point (bucket hat tool magazine, Z-axis safe height position) | 0.000 |
| 005 | Z-axis 3rd reference point (bucket hat tool magazine or disc tool magazine, Z-axis tool change position) | 0.000 |
| 006 | The delay filtering of the spindle orientation signal is turned off (unit: milliseconds) | 50 |
| 007 | Douli tool Magazine Rotation Counter | 16 |
| 008 | Maximum tool capacity of the tool magazine | 16 |
| 009 | Has the tool library entered debugging mode? (0: No 1: Yes) | No |
| 010 | Whether to use a tool clamping device (0: No 1: Yes) | Yes |
| 011 | Does the elastic tool button maintain output (0: not maintained 1: maintained) | not maintained |
| 012 | Did you check the tightness tool in place signal when the tool magazine is not in use (0: No 1: Yes) | No |
| 013 | Tightening tool in place X0.3 high and low level selection (0: high 1: low) | high |
| 014 | Loosening tool in place X0.0 high and low level selection (0: high 1: low) | high |
| 015 | Delay of spindle tool release completion (unit: milliseconds) | 0 |
| 016 | Delay of spindle tool clamping completion (unit: milliseconds) | 0 |
| 017 | Spindle tool release detection delay (unit: milliseconds) | 8000 |
| 018 | Delay in spindle tool clamping detection (unit: milliseconds) | 8000 |
| 021 | Is there a zero return switch in the tool magazine? Yes/No (0: 1: Yes) | yes |
| 022 | tool magazine No.1 tool zero point X2.0 high and low level selection (0: high 1: low) | high |
| 023 | Tool position counter X2.1 high and low level selection (0: high 1: low) | high |
| 024 | Delay detection during tool magazine rotation (unit: milliseconds) | 5000 |
| 025 | Tool library counting delay detection (unit: milliseconds) | 50 |
| 026 | Zero return delay completion of tool magazine (unit: milliseconds) | 50 |

| | | |
|---|---|---|
| 030 | Does the tool magazine use bidirectional valve output for forward and backward movement (0: No 1: Yes) | No |
| 031 | Is the output signal canceled after the tool magazine retreat solenoid valve is in place (0: No 1: Yes) | Yes |
| 032 | tool magazine advances to position X2.2 high and low level selection (0: high 1: low) | high |
| 033 | tool magazine retreats to position X2.3 high and low level selection (0: high 1: low) | high |
| 034 | Delay in tool magazine feed completion (unit: milliseconds) | 50 |
| 035 | Delay in completing the return of knives from the tool magazine (unit: milliseconds) | 50 |
| 036 | Tool library feed delay detection (unit: milliseconds) | 15000 |
| 037 | Tool library return delay detection (unit: milliseconds) | 15000 |

- Parameter debugging instructions

① **Manual tool magazine forward/reverse control operation:**

Long press the [Handwheel] button. When the handwheel mode light is flashing, press the [Tool magazine forward rotation] button to rotate the tool magazine clockwise (forward rotation). Y2.2 output, turn off Y2.3; Press the [Tool Magazine Reverse] key to rotate the tool magazine counterclockwise (reverse). Y2.3 output, turn off Y2.2

X2.1 library rotation counter input signal. When the magazine rotates one tool position, X2.1=1 (i.e. the system+24V is connected to X2.1) indicates that the tool position signal is correct. If the high and low levels of the magazine rotation counter input signal X2.1 are incorrect, please set [magazine parameters]: select the high and low levels of the magazine counter X2.1;

Determination of the rotation direction of the tool magazine: Press the [Tool Magazine Forward] button, and the tool magazine should rotate clockwise. Press the [Tool Magazine Reverse] button, and the tool magazine should rotate counterclockwise. If the opposite phenomenon occurs, the tool magazine count will be disordered, causing tool exchange errors. Adjusting the phase sequence of the tool magazine rotating motor can solve this problem.

② **tool magazine push out/return control (tool magazine in/out): M23/M24**

Set the forward and backward parameters of the tool magazine, and modify the tool magazine parameter P006 for dual solenoid valves when moving forward and backward;

When controlling the dual solenoid valve of the tool magazine, whether to turn off the reverse output after the reverse is in place, and modify the tool magazine parameter P007;

Long press the [Handwheel] key, when the handwheel mode light is flashing

Press the [Tool Magazine Front (Inverted)] key to advance the tool magazine. Y2.0 output, close Y2.1

Press the [Back] key to retract the tool magazine. Y2.1 output, turn off Y2.0

Or execute the instruction: M23 system outputs Y2.0, detects X2.2 (i.e. system+24V is connected to X2.2) in place, then the tool magazine is pushed out.

Instruction: M24 system closes Y2.1, detects X2.3 (i.e. system+24V and X2.3 are connected) in place, then the tool magazine return is complete. If the system does not detect the tool magazine return in place signal X2.3=1, then the Z manual axis cannot be moved and the handwheel can be moved. Or: [Tool Library Parameters] Tool Library Yes/No: Enter debugging mode, set to: Yes

③ **Spindle clamping and releasing control: M16/M17**

Instruction: M16 system outputs Y0.2, the spindle is in a loose state, and X1.1=1 is detected (i.e. the system+24V is connected to X1.1), indicating that the loose state is complete;

Instruction: M17 system closes Y0.2, the spindle is in tight cutting, and X0.3=1 is detected (i.e. system+24V is connected to X0.3), indicating that the tight cutting is complete.

④ **Main axis orientation M19, orientation cancellation: M18 or M05**

Directional control output: Y1.0

Targeted completion detection: X0.7

Explanation: When the instruction M19 is executed, the system outputs Y1.0 (which can control the relay or be designed according to the actual control principle), and the relay controls the orientation of the spindle servo. After the spindle servo orientation is completed, a orientation completion signal will be output (this completion signal controls the X0.7 of the system to be connected to the+24V of the system according to the actual control principle, that is, the orientation is completed).

Under [Handwheel or Manual Mode], press the [Spindle Alignment Stop] button to output Y1.0 to control spindle orientation. When the system detects the positioning completion signal from the spindle servo feedback (i.e. system+24V and X0.7 are connected), the [Spindle Alignment Stop] button indicator light on the panel will light up.

⑤ **Set the tool magazine to zero**

When the tool magazine has a return to zero switch, [Tool Magazine Parameters] Does the tool magazine have a return to zero switch? Yes

Tool magazine zeroing control (finding tool control for tool # 1): M20 (Before debugging the tool magazine, it is necessary to ensure that it can return to zero normally)

Command M20 tool library to rotate forward (Y2.2 output) to find the first tool. When X2.0=1 (i.e. system+24V is connected to X2.0), it returns to zero.

When the tool magazine does not have a return to zero switch, [Tool Magazine Parameters] Does the tool magazine have a return to zero switch: No

Long press the [Handwheel] button. When the handwheel mode light is flashing, press the [Tool magazine forward rotation] button to rotate the tool magazine clockwise (forward rotation). Output Y2.2. When the tool number on the disc is found to be 1, execute M20 and the disc returns to zero successfully.

When there are disorderly knives in the tool magazine, M20 can be executed to reset the tool magazine to zero to solve the problem

⑥ **Display the tool change program when the tool library is running**

When trying to run the tool library, the program for changing tools must be displayed in automatic mode with a level 2 password. If a single segment is required, the tool library debugging function K30.7 must be enabled to allow such operation.

⑦ **Ladder diagram tool number display**, Convenient diagnostic tool number signal D80 is assigned by the T command, D92 is the spindle tool number G37, C100 is the counter G39, D88 is the tool head position used for executing M25 rotary cutting; D86 is the position of the cutterhead. In manual mode, the position of the cutterhead recorded when rotating the magazine is used in M26.

➢ **Tool changing program**

```
O9102;
G80
G65H01P#100Q0;//#100:=0
G65H01P#101Q1;//#101:=1
G65H81P40Q#1000R#100;//IF #1000=#100 GOTO P40
G65H01P#103Q#4002;//#103:=#4002
G65H01P#104Q#4003;//#104:=#4003
G65H01P#1107Q#101;//#1107:=#101
//#1107 corresponds to F54.7, starting the tool change in the ladder diagram
M05;
G4P200;//#1001 corresponds to G54.1
G65H81P30Q#1001R#101;//Programming tool number=spindle tool number jumps to P30
M19;//Main axis orientation
G91G30P3Z0;//Return the tool point
N23M26;//Return tool number calculation
N25;
G65H81P25Q#1002R#100;//The current tool number is consistent with the spindle tool number
M23;//tool library launched
G65H01P#1106Q#101;//#1106 corresponds to F54.6,
M16;//TOOL UNCLAMP
G04P300;
G30P2Z0;//tool seeking point
M25;//tool taking operation
G30P3Z0;//tool point
M17;//Tool clamping
G65H01P#1106Q#100;//#1106 corresponds to F54.6,
M24;//tool magazine return
N30;
M05;
G65H01P#1106Q#100;
G65H01P#1107Q#100;
G#103;
G#104;
```

```
N40;
M99;
```

### 1.6.8.3 Parameter 001=3, disk manipulator tool library

| | | |
|---|---|---|
| 002 | Use of disc tool library (0: prohibited 1: allowed) | prohibited |
| 003 | Is it forbidden to edit programs with program numbers 9000-9999? (0: No 1: Yes) | No |
| 004 | Z-axis 2nd reference point (bucket hat tool magazine, Z-axis safe height position) | 0.000 |
| 005 | Z-axis 3rd reference point (bucket hat tool magazine or disc tool magazine, Z-axis tool change position) | 0.00 |
| 006 | The delay filtering of the spindle orientation signal is turned off (unit: milliseconds) | 50 |
| 007 | Douli tool Magazine Rotation Counter | 16 |
| 008 | Maximum tool capacity of the tool magazine | 16 |
| 009 | Has the tool library entered debugging mode? (0: No 1: Yes) | No |
| 010 | Whether to use a tool clamping device (0: No 1: Yes) | No |
| 011 | Does the elastic tool button maintain output (0: not maintained 1: maintained) | not maintained |
| 012 | Did you check the tightness tool in place signal when the tool magazine is not in use (0: No 1: Yes) | No |
| 013 | Tightening tool in place X0.3 high and low level selection (0: high 1: low) | high |
| 014 | Loosening tool in place X0.0 high and low level selection (0: high 1: low) | high |
| 015 | Delay of spindle tool release completion (unit: milliseconds) | 50 |
| 016 | Delay of spindle tool clamping completion (unit: milliseconds) | 50 |
| 017 | Spindle tool release detection delay (unit: milliseconds) | 8000 |
| 018 | Delay in spindle tool clamping detection (unit: milliseconds) | 8000 |
| 021 | Is there a zero return switch in the tool magazine? Yes/No (0: 1: Yes) | Yes |
| 022 | tool magazine No.1 tool zero point X2.0 high and low level selection (0: high 1: low) | high |
| 023 | Tool position counter X2.1 high and low level selection (0: high 1: low) | high |
| 024 | Delay detection during tool magazine rotation (unit: milliseconds) | 5000 |
| 025 | Tool library counting delay detection (unit: milliseconds) | 50 |
| 026 | Zero return delay completion of tool magazine (unit: milliseconds) | 50 |
| 029 | Does the tool holder automatically fall down after the tool magazine rotates into place (0: No 1: Yes) | Yes |
| 030 | Does the tool magazine use bidirectional valve output for forward and backward movement (0: No 1: Yes) | No |
| 031 | Is the output signal canceled after the tool magazine retreat solenoid valve is in place (0: No 1: Yes) | No |
| 032 | tool magazine advances to position X2.2 high and low level selection (0: high 1: low) | high |

| | | |
|---|---|---|
| 033 | tool magazine retreats to position X2.3 high and low level selection (0: high 1: low) | high |
| 034 | Delay in tool magazine feed completion (unit: milliseconds) | 50 |
| 035 | Delay in completing the return of knives from the tool magazine (unit: milliseconds) | 50 |
| 036 | Tool library feed delay detection (unit: milliseconds) | 15000 |
| 037 | Tool library return delay detection (unit: milliseconds) | 15000 |
| 041 | tool arm buckle tool X2.4 high and low level selection (0: high 1: low) | high |
| 042 | tool arm zero point X2.5 high and low level selection (0: high 1: low) | high |
| 043 | Blade arm brake X2.6 high and low level selection (0: high 1: low) | high |

➢ Parameter debugging instructions

① Manual tool magazine forward/reverse control operation

Long press the [Handwheel] button. When the handwheel mode light is flashing, press the [Tool magazine forward rotation] button to rotate the tool magazine clockwise (forward rotation) Y2.2 output and turn off Y2.3; Press the [Tool Magazine Reverse] key to rotate the tool magazine counterclockwise (reverse) Y2.3 output and turn off Y2.2;

X2.1 Tool magazine rotation counter input signal. When the tool magazine rotates one tool position, X2.1=1 (i.e. the system+24V is connected to X2.1) indicates that the tool position signal is correct. If the high and low levels of the tool magazine rotation counter input signal X2.1 are incorrect, please set [Tool magazine parameters]: Tool magazine counter X2.1 high and low level selection.

Determination of the rotation direction of the tool magazine: Press the [Tool Magazine Forward] button, and the tool magazine should rotate clockwise. Press the [Tool Magazine Reverse] button, and the tool magazine should rotate counterclockwise. If the opposite phenomenon occurs, the tool magazine count will be disordered, causing tool exchange errors. Adjusting the phase sequence of the tool magazine rotating motor can solve this problem.

② tool magazine push out/return control (tool magazine in/out): M23/M24

Set the forward and backward parameters of the tool magazine. When the forward and backward are dual solenoid valves, modify the tool magazine parameter P008;

When controlling the dual solenoid valve of the tool magazine, whether to turn off the reverse output after the reverse is in place, and modify the tool magazine parameter P009;

Long press the [Handwheel] key, when the handwheel mode light is flashing

Press the [Tool Magazine Front (Inverted)] key to advance the tool magazine. Y2.0 output, close Y2.1

Press the [Back] key to retract the tool magazine. Y2.1 output, turn off Y2.0

Execute instruction: M23 system outputs Y2.0, detects X2.2 (i.e. system+24V is connected to X2.2) in place, then the tool magazine is pushed out.

Instruction: M24 system closes Y2.1, detects X2.3 (i.e. system+24V and X2.3 are connected) in place, then the tool magazine return is complete. If the system does not detect the tool magazine return in place signal X2.3=1, then the Z manual axis cannot be moved and the handwheel can be moved.

③ **Spindle Puller Control: M16/M17**

Instruction: M16 system outputs Y0.2, the spindle is in a loose state, and X1.1=1 is detected (i.e. the system+24V is connected to X1.1), indicating that the loose state is complete;

Instruction: M17 system closes Y0.2, the spindle is in tight cutting, and X0.3=1 is detected (i.e. system+24V is connected to X0.3), indicating that the tight cutting is complete.

④ **Main axis orientation M19, orientation cancellation: M18 or M05**

Directional control output: Y1.0

Targeted completion detection: X0.7

Explanation: When the instruction M19 is executed, the system outputs Y1.0 (which can control the relay or be designed according to the actual control principle), and the relay controls the orientation of the spindle servo. After the spindle servo orientation is completed, a orientation completion signal will be output (this completion signal controls the X0.7 of the system to be connected to the+24V of the system according to the actual control principle, that is, the orientation is completed).

Under [Handwheel or Manual Mode], press the [Spindle Alignment Stop] button to output Y1.0 to control spindle orientation. When the system detects the positioning completion signal from the spindle servo feedback (i.e. system+24V and X0.7 are connected), the [Spindle Alignment Stop] button indicator light on the attached panel will light up.

⑤ **tool magazine**[Tool library parameters]: Maximum tool capacity of the tool library: 24 (this parameter is equal to: D100)

⑥ **Tool changing point in the tool magazine**[Tool library parameters], the third reference point on the Z-axis (Z-axis position of the tool arm buckle) is set according to the actual machine coordinates;

⑦ **Related M function codes and debugging steps:**

G91 G30 P3 Z0; // Return to the third reference point

M19;// Spindle orientation X0.7 completed

M23;// Blade sleeve vertical Y2.0 X2.2 completed

M45;// tool arm buckle tool Y2.4 X2.4 completed

M16;// Spindle loosening tool Y0.2 X1.1 completed

M46;// tool arm exchange Y2.4 X2.6 completed

M17;// tool X0.3 completed

M47;// tool arm return Y2.4 X2.5 completed

M24;// tool cover flipped up Y2.1 X2.1 completed

M18 or M05// Targeted cancellation

⑧ **M45 arm buckle tool action**

When PLC parameter K42.4=0 (the tool arm buckle tool in place signal is low and effective)

Process: When the tool is in place and the signal X2.4=1, execute M45 to output Y2.4 and rotate the tool arm. When the tool is in place and X2.4=0, the tool is fully engaged.

⑨ **M46 blade arm exchange action**

When PLC parameter K42.6=0 (tool arm exchange in place signal low level is valid)

Blade arm brake signal X2.6=1, execute M46 output Y2.4 blade arm rotation buckle, when the buckle is in place, X2.6=0 buckle is completed.

⑩ **M47 arm zeroing action**

After the M46 exchange is completed, the system executes the M17 tightening tool. Only when X0.3=1 is in place can the M47 tool arm return to zero be executed; When PLC parameter K42.5=0 (the zero point signal of the tool arm is valid at low level)

Process: The zero point signal of the tool arm is X2.5=1. Execute M47 to output Y2.4 and rotate the tool arm back to zero. When the return to zero is in place, X2.4=0 completes the return to zero.

**Set the zero return parameter for the tool magazine:**When the tool magazine has a return to zero switch, [Tool Magazine Parameters] Does the tool magazine have a return to zero switch: Yes; Tool magazine zeroing control (finding tool 1 control): M20 (before debugging the tool magazine, it is necessary to ensure that the tool magazine can return to zero normally), instruct M20 tool magazine to rotate forward (Y2.2 output) to find tool 1, and when X2.0=1 (i.e. the system+24V is connected to X2.0), the zeroing is completed.

When the tool magazine does not have a return to zero switch, [Tool Magazine Parameter] Does the tool magazine have a return to zero switch: No (this parameter is equal to K30.1)

Long press the [Handwheel] button. When the handwheel mode light is flashing, press the [Tool magazine forward rotation] button to rotate the tool magazine clockwise (forward rotation). Y2.2 output. When the tool number on the disc is found to be 1, execute M20 and the disc returns to zero successfully

When there are disorderly knives in the tool magazine, M20 can be executed to reset the tool magazine to zero to solve the problem

⑪ **Automatic pre selection tool reversing function:**[Tool library parameters] After the tool library rotates to the correct position, does the tool sleeve automatically fall down? After modifying this parameter, after executing the T code, the tool library rotates to the correct position and the tool sleeve automatically falls down, which facilitates tool changing efficiency

➢ **Tool changing program**

```
G80;
G65H01P#100Q0;//#100:=0
G65H01P#101Q1;//#101:=1
G65H81P40Q#1000R#100;//IF #1000=#100 GOTO P40
G65H01P#103Q#4002;
G65H01P#104Q#4003;
M05;
G4P200;
G65H81P30Q#1001R#101;//#1001 对应是 G54.1
G65H01P#1107Q#101;
//#1107 corresponds to F54.7 at the beginning of tool change in the ladder diagram
M19;//Main axis orientation
G91G30P3Z0;//Reference point for tool change
N25;
G65H81P25Q#1003R#100;//Current tool set=target tool number
M23;//tool cover Y2.0
G04P200;
M45;//tool arm buckle tool Y2.4
G04P200;
M16;//Spindle loosening tool Y0.2
G04P500;
M46;//tool arm exchange
G04P200;
M17;//Spindle clamping
G04P300;
M47;//tool arm return
M24;//Flip the tool cover up Y2.1
M05;//
G65H01P#1107Q#100;
N30;
G#104;
G#103;
N40;
M99;
```

### 1.6.8.4 Parameter 001=4, servo bucket hat tool library

① Related parameter settings

[Tool library parameters] Set the ladder diagram number (1: CNC milling 2: bucket hat 3: disc 4: A-axis bucket hat) to 4

[Tool Library Parameters] The allowed/prohibited use of the servo bucket hat tool library is set to: allowed

[Rotation axis parameter] 4TH axis is (0: Linear axis 1: Rotation axis), set as: Rotation axis

[Feed axis parameters] Control the number of axes, set to 4

[Feed axis parameters] 4TH axis instruction multiplication factor, set according to actual conditions (ensure that the tool magazine rotates once and the A-axis coordinates are 0~360)

[Feed axis parameters] 4TH axis instruction frequency division coefficient, set according to actual conditions (ensure that the tool magazine rotates once and the A-axis coordinates are 0~360)

② The following tool library parameters can be set in the [Common Variables] section of the [Tool Supplement] interface, for example: # 600 represents variable number 600.

Total number of knives in the tool magazine # 600 (can only be set to: 8, 16, 24)

Coordinate of tool magazine 1: # 610. After the A-axis returns to zero, manually move the A-axis to determine the current position of tool 1 and copy the machine coordinates of tool 1 to variable # 610.

Note: Only manually set the machine coordinates for tool number 1, while the coordinates for other tool numbers are automatically generated by the system and do not require manual setting.

Does the spindle use directional function: # 601 (1: Yes, 0: No)

Spindle directional output: Y1.0 (instruction M19 spindle orientation, M18 cancel orientation)

Spindle orientation completed: X0.7

③ tool magazine push out/return control (tool magazine in/out): M23/M24

Set the forward and backward parameters of the tool magazine. When the forward and backward are dual solenoid valves, modify the tool magazine parameter P008;

When controlling the dual solenoid valve of the tool magazine, whether to turn off the reverse output after the reverse is in place, and modify the tool magazine parameter P009;

Long press the [Handwheel] key, when the handwheel mode light is flashing

Press the [Tool Magazine Front (Inverted)] key to advance the tool magazine. Y2.0 output, close Y2.1

Press the [Back] key to retract the tool magazine. Y2.1 output, turn off Y2.0

Or execute the instruction: M23 system outputs Y2.0, detects X2.2 (i.e. system+24V is connected to X2.2) in place, then the tool magazine is pushed out.

Instruction: M24 system closes Y2.1, detects X2.3 (i.e. system+24V and X2.3 are connected) in place, then the tool magazine return is complete. If the system does not detect the tool magazine return in place signal X2.3=1, then the Z manual axis cannot be moved and the handwheel can be moved.

④ Spindle Puller Control: M16/M17

Instruction: M16 system outputs Y0.2, the spindle is in a loose state, and X1.1=1 is detected (i.e. the system+24V is connected to X1.1), indicating that the loose state is complete;

Instruction: M17 system closes Y0.2, the spindle is in tight cutting, and X0.3=1 is detected (i.e. system+24V is connected to X0.3), indicating that the tight cutting is complete.

⑤ Does the spindle use directional function: # 601 (1: Yes, 0: No)

Main axis orientation M19, orientation cancellation: M18 or M05

Directional control output: Y1.0

Targeted completion detection: X0.7

Explanation: When the instruction M19 is executed, the system outputs Y1.0 (which can control the relay or be designed according to the actual control principle), and the relay controls the orientation of the spindle servo. After the spindle servo orientation is completed, a orientation completion signal will be output (this completion signal controls the X0.7 of the system to be connected to the+24V of the system according to the actual control principle, that is, the orientation is completed).

Under [Handwheel or Manual Mode], press the [Spindle Alignment Stop] button to output Y1.0 to control spindle orientation. When the system detects the positioning completion signal from the spindle servo feedback (i.e. system+24V and X0.7 are connected), the [Spindle Alignment Stop] button indicator light on the attached panel will light up.

⑥ [Tool library parameters], Z-axis second reference point (bucket hat tool library, Z-axis safe height position).

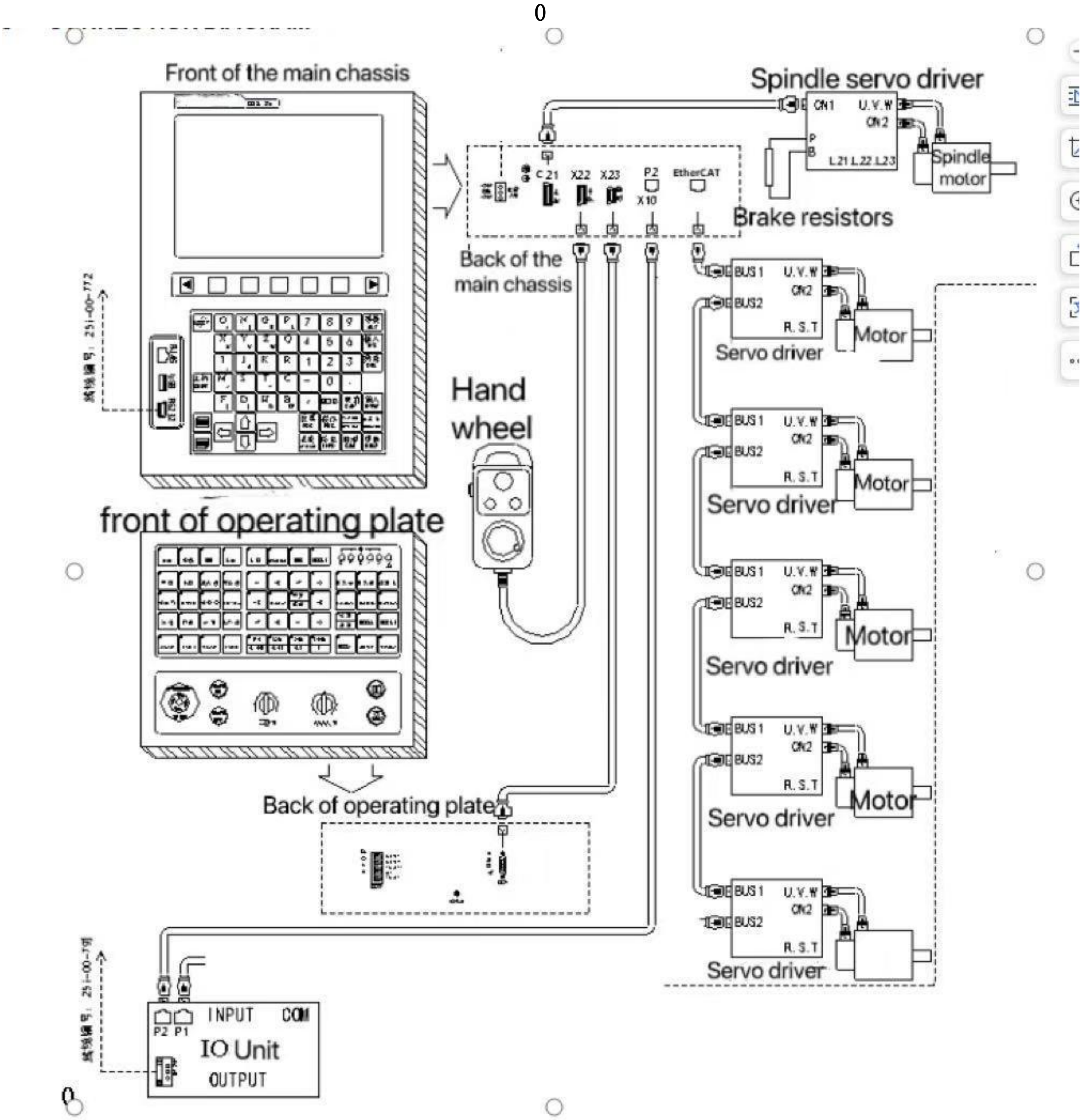
⑦ [Tool library parameters], Z-axis 3rd reference point (bucket hat tool library, Z-axis downward pull tool position)

⑧ When the tool magazine is allowed to be used, the system will warn immediately upon power on: the servo shaft has not returned to the mechanical zero point! When each axis returns to zero, the warning is automatically cleared and the system is in normal condition!

Final step: Ensure that the first tool change is empty (i.e. no handle is placed) and set D50=1 in the [PLC parameter] for the first tool change:

## 1.6.9 EtherCAT bus wiring instructions

- **CONNECTION DIAGRAM**




Lathe system:

System connection driver timing: The system network cable is connected to the X-axis driver, the X-axis driver is connected to the Z-axis driver, the Z-axis driver is connected to the Y-axis driver, the Y-axis driver is connected to the A-axis driver, and the A-axis driver is connected to the B-axis driver.

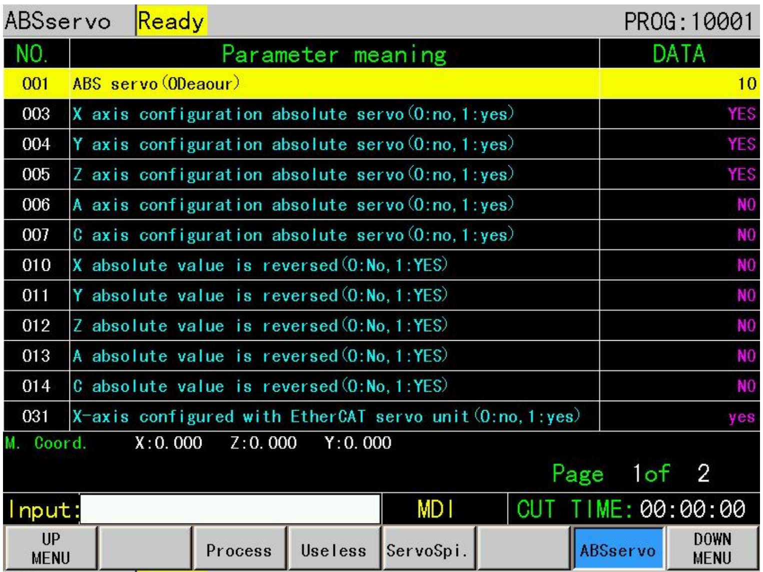
Milling machine system:

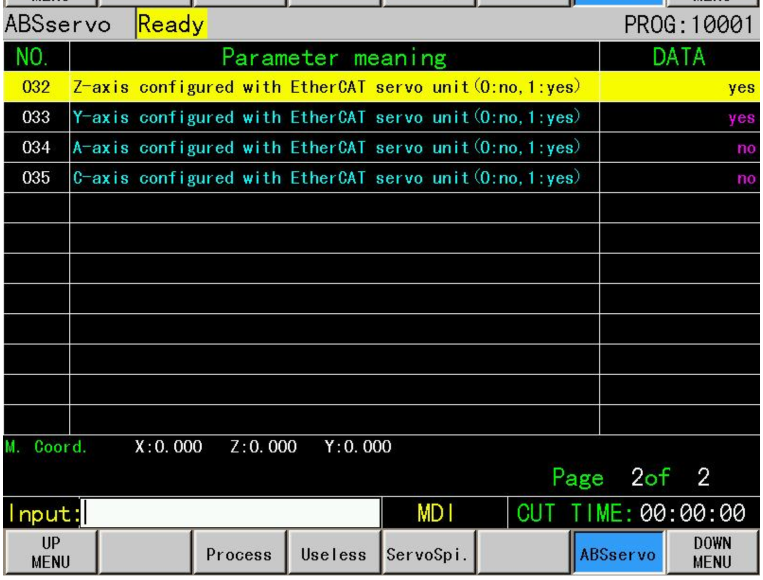
System connection driver timing: The system network cable is connected to the X-axis driver, the X-axis driver is connected to the Y-axis driver, the Y-axis driver is connected to the Z-axis driver, the Z-axis driver is connected to the A-axis driver, and the A-axis driver is connected to the B-axis driver.

Note: The drive does not require a station number to be set. The system automatically identifies the axis number of the drive based on its sequence.

- **Parameter settings**

System Parameters - [ethercat Parameters]




- **Parameter Description**

1. Open the corresponding EtherCAT parameter for the axis,After parameter modification, power off and restart will take effect.

2. Whether to display the absolute value zero point setting is to determine whether the bus remembers the coordinates of the motor, such as parameter selection

When the system does not read the absolute position of the motor, it is equivalent to using incremental values. Change the parameter to yes, the system reads coordinates from power on and resets coordinates after emergency stop.

3. The meaning of reversing the direction of absolute value: The encoder rotation of the absolute value motor has direction counting. Since the system cannot read the direction of the encoder, it needs to rely on parameters to adjust the encoder direction read by the system to be consistent with the actual encoder direction. If the actual setting direction is reversed, it will cause the system to read the motor value differently from the actual value each time, and the most obvious difference is when the power is turned off and then turned on.

- **Gear ratio calculation**

17 bit encoder line count: 131072 lines (number of pulses per motor rotation)

23 bit encoder lines: 8388608 lines (number of pulses per motor rotation)

The servo motor travels one revolution with 1000 pulses emitted by the system, for example, the screw pitch is 10MM

Counting gear ratio:

**Linear axis**

17 bit 131072: 1000X10(Screw pitch)=131072: 10000=8192: 625

23 bit 8388608: 1000X10(Screw pitch)=8388608: 10000=524288: 625

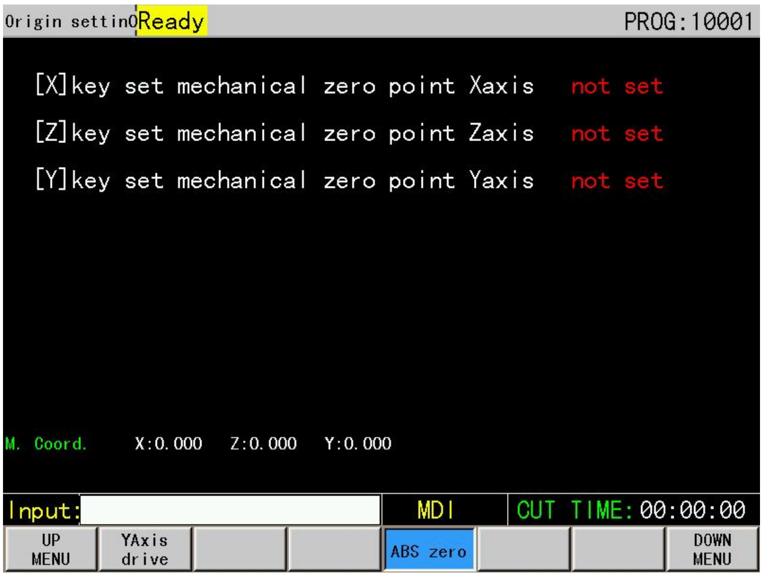
**rotation axis**

17 bit 131072: 1000X360(One rotation of the axis is 360 degrees)=131072: 360000=2048: 5625

23 bit 8388608: 1000X360(One rotation of the axis is 360 degrees)=8388608: 360000=131072: 5625

The calculated value is directly filled into the system gear ratio parameter

- **Adjust the direction of movement of the corresponding axis**
  1. 1. Execute the movement direction and coordinate values of G00 or G01. If the direction is incorrect, you can modify the axis movement direction to take the opposite direction
  2. 2. Adjust the movement direction manually by pressing the button to ensure that the movement direction matches the direction set by the machine tool
  3. Simultaneously adjust the direction of the handwheel
- **Set the zero point of the machine coordinate system**
  1. 1. The prerequisite for setting the zero point of the machine tool coordinate system, the direction of movement of each axis, and the gear ratio of each axis must be set before operation
  2. Parameter - [Absolute Zero] Interface




In MDI mode, press the corresponding letter key for the axis, such as for the X-axis. Pressing the X letter key will bring up a dialog box. Choose 'Yes' to set the zero point.

Note:After setting up, move each axis to approximately+50 position, power off and restart. After powering on, observe the change in machine coordinates. If the coordinates change to -50, reverse the direction of the absolute value encoder, and then power off and power on again. The coordinates will change to+50, indicating successful setting.

When the bus communication cannot be connected, the program cannot be started