I5T3 I5T5 I5T5E

CNC Controller for Lathe

User Manual V1.22

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 lathe!

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 lathe.

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.

VOLUME I Programming instructions

Chapter 1 Overview....9
1.1 Product introduction....9
1.2 technical specifications....9
1.3 G code table....12
Chapter 2 Fundamentals of Programming....13
2.1 control axis....13
2.2 Axis name....13
2.3 Axis Display....14
2.4 Machine zero and machine coordinates....14
2.5 Workpiece Coordinate System....14
2.6 Absolute value programming and incremental programming....15
2.7 Programming for diameter and radius methods....16
2.8 Modal, non modal, and initial state....17
Chapter 3 Composition of Part Program....18
3.1 Program composition....18
3.1.1 program name....18
3.1.2 code word....19
3.2 General structure of the program....20
3.2.1 Subprogram writing....22
3.2.2 Calling subprograms....22
3.2.3 End of program....24
Chapter 4 Preparation function G COMMANDS....25
4.1 Types of ready function G COMMANDS....25
4.2 Simple G COMMANDS....27
4.2.1 Rapid Traverse Movement G00....27
4.2.2 Linear Interpolation G01....28
4.2.3 Circular Interpolation G02, G03....29
4.2.4 Dwell G04....32
4.2.5 Chamfering Function....33
4.2.6 Workpiece Coordinate System G54～G59....38
4.2.7 Workpiece Coordinate System G50....39
4.2.8 Skip Interpolation G31....41
4.2.9 Machine 1st Reference Point G28....43
4.2.10 Machine 2nd, 3rd, 4th Reference Point G30....44
4.2.11 Axial Cutting Cycle G90....45
4.2.12 Radial Cutting Cycle G94....48
4.2.13 Axial Roughing Cycle G71....51
4.2.14 Radial Roughing Cycle G72....57
4.2.15 Closed Cutting Cycle G73....65
4.2.16 Finishing Cycle G70....69
4.2.17 Axial Grooving Multiple Cycle G74....70
4.2.18 Radial Grooving Multiple Cycle G75....73
4.2.19Thread Cutting with Constant Lead G32....77
4.2.20 Thread Cutting with Variable Lead G34....80

4.2.21 Z Thread Cutting G33.. 81
4.2.22 Thread Cutting Cycle G92.......................................82
4.2.23 Multiple Thread Cutting Cycle G76...............................86
4.2.24 Constant Surface Speed Control G96, Constant Rotational Speed Control G97.90
4.2.25 Feedrate per Minute G98, Feedrate per Rev G99...................90
4.2.26 Drilling Cycle G83..92
4.3 TOOL NOSE RADIUS COMPENSATION (G41, G42).................................93
4.3.1 Overview...93
4.3.2 Imaginary Tool Nose Direction....................................94
4.3.3 Compensation Value Setting.......................................93
4.3.4 Command Format...100
4.3.5 Compensation Direction..100
4.3.6 Notes...102
4.3.7 Application...103
4.3.8 Tool Nose Radius Compensation Offset Path.......................104
4.3.9 Tool Traversing when Starting Tool 105
4.3.10 Tool Traversing when Starting Tool 106
4.3.11 Tool Traversing in Offset Canceling Mode.......................109
4.3.12 Tool Interference Check..111
4.3.13 Commands for Canceling Compensation Vector Temporarily......111
4.3.14 exceptional case...113
4.4 Macro Commands...114
4.4.1 Macro Variables 114
4.4.2 Types of variables..115
4.4.3 Local variable..116
4.4.4.Notes on user macro program ontology...........................117
4.4.5 Non modal call G65..118
4.4.6 User Macro Program Function A..................................119
4.4.7 User Macro Program Functions B.................................123
Chapter 5 M COMMAND...128
5.1 M (Miscellaneous Function)...129
5.1.1 Spindle CW, CCW Control M03, M04...............................129
5.1.2 Spindle Stop Control M05..129
5.1.3 Cooling Control M08, M09..129
5.1.4 Tailstock Control M10, M11.......................................130
5.1.5 Chuck Control M12, M13...130
5.1.6 Spindle Position/Speed Control Switch M14, M15..................130
5.1.7 Spindle directional start and stop M19.M18......................130
5.1.8 Spindle Clamped/Released M20, M21...............................130
5.1.9 Rigid tapping M29..130
5.1.10 Lubricating Control M32, M33...................................130
5.1.11 Chip conveyor on/off M35,M36....................................130
5.1.12 Spindle Automatic Gear Change M41, M42, M43, M44...........130
5.1.13 Automatic feeding forward output and shutdown M70，M71..........131
5.1.14 Custom output, close M80, M81...................................131

5.1.15 Waiting for signal input M90....131
5.2 M (Miscellaneous Function)....131
5.2.1 End of Program run M30 M02....132
5.2.2 Program Stop M00....132
5.2.3 Program Optional Stop M01....132
5.2.4 Subprogram Call M98....132
5.2.5 Return From Subprogram M99....132
Chapter 6 Spindle Function....134
6.1 Spindle Speed Analog Voltage Control....134
6.2 Spindle Speed Series Control....134
6.3 Constant Surface Speed Control G96, Constant Rotational Speed Control G97.97....135
Chapter 7 Feed function F code....139
7.1 Quick movement....139
7.2 cut speed....139
7.2.1 Feed per revolution G99....140
7.3 Feed rate multiplier button....140
7.4 Automatic acceleration and deceleration....140
7.5 Acceleration and deceleration processing at the corner of the program section....141
Chapter 8 Tool Function....142
8.1 Tool Control....142

VOLUME II OPERATING AND CONNECTION

Chapter1 Operation Panel....146
1.1 Panel division....146
1.2 Panel Function Description....146
Chapter 2 Interface Display and Data Modification and Setting 152
2.1 Position display....152
2.2 program display....156
2.3 Tool compensation display, modification, and settings....159
2.4 Parameter switch, modification and setting....162
2.5 parameter display....167
2.6 Diagnostic display....168
2.7 Ladder diagram display and parameter modification....172
Chapter 3 debugging and connection....177
3.1 X. Connection of Y-axis and Z-axis and A-axis interfaces....177
3.2 Connection of spindle encoder interface....181
3.3 Connection of MPG interface....184
3.4 communication interface....185
3.5 I/O interface definition....186
3.6 MPG Box connection....192
3.7 CNC Installation dimensions....193
Chapter 4 I/O Function and Connection....194
4.1.1 Limit and emergency stop 194
4.1.2 External cycle start and feed hold....195
4.1.3 Three position start pause switch....195

4.1.4 Control of spindle counterclockwise and clockwise rotation..............196
4.1.5 SPINDLE JOG..197
4.1.6 Spindle orientation and position switching function..................197
4.1.7 Chuck control...199
4.1.8 Tail seat control...201
4.1.9 Lubrication control..202
4.2.0 Tri color lamp control...203
4.2.1 Tool change control..203
4.2.2 User M code control..206
4.2.3 M100、M101—User output and detection...............................207
4.2.4 Bus servo debugging..209
4.2.5 Automatic feeding M70, M71 functions................................214
4.2.6 Pulse spindle selection parameters 214

VOLUME I Programming instructions

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

- Standard absolute value motor, optional bus I/O, etc
- Minimum control accuracy 0.1um, maximum movement speed 60m/min
- Adaptive servo spindle can achieve functions such as spindle orientation and CS axis control
- Single head/multi head metric and British straight thread, taper thread, and end thread functions
- Equipped with hand pulse test cutting and hand pulse interruption functions
- Support servo turrets, four station electric tool holders, hydraulic tool holders, etc

1.2 technical norms

	Control axis: X-axis, Z-axis, Y-axis, C-axis; The C-axis can be used as the Cs axis
	Interpolation method: positioning (G00), straight line (G01), arc (G02, G03)

motion control function	Position instruction range: metric: -99999999.99mm~999999.99mm, minimum instruction unit: 0.0001mm English: -999.9999inch to 9999.9999inch, minimum instruction unit: 0.00001inch Note: When editing the diameter, the X-axis is reduced by half
	Maximum feed speed: linear 8000mm/min Feed rate: 0-150%, 15 level real-time adjustment
	Maximum fast speed: 60000 mm/min 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 ❖ 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 ❖ Support for statement based macro programs ❖ Support for 5-level subroutine calls and user macro program calls
screw thread function	❖ Plain thread (following the spindle) ❖ Single head/multi head metric British straight thread, taper thread, and end face thread, equal pitch thread and variable pitch thread ❖ The length, angle, and speed characteristics of thread unwinding can be set by programs and parameters ❖ Thread pitch: 0.001mm~500mm (metric system) 0.06 teeth/inch~25400 teeth/inch (British system)

T tool function	❖ Standard 4-station electric tool holder, optional tool holder: maximum setting for 8-station electric tool holder, Liuxin tool holder (12 stations), Dema tool holder (coding or counting type) ❖ Tool change method: MDI/automatic absolute tool change or manual relative tool change, forward rotation tool selection, reverse locking ❖ Tool change method: MDI/automatic absolute tool change or manual relative tool change, forward rotation tool selection, reverse locking ❖ Tool position signal input method: direct input
S principal axis function	❖ S2 digits (I/O gear control)/S5 digits (analog output) ❖ Spindle encoder: The number of encoder lines can be set (100-5000p/r) ❖ Encoder to spindle transmission ratio: (1-255): (1-255) ❖ Spindle magnification: 50%~120%, a total of eight levels of real-time adjustment ❖ 1 channel 0V~10V analog voltage output
M auxiliary function	❖ Specify with address M and 2 digits, M function can be customized ❖ System internal M instruction (non redefining): program end M02, M30; Program stop M00; Select to stop M01; Subprogram call M98; Subprogram end M99 ❖ Cooling liquid start/stop, lubrication start/stop, MDI/automatic control, chuck clamping/loosening, control tailstock in/out
program edit	❖ Program capacity: 400MB, 4000 programs ❖ Format: Relative/Absolute Mixed Programming ❖ Subprogram: editable, supports five fold subroutine nesting
Operation	❖ Method selection: edit, automatic, MDI, zero return, manual, single step, manual pulse ❖ 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; ❖ I/O unit input/output: 16/16, expandable

Security	❖ emergency stop ❖ Hardware travel limit ❖ Data backup and recovery
communication	❖ USB: USB file operation, direct processing of USB files, support for PLC program and system software USB upgrade

1.3 G code table

Command	Function	Command	Function
G00	Rapidtraverse (positioning)	G72	Radial roughing cycle
G01	Linear interpolation	G73	Closed cutting cycle
G02	CW arc interpolation	G70	Finishing cycle
G03	CCW arc interpolation	G74	Axial grooving cycle
G04	Dwell, exact stop	G75	Radial grooving cycle
G31	Skip function	G76	Multiple thread cutting cycle
G32	Constant pitch thread cutting	G83	Axial drilling cycle
G33	Constant pitch thread cutting	G80	Rigid tapping state cancel
G34	Thread cutting with variable lead	G90	Axial cutting cycle
G40	Tool nose radius compensation cancel	G92	Thread cutting cycle
G41	Tool nose radius compensation left	G94	Radial cutting cycle
G42	Tool nose radius compensation right	G96	Constant surface speed control
G50	Tool nose radius compensation right	G97	Constant surface speed control cancel
G54-G59	Workpiece coordinate system	G98	Feed per minute
G65	Macro program modal call	G99	Feed per revolution
G71	Axial roughing cycle (flute cycle)		

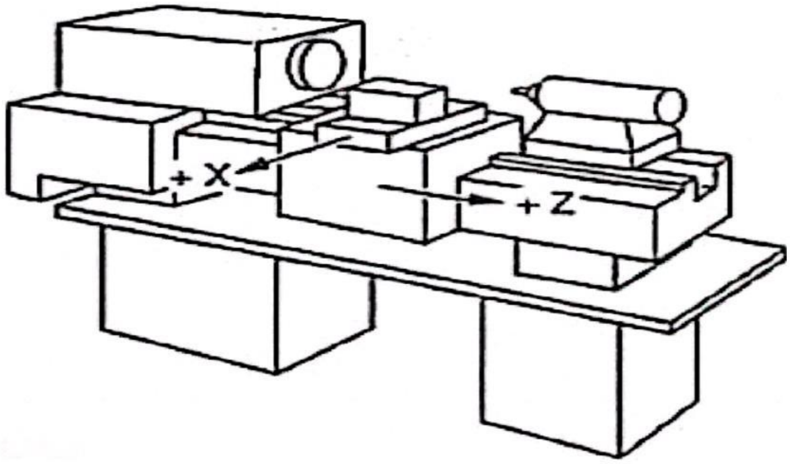
Chapter 2 Fundamentals of Programming

2.1 control axis

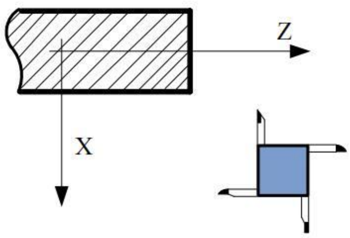
item	CNC system I5T3 I5T5
Number of basic control axes	2-axis (X, Z)

2.2 Axis name

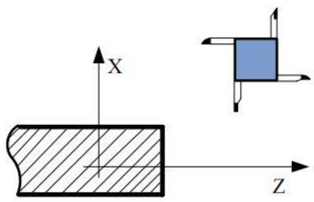
The default names for the two basic axes are X and Z



When the system uses a Cartesian coordinate system composed of the X-axis and Z-axis for positioning and interpolation motion, the X-axis represents the front and rear directions of the horizontal plane, and the Z-axis represents the left and right directions of the horizontal plane. The direction towards the workpiece is negative, and the direction away from the workpiece is positive. This system supports the functions of front and rear tool holders, and it is specified (from the front of the lathe) that the tool holder is called the front tool holder in front of the workpiece, and the tool holder is called the back tool holder behind the workpiece. The X direction of the coordinate system of the front and rear tool holders is exactly opposite, while the Z direction is the same. In future illustrations and examples, this manual will explain the coordinate system of the rear cutterbed, while the coordinate system of the front cutterbed can be inferred by analogy.

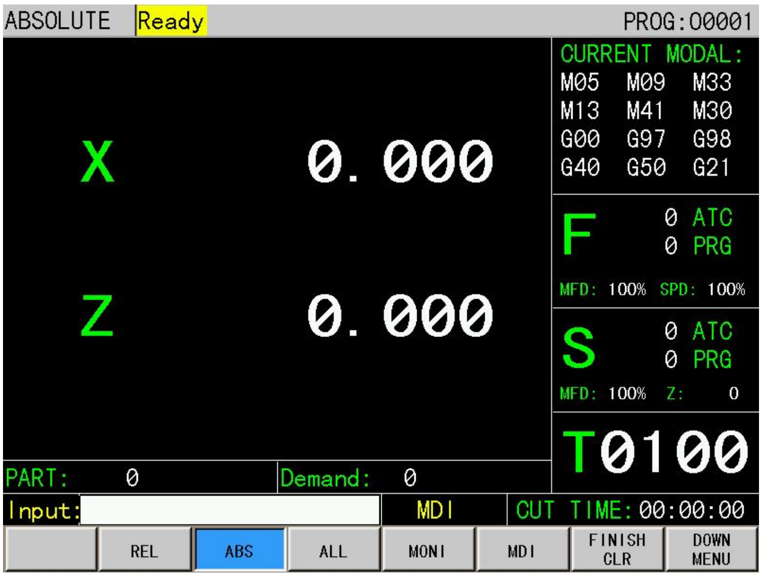


Front tool holder diagram



Rear tool holder diagram

2.3 Axis Display



2.4 Machine zero point and machine coordinate system

A specific point on a machine tool used as a machining reference is called the machine tool zero point. The machine tool manufacturer sets the zero point for each machine tool, usually at the maximum stroke in the positive direction of the X and Z axes. After setting it, it will not be moved or changed.

The coordinate system set with the zero point of the machine tool as the origin is called the machine coordinate system.

When CNC is powered on, it is not possible to determine the position of the machine tool zero point. Usually, it is necessary to automatically or manually return to the machine tool zero point to establish the machine tool coordinate system. After the machine tool returns to the machine tool zero point, CNC can automatically establish the machine tool coordinate system with the machine tool zero point as the origin.

2.5 Workpiece Coordinate System

The coordinate system used when processing workpieces is called the workpiece coordinate system (also known as the part coordinate system). The workpiece coordinate system is pre-set by the operator on the CNC side based on the installation position of the workpiece (setting the workpiece coordinate system).

Usually, all machining programs of a part set a common workpiece coordinate system (select the workpiece coordinate system).

Changing the workpiece coordinate system can be done by moving its origin (changing the position of the workpiece coordinate system).

The selection of reference position points for the workpiece coordinate system should try to meet the conditions of simple programming, less size conversion, and small machining errors caused. In general, the reference point should be selected on the benchmark for dimensioning or positioning. For lathe programming, the reference point is generally selected at the intersection point between the workpiece axis and the end face of the chuck (Figure 2-5-1) or the end face of the workpiece (Figure 2-5-2).

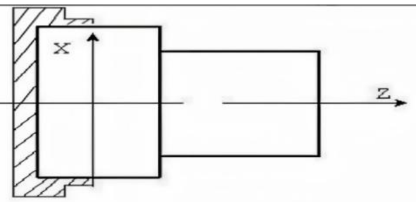


Fig. 2-5-1 Reference point on the end face of the chuck

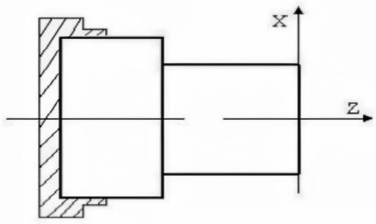


Fig. 2-5-2 Reference point on the end face of the workpiece

There are two ways to set the workpiece coordinate system:

1. Set with G50.
2. Set using G54 to G59.

2.6 Absolute value programming and incremental value programming

There are two ways to represent the movement of the code axis: absolute value code and relative value code. Absolute value code is a method of programming using the coordinate values of the endpoint position of the axis movement, known as absolute coordinate programming. Incremental value code is a method of directly programming with axis movement, known as incremental value programming. In this system, absolute coordinate programming uses instructions X and Z, and incremental value programming uses instructions U and W.

Absolute value code	Absolute value code	notes
X	U	X Axis movement code
Z	W	Z Axis movement code

Example: Write the A → B program in Figure 2-6-1 using a mixed programming method of absolute value, incremental value, absolute value, and incremental value.

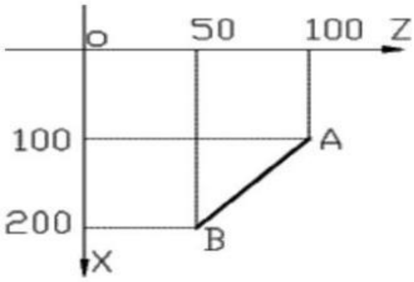


Fig. 2-6-1

Absolute coordinate programming: G01 X200. Z50.;

Relative coordinate programming: G01 U200. W50.;

Mixed coordinate programming: G01 X200. W-50.; or G01 U100. Z50.

Note: When X/Z and U/W are executed together in a program segment, an alarm will be generated

G01 X20. W30. U20. Z30.;

2.7 Programming for diameter and radius methods

According to whether the X-axis coordinate value is input as a diameter value or a radius value during
programming, it can be divided into: diameter programming and radius programming.

Diameter programming: When the (0: diameter 1: radius) programming of the [Feed Axis Parameters] class is set to "diameter", the programming value of the X-axis in the program is entered according to the diameter value, and the coordinates of the X-axis are displayed as the diameter value.

Radius programming: When programming (0: diameter 1: radius) for the [Feed Axis Parameters] class as "radius", the programming value of the X-axis in the program is entered according to the radius value, and the coordinates of the X-axis are displayed as the radius value.

The addresses related to diameter programming or radius programming are shown in the table below:

site	explanation	Diameter Programming	Radius Programming
X	X-axis coordinates	Diameter value representation	Radius value representation
	G50 sets X-axis coordinates		
U	X-axis movement increment	Diameter value representation	Radius value representation
	X-axis precision machining allowance in codes G71, G72, and G73	Diameter value representation	Radius value representation
R	The amount of tool retraction when cutting to the end point in G74	Diameter value representation	Diameter value representation

Except for the addresses listed in the above table, all other addresses and data, such as the radius of an arc, the taper of G90, and other programming values for the X-axis, are input based on the radius value, regardless of the diameter or radius programming settings.

Note: Now almost all CNC lathe debugging is to set the (0: diameter 1: radius) of parameter - [position parameter] P1.2 as "diameter", and then set the electronic gear ratio of axis X as: 1/2 output, (Usually, this 1/2 gear ratio is set at the servo actuator end, as the servo driver parameters are rarely changed after setting, which can reduce the risk of incorrect modifications. However, when equipped with a stepper driver, as the stepper driver may not have the function of electronic gear ratio, it is necessary to set the 1/2 gear ratio at the system end.). For example, when the gear ratio of the X-axis of a CNC lathe is set to 4/5, it happens to be 1/1 output, that is, when the system runs G00 U1.0, the drag plate of the X-axis also moves 1.0mm in the forward direction. To make the drag plate only move 0.5mm, we multiply 4/5 by 1/2 to obtain 4/10, which is approximately 2/5, and set 2/5 as the gear ratio in the X-direction. Then, when the system runs again: G00 U1.0, the X-axis drag plate runs at 0.5mm, so the cutting on both sides during machining is exactly 1mm.

Note: In the instructions described later in this user manual, unless otherwise specified, diameter programming is used.

2. 8 Modal, non modal, and initial state

Modality refers to the execution of the corresponding function and state of a word, which remains valid until its function and state are re executed. In other words, if the same function and state are used in future program segments, there is no need to input this field again.

For example, the following program:

G00 X150 Z100; (Quickly locate to X150 Z100)

X170 Z30; (Quickly locate to X170 Z30, G00 is the modal code, which can be omitted without input)

G01 X55 Z50 F200; (Linear interpolation to X55 Z50, feed speed 200mm/min G00 → G01)

X100; (Linear interpolation to X100 Z50, with a feed rate of 200 mm/min. G01, Z50, and F200 are all modal codes and can be omitted without input)

G00 X0 Z0; (Quickly locate to G01 → G00 at X0 Z0)

Non modal refers to the fact that the function and state of the corresponding field are only valid once executed, and in the future, the same function and state must be executed again. In other words, if the same function and state are used in future program segments, the field must be inputted again.

The initial state refers to the default functions and states of the system after power on, which means that if no corresponding functional state is specified after power on, the system will execute according to the initial state functions and states. The initial states of this system are G00, G18, G21, G40, G54, G97, and G98.

For example, the following program:

O0001;

G0 X100 Z100; (Quickly locate to X100 Z100, where G0 is the initial state of the system. When the system does not command other modes, the system executes the movement in the initial state G00 mode)

G1 X0 Z0 F100; (The system is linearly interpolated to X0 Z0 at a feed rate of 100 mm/min, as G98 is a hourly feed mode and G98 is also the initial state of system power on)

Chapter 3 Composition of Part Program

3.1 Program composition

A program is composed of multiple program segments, which in turn are composed of words. Each program segment is separated by a program segment end code (ISO for LF, EIA for CR). The character ";" is used in this manual to indicate the end of program segment code.

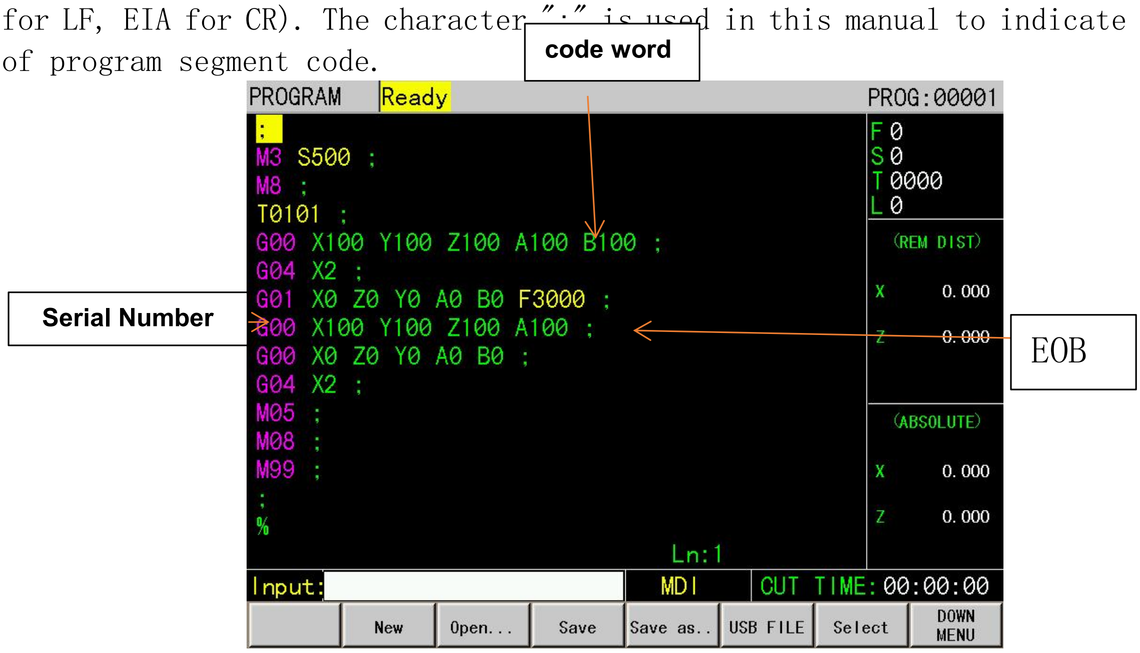


Fig. 3-1-1 Program Structure

The set of codes that control CNC machine tools to complete part processing is called a program. After the written program is input into the CNC system, the system controls the tool to move along a straight line or arc based on the code, or to rotate or stop the spindle. In the program, these codes should be written according to the actual movement sequence of the machine tool. The structure of the program is shown in Figure 3-1-1.

3.1.1 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

01.CNC

If the program is to be called (such as M98), the program name should be composed of the address O and subsequent four digit values, as shown in Figure 3-1-1.

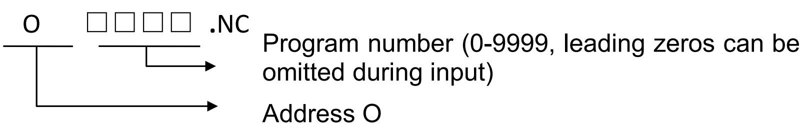


Fig. 2-1-1-1 Composition of program

3.1.2 code word

Code words (Figure 3-1-1) are the elements that make up the program segment. A word is composed of an address and the number after it (sometimes preceded by a+or - sign)

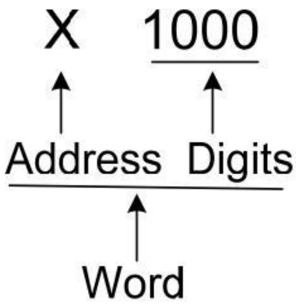


Fig. 3-1-2 Word table

The address is one of the English alphabet (A~Z). It specifies the meaning of the subsequent values. In this system, the addresses that can be used, their meanings, and value ranges are shown in the table:

According to different preparation functions, sometimes an address also has different meanings.

ADDRESS	Value range	Functional significance
O	0～9999	Program Name
N	0～9999	Block number
G	00～99	Preparatory function
X	-9999.9999～9999.9999（mm）	X coordinate
	0～999.999（S）	Pause time
Y	-9999.9999～9999.9999（mm）	Y coordinate
Z	-9999.9999～9999.9999（mm）	Z coordinate
R	-9999.9999～9999.9999（mm）	Arc radius/angular displacement
	-9999.9999～9999.9999（mm）	R-plane in a fixed cycle
	-9999.9999～9999.9999（mm）	Tool retraction clearance in G74, G75
	-99999.9999～9999.9999（mm）	Tool retraction clearance from end point in G74, G75
	-9999.9999～9999.9999（mm）	Taper in G90, G92, G94, G96
	0～9999.9999（mm）	Tool retraction in G71, G72
	1～999（Times）	Roughing cycle times in G73
	0～999.999（mm）	Finishing allowance in G76
I	-9999.9999～9999.9999（mm）	X vector between arc center and starting point
K	-9999.9999～9999.9999（mm）	Z vector between arc center and starting point
F	0～9999（mm/min）	Feedrate per minute
	0.001～500(mm/r)	Feedrate per rev

ADDRESS	Value range	Functional significance
S	0～9999（r/min）	Spindle speed specified
	00～04	Multi-gear spindle output
T	0～9999	Tool function
M	00～99	Miscellaneous function output, M program execution flow
P	0～9999.9999（ms）	Pause time
	1～9999	Subprogram call times
	0～9999.99（mm）	X circle movement in G74, G75
	0～999999	Calling times of subprogram number
Q	0～999999	End block number of finishing in the compound cycle
Q	0～9999.99（mm）	Z circle movement in G74, G75
H	01～99	Operand in G65
L	1～9999	The number of times a subroutine is repeatedly called

All the limits shown in the above table are for CNC devices, while the restrictions on machine tools are not included. Please pay special attention. Therefore, when programming, in addition to referring to this manual, it is also necessary to refer to the machine tool manufacturer's user manual and program based on an understanding of programming limitations.

3.2 General structure of the program

Programs are divided into main programs and subroutines. Usually, CNC moves according to the instructions of the main program. If there is code calling a subroutine in the main program, CNC moves according to the subroutine. When encountering code returning to the main program in the subroutine, CNC returns to the main program to continue executing. The sequence of program actions is shown in Figure 3-2-1.

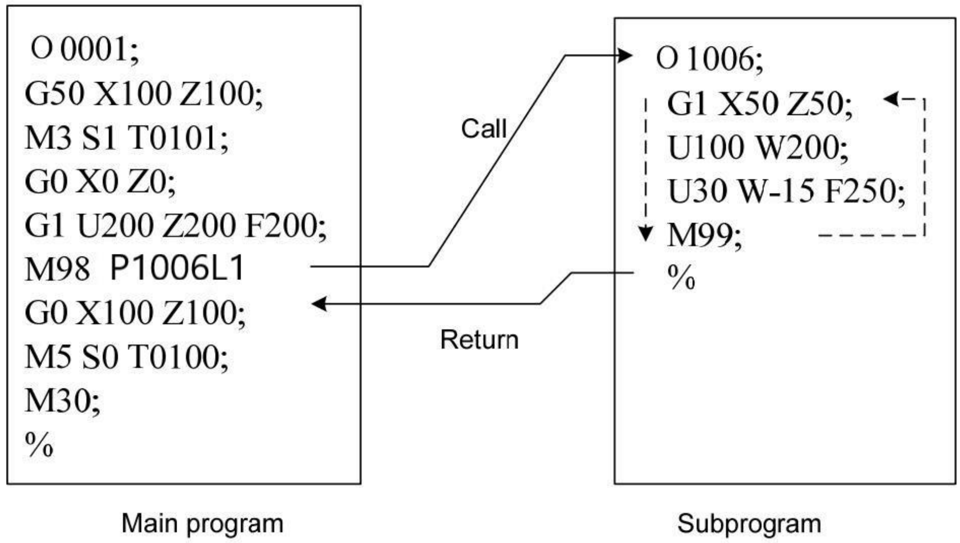


Fig. 3-2-1

The composition structure of the main program and subroutines is consistent.

When there is a fixed sequence in the program and it appears repeatedly, it can be used as a subroutine and stored in memory in advance, without the need to write repeatedly to simplify the program. The subroutine can be automatically called out, usually using M98 in the main program, and the called subroutine can also call other subroutines. The subroutine called out from the main program is called a single subroutine, and a total of four subroutines can be called (as shown in Figure 3-2). The last segment of the subprogram is returned to the main program with M99 code, and the next segment of the subprogram is called to continue execution. (If the last segment of the subprogram ends with M02 or M30 code, the function returns to the main program as M99, and calls the next segment of the subprogram segment to continue execution.)

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

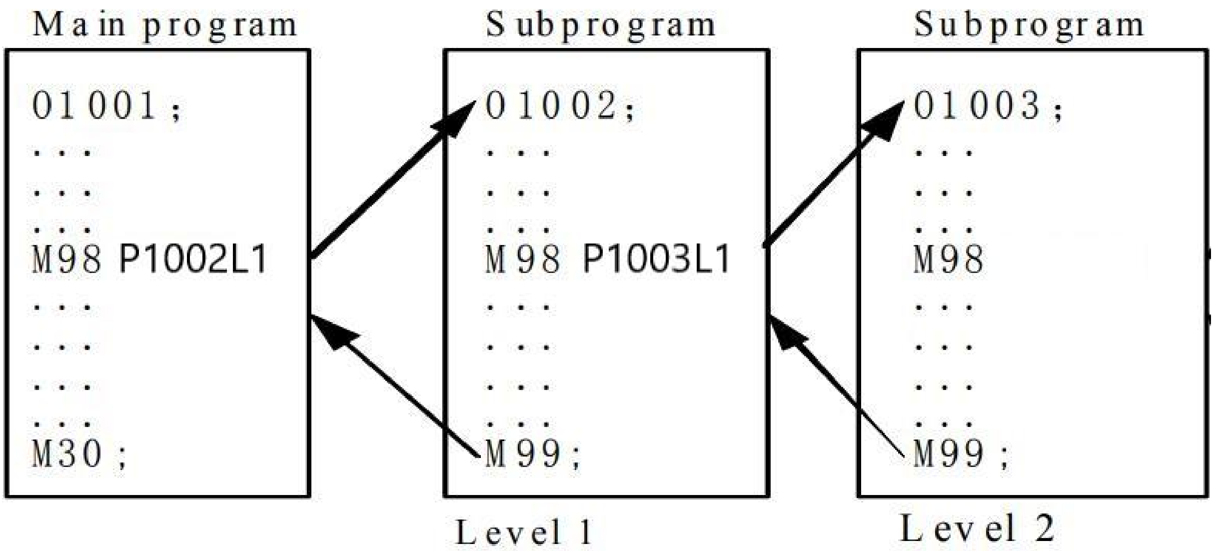


Fig. 3-2-2 Double subroutine nesting

can use a subroutine to call the same subroutine continuously and repeatedly, up to 9999 times.

3.2.1 Subprogram writing

Write a subroutine in the following format

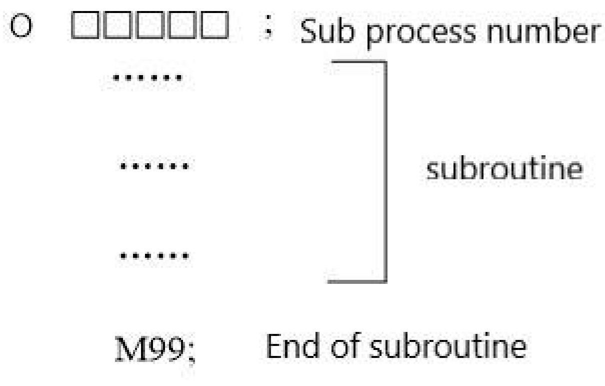


Fig. 3-2-1-1

At the beginning of the subroutine, write the subroutine number after the address O, and at the end of the subroutine, write the M99 code (in the format shown above).

3.2.2 Calling subprograms

The subroutine is called out and executed by the main program or subroutine calling code. The code format for calling subroutines is as follows:

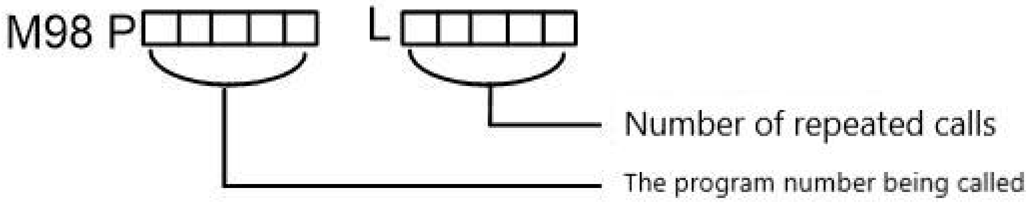


Fig. 3-2-2-1

● If the number of repetitions is omitted, it is considered as 1 repetition.

(Example) M98 P1002L5; (Indicates that the subroutine with number 1002 has been called 5 times in a row.)

● The order in which subroutines are called from the main program for execution

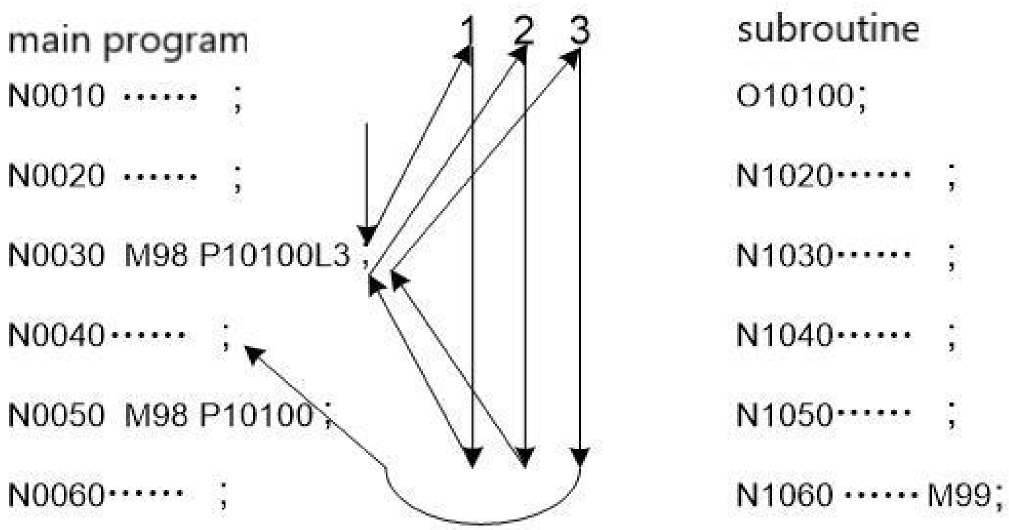


Fig. 3-2-2-2

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

Note 1: When the subroutine number specified with address P cannot be retrieved, an alarm is generated.

Note 2: Subprograms 9000~9999 are reserved by the system. When a user calls this type of subroutine, the system can execute the content of the subroutine, and the system will not display the content of the subroutine.

3.2.3 End of program

The program starts with the program name and ends with M02, M30, or M99 (see Figure 3-2). During the execution of the program, if the program end code is detected as M02, M30, or M99, and if the M02, M30 code ends, the program ends and becomes reset; M30 can use parameter - [Quick Debugging] P030 to control whether to return to the beginning of the program, while M02 can use parameter - [Quick Debugging] P029 to control whether to return to the beginning of the program. If the M99 code ends, the program header is returned and the program is executed in a loop; If M99, M02, and M30 are at the end of the subroutine, return to the program that called the subroutine and continue executing the subsequent program segments.

Chapter 4 Preparation function G COMMANDS

4.1 Types of ready function G-code

The preparation function is represented by the G-code and the following number, specifying the meaning of the program segment where it is located. There are two types of G-code:

Category	Meaning
Non modal G-code	Only valid in the program segment being instructed
Modal G-code	Valid until other G-code in the same group

Fig. 4-1-1

(Example) G01 and G00 are modal G-code of the same group

G01 X __ ；

Z ______ ； G01 **effective**

X ______ ； G01 **effective**

G00 Z____； G00 **effective**

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

Word	Group	Function
G04	**00**	Dwell time preset
G28		Dwell time preset
G31		Skip interpolation
G50		Setting workpiece coordinate system
G65		Macro command
G70		Finishing cycle
G71		Axial roughing cycle
G72		Radial roughing cycle
G73		Closed c
G74		Axial grooving cycle
G75		Radial grooving cycle
G76		Multiple thread cutting cycle
G00*	**01**	Rapid traverse movement
G01		Linear interpolation
G02		Circular interpolation(CW)
G03		Circular interpolation(CCW)

G32		Thread cutting
G34		Variable pitch thread cutting
G90		Axial cutting cycle
G92		Thread cutting cycle
G94		Radial cutting cycle
G96	02	Constant surface speed ON
G97*		Constant surface speed OFF
G98*	03	Feed per minute
G99		Feed per rev
G20	06	Inch select
G21*		Metric select
G40*	07	Cancel cutter radius compensation
G41		Tool nose radius compensation left contour (option)
G42		Tool nose radius compensation right contour(option)
G54*	14	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

Note 1: If the modal code is in the same segment as the non modal code, the non modal code takes priority, and the corresponding modal code is changed based on other modal codes in the same segment, but they are not executed.
Note 2: For G-code marked with *, when the power is turned on, the system is in this G-code state.
Note 3: All G-code in the 00 group are modeless G-code.
Note 4: If the G-code not listed in the list of G-code is used, an alarm will appear, or the G-code that does not have the selection function will also give an alarm.
Note 5: Several G-code of different groups can be instructed in the same program segment. In principle, more than two G-code of the same group cannot be instructed in the same program segment. If the same group of codes is set to not alarm in the same segment, the G-code appearing later shall prevail.
Note 6: G codes are represented by group numbers according to different types.

4.2 G COMMANDS

4.2.1 Rapid Traverse Movement G00

Command format: G00 X(U) Z(W) ;

Command function: X, Z rapidly traverses at the respective traverse speed to the end points from their starting point. G00 is initial command as Fig.4-2-1-1. X, Z traverses at the respective traverse speed, the short axis arrives the end point and the length axis continuously moves to the end point and the compound path may be not linear.

Command specification: G00 is initial mode; X, U, Z, W range: ±99999999× least input increment; Can omit one or all command addresses X(U), Z(W). The coordinate values of starting point and end point are the same when omitting one command address; the end point and the starting point are in the same position when all are omitted. X, Z are valid, and U, W are invalid when X, U, Z and W are in the same one block.

Command path:

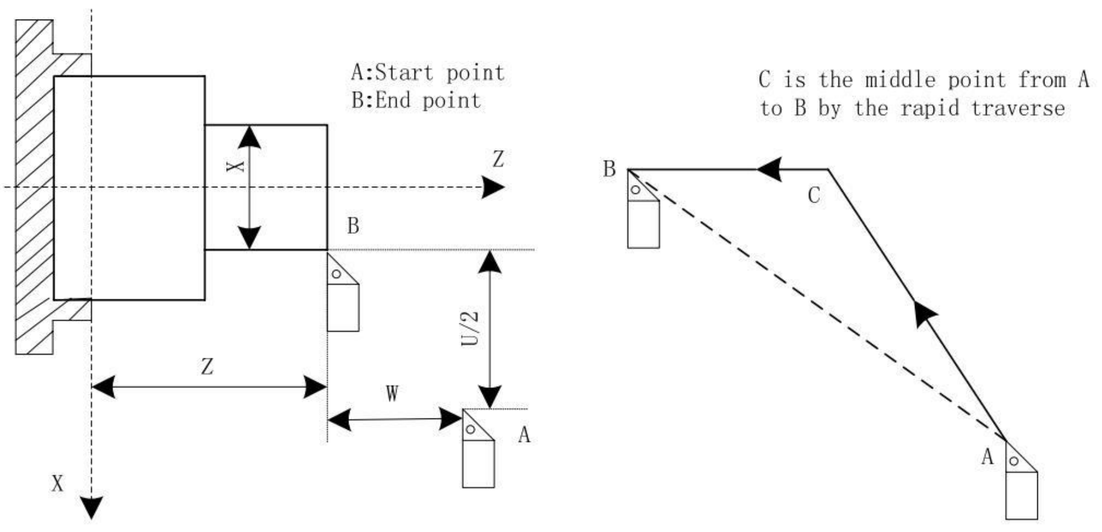


Fig. 4-2-1-1

Remarks：

1. The tool does not move without specifying positioning parameters, and the system only changes the current mode of tool movement to G00.
2. G00 and G0 are equivalent formats.
3. The G00 speed of the X and Z axes is set by the parameter - [Feed Axis Parameter] P021-P023.

Restrictions：

The fast moving speed is set by parameters, such as setting the F speed in G00 code, which is the cutting feed speed for the subsequent machining segment.

Example：

G00 X0 Z10 F500； Fast feed using system parameter settings

G1 X20 Z50； Using F500 feed rate

Adjust the fast feed speed using the buttons on the operation panel (as shown in Figure 4-2-1-2), F0, 25, 50100%; The speed corresponding to F0 is set by parameter - [Feed Axis Parameter] P030, which is common to all axes.

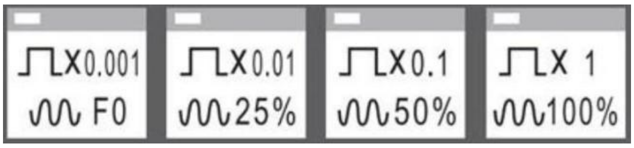


Fig. 4-2-1-2 Fast feed rate button

Attention: Pay attention to the position of the workbench and workpiece during programming to prevent collision with the Tool.

Example: The tool quickly moves from point A to point B. As shown in Figures 4-2-1-3

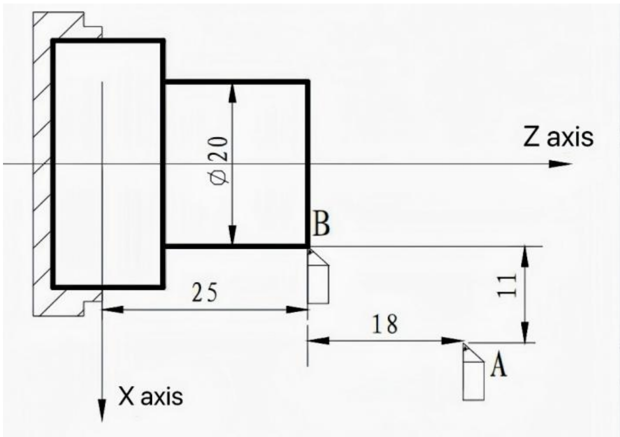


Fig. 4-2-1-3

G0 X20 Z25; (absolute programming)

G0 U-22 W-18; (Relative programming)
G0 X20 W-18; (mix-programming)
G0 U-22 Z25; (mix-programming)

4.2.2 Linear Interpolation G01

Command format: G01 X(U) _ Z(W) _ F_;

Command function: The movement path is a straight line from starting point to end point as Fig.4-2-2-1

Command specification: G01 is modal.

Can omit one or all command addresses X (U), Z (W). The coordinate values of starting point and end point are the same when omitting one command address; the end point and the starting point are in thesame position when all are omitted. F command value is the vector compound speed of X and Z instantaneous speed and the actual cutting feedrate is the product between the feedrate override and F command value. After F command value is executed, it has been reserved unless the new one is executed. Do not repeat it when the following G commands adopt functions of F word.

Note: In the G98 state, the maximum value of F does not exceed the set value of the data parameter - [Feed Axis] P024 (Cutting Feed Upper Limit Speed).

Command path:

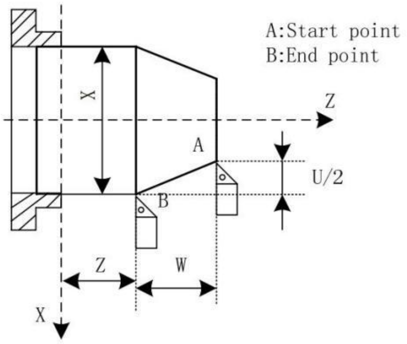


Fig. 4-2-2-1

Example: Cutting path from Φ40 to Φ60 as Fig.4-2-2-2

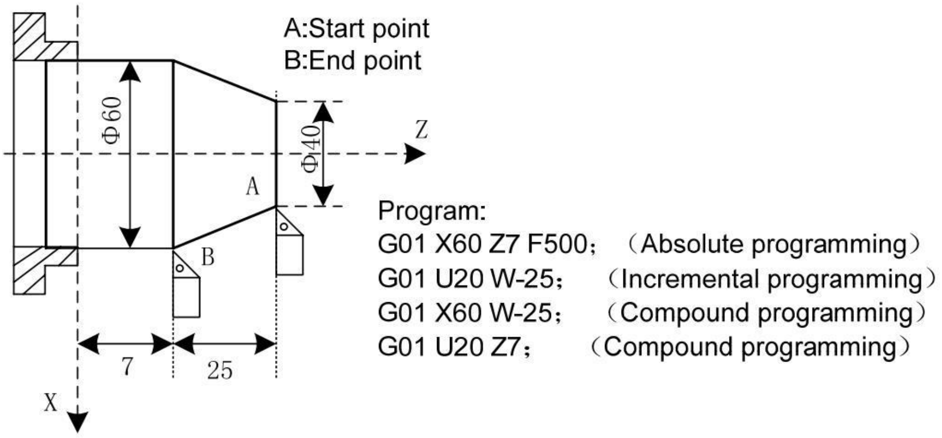


Fig. 4-2-2-2

4.2.3 Circular Interpolation G02, G03

Command format: G02 X(U)__ Z(W)__ (R__) or I__ K__

G03 X(U)__ Z(W)__ (R__) or I__ K__

Command function:

G02 movement path is clockwise (rear tool post coordinate system)/counterclockwise (front tool post coordinate system) arc from starting point to end point as Fig.4-2-3-1.

G03 movement path is counterclockwise (rear tool post coordinate system/clockwise (front tool post coordinate system) arc from starting point to end point as Fig.4-2-3-2.

Command path:

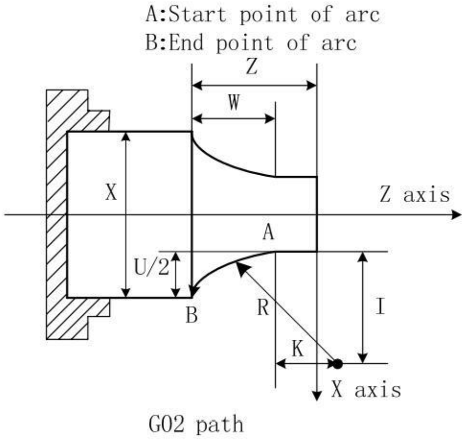


Fig. 4-2-3-1

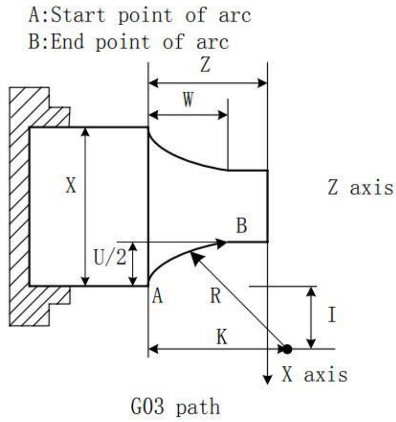


Fig. 4-2-3-2

Command specification:

G02, G03 are modal,

R is arc radius, range: ±99999999× least input increment;

I: X difference value between circle center and starting point of arc in radius;

K: Z difference value between circle center and starting point of arc;

Center point of arc is specified by address I, K which separately corresponds to X, Z, I, K expresses the vector (it is the increment value) from starting point to center point of arc as the following figure; I=Coordinates of center point-that of starting point in X direction; K= Coordinates of center point-that of starting point in Z direction;

I, K are with sign symbol. When directions of I, K are the same as those of X, Z, they are positive, otherwise, they are negative.

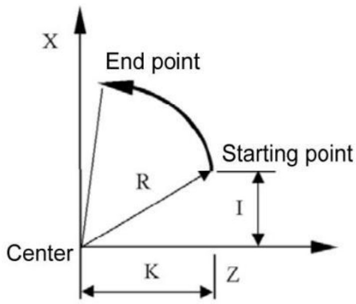


Fig. 4-2-3-3

Arc direction: G02/G03 direction (clockwise/counterclockwise) is opposite on the front tool post coordinate system and the rear one as Fig.4-2-3-4

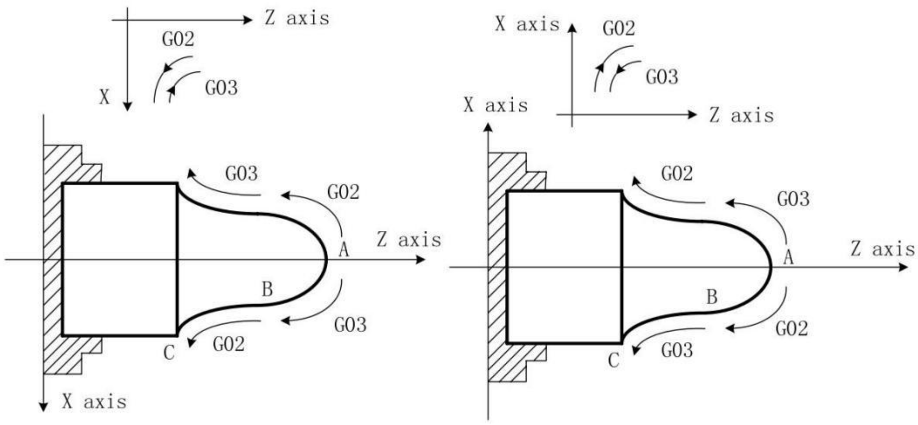


Fig. 4-2-3-4

Notes:

- When I = 0 or K = 0, they can be omitted; one of I, K or R must be input, otherwise the system alarms.
- R is valid and I, K are invalid when they are input at the same time.
- R value must be equal to or more than half distance from starting point to end point, and the system alarms if the end point is not on the arc defined by R command;
- Omit all or one of X(U), Z(W); coordinates of starting point and end point of this axis are the same when omitting ones, the path is a full circle(360°) in G02/G03 when center point are specified by I,K; the path is 0(0°) when center point is specified by R.
- R should be used for programming. The system executes in R= 2 2 I + K to ensure starting point and end point of arc path are the specified ones in I, K programming.
- When the distance from center point to end point is not equal to R(R= 2 2 I + K) in I,K programming, the system automatically adjusts position of center point to ensure starting point and end point of arc path are the specified ones; When the distance from center point to end point is more than 2R, and the system alarms.
- Arc is less than 360° when R is commanded, the arc is more than 180° when R is negative, and it is less than or equal to 180° when R is positive.

Example: Arc cutting path from Φ45.25 to Φ63.06 shown in Fig.4-2-3-5

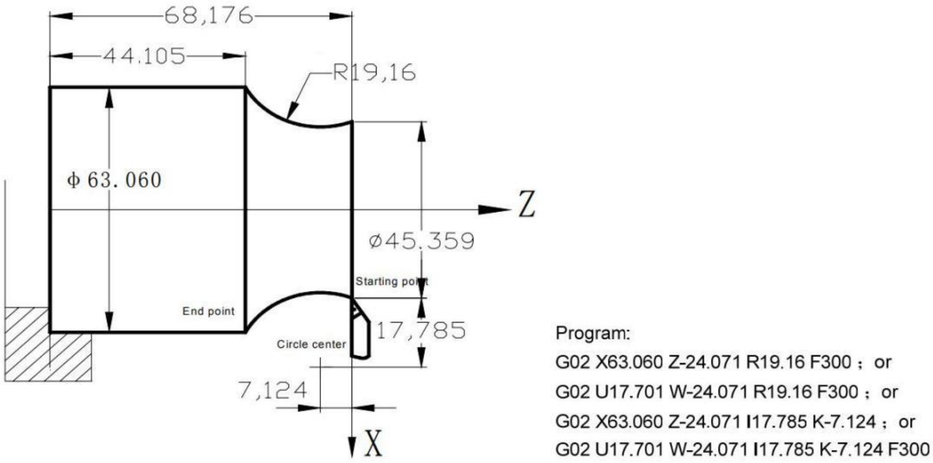


Fig. 4-2-3-5

Compound programming in G02/G03:

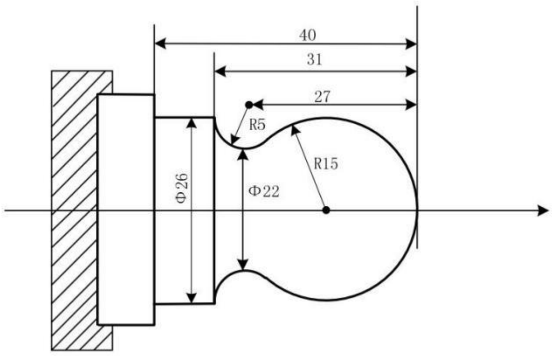


Fig. 4-2-3-6 **Circular programming example**

```
Program: O0001
N001 G0 X40 Z5;         (Rapidly traverse)
N002 M03 S200;          (Start spindle)
N003 G01 X0 Z0 F900;    (Approach workpiece)
N005 G03 U24 W-24 R15;  (Cut R15 arc)
N006 G02 X26 Z-31 R5;   (Cut R5 arc)
N007 G01 Z-40;          (Cut Φ26)
N008 X40 Z5;            (Return to starting point)
N009 M30;               (End of program)
```

4.2.4 Dwell G04

Command format: G04 X(U)_ or P_

Command function:each axis stops the motion, the modal of G commands and the reserved data, state are not changed, and execute the next block after dwelling the defined time. Within the G04 dwell time, receive the skip signal (it is determined by Q value), intermit the dwell and then perform the next block.

Command specification: 1、: G04 is non-modal.

2、 G04 dwell time is defined by the word P__, X__ or U__.

3、 When P, X or U specifies a negative value, it means that the dwell time is 0.

4、 The system exactly stop a block when P, X, U, Q are not input.

5、 Range is -99999999～99999999 （unit: ms）.

6、 X、U range is -9999.999～9999.999 (unit: s)。

4.2.5 Chamfering Function

Chamfering function is to insert one straight line or circular between two contours to make the tool smoothly transmit from one contour to another one.

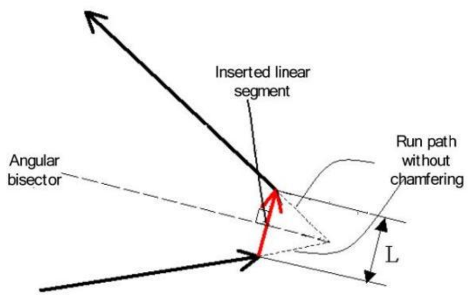
1) Linear Chamfering

Linear chamfering: insert one straight line in the linear contours, arc contours, linear contour and arc contour. The command address of linear chamfering is L, behind which data is the length of chamfering straight line. The linear chamfering must be used in G01, G02 or G03 command

A. Linear to linear

Command format: G01 X(U)_ Z(W)_ L_ ；
G01 X(U)_ Z(W)_ ；

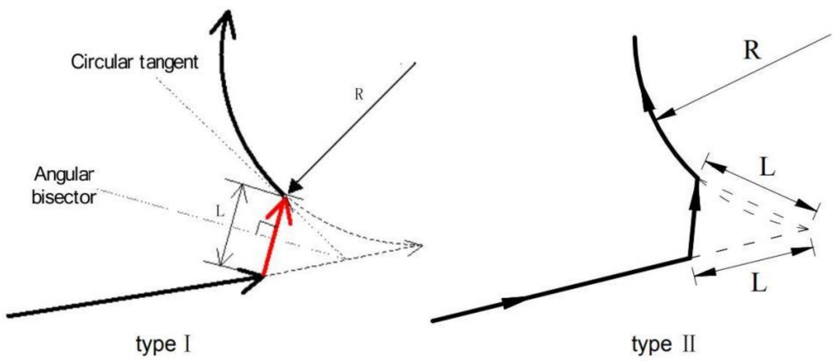
Command function: insert one straight line between two linear interpolation blocks.



B. Linear to circular

Command format: G01 X(U)_ Z(W)_ L_;
G02/G03 X(U)_ Z(W)_ R_;
OrG01 X(U)_ Z(W)_ L_;
G02/G03 X(U)_ Z(W)_ I_ K_;

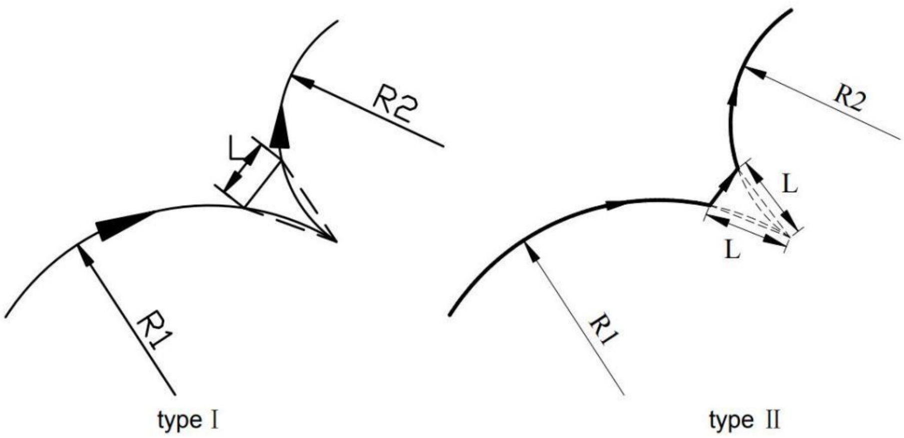
Command function: insert one straight line between the linear and circular interpolation blocks.



C. Circular to circular

Command format： G02/G03 X(U)_ Z(W)_ R_ L_；
G02/G03 X(U)_ Z(W)_ R_；
Or G02/G03 X(U)_ Z(W)_ I_ K_ L_；
G02/G03 X(U)_ Z(W)_ I_ K_；

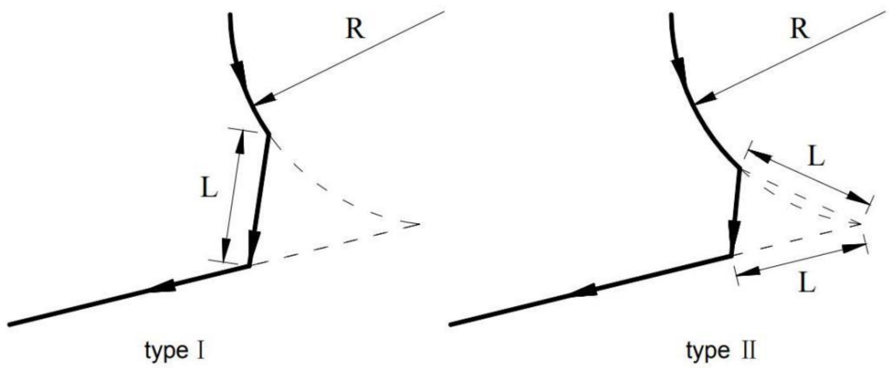
Command function： insert one straight line between two circular interpolation blocks.



D. Circular to linear

Command format： G02/G03 X(U)_ Z(W)_ R_ L_；
G01 X(U)_ Z(W)_；
Or G02/G03 X(U)_ Z(W)_ I_ K_ L_；
G01 X(U)_ Z(W)_；

Command function： insert one straight line block between circular and linear interpolation block.



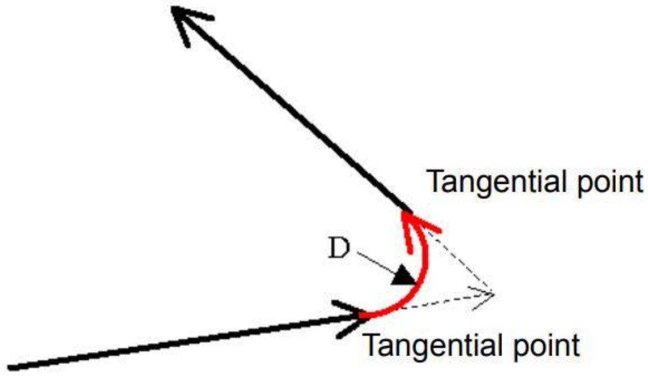
2) Circular Chamfering

Circular chamfering: insert one circular between linear contours, circular contours, linear contour and circular contour, the circular and the contour line are transited by the tangent. The command of circular chamfering is D, and the data behind the command is the radius of chamfering circular. The circular chamfering must be used in G01, G02 or G03.

A. Linear to linear

Command format： G01 X(U)_ Z(W)_ D_；
G01 X(U)_ Z(W)_；

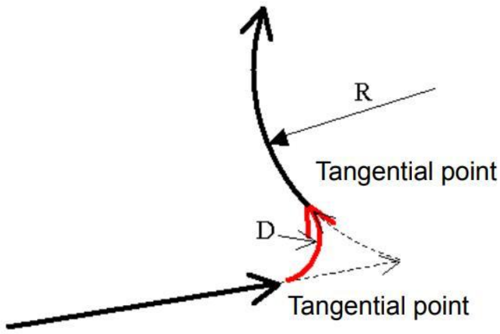
Command function: insert one circular between two straight lines, the inserted circular block and two straight lines are tangent, the radius is the data behind the command address D.



B. **Linear to circular**

Command format: G01 X(U)_ Z(W)_ D_;
G02/G03 X(U)_ Z(W)_ R_;
Or G01 X(U)_ Z(W)_ D_;
G02/G03 X(U)_ Z(W)_ I_ K_;

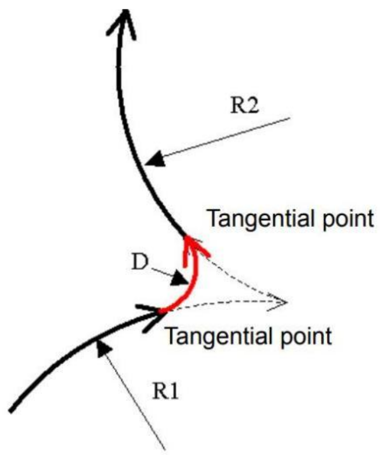
Command function: insert one circular between linear and circular, the inserted circular is tangent to the linear and the circular, and the radius is the data behind the command address D.



C. **Circular to circular**

Command format: G02/G03 X(U)_ Z(W)_ R_ D_;
G02/G03 X(U)_ Z(W)_ R_;
Or G02/G03 X(U)_ Z(W)_ R_ D_;
G02/G03 X(U)_ Z(W)_ I_ K_;
Or G02/G03 X(U)_ Z(W)_ I_ K_ D_;
G02/G03 X(U)_ Z(W)_ I_ K_;
Or G02/G03 X(U)_ Z(W)_ I_ K_ D_;
G02/G03 X(U)_ Z(W)_ R_;

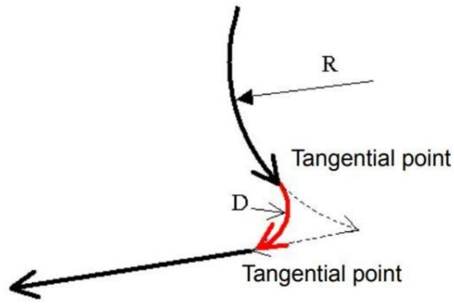
Command function: insert one circular between two circular blocks, the inserted circular is tangent to the two circular blocks, and the radius is the data behind the command address D.



d. **Circular to linear**

Command format: G02/G03 X(U)_ Z(W)_ R_ D_;
G01 X(U)_ Z(W)_;
或 G02/G03 X(U)_ Z(W)_ I_ K_ D_;
G01 X(U)_ Z(W)_;

Command function: insert one circular block between the circular and the linear, the inserted circular block is tangent to the circular and the linear, and the radius is the data behind the command address D.



3) Special Cases

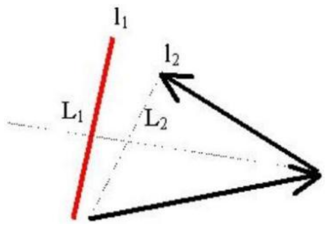
The chamfering function is invalid or alarms as follows:

① **Linear chamfering**

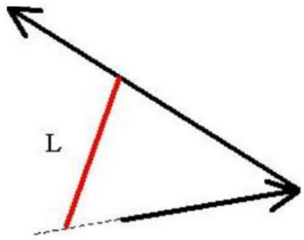
A. The chamfering function is invalid when two interpolation straight lines are in the same linear

B. CNC alarms when the chamfering linear is too long.

L1 i is the chamfering linear, and the length is L1; l2 is the third edge of the triangle which is formed by two interpolation straight lines, the length is L2, CNC alarms when L1 is bigger than L2 as follows:



C. Some linear block is too short
The chamfering linear length is L, CNC alarms when other end of the caculated chamfering linear is not in the interpolation linear(in the extension line of the interpolation linear).

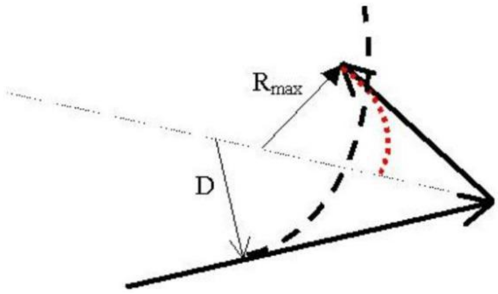


② **Circular chamfering**

A. The circular chamfering function is invalid when two interpolation straight lines are in the same block.

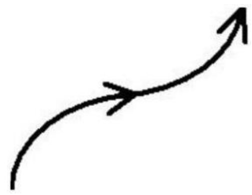
B. CNC alarms when the chamfering circular radius is too big.

CNC alarms when the chamfering circular radius is D, max. circular radius of the tangential linear lines is Rmax which is less than D as follows.

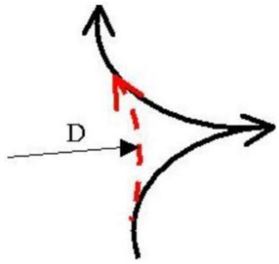


C. The circular chamfering function is invalid when the linear and the circular, or the circular and the linear are tangential.

D. The circular chamfering function is invalid when one circular and another one are tangential. ;



The circular chamfering function is valid when the circular tangency is as follows:



4.2.6 Workpiece Coordinate System G54～G59

Command function: Specify the current workpiece coordinate system, and select the workpiece coordinate system by specifying the G-code of the workpiece coordinate system in the program.

Command format: G54～G59

Explanation:

1、 No code parameters

2、 The system itself can set six workpiece coordinate systems, and any one of them can be selected by 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、 When starting up, the system displays the workpiece coordinate systems G54~G59, G50 or additional workpiece coordinate systems that were executed before the power outage.

4、 When calling different workpiece coordinate systems in the program section, the axis of the command movement will be positioned to the coordinate point under the new workpiece coordinate system; If there is no command to move the axis, the coordinates will jump to the corresponding coordinate values in the new workpiece coordinate system, while the actual machine position will not change.

Example: The machine tool coordinates corresponding to the origin of the G54 coordinate system are (10, 10)

The machine tool coordinates corresponding to the origin of the G55 coordinate system are (30, 30)

When executing the program in sequence, the absolute coordinates of the endpoint and the machine coordinate are displayed as follows:

program	Absolute coordinates	Machine tool coordinates
G0 G54 X50 Z50	50, 50	60, 60
G55 X100	100, 30	130, 60
X120 Z80	120, 80	150, 110

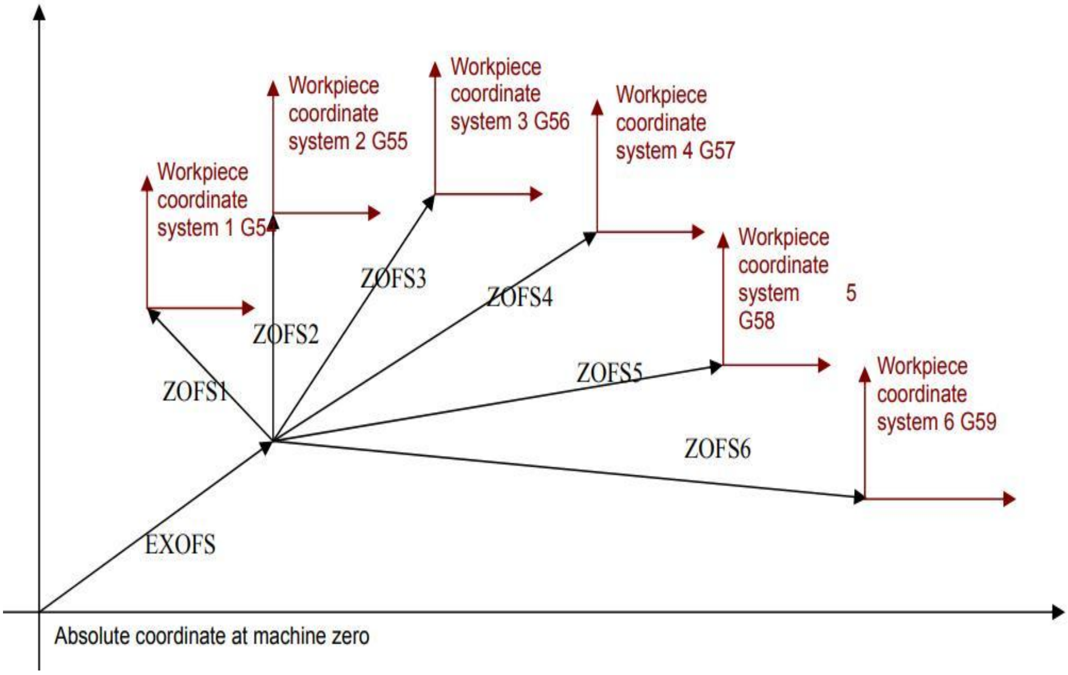


Fig. 4-2-6-1

Fig. **4-2-6-1**, After starting the machine tool, manually return to the mechanical zero point and establish the machine tool coordinate system from the mechanical zero point, thereby generating the machine tool reference point and determining the workpiece coordinate system. The origin of six workpiece coordinate systems can be specified by inputting coordinate offset or setting corresponding parameters in the input method. These six workpiece coordinate systems are set based on the distance from the mechanical zero point to the respective coordinate system zero point.

example: N10 G55 G00 X100 Z20;
N20 G56 X80.5 Z25.5;

In the above example, when the N10 program segment starts executing, it quickly locates to the position of the workpiece coordinate system G55 (X=100, Z=20). When the N20 program segment starts executing, it quickly locates the position of the workpiece coordinate system G56, and the absolute coordinate value automatically becomes the coordinate value under the G56 workpiece coordinate system (X=80.5, Z=25.5)

4.2.7 Workpiece Coordinate System G50

Command format: G50 X(U) Z(W) ;

Command function: define the absolute coordinates of current position and create the workpiece coordinates system (called floating coordinates system) by setting the absolute coordinates of current position in the system. After G50 is executed, the system takes the current position as the program zero (program reference point), and the system returns to the point after executing the program zero return. After the workpiece coordinate system is created, input the coordinate values with the coordinate system in the absolute coordinates programming until the next workpiece coordinate system is created again (using G50).

Command specifications:

G50 is non-modal;

X: New absolute coordinates of current position in X direction;

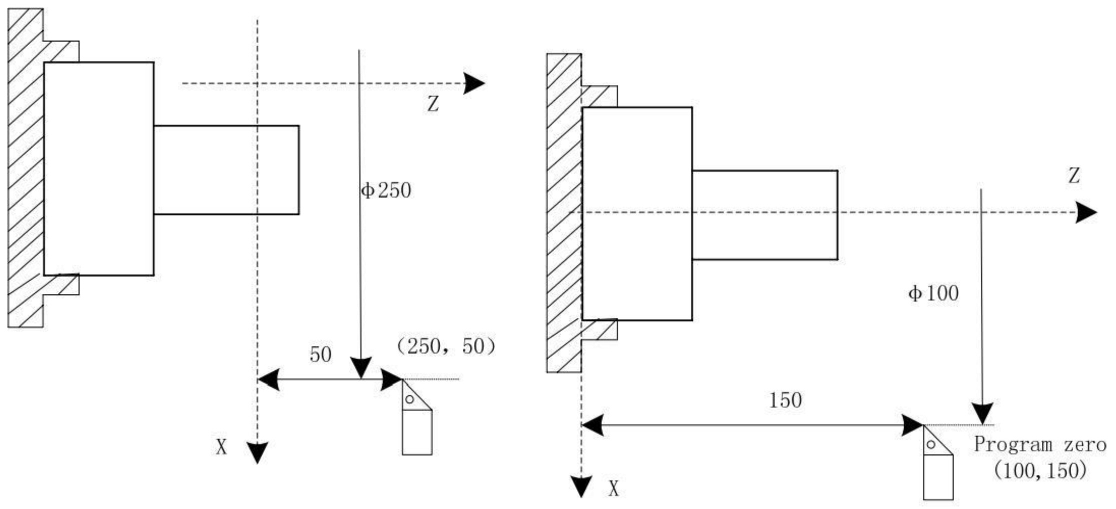
U: Different value between the new absolute coordinates of current position in X direction and the absolute coordinates before executing commands;

Z: New absolute coordinates of current position in Z direction;

W: Different value between the new absolute coordinates of current position in X direction and the absolute coordinates before executing commands;

In G50, when X(U) or Z(W) are not input, the system does not change current coordinates position as program zero; when X(U) and Z(W) are not input, the system takes the previous setting position as program zero.

Example:



Before setting coordinate system with G50　　After setting coordinate system with G50

create the above-mentioned workpiece coordinate system and set (X100 Z150) to the reference point of program after executing "G50 X100 Z150".

Note 1: When setting the coordinate system with G50, it should be done in the state of tool deviation cancellation, and the absolute coordinate after setting is the G50 set value; The tool deviation cancellation can be executed in MDI mode: "T0100 G00 U0 W0". Assuming the current tool deviation status is T0101.

Note 2: When using G50 to set the coordinate system in the state of tool deviation, the absolute coordinate display has the following two situations:

A、The tool deviation has been executed (there is a movement command after the tool deviation), and the absolute coordinate after setting is the G50 set value. The following table shows:

Program (execute tool compensation by coordinate offset)	Absolute coordinate display value	01 # cutter compensation value
G00 X0 Z0	X0　Z0	X-12　Z-23

T0101	X12 Z23	
G00 X0 Z0	X0 Z0	
G50 X20 Z20	X20 Z20	

B、The tool deviation has not been executed yet (there is no movement command after the tool deviation), including canceling the tool deviation and setting the tool deviation. The absolute coordinates after setting reflect the tool deviation value. The following table shows:

Program (execute tool compensation by coordinate offset)	Absolute coordinate display value	01 # cutter compensation value
G00 X0 Z0	X0 Z0	X-12 Z-23
T0101	X12 Z23	
G50 X50 Z50	X50 Z50	
T0100	X38 Z27	
G50 X20 Z20	X8 Z-3	

Program (execute tool compensation by coordinate offset)	Absolute coordinate display value	01 # cutter compensation value
G00 X0 Z0	X0 Z0	X-12 Z-23
T0101	X12 Z23	
G50 X20 Z20	X32 Z43	

2) Coordinate system translation

Command format:G50 U_ W_ ;

Command function: According to the above code, translate the position of the tool tip on the tool holder to a distance specified by a parameter in the original absolute coordinate system. The position of the new coordinate tool tip relative to the original absolute coordinate system is X+U, Z+W.

Command specifications: When the parameter is set to diameter programming, the X direction is specified as the diameter, and when the parameter is set to radius programming, the X direction is specified as the radius.

4.2.8 Skip Interpolation G31

Command format: G31 X(U)_ Z(W)_ F_;

Command function: During the execution of the code, if an external jump signal is input, the execution of the code is interrupted and the next program segment is executed. This function can be used for dynamic measurement of workpiece size (such as grinding machines), tool setting measurement, etc.

Command specifications: 1. non-modal G command (00 group);

2. The address format is consistent with G01 code, and the usage is also similar.

3. Before using this code, the tool tip radius compensation needs to be revoked;

4. Feedrate should not be set to too big to get the precise stop position;

a .Following Block Execution After Skip

1. The next block of G31 is the incremental coordinate programming shown in Fig. 4-2-9-1

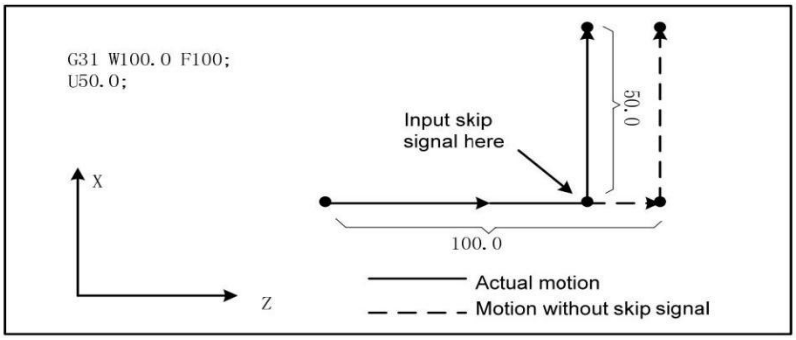


Fig. 4-2-9-1

2. The next block of G31 is the absolute coordinate programming of one axis as Fig.4-2-9-2

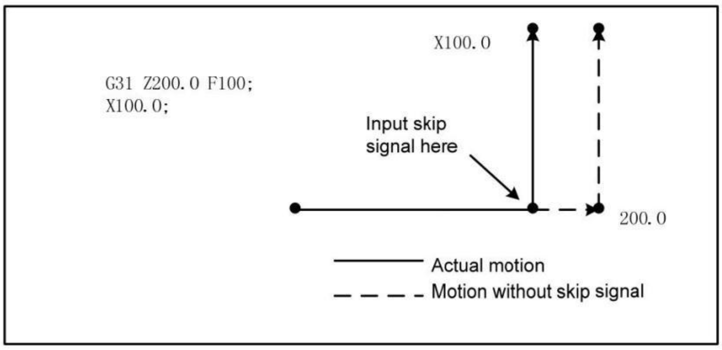


Fig. 4-2-9-2

3. The next block of G31 is the absolute coordinate programming of two axes shown in Fig4-2-9-3

Program: G31 Z200 F100

G01 X100 Z300

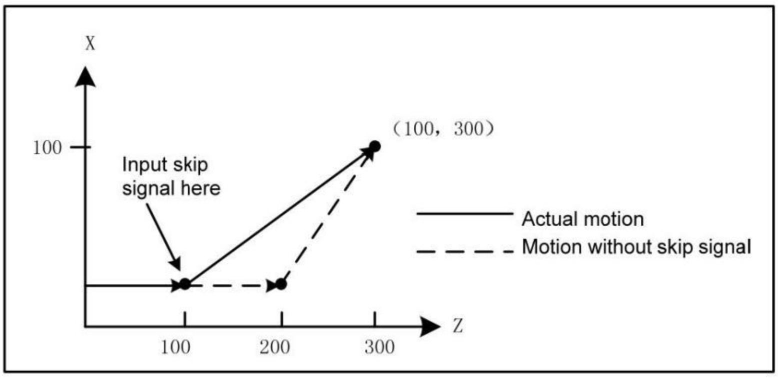


Fig. 4-2-9-3

b. Signals Relevant to G31

Skip signal:

SKIP: X2.0

Type: input signal

Function: X2.0 ends the skip cutting. I.e. in a block containing G31, the skip signal becoming the absolute coordinate position of "1" is to be stored in the macro variable (#5011～#5013 separately corresponds to X, Z, Y)

Operation: when the skip signal becomes "0", CNC executes as follows: When the block is executing G31, CNC stores the current absolute coordinates of each axis. CNC stops G31 to execute the next block, the skip signal detects its state instead of its RISING EDGE. So when the skip signal is "1", it meets the skip conditions.

Note: If G31 is not used, X2.0 input interface is used to the common input interface. The skip signal is valid, CNC immediately stops the feed axis (without acceleration/deceleration execution), and G31 feedrate should be as low as possible below 1000 mm/min to get the precise stop position.

4.2.9 Machine 1st Reference Point G28

Command format: G28 X(U) Z(W) ;

Command function: the tool rapid traverses to the middle point defined by X(U), Z(W) from starting point and then return to the machine zero.

Command specifications: G28 is non-modal. X, Z, Y: absolute coordinates of middle point; U, W, V: Z absolute coordinates of middle point; W: Difference value of absolute coordinates between middle point and starting point in Z direction. Omit all or one of X(U), Z(W) as follows:

Command	Function
G28 X(U) __	X returns to machine zero and Z axis remains in the previous position
G28 Z(W) __	Z returns to machine zero and X axis remains in the previous position
G28	in the previous positions and continuously execute the next block
G28 X(U) __ Z(W) __	X, Z return to machine zero simultaneously

Running path(as Fig.4-2-10-1):

(1) Rapid traverse to middle point of specified axis from current position(A point→B point) ;

(2) Rapid traverse to reference point from the middle point(B point→R point) ;

(3) If the machine is not locked, LED is ON when the machine reference point return is completed.

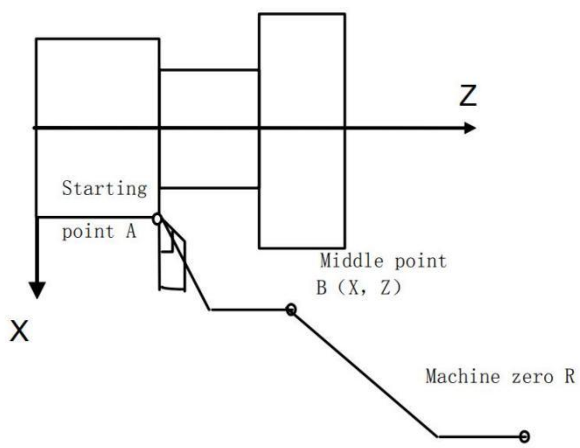


Fig. 4-2-10-1

Note 1: Do not execute G28 and machine zero return without the zero switch on the machine.
Note 2: Machine zero returns in Jog mode and in G28 are the same and their deceleration signals and the signal every rotation must be detected;
Note 3: X and Z move at the respectively rapid traverse speed from A to B and from B to R, and so the path is not always a straight line;
Note 4: The system cancels the tool length compensation after executing G28 to perform the machine zero return;

4.2.10 Machine 2nd, 3rd, 4th Reference Point G30

Command format: G30 P2 X(U) __ Z(W) __;
G30 P3 X(U) __ Z(W) __;
G30 P4 X(U) Z(W) ;

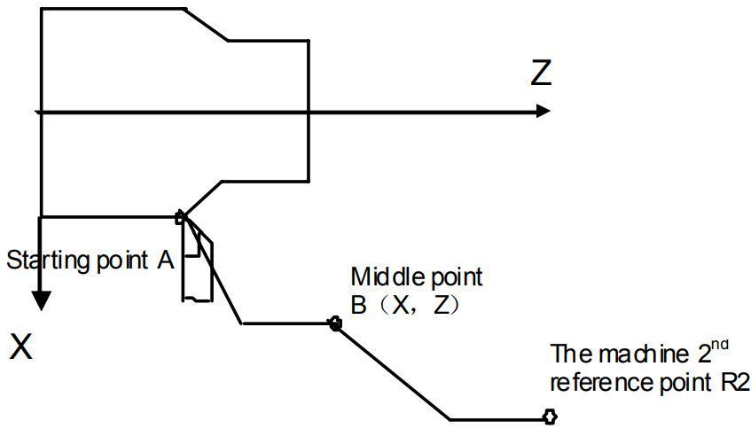
Command function: the tool rapidly traverses with the rapid traverse speed to the middle point specified by X(U) , Z(W)

Command specifications: G30 is non-modal. X: X absolute coordinate of the middle point; U: difference value of X absolute coordinate value between the middle point and starting point; Z: Z absolute coordinate of the middle point; W: difference point of Z absolute coordinate between the middle point and starting point. Omit one or all of X (U) , Z(W) as follows:

Command	Function
G30 Pn X(U)	X returns to the machine nth reference point, Z axis retains
G30 Pn Z(W)	Z return to the nth machine reference point, X axis retains
G30	X and Z retain, go on executing the next program block
G30 Pn X(U) Z(W)	X and Z return to the machine nth reference point simultaneously

Note 1: n in the above table is 2, 3 or 4;
Note 2: Do not check the deceleration, zero signal when you execute the machine 2nd, 3rd, 4th reference point.

Command operations: (taking example of returning to machine 2nd reference point as follows):
(1) Rapidly traverse to the middle position of command axis from the current position (A point →B point);
(2) Traverse from the middle point with the speed set by No.113 to the 2nd reference point set by No.122 and No.123 (B point →R2 point);
(3) When CNC is not in the machine lock state, the completion signal of reference point return ZP21 Bit0, Bit1 is high.



Note 1: Execute the machine 2nd, 3rd, 4th reference point return after you manually execute the machine reference point return or G28 (machine reference point return).
Note 2: A→B and B0→R2, two axes separately traverse, and so their trails are linear or not.
Note 3: CNC cancels the tool length compensation after you execute G30 to return 2nd, 3rd, and 4th reference point.
Note 4: Must not execute G30 (machine 2nd, 3rd, 4th reference point return) when the zero switch is not installed on the machine. Note 5: Do not set the workpiece coordinate system when you execute the 2nd, 3rd, and the machine 4th reference point return.

Fixed Cycle Command

To simplify programming, the system defines G command of single machining cycle with one block to complete the rapid traverse to position, linear/thread cutting and rapid traverse to return to the starting point:

G90: axial cutting cycle;

G92: thread cutting cycle;

G94: radial cutting cycle;

G92 will be introduced in section Thread Function.

4.2.11 Axial Cutting Cycle G90

Command format: G90 X(U) __ Z(W) __ F__; (cylinder cutting)

G90 X(U) __ Z(W) __ R__ F__; (taper cutting)

Command function: From starting point, the cutting cycle of cylindrical surface or taper surface is completed by radial feeding(X) and axial (Z or X and Z) cutting.

Command specifications:

G90 is modal;

Starting point of cutting: starting position of linear interpolation (cutting feed)

End point of cutting: end position of linear interpolation (cutting feed)

X: X absolute coordinates of cutting end point

U: Different value of X absolute coordinate between end point and starting point of cutting
Z: Different value of Z absolute coordinate between end point and starting point of cutting
W: Different value of Z absolute coordinate between end point and starting point of cutting
R: Different value (radius value) of X absolute coordinates between end point and start point of cutting. When the signs of R is not the same that of U, R|≤|U/2|; when R=0 or the input is default, the cylinder cutting is executed as Fig. 4-2-13-1, otherwise, the cone cutting is executed as Fig. 4-2-13-2

Cycle process:
① X rapidly traverses from starting point to cutting starting point;
② Cutting feed (linear interpolation) from the cutting starting point to cutting end point;
③ X executes the tool retraction at feedrate (opposite direction to the above-mentioned ①), and return to the position which the absolute coordinates and the starting point are the same;
④ Z rapidly traverses to return to the starting point and the cycle is completed.

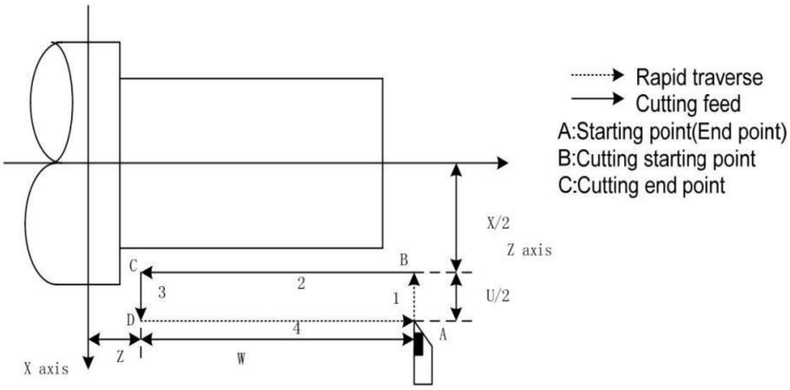


Fig. 4-2-13-1

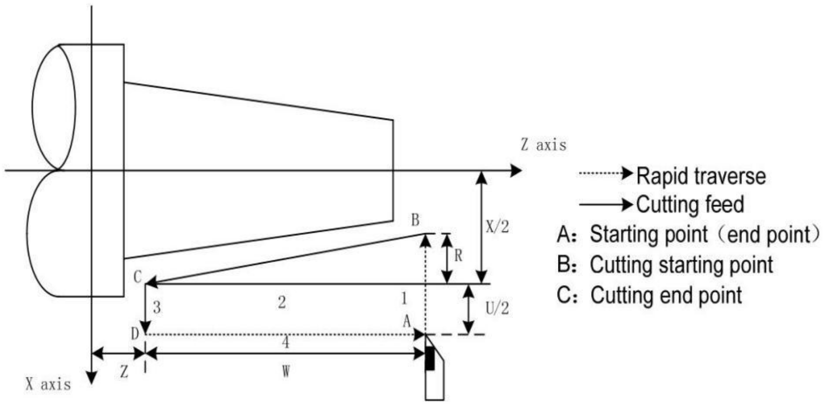


Fig. 4-2-13-2

Cutting path: Relative position between cutting end point and starting point with U, W, R, and tool path of U, W, R with different signs are shown in Fig 4-2-13-3。

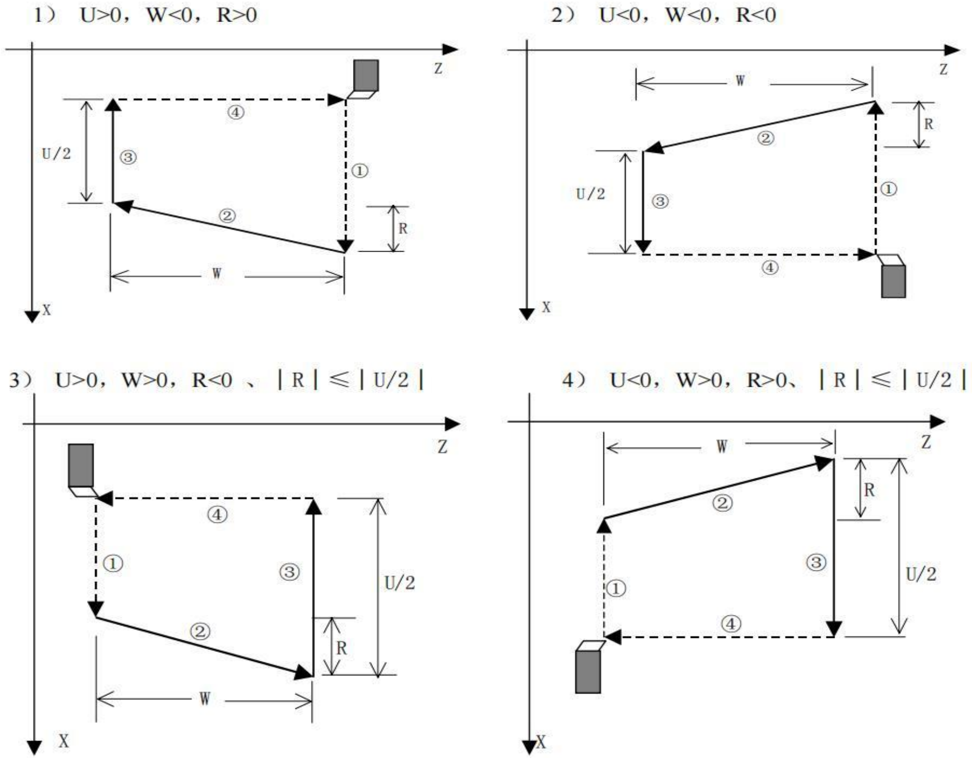


Fig. 4-2-13-3

Example: Fig. 4-2-13-4 rod Φ125×110

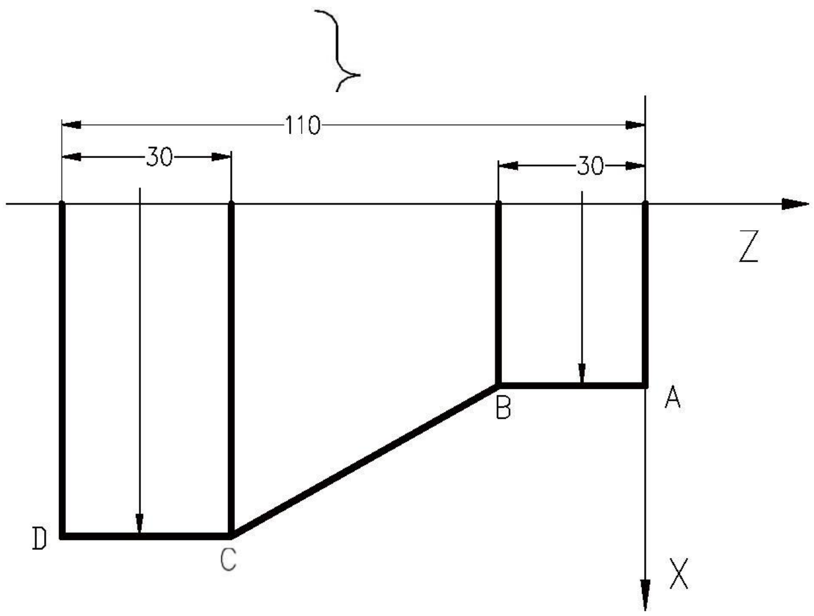


Fig. 4-2-13-4

Program：02138；

M3 S300 G0 X130 Z3；

G90 X120 Z-110 F200； (A→D，cut Φ120)

X110 Z-30；

X100； (A→B，6 times cutting cycle Φ60, increment of 10mm)

X90；

X80；

X70；

X60；

```
G0 X120 Z-30;
G90 X120 Z-44 R-7.5 F150;
Z-56 R-15
Z-68 R-22.5                              (B→C, 4 times taper cutting);
Z-80 R-30
M30;
```

4.2.12 Radial Cutting Cycle G94

Command format: G94 X(U) __ Z(W) __ F__; (face cutting)

G94 X(U) __ Z(W) __ R__ F__; (taper face cutting)

Command function: From starting point, the cutting cycle of cylindrical surface or taper surface is completed by radial feeding(X) and axial (Z or X and Z) cutting.

Command specifications:

G94 is modal;

Starting point of cutting: starting position of linear interpolation (cutting feed). Unit: mm;

End point of cutting: end position of linear interpolation (cutting feed). Unit: mm;

X: X absolute coordinate of end point of cutting. Unit: mm;

U: Different value of absolute coordinate from end point to starting point of cutting in X direction .Unit: mm;

Z: Z absolute coordinates of end point of cutting, Unit: mm;

W: Different value of X absolute coordinate from end point to starting point of cutting, Unit: mm;

R: Different value(R value) of X absolute coordinates from end point to starting point of cutting. When the sign of R is not the same as that of U, R, ∣R∣≤∣W∣. Radial linear cutting is shown in Fig. 4-2-14-1, radial taper cutting is as Fig. 4-2-14-2. Ranges of X, U, Z, W, R are referred to Table 1-2 of Section 1.4.1, unit: mm/inch. **Cycle process:**

① Z rapidly traverses from starting point to cutting starting point;

② Cutting feed (linear interpolation) from the cutting starting point to cutting end point;

③ Z executes the tool retraction at the cutting feedrate (opposite direction to the above-mentioned ①), and returns to the position which the absolute coordinates and the starting point are the same;

④ The tool rapidly traverses to return to the starting point and the cycle is completed.

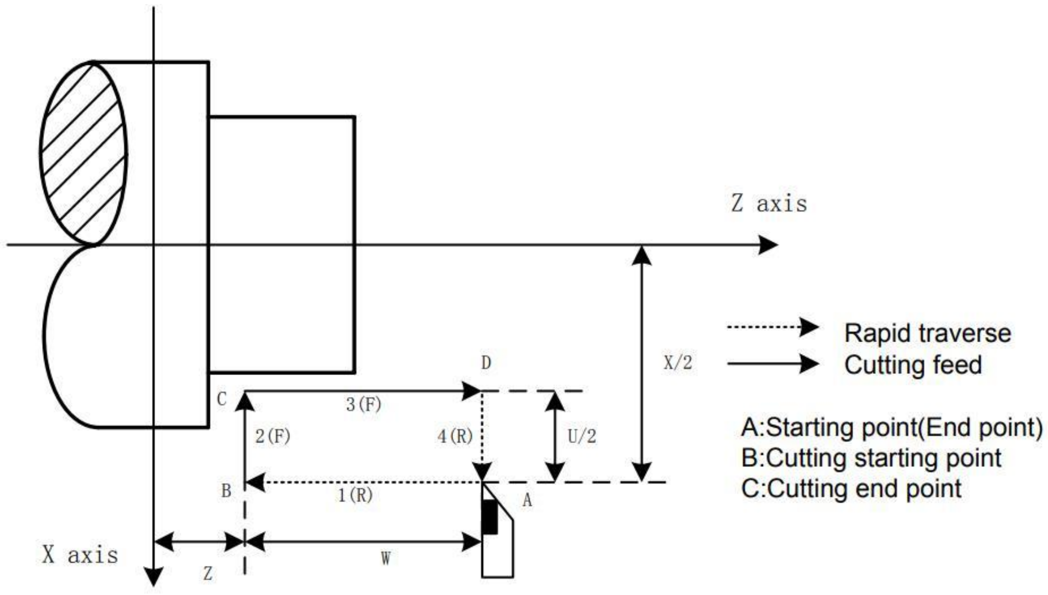


Fig. 4-2-14-1

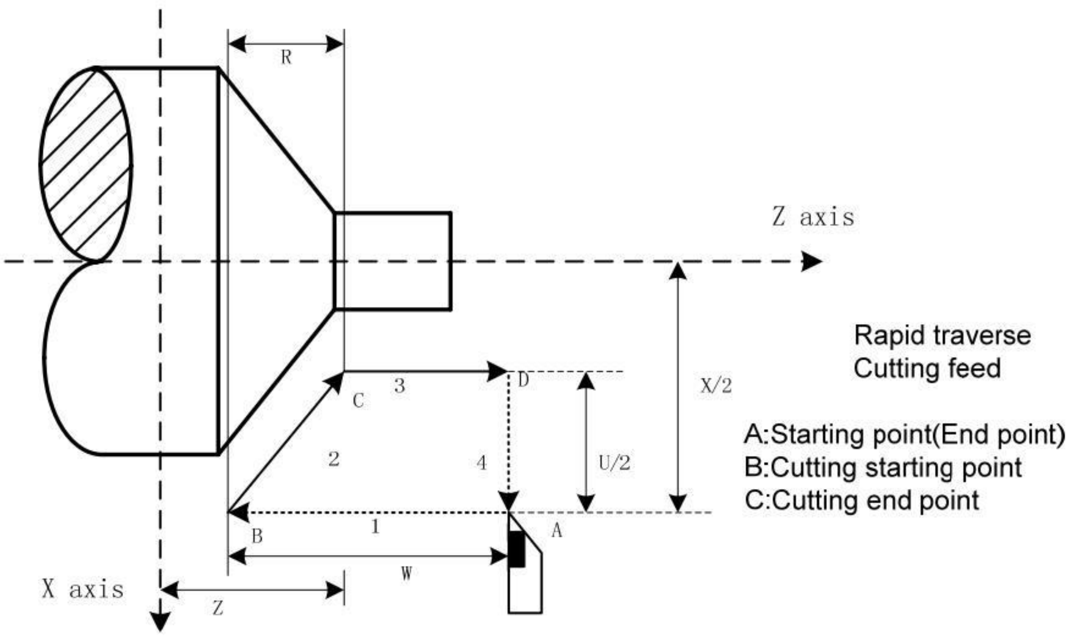


Fig. 4-1-14-2

Code trajectory: Cutting path: Relative position between cutting end point and starting point with U, W is shown in 4-2-14-3:

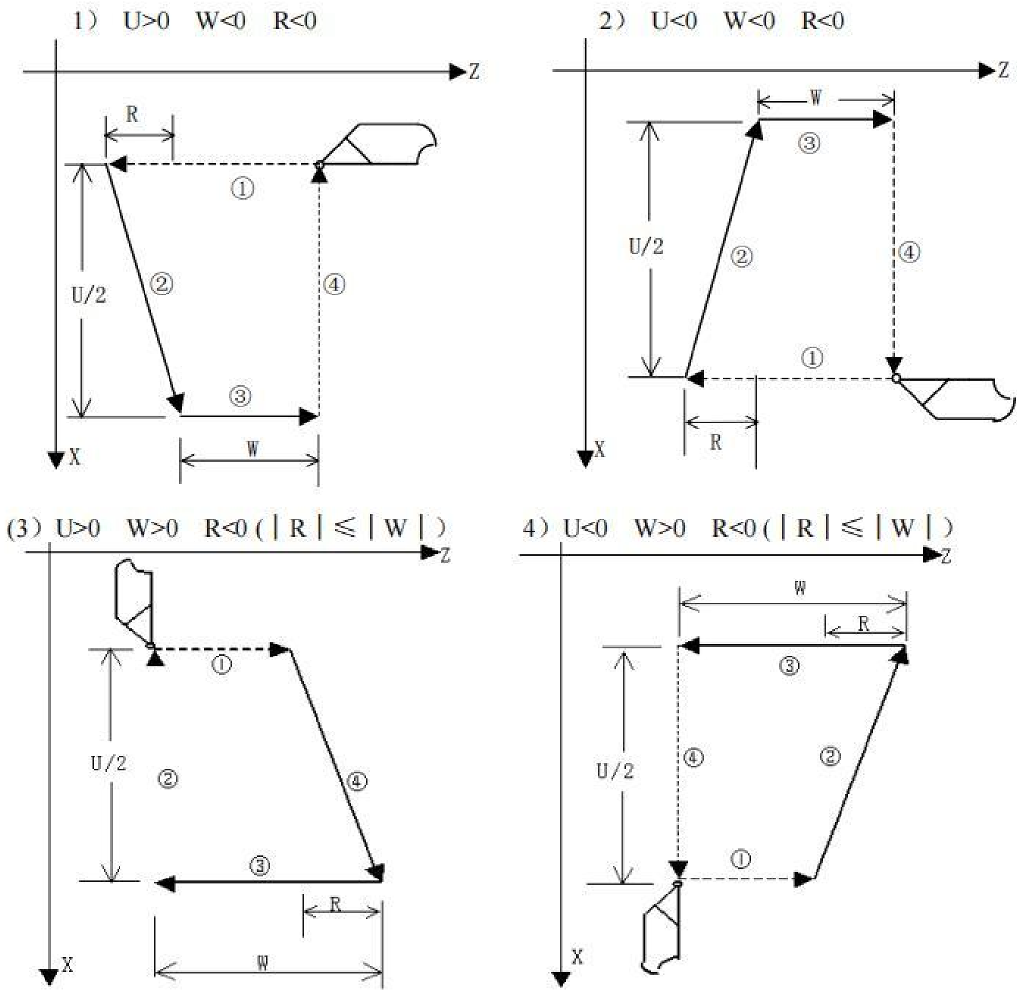


Fig. 4-2-14-3

Example: Fig.4-2-14-4, rod Φ125×112

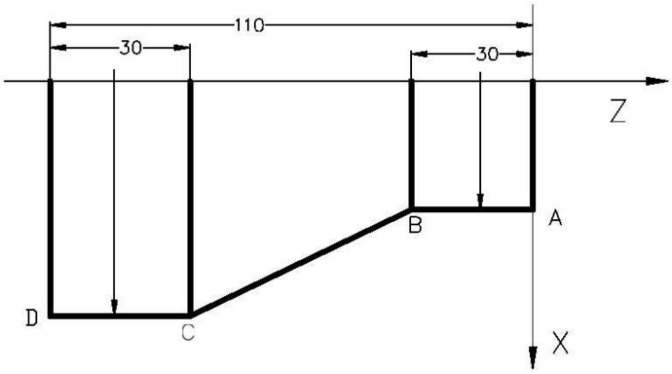


Fig. 4-2-14-4

Program:06212;

```
G00 X130 Z5 M3 S1;
G94 X0 Z0 F200          End face cutting
X120 Z-110 F300;        (Outer cutting Φ120)
G00 X120 Z0
G94 X108 Z-30 R-10
X96 R-20
```

```
X84 R-30                (C→B→A, cutting Φ60)
X72 R-40
X60 R-50;
M30;
```

4.2.13 Axial Roughing Cycle G71

G71 has two kinds of roughing cycle: type I and type II.

Command format: G71 U(Δd) R(e) F S T ; (1)

G71 P(ns) Q(nf) U(Δu) W(Δw) K0/1 J0/1;

```
N(ns) G0/G1 X(U)..;                    N(ns) G0/G1 X(U) Z(W)...;
. . . . . . . . . ;                    . . . . . . . . . ;
 . . . . F;                             . . . . F;
 . . . . S;        (3)Type I                 . . . . S;        (3)Type II
 . . . .                                . .
N(nf). . . . . ;                        N(nf). . . . . ;
```

Command function: G71 is divided into three parts:

(1) 1st blocks for defining the travels of tool infeed and retract tool, the cutting feedrate, the spindle speed and the tool function when roughing;

(2) 2nd blocks for defining the block interval, finishing allowance;

(3) 3rd blocks for some continuous finishing path, counting the roughing path without being executed actually when executing G71.

According to the finishing path, the finishing allowance, the path of tool infeed and tool retract, the system automatically counts the path of roughing, the tool cuts the workpiece in paralleling with Z, and the roughing is completed by multiple executing the cutting cycle tool infeed→cutting→tool retraction. The starting point and the end point are the same one. The command is applied to the formed roughing of non-formed rod.

Relevant definitions:

Finishing path: The above-mentioned Part 3 of G71(ns～nf block)defines the finishing path, and the starting point of finishing path (starting point of ns block)is the same these of starting point and end point of G71, called A point; the first block of finishing path(ns block)is used for X rapid traversing or tool infeed, and the end point of finishing path is called to B point; the end point of finishing path(end point of nf block)is called to C point. The finishing path is A→B→C.

Finishing path: The above-mentioned Part 3 of G71(ns～nf block)defines the finishing path, and the starting point of finishing path (starting point of ns block)is the same these of starting point and end point of G71, called A point; the first block of finishing path(ns block)is used for X rapid traversing or tool infeed, and the end point of finishing path is called to B point; the end point of finishing path(end point of nf block)is called to C point. The finishing path is A→B→C.

Δd: It is each travel of X tool infeed in roughing, its value: 0.001~99.999 (IS_B) /0.0001~99.9999 (IS_C) (unit: mm/inch, radius value) without sign, and the direction of tool infeed is defined by move direction of ns block. The command value Δd is reserved after executing U(Δd). The value of system parameter No.051 is regarded as the travel of tool infeed when U(Δd) is not input.

e: It is travel of X tool retraction in roughing its value: 0~99.999 (IS_B) /0~99.9999 (IS_C) (unit: mm/inch, radius value) without sign, and the direction of tool retraction is opposite to that of tool infeed, the command value e is reserved after R(e) is executed. The value of system parameter No.052 is regarded as the travel of tool retraction when R(e) is not input.

ns: Block number of the first block of finishing path.

nf: Block number of the last block of finishing path. I5T3 I5T5 Turning CNC System User Manual 108 Ⅰ Programming

Δu: X finishing allowance is -99999.999~99999.999(IS_B) /-9999.9999~9999.9999(IS_C) (diameter, unit: mm/inch, with sign). X coordinate offset of roughing path compared to finishing path, i.e. the different value of X absolute coordinates between A’ and A. The system defaults Δu=0 when U(Δu) is not input, i.e. there is no finishing allowance in X direction for roughing cycle.

Δw: Z finishing allowance is -99999.999~99999.999(IS_B) /-9999.9999~9999.9999(IS_C) (diameter, unit: mm/inch, with sign). the Z coordinate offset of roughing path compared to finishing path, i.e. the different value of Z absolute coordinate between A’ and A. The system defaults Δw=0 when W(Δw) is not input, i.e. there is no Z finishing allowance for roughing cycle.

K: When K is not input or is not 1, the system does not check the program monotonicity except that the Z value of starting point and end point of the arc or ellipse or parabola or the arc is more than 180 degree; K=1, the system checks the program monotonicity.

F: Feedrate; S: Spindle speed; T: Tool number, tool offset number. M, S, T, F: They can be specified in the first G71 or the second ones or program ns～nf. M, S, T, F functions of M, S, T, F blocks are invalid in G71, and they are valid in G70 finishing blocks.

Type I:

1) **Execution process:** Fig.4-2-15-1。

① X rapidly traverses to A’ from A point, X travel is Δu, and Z travel is Δw;

② X moves from A’ is Δd(tool infeed), ns block is for tool infeed at rapid traverse speed with G0, is for tool infeed at feedrate F with G71, and its direction of tool infeed is that of A→B point;

③ Z executes the cutting feeds to the roughing path, and its direction is the same that of Z coordinate A→B point;

④ X, Z execute the tool retraction e (45° straight line) at feedrate, the directions of tool retraction is opposite to that of too infeed;

⑤ Z rapidly retracts at rapid traverse speed to the position which is the same that of Z coordinate;

⑥ After executing X tool infeed (Δd+e)again, the end point of traversing tool is still on the middle point of straight line between A’ and B’(the tool does not reach or exceed B’), and after executing the tool infeed (Δd+e)again, execute ; ③ after executing the tool infeed (Δd+e)again, the end point of tool traversing reaches B’ point or exceeds the straight line between A’ →B’ point and X executes the tool infeed to B’ point, and then the next step is executed;

⑦ Cutting feed from B’ to C’ point along the roughing path;

⑧ Rapid traverse to A from C’ point and the program jumps to the next clock following nf block after G71 cycle is ended.

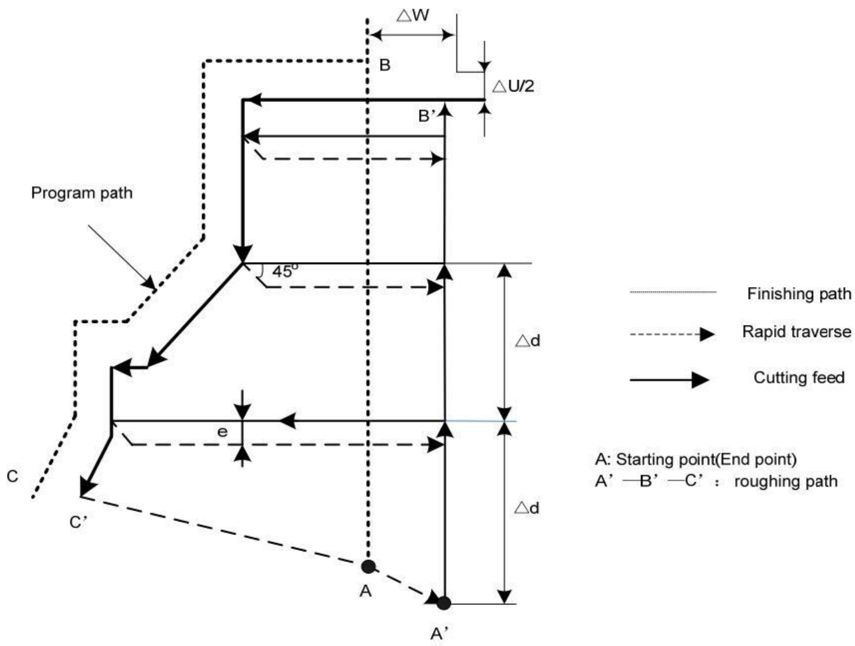


Fig. 4-2-15-1 G71 cycle path

2) Coordinate offset direction with finishing allowance:

Δu, Δw define the coordinate offset and cut-in direction in finishing, and their sign symbol are as follows Fig. 4-2-15-2: B→C for finishing path, B’ →C’ for roughing path and A is the tool start-up point.

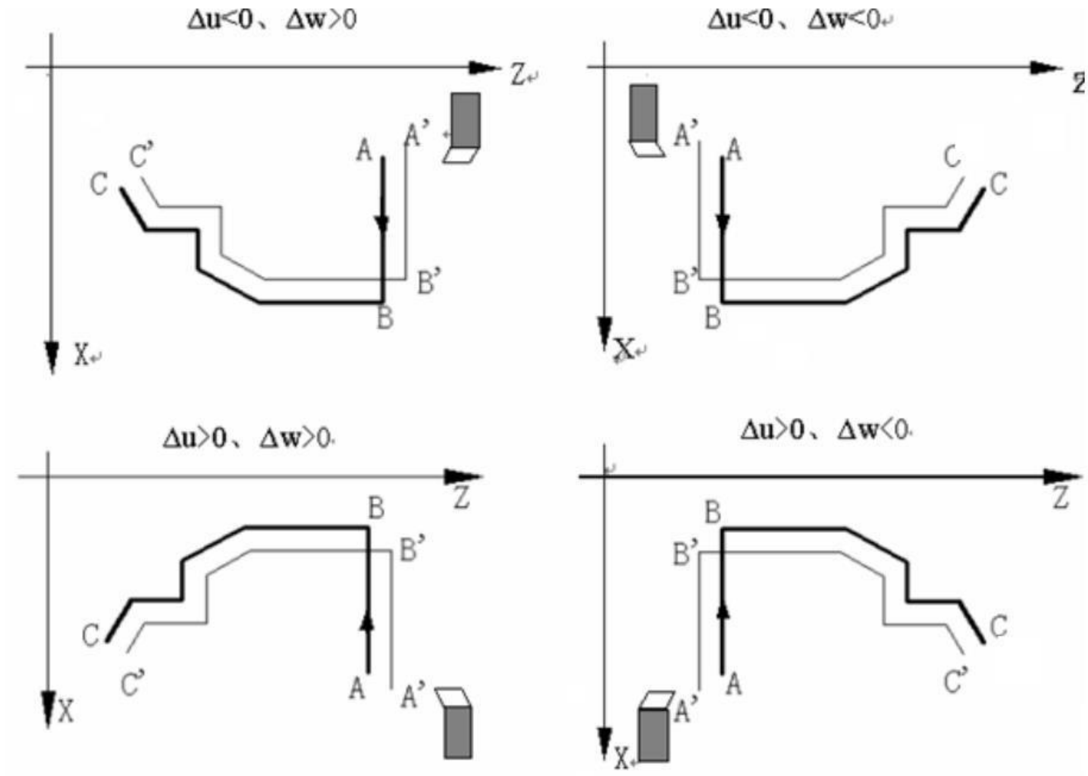


Fig. 4-2-15-2

Type II:

The type II is different from the type I as follows:

1) Relative definition: More one parameter than the type I.

J: When J is not input or J is not 1, the system does not execute the run along the roughing contour; J=1: the system executes the run along the roughing contour.

2) The system does not execute the monotonous increasing or the monotonous decreasing along X external contour, and the workpiece can be up to 10 grooves as follows:

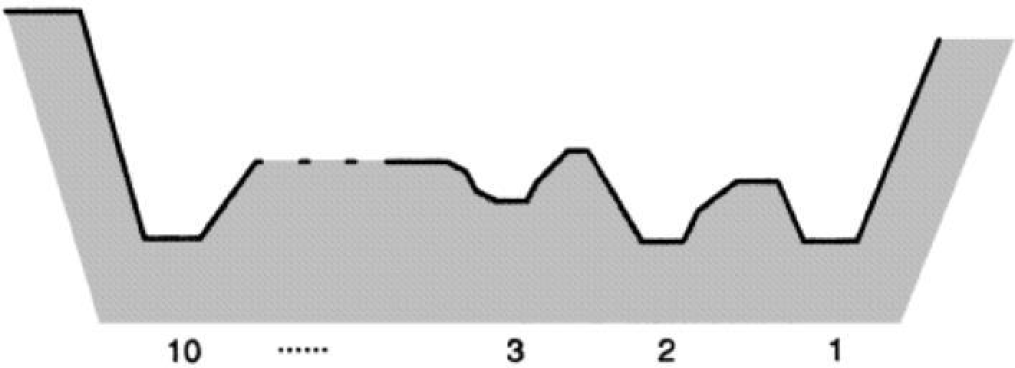


Fig. **4-2-15-3（type Ⅱ）**

But, the Z external contour must be the monotonous increasing or the monotonous decreasing, and the following contour cannot be machined:

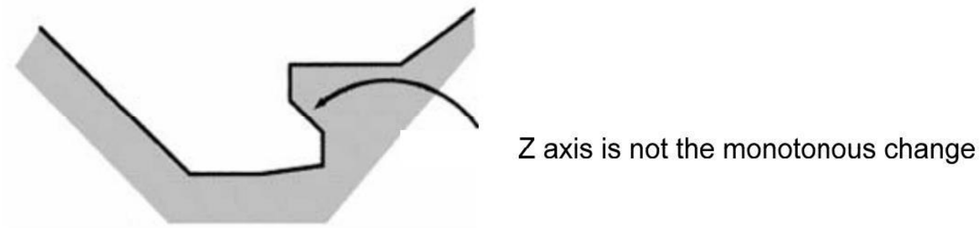


Fig. **4-2-15-4**（type Ⅱ）

3）The first tool cutting need not the vertical: The machining can be executed when Z is the monotonous change shape as follows:

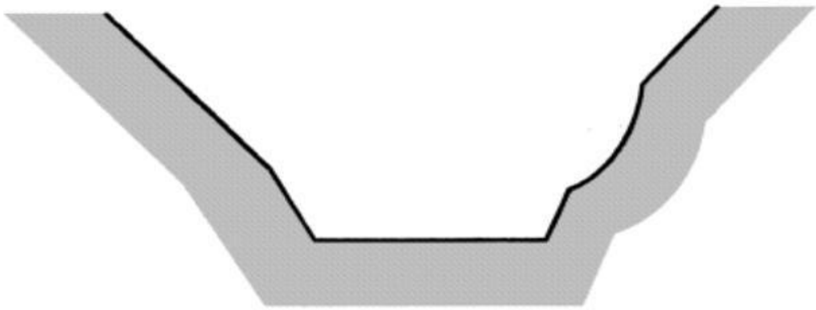


Fig. **4-2-15-5**（type Ⅱ）

4）After the turning, the system should execute the tool retraction, the retraction travel is specified by R (e) or No.40 as follows:

e is set by parameter

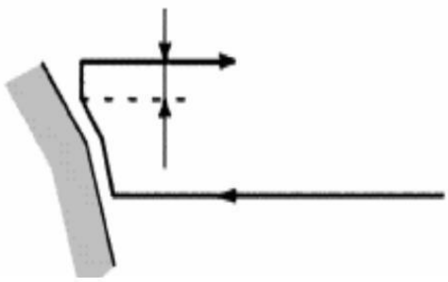


Fig. **4-2-15-6**（type Ⅱ）

5）**Command execution process：** roughing path A->H

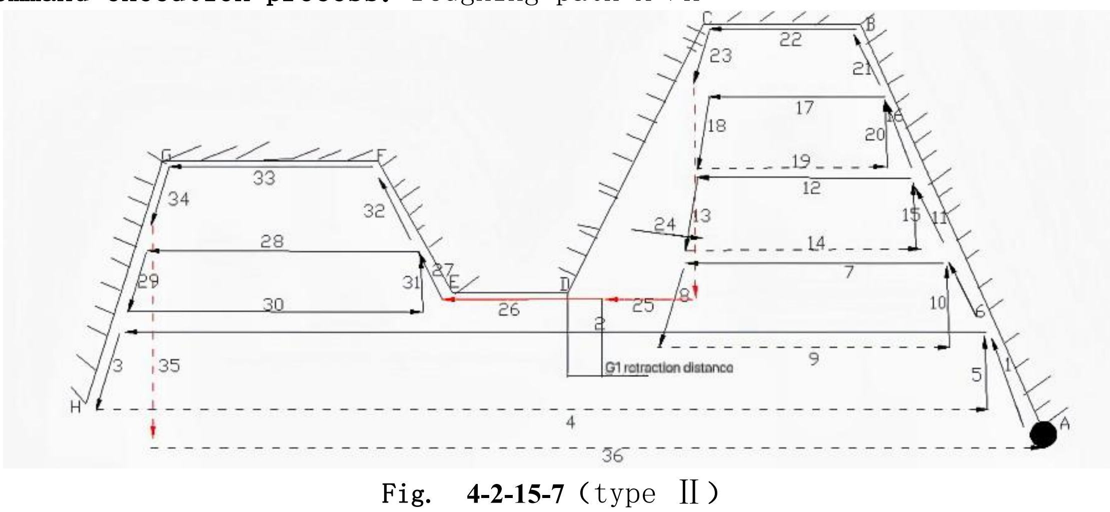


Fig. **4-2-15-7**（type Ⅱ）

Notes:

1. ns block is only G00, G01. When the workpiece is type II, the system must specify the two axes X(U) and Z(W), and W0 must be specified when Z does not move;
2. For type II, only X finishing allowance can be specified; when Z finishing allowance is specified, the whole machining path offsets, and it can be specified to 0;
3. For type II, after the current grooving is completed to execute the next, the tool approaches the workpiece(remark 25 and 26) in the remainder tool retraction distance at G1 speed; when the tool retraction is 0 or the remainder distance is less than the tool retraction, and the tool approaches the workpiece at G1 speed;
4. Some workpiece without remarking the type I or the type II adapts the both;
5. For the finishing path(ns～nf block), Z dimension must be monotonous change(always increasing or decreasing), X dimension in the type I must be monotonous change and does not need in the type II;
6. ns～nf blocks in programming must be followed G71 blocks. If they are in front of G71 blocks, the system automatically searches and executes ns～nf blocks, and then executes the next program following nf block after they are executed, which causes the system executes ns～nf blocks repetitively;
7. ns～nf blocks are used for counting the roughing path and the blocks are not executed when G71 is executed. F, S, T commands of ns～nf blocks are invalid when G71 is executed, at the moment, F, S, T commands of G71 blocks are valid. F, S, T of ns～nf blocks are valid when executing ns～nf to command G70 finishing cycle;
8. In ns～nf blocks, there are only G commands: G00, G01, G02, G03, G04, G05, G6.2, G6.3, G7.2, G7.3, G96, G97, G98, G99, G40, G41, G42 and the system cannot call subprograms (M98/M99);
8. In ns～nf ,the program block quantity cannot exceed 100;
9. G96, G97, G98, G99, G40, G41, G42 are invalid when G71 is executed, and are valid when G70 is executed;
10. When G71 is executed, the system can stop the automatic run and manual traverse, but return to the position before manual traversing when G71 is executed again, otherwise, the following path will be wrong;
11. When the system is executing the feed hold or single block, the program pauses after the system has executed end point of current path;
12. d △ , u are specified by the same U and differ △ ent with or without being specified P, Q commands;
13. There are no the same block number in ns˜nf when compound cycle commands are executed repetitively in one program;
15. The tool retraction point should be high or low as possible to avoid crashing the workpiece.

Example: 4-2-15-8 (Type I)

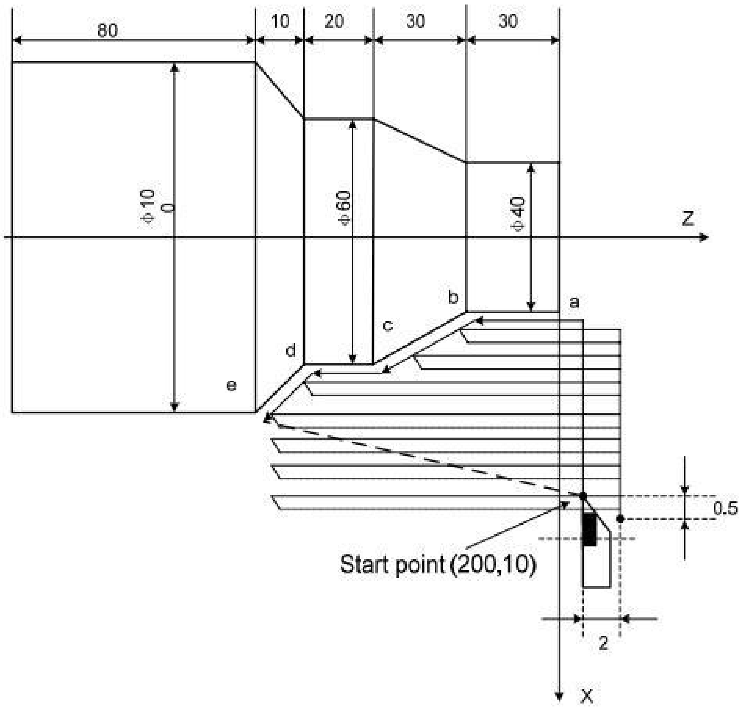


Fig. **4-2-15-8**

Program:：O2158；

G00 X200 Z10 M3 S800； (Spindle clockwise with 800 r/min)

G71 U2 R1 F200； (Cutting depth each time 4mm, tool retraction 2mm [in diameter])

G71 P80 Q120 U1 W2； (roughing a---e, machining allowance: X, 1mm;Z, 2mm)

N80 G00 X40 S1200； (Positioning)

G01 Z-30 F100 ； (a→b)

X60 W-30； (b→c) a→b→c→d→e blocks for finishing path

W-20； (c→d)

N120 X100 W-10； (d→e)

G70 P80 Q120； (a---e blocks for finishing path)

M30； (End of block)

4.2.14 Radial Roughing Cycle G72

Command format: G72 W(Δd) R(e) F S T ； (1)

G72 P(ns) Q(nf) U(Δu) W(Δw)； (2)

N (ns) . . . . . ;

. . . . ;

. . . . F;

. . . . S;

. . . . ; (3)

N (nf). . . . . ;

Command function: G72 is divided into three parts:

(1): 1st blocks for defining the travels of tool infeed and tool retraction, the cutting speed, the spindle speed and the tool function in roughing;

⑵: 2nd blocks for defining the block interval, finishing allowance;

⑶: 3rd blocks for some continuous finishing path, counting the roughing path without being executed actually when G72 is executed.

According to the finishing path, the finishing allowance, the path of tool infeed and retract tool, the system automatically counts the path of roughing, the tool cuts the workpiece in paralleling with Z, and the roughing is completed by multiple executing the cutting cycle tool infeed→cutting feed→tool retraction. The starting point and the end point of G72 are the same one. The command is applied to the formed roughing of non-formed rod.

Relevant definitions:

Finishing path: the above-mentioned Part ⑶ of G71(ns～nf block)defines the finishing path, and the starting point of finishing path (i.e. starting point of ns block)is the same these of starting point and end point of G72, called A point; the first block of finishing path(ns block)is used for Z rapid traversing or cutting feed, and the end point of finishing path is called to B point; the end point of finishing path(end point of nf block)is called to C point. The finishing path is A→B→C.

Roughing path: The finishing path is the one after offsetting the finishing allowance Δu, Δw) and is the path contour formed by executing G72. A, B, C point of finishing path after offset corresponds separately to A', B', C' point of roughing path, and the final continuous cutting path of G72 is B'→C' point.

Δd: it is Z cutting in roughing, its value: 0.001~99.999 (IS_B) /0.0001~99.9999 (IS_C) (unit: mm/inch) without sign symbol, and the direction of tool infeed is determined by ns block traverse direction. The value of system parameter No.051 is regarded as the tool infeed clearance when W(Δd) is not input.

e: it is Z tool retraction clearance in roughing, its value: 0~99.999 (IS_B) /0~99.9999 (IS_C) (unit: mm) without sign symbol, and the direction of tool retraction is opposite to that of tool infeed, the specified value e is reserved after R(e) is executed. The value of system parameter No.052 is regarded as the tool retraction clearance when R(e) is not input.

ns: Block number of the first block of finishing path.

nf: Block number of the last block of finishing path.

Δu : it is X finishing allowance in roughing, its range: -99999.999~99999.999(IS_B) /-9999.9999~9999.9999(IS_C) (X coordinate offset of roughing contour corresponding to the finishing path, i.e. X absolute coordinate difference between A' and A.(diameter, unit: mm/inch, with sign symbol).

Δw : it is Z finishing allowance in roughing, its range: -99999.999~99999.999(IS_B) /-9999.9999~9999.9999(IS_C) (Z coordinate offset of roughing contour corresponding to the finishing path, i.e. Z absolute coordinate difference between A' and A.(diameter, unit: mm/inch, with sign symbol).

When K is not input or is not 1, the system does not check the program monotonicity except that the Z value of starting point and end point of the arc or ellipse or parabola or the arc is more than 180 degree; K=1, the system checks the program monotonicity.

F: Cutting feedrate;

S: Spindle speed;

T: Tool number, tool offset number.

M, S, T, F: They can be specified in the first G72 or the second ones or program ns～nf. M, S, T, F functions of M, S, T, F blocks are invalid in G72, and they are valid in G70 finishing blocks.

Execution process: ①X rapidly traverses to A' from A point, X travel is Δu, and Z travel is Δw;

②X moves from A' is Δd(tool infeed), ns block is for tool infeed at rapid traverse speed with G0, is for tool infeed at G72 feedrate F in G1, and its direction of tool infeed is that of A→B point;

③X executes the cutting feeds to the roughing path, and its direction is the same that of X coordinate B→C point;

④X, Z execute the tool retraction e (45° straight line)at feedrate, the directions of tool retraction is opposite to that of tool infeed ;

⑤X rapidly retracts at rapid traverse speed to the position which is the same that of Z coordinate;

⑥After Z tool infeed (Δd+e)again is executed, the end point of traversing tool is still on the middle point of straight line between A' and B' (the tool does not reach or exceed B'), and after Z executes the tool infeed (Δd+e)again, is executed; ③ after the tool infeed (Δd+e) is executed again, the end point of tool traversing reaches B' point or exceeds the straight line between A' →B' point and Z executes the tool infeed to B' point, and then the next step is executed;

⑦Cutting feed from B' to C' point along the roughing path;

⑧Rapidly traverse to A from C' point and the program jumps to the next clock following nf block after G71 cycle is completed.

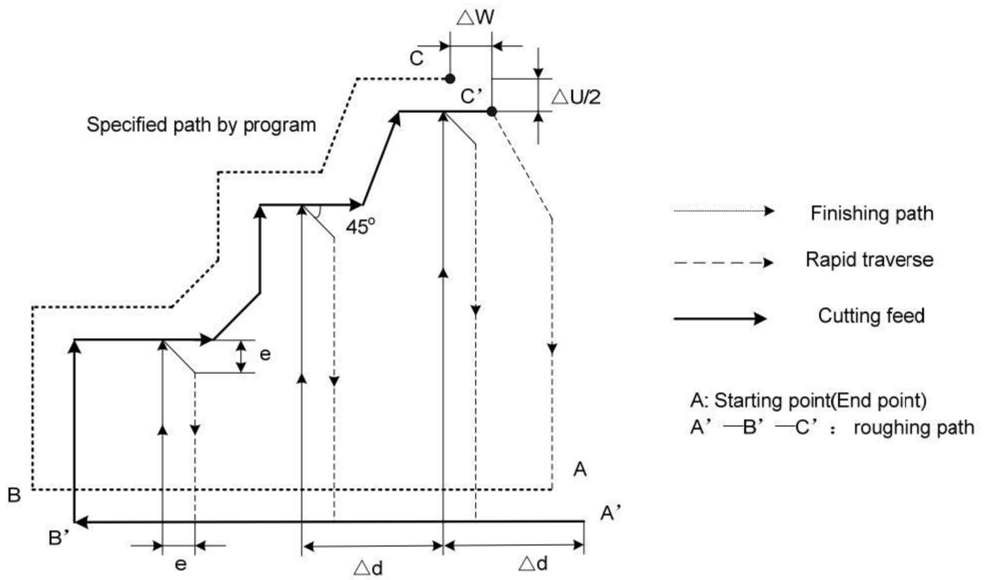


Fig. 4-2-16-1

Command specifications:

1. ns～nf blocks in programming must be followed G72 blocks. If they are in the front of G72 blocks, the system automatically searches and executes ns～nf blocks,

and then executes the next program following nf block after they are executed, which causes the system executes ns～nf blocks repetitively;

2. ns～nf blocks are used for counting the roughing path and the blocks are not executed when G72 is executed. F, S, T commands of ns～nf blocks are invalid when G72 is executed, at the moment, F, S, T commands of G72 blocks are valid. F, S, T of ns～nf blocks are valid when executing ns～nf to command G70 finishing cycle;

3. The dimensions in X, Z direction must be changed monotonously (always increasing or reducing) for the finishing path;

4. In ns～nf blocks, there are only G commands: G01, G02, G03, G04, G05, G6.2, G6.3, G7.2, G7.3, G96, G97, G98, G99, G40, G41, G42 and the system cannot call subprograms (M98/M99);

5. G96, G97, G98, G99, G40, G41, G42 are invalid when G72 is executed, and are valid when G70 is done;

6. In the ns～nf, the program block quantity cannot exceed 100;

7. When G72 is executed, the system can stop the automatic run and manual traverse, but return to the position before manual traversing when G72 is executed again, otherwise, the following path will be wrong;

8. When the system is executing the feed hold or single block, the program pauses after the system has executed end point of current path;

9. d △ , w are specified by the same W and differen △ t with or without being specified P, Q commands;

10. There are no the same block number in ns~nf when compound cycle commands are executed repetitively in one program;

11. The tool retraction point should be high or low as possible to avoid crashing the workpiece.

Coordinate offset direction with finishing allowance:

Δu, Δw define the coordinate offset and its direction of cut-in in finishing, and their sign symbol are as follows Fig. 4-2-16-2: B→C for finishing path, B’ →C’ for roughing path and A is the tool start-up point.

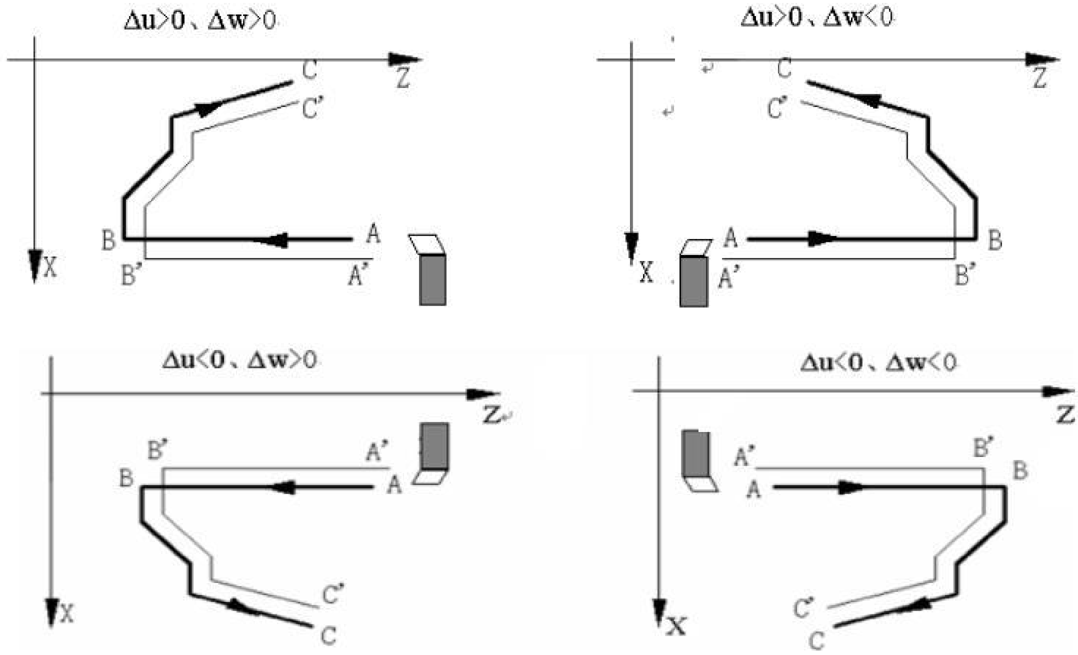


Fig. 4-2-16-2

G72 code machining example (Type I):

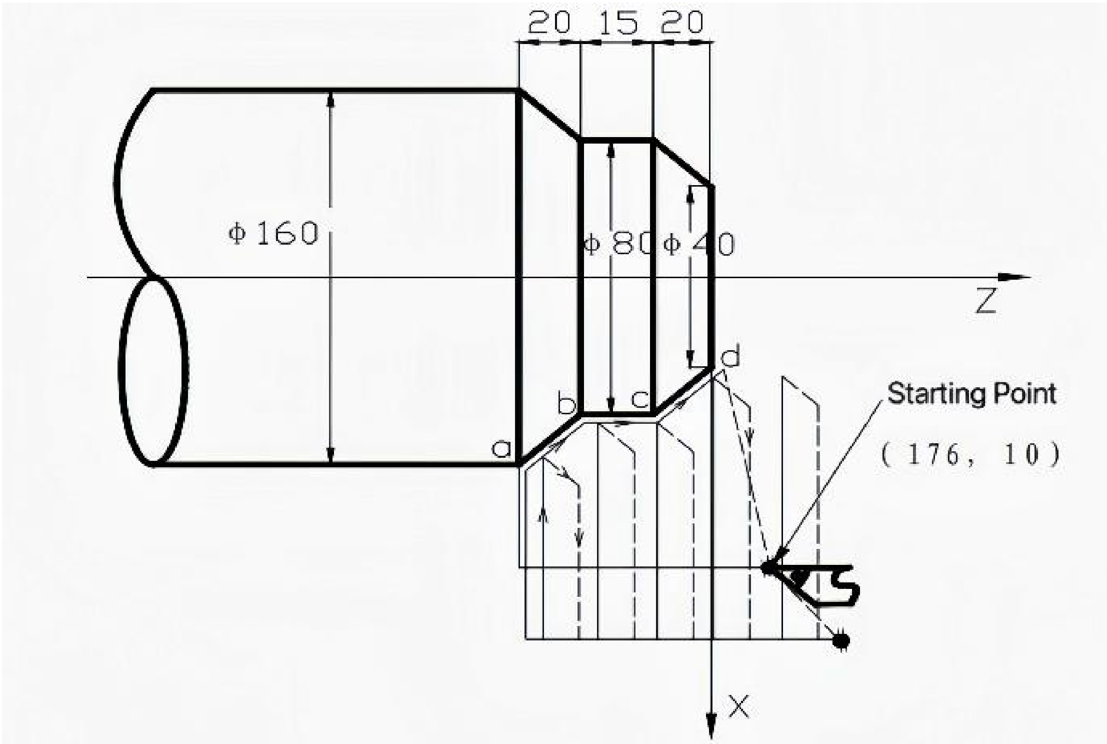


Fig. 4-2-16-3

Program：02163；

G00 X176 Z10 M03 S500 (Change No.2 tool and execute its compensation, pindle CW rotation with 500 r/min)

G72 W2.0 R0.5 F300； (Tool infeed 2mm, tool retraction 0.5mm)

G72 P10 Q20 U0.2 W0.1； (Roughing a--d,X roughing allowance 0.2mm and Z 0.1mm)

N10 G00 Z-55 S800 ； (Rapid traverse)

G01 X160 F120； (Infeed to a point)

X80 W20； (Machining a—b) Blocks for finishing path

W15； (Machining b—c)

N20 X40 W20 ； (Machining c—d)

G70 P010 Q020 M30； (Finishing a—d)

Circular machining of grooves (G72 type II)

Type II is different from Type I as follows:

1）Related definition: 1 more parameter than type I.

2）The contour along the X-axis does not need to monotonically increase or decrease, and can have up to 10 grooves, as shown below:

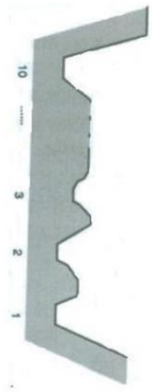


Fig. 4-2-16-4

However, the outer contour along the X-axis must monotonically increase or decrease, and the following contour cannot be processed:

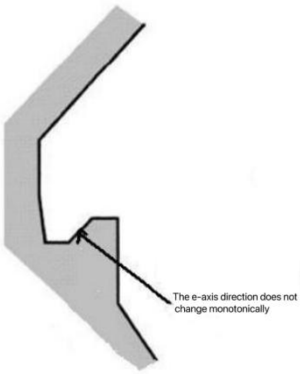


Fig. 4-2-16-5

1) The first Tool does not need to be vertical: if the shape changes monotonically along the Z-axis, processing can be carried out, as shown below:

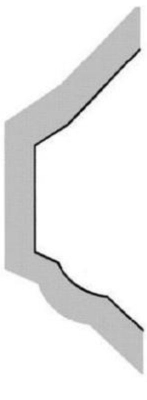


Fig. 4-2-16-6

2) After turning, the tool should be retracted, and the amount of retracted tool should be specified by the R (e) parameter or by the corresponding parameter setting value of the data parameter. The schematic diagram is as follows

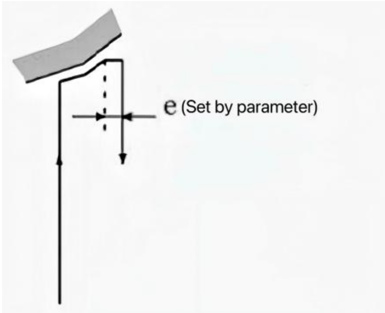


Fig. 4-2-16-7

3) The finish machining allowance can only be specified in the Z direction. If the finish machining allowance in the X direction is specified, it will cause the entire machining trajectory to shift. If specified, it is best to specify it as 0

代码格式：G72 W（Δd）R （e）F_ S_ T_;

```
G72 P（NS）Q（NF）U（Δu）W（Δw）;
    N（NS）G0/G1 X(U) Z(W) . . ;
    . . . . . . . . ;
    . . . .  F;
    . . . .  S;                    Precision machining route program section
    . . . .  T;
   N（NF）. . . . . . ;
```

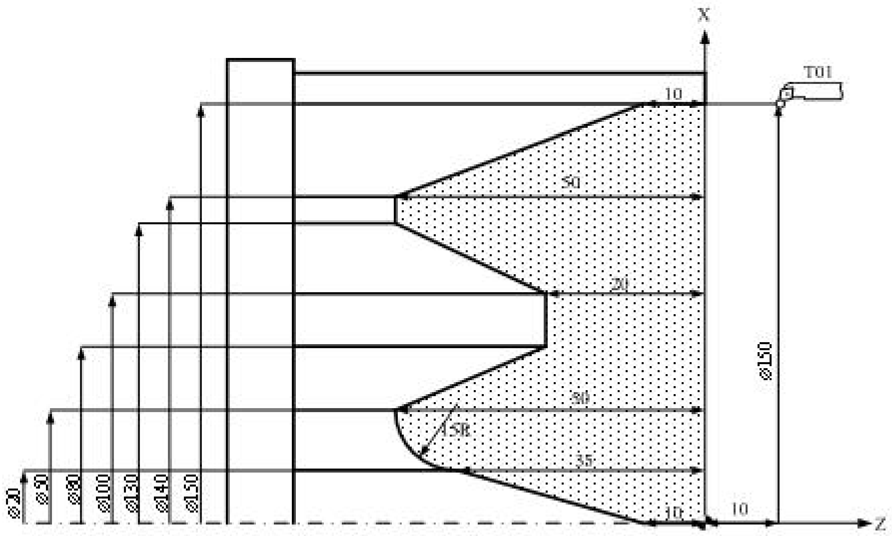


Fig. 4-2-16-8（G72 type II machining trajectory）

program：02168

```
G50 S5000;             //Maximum speed limit 5000 rpm
G96 S130 M03;          //Constant speed, surface speed of 130 m/min, spindle forward rotation
M08;                   //Turn on cutting agent
G00 X150.0 Z10.0;      //Quickly navigate to the starting point
```

```
G72 W2.0 R1.0 ;        //Z-axis cutting depth 2.0 mm, tool retraction 1.0 mm,

G72 P01 Q02 U0.8 W0.1 F0.6;

          //Perform a radial (end face) rough turning cycle, with blocks numbered N01 to N02, X-axis

          //The reserved amount for precision turning is 0.8 mm, and the reserved amount for precision turning in the Z-axis is 0.1 mm,

          //Feed rate 0.6 mm/rev

N01 G00 X150.0 Z0.0; //Rough turning contour

G01 Z-10.0;

X140.0 Z-50.0;

X130.0;

X100.0 Z-20.0;

X80.0;

X50.0 Z-50.0;

G03 X20.0 Z-35.0 R15.0;

G01 X20.0;

X0.0 Z-10.0;

N02 X0.0 Z0.0;

M09;

M05;

M30;                   //END
```

4.2.15 Closed Cutting Cycle G73

Command format: G73 U(Δi) W (Δk) R (d) F S T ; ⑴

G73 P(ns) Q(nf) U(Δu) W(Δw); ⑵

N (ns) . . . . . ;

. . . . ;

. . . . F;

. . . . S;

. . . . ; ⑶

•

N (nf). . . . . ;

Command functions: G73 is divided into three parts:

⑴ Blocks for defining the travels of tool infeed and tool retraction, the cutting speed, the spindle speed and the tool function when roughing;

⑵ Blocks for defining the block interval, finishing allowance;

⑶ Blocks for some continuous finishing path, counting the roughing path without being executed actually when executing G73.

According to the finishing allowance, the travel of tool retraction and the cutting times, the system automatically counts the travel of roughing offset, the travel of each tool infeed and the path of roughing, the path of each cutting is the offset travel of finishing path, the cutting path approaches gradually the finishing one, and last cutting path is the finishing one according to the finishing allowance. The starting point and end point of G73 are the same one, and G73 is applied to roughing for the formed rod. G73 is non-modal and its path is shown in Fig.3-31

Relevant definitions:

Finishing path: The above-mentioned Part 3 of G73 (ns～nf block)defines the finishing path, and the starting point of finishing path (start point of ns block)is the same these of starting point and end point of G73, called A point; the end point of the first block of finishing path(ns block)is called B point; the end point of finishing path(end point of nf block) is called C point. The finishing path is A→B→C.

Roughing path: It is one group of offset path of finishing one, and the roughing path times are the same that of cutting. After the coordinates offset, A, B, C of finishing path separately corresponds to An, Bn , Cn of roughing path(n is the cutting times, the first cutting path is A1, B1, C1 and the last one is Ad, Bd, Cd). The coordinates offset value of the first cutting compared to finishing path is (Δi×2+Δu, Δw+Δk) (diameter programming) , the coordinates offset value of the last cutting compared to finishing path is(Δu, Δw) , the coordinates offset value of each cutting compared to the previous one is as follows:

$$(-\frac{\Delta i\times 2}{d-1},-\frac{\Delta k}{d-1})$$

Δi: It is X tool retraction clearance in roughing, and its range is ±99999999× least input increment (radius, unit: mm/inch, with sign symbol) , Δi is equal to X coordinate offset value (radius value) of A1 point compared to Ad point. The X total cutting travel(radius value) is equal to |Δi| in roughing, and X cutting direction is opposite to the sign of Δi: Δi >0, the system executes X negative cutting in roughing. The No.053 value is regarded as X tool retraction clearance in roughing when U(Δi) is not input.

Δk: It is Z tool retraction clearance in roughing, and its range is -99999.999 ～ 99999.999(IS_B) /-9999.9999～9999.9999(IS_C) (radius, unit: mm/inch, with sign symbol) , Δk is equal to Z coordinate offset value (radius value) of A1 point compared to Ad point. Z total cutting travel(radius value) is equal to |Δk| in roughing, and Z cutting direction is opposite to the sign of Δk: Δk>0, the system executes Z negative cutting in roughing. The No.054 value is regarded as Z tool retraction clearance in roughing when W(Δk) is not input.

d: It is the cutting times 1~9999 (unit: times). R5 means the closed cutting cycle is completed by 5 times cutting. No.041 value is regarded as the cutting times when R(d) is not input. When the cutting times is 1, the system completes the closed cutting cycle based on 2 times cutting.

ns: Block number of the first block of finishing path.

nf: Block number of the last block of finishing path.

Δu: It is X finishing allowance and its range is -99999.999~99999.999(IS_B) /-9999.9999~9999.9999(IS_C) (diameter, unit: mm/inch, with sign symbol) and is the X coordinate offset of roughing path compared to finishing path, i.e. the different value of X absolute coordinates of A1 compared to A. Δu>0, it is the offset of the last X positive roughing path compared to finishing path. The system defaults Δu=0 when U(Δu) is not input, i.e. there is no X finishing allowance for roughing cycle.

Δw: It is Z finishing allowance and its range is-99999.999~99999.999(IS_B) /-9999.9999~9999.9999(IS_C) (diameter, unit: mm/inch, with sign symbol) and is the X coordinate offset of roughing path compared to finishing path, i.e. the different value of Z absolute coordinates of A1 compared to A. Δw>0, it is the offset of the last X positive roughing path compared to finishing path. The system defaults Δw=0 when W(Δw) is not input, i.e. there is no Z finishing allowance for roughing cycle.

F: Feedrate;

S: Spindle speed;

T: Tool number, tool offset number.

M, S, T, F: They can be specified in the first G73 or the second ones or program ns～nf. M, S, T, F functions of M, S, T, F blocks are invalid in G73, and they are valid in G70 finishing blocks.

Execution process:(Fig.4-2-17-1)

① A→A1: Rapid traverse;

② First roughing A1→B1→C1 :

A1→B1: Rapid traverse speed in ns block in G0, cutting feedrate specified by G73 in ns block in G1;

B1→C1: Cutting feed.

③ C1→A2: Rapid traverse.

④ Second roughing A2→B2→C2 :

A2→B2: Rapid traverse speed in ns block in G0, cutting feedrate specified by G73 in ns block in G1;
B2→C2: Cutting feed.
⑤ C2→A3: Rapid traverse:
…………
No. n times roughing, An→Bn→Cn :
An→Bn: ns Rapid traverse speed in ns block in G0, cutting feedrate specified by G73 in ns block in G1;
Bn→Cn: Cutting feed.
Cn→An+1: Rapid traverse;
…………
Last roughing, Ad→Bd→Cd : Ad→Bd:
Rapid traverse speed in ns block in G0, cutting feedrate specified by G73 in ns block in G1;
Bd→Cd: Cutting feed.
Cd→A: Rapid traverse to starting point;

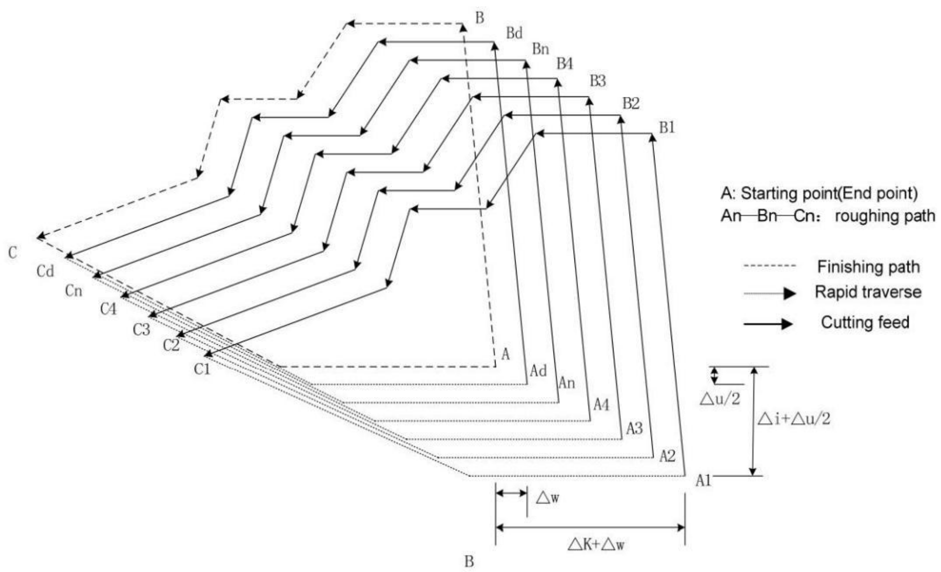


Fig. 4-2-17-1 G73 path

Command specifications:
1. ns～nf blocks in programming must be followed G73 blocks. If they are in the front of G73 blocks, the system automatically searches and executes ns～nf blocks, and then executes the next program following nf block after they are executed, which causes the system executes ns～nf blocks repetitively.
2. ns～nf blocks are used for counting the roughing path and the blocks are not executed when G73 is executed. F, S, T commands of ns～nf blocks are invalid when G71 is executed, at the moment, F, S, T commands of G73 blocks are valid. F, S, T of ns～nf blocks are valid when executing ns～nf to command G70 finishing cycle.
3. There are only G00, G01 in ns block.
4. In ns～nf blocks, there are only G commands: G00, G01, G02, G03, G04, G05, G6.2, G6.3, G7.2, G7.3, G96, G97, G98, G99, G40, G41, G42 and the system cannot call subprograms (M98/M99) .

5. In ns～nf, the program block quantity cannot exceed 100;
6. G96, G97, G98, G99, G40, G41, G42 are invalid when G73 is executed, and are valid when G70 is executed.
7. When G73 is executed, the system can stop the automatic run and manual traverse, but return to the position before manual traversing when G73 is executed again, otherwise, the following path will be wrong.
8. When the system is executing the feed hold or single block, the program pauses after the system has executed end point of current path.
9. i △ , u are specified by the same U and △ Δk, Δw are specified by the same U, and they are different with or without being specified P,Q commands.
10. G73 cannot be executed in MDI, otherwise, the system alarms.
11. There are no the same block number in ns~nf when compound cycle commands are executed repetitively in one program.
12. The tool retraction point should be high or low as possible to avoid crashing the workpiece. Coordinate offset direction with finishing allowance:
Δi, Δk define the coordinates offset and its direction of roughing; Δu, Δw define the coordinate offset and the cut-in direction in finishing, and their sign symbols are as follows Fig. 3-32: A is tool start-up point, B→C for workpiece contour, B’ →C’ for roughing contour and B’’ →C’’ for finishing path.

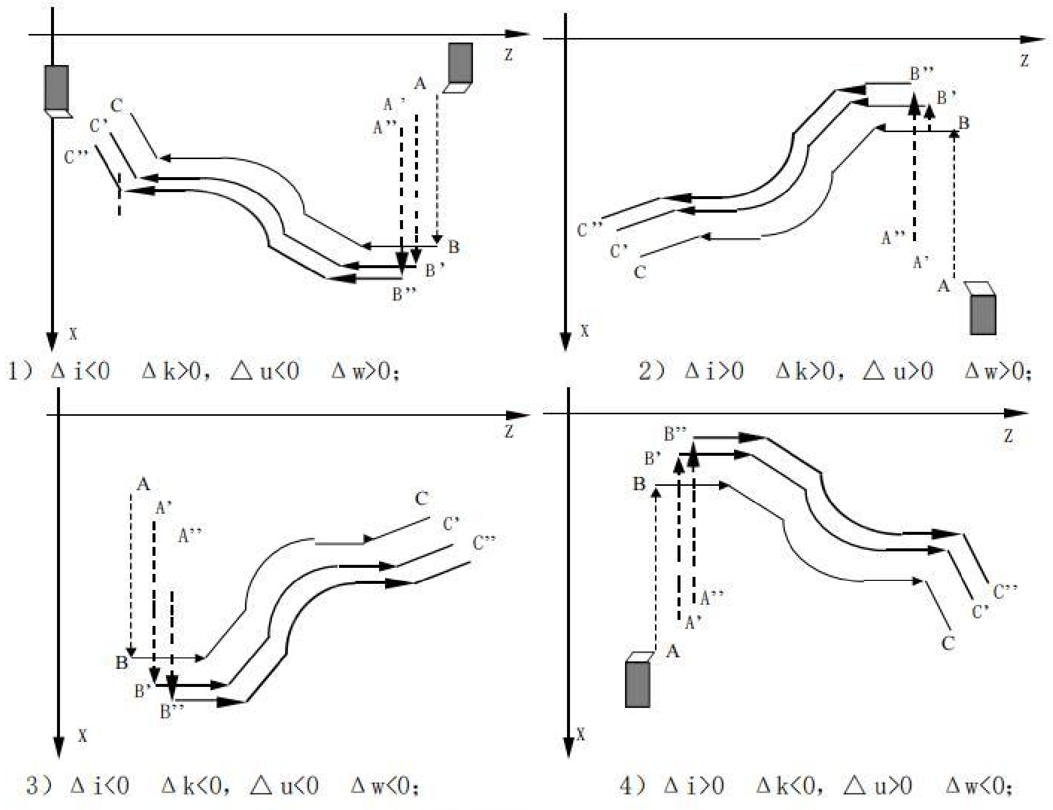


Fig. 4-2-17-2

G73 code machining example:

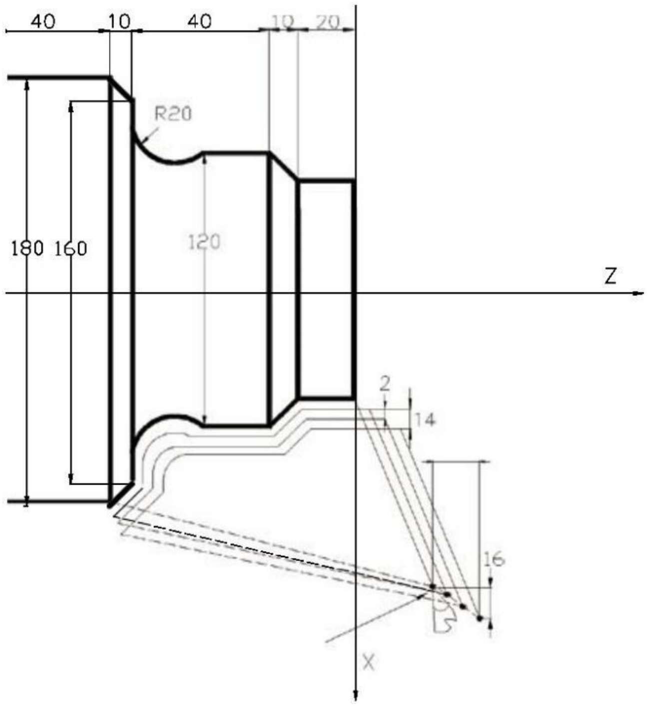


Fig. 4-2-17-3

Program: 02173;

G99 G00 X200 Z10 M03 S500; (Specify feedrate per rev and position starting point and start spindle)

G73 U1.0 W1.0 R3 ; ((X tool retraction with 2mm, Z 1mm)

G73 P14 Q19 U0.5 W0.3 F0.3 ; ((X roughing with 0.5 allowance and Z 0.3mm)

N14 G00 X80 Z0 ;

G01 W-20 F0.15 S600 ;

X120 W-10 ;

W-20 ; Blocks for finishing

G02 X160 W-20 R20 ;

N19 G01 X180 W-10 ;

G70 P14 Q19 M30; (Finishing)

4.2.16 Finishing Cycle G70

Command format: G70 P(ns) Q(nf);

Command function: The tool executes the finishing of workpiece from starting point along with the finishing path defined by ns～nf blocks. After executing G71, G72 or G73 to roughing, execute G70 to finishing and single cutting of finishing allowance is completed. The tool returns to starting point and execute the next block following G70 block after G70 cycle is completed.

ns: Block number of the first block of finishing path.

nf: Block number of the last block of finishing path.

G70 path is defined by programmed one of ns～nf blocks. Relationships of relative position of ns, nf block in G70～G73 blocks are as follows:

G71/G72/G73……;

```
N (ns) . . . . . .
       . . . . . . . .
       . F
       .                         Blocks for finishing path
       .
       .
N (nf) . . . . . .
G70 P(ns) Q(nf);
```

Command specifications:

1. ns～nf blocks in programming must be followed G70 blocks.
2. F, S, T in ns～nf blocks are valid when executing ns～nf to command G70 finishing cycle.
3. G96, G97, G98, G99, G40, G41, G42 are valid in G70;
4. When G70 is executed, the system can stop the automatic run and manual traverse, but return to the position before manual traversing when G70 is executed again, otherwise, the following path will be wrong.
5. When the system is executing the feed hold or single block, the program pauses after the system has executed end point of current path.
6. There are no the same block number in ns~nf when compound cycle commands are executed repetitively in one program.
7. In ns～nf, the program block quantity cannot exceed 100;
8. The tool retraction point should be high or low as possible to avoid crashing the workpiece.

4.2.17 Axial Grooving Multiple Cycle G74

Command format: G74 R(e);

G74 X(U) Z(W) P(Δi) Q(Δk) R(Δd) F ;

Command function: Axial (X axis) tool infeed cycle compounds radial discontinuous cutting cycle: Tool infeeds from starting point in radial direction(Z), retracts, infeeds again, and again and again, and last tool retracts in axial direction, and retracts to the Z position in radial direction, which is called one radial cutting cycle; tool infeeds in axial direction and execute the next radial cutting cycle; cut to end point of cutting, and then return to starting point (starting point and end point are the same one in G74), which is called one radial grooving compound cycle. Directions of axial tool infeed and radial tool infeed are defined by relative position between end point X(U) Z(W) and starting point of cutting. G75 is used for machining radial loop groove or column surface by radial discontinuously cutting, breaking stock and stock removal.

Relevant definitions:

Starting point of axial cutting cycle: starting position of axial tool infeed for each axial cutting cycle, defining with An(n=1,2,3……), Z coordinate of An is the same that of starting point A, the different value of X coordinate between An and An-1 is Δi. The starting point A1 of the first axial cutting cycle is the same as the starting point A, and the X coordinate of starting point (Af) of the last axial cutting cycle is the same that of cutting end point.

End point of axial tool infeed: starting position of axial tool infeed for each axial cutting cycle, defining with Bn(n=1,2,3……), Z coordinate of Bn is the same that of cutting end point, X coordinate of Bn is the same that of An , and the end point (Bf) of the last axial tool infeed is the same that of cutting end point.
End point of radius tool retraction: end position of radius tool infeed (travel of tool infeed is Δd) after each axial cutting cycle reaches the end point of axial tool infeed, defining with Cn(n=1,2,3……), Z coordinate of Cn is the same that of cutting end point, and the different value of X coordinate between Cn and An is Δd;
End point of axial cutting cycle: end position of axial tool retraction from the end point of radius tool retraction, defining with Dn(n=1, 2, 3……), Z coordinate of Dn is the same that of starting point, X coordinate of Dn is the same that of Cn (the different value of X coordinate between it and An is Δd);
Cutting end point: it is defined by X(U) Z(W) , and is defined with Bf of the last axial tool infeed.

R(e) : it is the tool retraction clearance after each axial(Z) tool infeed, and its range is 0~99.999(IS-B) without sign symbols. The specified value is reserved validly after R(e) is executed. The NO.056 value is regarded as the tool retraction clearance when R(e) is not input.

X: X absolute coordinate value of cutting end point Bf (unit: mm).

U: Different value of X absolute coordinate between cutting end point Bf and starting point.

Z: Z absolute coordinate value of cutting end point Bf (unit: mm).

W: Different value of Z absolute coordinates between cutting end point Bf and starting point

P(Δi) : radial(X) cutting for each axial cutting cycle , range: 0<Δi≤9999999(IS_B)/0<Δi≤ 99999999 (IS_C) (unit: least input increment, diameter value, without sign symbol).

Q(Δk): radial(Z) cutting for each axial cutting cycle, range: 0<Δk≤9999999(IS_B)/0<Δk ≤99999999 (IS_C) (unit: least input increment, diameter value, without sign symbol).

R(Δd) : radial (X) tool retraction after cutting to end point of axial cutting, range: 0~99999999× least input increment (unit: mm/inch, diameter value, without sign symbol).. The radial (X) tool retraction clearance is 0 when the system defaults the axial cutting end point. The system defaults the tool retraction is executed in positive direction when X(U) and P(Δi) are omitted.

Execution process: Fig. 4-2-19-1。

①Axial (Z) cutting feed △k from the starting point of axial cutting cycle, feed in Z negative direction when the coordinates of cutting end point is less than that of starting point in Z direction, otherwise, feed in Z positive direction;

② Axial (Z) rapid tool retraction e and its direction is opposite to the feed direction of ①;

③ X executes the cutting feed (Δk+e) again, the end point of cutting feed is still in it between starting point An of axial cutting cycle and end point of axial tool infeed, Z executes the cutting feed (Δk+e)again and execute ②; after Z executing the cutting feed (Δk+e)again, the end point of cutting feed is on Bn or is not on it between An and Bn cutting feed to Bn in Z direction and then execute ④;

④ Radial(X) rapid tool retraction △d/2 to Cn , when X coordinate of Bf (cutting end point) is less than that of A (starting point), retract tool in X positive, otherwise, retract tool in X negative direction;

⑤ Axial(Z axial) rapid retract tool to Dn, No. n axial cutting cycle is completed. If the current axial cutting cycle is not the last one, execute ⑥ ; if it is the previous one before the last axial cutting cycle, execute ⑦;

⑥ Radial(X axial)rapid tool infeed, and it direction is opposite to ④ retract tool. If the end point of tool infeed is still on it between A and Af (starting point of last axial cutting cycle) after X executes the tool infeed (△d/2+△i/2) , i.e. Dn→An+1 and then execute ① (start the next axial cutting cycle); if X end point of tool infeed is not on it between Dn and Af after tool infeed (△d/2+△i/2), rapidly traverse to Af and execute ① to start the first axial cutting cycle;

⑦ X rapidly traverse to return to A, and G74 is completed.

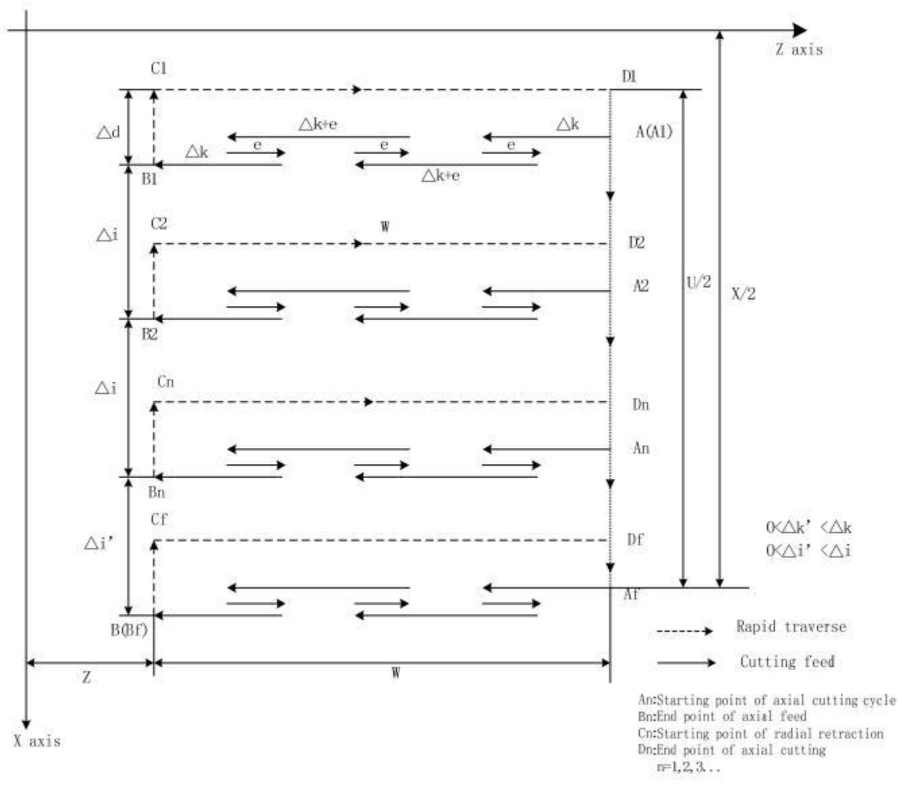


Fig. 4-2-19-1

Command specifications:

1. The cycle movement is executed by Z(W) and P(Δk) blocks of G74, and the movement is not executed if only "G74 R(e) ; " block is executed;

2. Δd and e are specified by the same address and whether there are Z(W) and P(Δk) word or not in blocks to distinguish them;

3. The tool can stop in Auto mode and traverse in Manual mode when G74 is executed, but the tool must return to the position before executing in Manual mode when G74 is executed again, otherwise the following path will be wrong.
4. When the single block is running, programs dwell after each axial cutting cycle is completed.
5. R(Δd) must be omitted in blind hole cutting, and so there is no distance of tool retraction when the tool cuts to axial end point of cutting.
Example：Fig.4-2-19-2

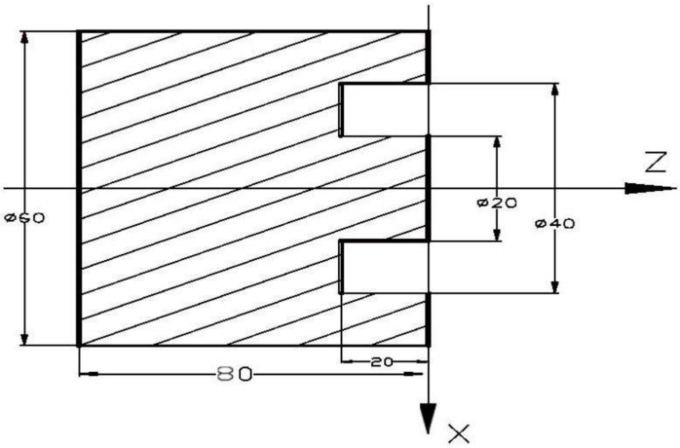


Fig. 4-2-19-2

Program (suppose that the grooving tool width is 4mm, system least increment is 0.001mm):

O2192；
G0 X36 Z5 M3 S500； (Start spindle and position to starting point of machining)
G74 R0.5 ； (Machining cycle)
G74 X20 Z-20 P3000 Q5000 F50；(Z tool infeed 5mm and tool retraction 0.5mm each time; rapid return to starting point (Z5) after cutting feed to end point (Z-20), X tool infeed 3mm and cycle the above-mentioned steps)
M30； (End of program)

4.2.18 Radial Grooving Multiple Cycle G75

Command format：G75 R(e)；
G75 X(U) Z(W) P(Δi) Q(Δk) R(Δd) F ；

Command function：Axial (Z) tool infeed cycle compounds radial discontinuous cutting cycle: Tool infeeds from starting point in radial direction, retracts, infeeds again, and again and again, and last tool retracts in axial direction, and retracts to position in radial direction, which is called one radial cutting cycle; tool infeeds in axial direction and execute the next radial cutting cycle; cut to end point of cutting, and then return to starting point (starting point and end point are the same one in G75), which is called one radial grooving compound cycle. Directions of axial tool infeed and radial tool infeed are defined by relative position between end point X(U) Z(W) and starting point of cutting. G75 is used for machining radial loop groove or column surface by radial discontinuously cutting, breaking stock and stock removal.

Relevant definitions:

Starting point of radial cutting cycle: Starting position of axial tool infeed for each radial cutting cycle, defined by An(n=1, 2, 3……), X coordinate of An is the same that of starting point A, the different value of X coordinate between An and An-1 is Δk. The starting point A1 of the first radial cutting cycle is the same as the starting point A, and Z starting point (Af) of the last axial cutting cycle is the same that of cutting end point.

End point of radial tool infeed: Starting position of radial tool infeed for each radial cutting cycle, defined by Bn (n=1, 2, 3……), X coordinates of Bn is the same that of cutting end point, Z coordinates of Bn is the same that of An, and the end point (Bf) of the last radial tool infeed is the same that of cutting end point.

End point of axial tool retraction: End position of axial tool infeed (travel of tool infeed is Δd) after each axial cutting cycle reaches the end point of axial tool infeed, defining with Cn(n=1,2,3……), X coordinate of Cn is the same that of cutting end point, and the different value of Z coordinate between Cn and An is Δd;

End point of radial cutting cycle: End position of radial tool retraction from the end point of axial tool retraction, defined by Dn(n=1, 2, 3……), X coordinate of Dn is the same that of starting point, Z coordinates of Dn is the same that of Cn (the different value of Z coordinate between it and An is Δd);

Cutting end point: It is defined by X(U) Z(W) , and is defined with Bf of the last radial tool infeed.

R(e) : It is the tool retraction clearance after each radial(X) tool infeed, its range is 0~99.999 (unit: mm, radius value) without sign symbols. The specified value is reserved validly after R(e) is executed and the data is switched and saved to No.042. NO.042 value is regarded as the tool retraction clearance when R(e) is not input.

X: X absolute coordinate value of cutting end point Bf (unit: mm).

U: Different value of X absolute coordinate between cutting end point Bf and starting point.

Z: Z absolute coordinate value of cutting end point Bf (unit: mm).

W: Different value of Z absolute coordinate between cutting end point Bf and starting point A(unit: mm).

P(Δi) : Radial(X) discontinuous tool infeed of each axial cutting cycle, its range: 0<Δi≤ 9999999 (IS_B)/ 0<Δk≤ 99999999 (IS_C) (unit: least input increment, without sign symbol).

Q(Δk) : Axial(Z) discontinuous tool infeed of each radial cutting cycle, its range: 0<Δk≤ 9999999(IS_B)/0<Δk≤ 99999999 (IS_C) (unit: least input increment, without sign symbol).

R(Δd): Axial (Z) tool retraction clearance after cutting to end point of radial cutting, its range: 0~99999999×least input increment (unit: mm/inch, without sign symbol).

The system defaults the tool retraction clearance is 0 after the radial cutting end point is completed when R (Δd) is omitted. The system defaults it executes the positive tool retraction when Z(W) and Q(Δk) are omitted.

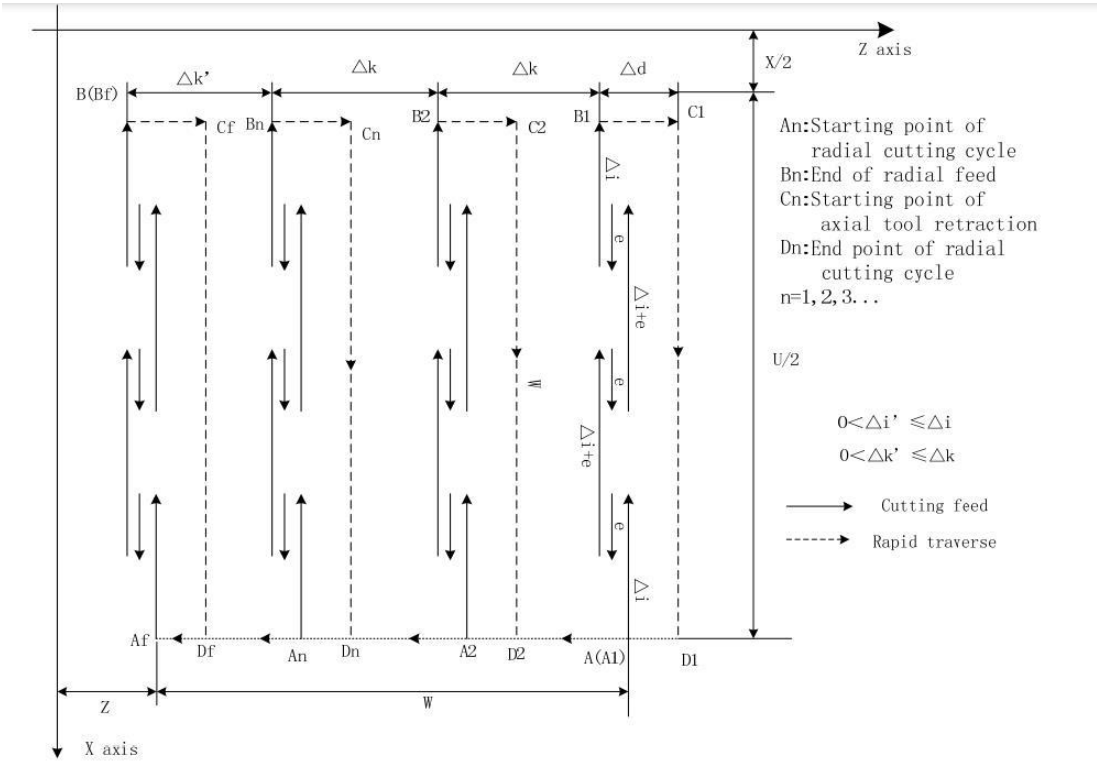


Fig. 4-2-20-1

Execution process: Fig. 4-2-20-1

① Radial (X) cutting feed △i from the starting point of radial cutting cycle, feed in X negative direction when the coordinates of cutting end point is less than that of starting point in X direction, otherwise, feed in X positive direction;

② Radial(X) rapid tool retraction e and its direction is opposite to the feed direction of ①;

③ X executes the cutting feed (Δi+e) again, the end point of cutting feed is still in it between starting point An of radial cutting cycle and end point of radial tool infeed, X executes the cutting feed (Δi+e) again and executes ②; after X cutting feed (Δi+e) is executed again, the end point of X cutting feed is on Bn or is not on it between An and Bn cutting feed to Bn and then execute○4 ;

④ Axial(Z) rapid tool retraction △d to Cn , when Z coordinate of Bf (cutting end point) is less than that of A (starting point), retract tool in Z positive, otherwise, retract tool in Z negative direction;

⑤ Radial (X) rapid retract tool to Dn, No. n radial cutting cycle is completed. The current radial cutting cycle is not the last one, execute ⑥; if it is the previous one before the last radial cutting cycle, execute ⑦;

⑥ Axial(X) rapid tool infeed, and it direction is opposite to ④ retract tool. If the end point of tool infeed is still on it between A and Af (starting point of last radial cutting cycle) after Z tool infeed (△d+△k), i.e. Dn→An+1 and then execute ① (start the next radial cutting cycle); if the end point of tool infeed is not on it between Dn and Af after Z tool infeed (△d+△k), rapidly traverse to Af and execute ① to start the first radial cutting cycle;

Explanation ：

1. The cycle movement is executed by X(W) and P(Δi) blocks of G75, and the movement is not executed if only "G75 R(e) ; " block is executed;
2. Δd and e are specified by the same address R and whether there are X(U) and P(Δi) words or not in blocks to distinguish them;
3. The tool can stop in Auto mode and traverse in Manual mode when G75 is executed, but the tool must return to the position before executing in Manual mode when G75 is executed again, otherwise the following path will be wrong;
4. When the system is executing the feed hold or single block, the program pauses after the system has executed end point of current path;
5. R(Δd) must be omitted in grooving, and so there is no tool retraction clearance when the tool cuts to radial cutting end point.

Example：

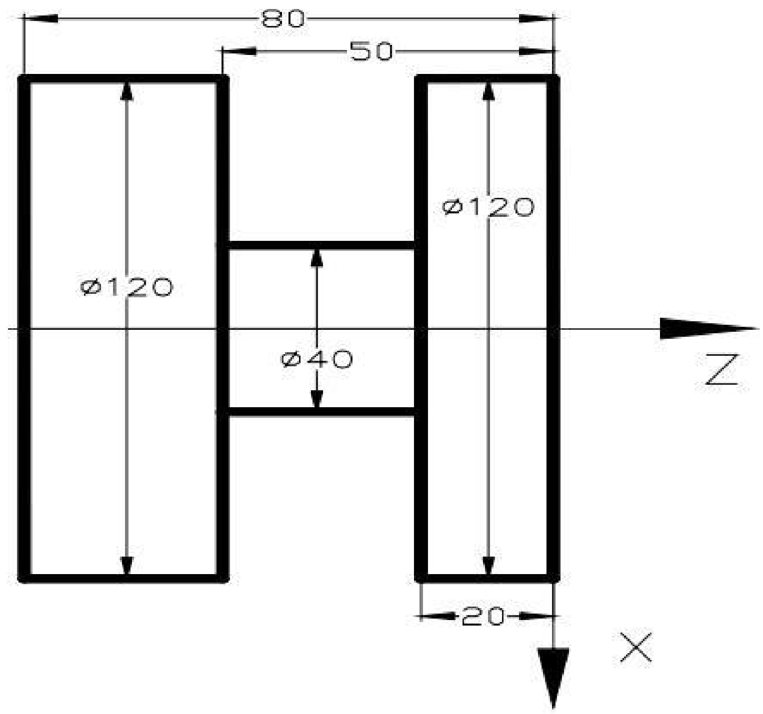


Fig. 4-2-20-2

Program (suppose the grooving tool width is 4mm, the system least increment is 0.001mm):

O2202;

G00 X150 Z50 M3 S500; (Start spindle with 500 r/min)

G0 X125 Z-24; (Position to starting point of machining)

G75 R0.5 F150; (Machining cycle)

G75 X40 Z-50 P6000 Q3000; (X tool infeed 6mm every time, tool retraction 0.5mm, rapid returning to starting point (X125) after infeeding to end point (X40), Z tool infeed 3mm and cycle the above-mentioned steps to continuously run programs)

G0 X150 Z50； ((Return to starting point of machining)

M30； (End of program)

4.2.19 Thread Cutting with Constant Lead G32

Command format: G32 X(U)_ Z(W)_ F(I)_ J_ K_ Q_

Command function: The path of tool traversing is a straight line from starting point to end point as Fig.4-2-21-1; the longer moving distance from starting point to end point(X in radius value) is called as the long axis and another is called as the short axis. In course of motion, the long axis traverses one lead when the spindle rotates one revolution, and the short axis and the long axis execute the linear interpolation. Form one spiral grooving with variable lead on the surface of workpiece to realize thread cutting with constant lead. Metric pitch and inch pitch are defined respectively by F, I. Metric or inch straight, taper, end face thread and continuous multi-section thread can by machined in G32.

Command specifications:G32 is modal;

Pitch is defined to moving distance when the spindle rotates one rev(X in radius);

Execute the straight thread cutting when X coordinates of starting point and end point are the same one(not input X or U);

Execute the end face thread cutting when X coordinates of starting point and end point are the same one(not input Z or W);

Execute the cutting taper thread when X and Z coordinates of starting point and end point are different;

Related definitions ：

相关定义：

F: Metric pitch is moving distance of long axis when the spindle rotates one rev: 0.001 mm～500 mm. After F is executed, it is valid until F with specified pitch is executed again.

I: Teeth per inch. It is ones per inch(25.4 mm) in long axis, and also is circles of spindle rotation when the long axis traverses one inch(25.4 mm) :0.06tooth/inch～25400tooth/inch. After I is executed, it is valid until I with specified pitch is executed again. The metric, inch input both express the teeth per inch thread.

J: Movement in the short axis in thread run-out, its range: (-99999999~99999999) × least input increment with negative sign; if the short axis is X, its value is specified with the radius; J value is the modal parameter.

K: Length in the long axis in thread run-out, its range: 0~99999999× least input increment. If the long axis is X, its value is in radius without direction; K is modal parameter.

Q: Initial angle(offset angle)between spindle rotation one rev and starting point of thread cutting: 0～360000 (unit: 0.001 degree). Q is non-modal parameter, must be defined every time, otherwise it is 00 .

Q rules: 1. Its initial angle is 0° if Q is not specified;

2. For continuous thread cutting, Q specified by its following thread cutting block except for the first block is invalid, namely Q is omitted even if it is specified;

3. Multi threads formed by initial angle is not more than 65535;

4. Q unit : 0.0010 . Q180000 is input in program if it offsets 1800 with spindle one-turn; if Q180 or Q180.0, it is 0.18°

Difference between long axis and short axis is shown in Fig.4-2-21-1

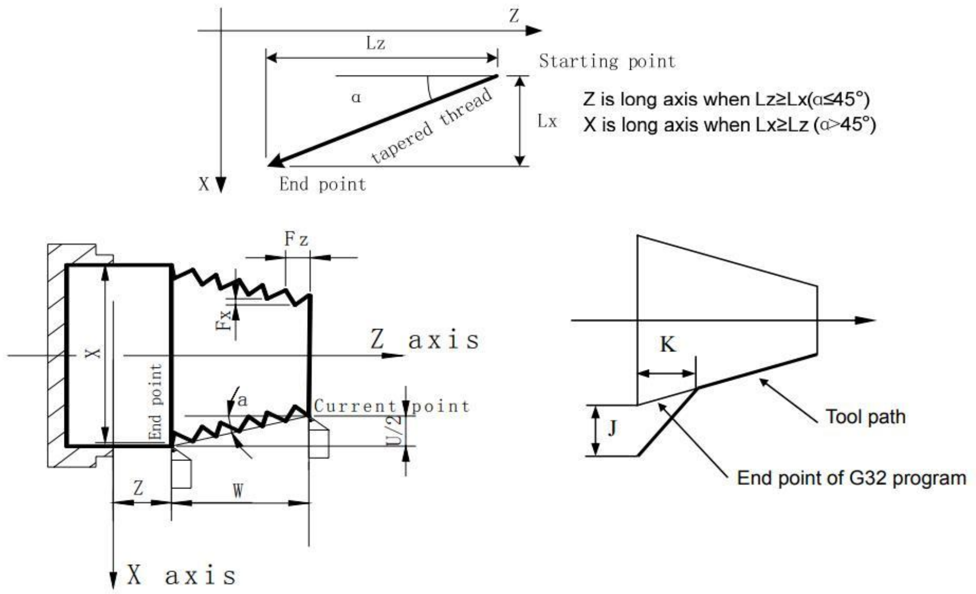


Fig. 4-2-21-1 G32 path

Notes:

1. J, K are modal. The thread run-out is previous J, K value when they are omitted in the next block in continuous thread cutting. Their mode are cancelled when no thread Tool path End point of G32 program J K Chapter 3 G Commands 135 Ⅰ Programming cutting are executed;

2. There is no thread run-out when J, or K are omitted; K=J is the thread run-out value when K is omitted;

3. There is no thread run-out when J=0 or J=0, K=0;

4. The thread run-out value J=K when J≠0,K=0;

5. There is no thread run-out when J=0 or K≠0;

6. When parameter No.107 set to 0, its tail-retraction angle is matched with the J or K ratio. When the parameter No.107 does not set to 0, the tail-retraction speed of the short axis is set by parameter value, and the tail-retraction angle is determined by thread cutting speed and short axis tail-retraction speed;

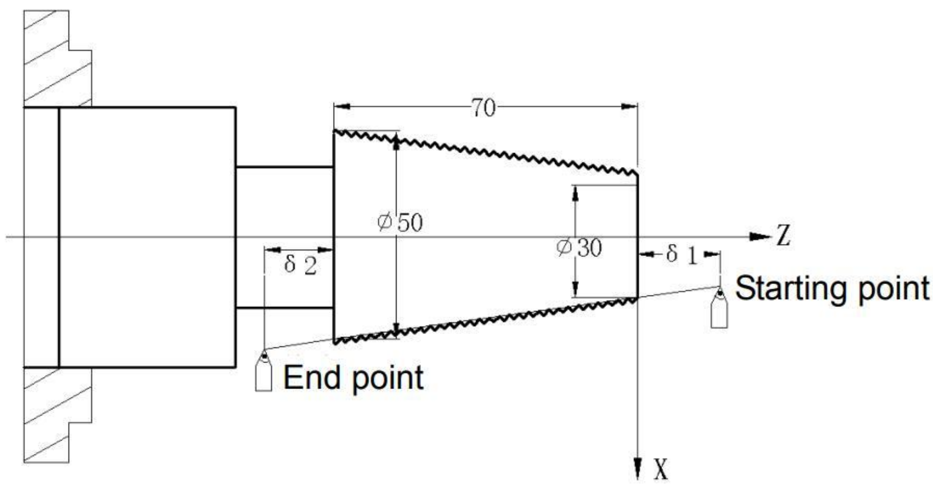
7. If the current block is for thread and the next block is the same, the system does not test the spindle encoder signal per rev at starting the next block to execute the direct thread cutting, which function is called as continuous thread machining;

8. After the feed hold is executed, the system displays "Pause" and the thread cutting continuously executes not to stop until the current block is executed completely; if the continuous thread cutting is executed, the program run pauses after thread cutting blocks are executed completely;

9. In Single block, the program stops run after the current block is executed. The program stops running after all blocks for thread cutting are executed;

10. The thread cutting decelerates to stop when the system resets, emergently stop or its drive unit alarms.

Example: Pitch: 2mm. δ1 = 3mm, δ2 = 2mm, total cutting depth 2mm divided into two times cut-in.



Program:

```
O0009;
G00 X27.4286 Z3; (First cut-in 1mm)
G32 X48.8571 W-75 F2.0; (First taper cutting)
G00 X55; (Tool retraction)
W75; (Z returns to the starting point)
X26.4286; (Second tool infeed 0.5mm)
G32 X47.4286 W-75 F2.0; (Second taper thread cutting )
G00 X55; (Tool retraction)
W75 ; (Z returns to the starting point)
M30;
```

4.2.20 Thread Cutting with Variable Lead G34

Command format: G34 X(U)__ Z(W)__ F(I)__ J__ K__ R__ ;

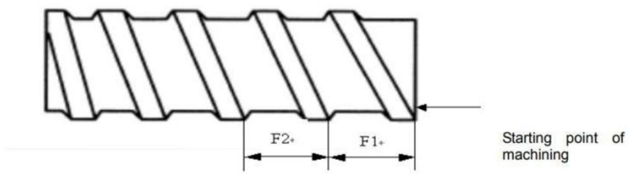
Command function: The motion path of tool is a straight line from starting point of X, Z to end point specified by the block, the longer moving distance from starting point to end point(X in radius value) is called as the long axis and another is called as the short axis. In course of motion, the long axis traverses one lead when the spindle rotates one rev, the pitch increases or decreases a specified value per rev and one spiral grooving with variable lead on the surface of workpiece to realize thread cutting with variable lead. Tool retraction can be set in thread cutting. F, I are specified separately to metric, inch pitch. Executing G34 can machine metric or inch straight, taper, end face thread with variable pitch.

Command specifications:

G34 is modal;

1. Meanings of X(U) , Z(W) , J, K are the same that of G32;
2. Specify lead, and its range is referred to Table 1-9;
3. Specify thread teeth per inch, and its range is referred to Table 1-9;
4. Increment or decrement of pitch per rev, R=F1- F2, with direction; F1>F2, pitch decreases when R is negative; F1<F2, pitch increases when R is positive ;
5. ±0.001~±500.000 mm/pitch (metric thread);;
   ±0.060~±2540 tooth/inch (inch thread).

The system alarms when R exceeds the above-mentioned range or the pitch exceeds permissive value or is negative owing to R increases or decreases.



Note: It is the same as that of G32.

Example: First pitch of starting point: 4mm, increment 0.2mm per rev of spindle

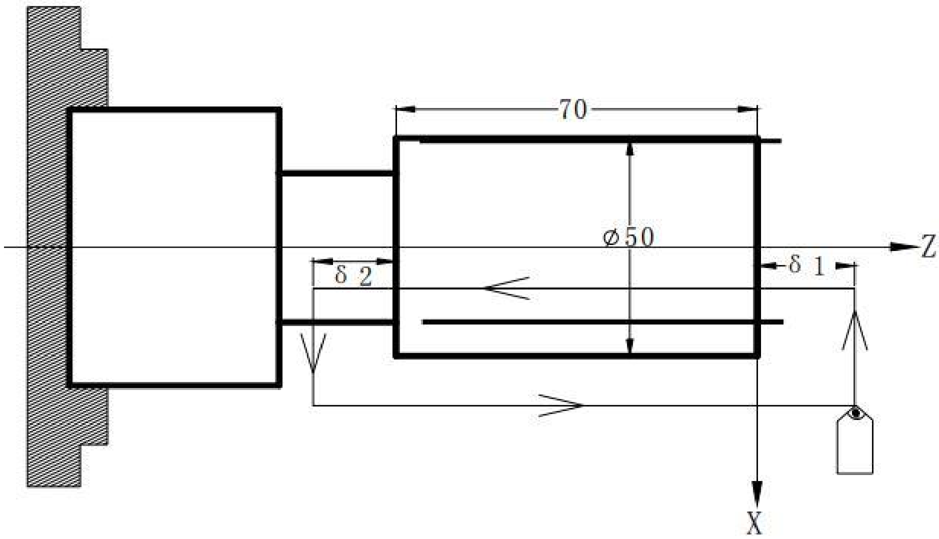


Fig. 4-2-22-1

Use macro variables to simplify programming when G34 is used many times. δ1 = 4mm, δ2 = 4mm, total cutting depth 4mm, total cutting cycle 15 times; first tool infeed 0.8mm, gradual decreasing cutting every time 0.2mm, min. infeed 0.2mm.

Program:

```
O2221;
G00 X60 Z4 M03 S500;
G65 H01 P#102 Q0.8;      //First tool infeed: assignment #102=0.8mm
G65 H01 P#103 Q0;        //Cycle count: assignment #103=0
N10 G65 H02 P#104 Q#103 R1;  //Cycle count starting: #104=#103+1
G65 H01 P#103 Q#104;     // #103=#104
G65 H81 P30 Q#104 R15;   //Total cutting cycle times: #104=15,jump to block N30
G00 U-10;                //Tool infeed to Φ50
G65 H01 P#100 Q#102;     //Cutting infeed: #100=#102
G00 U-#100;              //Tool infeed
G34 W-78 F3.8 J5 K2 R0.2;  //Variable pitch cutting
G00 U10;                 //Tool retraction
Z4;                      //Z returns to starting point
G65 H03 P#101 Q#100 R0.2; //Decreasing of cutting feed again: #101=#100-0.2
G65 H01 P#102 Q#101;     //Assignment again #102=#101
G65 H86 P20 Q#102 R0.2;  //Infeed: Jump to block N20 when #102 0.2mm
G65 H80 P10;             //Unconditionally jump to block N10
N20 G65 H01 P#102 Q0.2;  //Min. infeed: #102=0.2
G65 H80 P10;             //Unconditionally jump to block N10
N30 M30;
```

4.2.21 Z Thread Cutting G33

Command format: G33 Z(W)__ F(I)__ L__ ;

Command function: Tool path is from starting point to end point and then from end point to starting point. The tool traverses one pitch when the spindle rotates one rev, the pitch is consistent with pitch of tool and there is spiral grooving in internal hole of workpiece and the internal machining can be completed one time.

Command specification: G33 is modal command;

Z(W): When Z or W is not input and starting point and end point of Z axis are the same one, the thread cutting must not be executed;

F: Thread pitch, and its range is referred to Table 1-2;

I: Specify thread teeth per inch, and its range is referred to Table 1-2;

L: Multi threads: 1～99 and it is modal parameter. (The system defaults it is single thread when L is omitted)

Cycle process:

① Z tool infeed (start spindle before G33 is executed);

② M05 signal outputs after Z reaches the specified end point in programming;

③ Test spindle after completely stopping;

④ Spindle rotation (CCW) signal outputs (reverse to the original rotation direction);

⑤ Z executes the tool retracts to starting point;
⑥ M05 signal outputs and the spindle stops;
⑦ Repeat the steps ①～⑥ if multi threads are machined.

Example: Fig. 4-2-23-1, thread M10×1.5

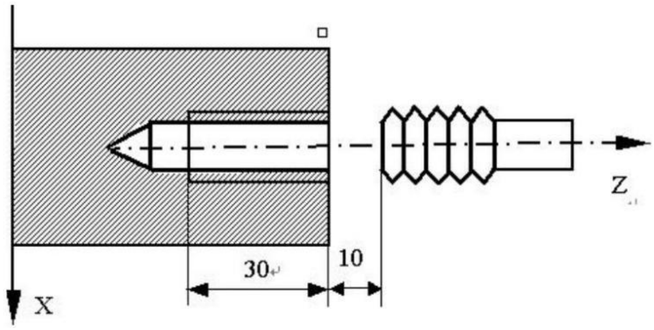


Fig. 4-2-23-1

Program: O2231;

```
G00 Z90 X0 M03;  Start spindle
G33 Z50 F1.5;    Start spindle
M03              Start spindle
G00 X60 Z100;    Machine continuously
M30
```

Note 1: Before tapping, define rotation direction of spindle according to tool rotating. The spindle stops rotation after the tapping is completed and the spindle is started again when machining thread continuously.

Note 2: G33 is for rigid tapping. The spindle decelerates to stop after its stop signal is valid, at the moment, Z executes continuously infeeds along with the spindle rotating, and so the actual cutting bottom hole is deeper than requirement and the length is defined by the spindle speed and its brake in tapping.

Note 3: Z rapid traverse speed in tapping is defined by spindle speed and pitch is not relevant to cutting feedrate override.

Note 4: In Single block to feed hold, the tapping cycle continuously executes not to stop until the tool returns to starting point when the system displays "Pause".

Note 5: The thread cutting decelerates to stop when the system resets, emergently stop or its driver alarms.

4.2.22 Thread Cutting Cycle G92

Command format: G92 X(U) _ Z(W) _ F_ J_ K_ L ; (Metric straight thread cutting cycle)
G92 X(U) _ Z(W) _ I_ J_ K_ L ; (Inch straight thread cutting cycle)
G92 X(U) _ Z(W) _ R_ F_ J_ K_ L ; (Metric taper thread cutting cycle)
G92 X(U) _ Z(W) _ R_ I_ J_ K_ L ; (Metric taper thread cutting cycle)

Command function: Tool infeeds in radial(X) direction and cuts in axial(Z or X, Z) direction from starting point of cutting to realize straight thread, taper thread cutting cycle with constant thread pitch. Thread run-out in G92: at the fixed distance from end point of thread cutting, Z executes thread interpolation and X retracts with exponential or linear acceleration, and X retracts at rapidly traverse speed after Z reaches to end point of cutting as Fig.4-2-24-1。

Command specifications:

G92 is modal;

Starting point of cutting: starting position of thread interpolation;

End point of cutting: end position of thread interpolation;

X: X absolute coordinate of end point of cutting, unit: mm;

U: different value of X absolute coordinate from end point to starting point of cutting, unit: mm;

Z: Z absolute coordinate of end point of cutting, unit: mm;

W: Different value of X absolute coordinate from end point to starting point of cutting, unit: mm;

R: Different value(radius value) of X absolute coordinate from end point to starting point of cutting. When the sign of R is not the same that of U, R|≤|U/2|, unit: mm;

F: Thread lead, its range: 0< F ≤500 mm. After F value is executed, it is reserved and can be omitted;

I: Thread teeth per inch, its range: 0.06tooth/inch～25400tooth/inch , it is reserved and it can be omitted not to input after I specified value is executed;

J: Movement in the short axis in thread run-out, its range 0~99999999× least input increment , unit: mm/inch, without direction (automatically define its direction according to starting position of program), and it is modal parameter. If the short axis is X, its value is specified by radius;

K: Movement in the long axis in thread run-out, its range: 0~99999999× least input increment , unit: mm/inch, without direction (automatically define its direction according to starting position of program), and it is modal parameter. If the long axis is X, its value is specified by radius;

L: Multi threads: 1～99 and it is modal parameter. (The system defaults it is single thread when L is omitted).

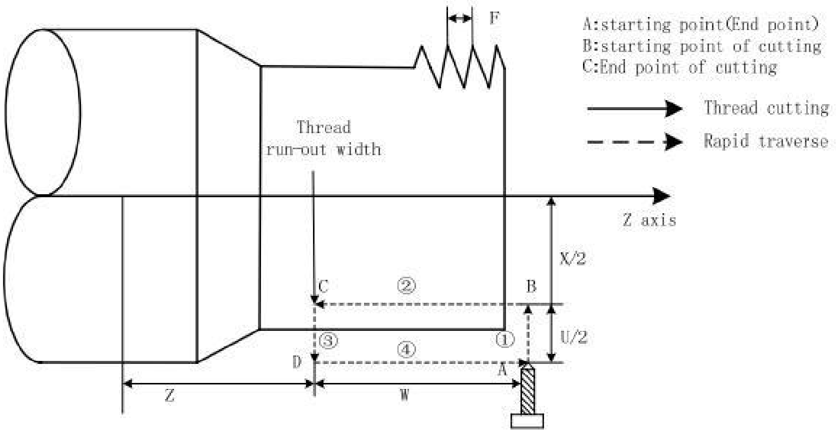


Fig. 4-2-24-1

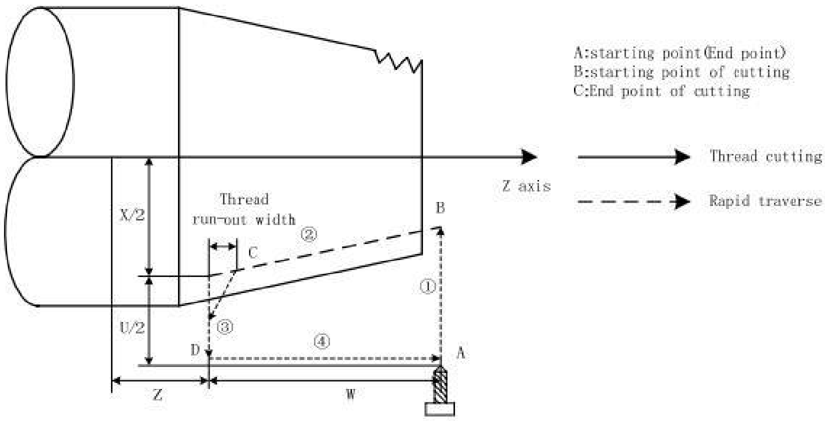


Fig. 4-2-24-2

The system can machine one thread with many tool infeed in G92, but cannot do continuous two thread and end face thread. Definition of thread pitch in G92 is the same that of G32, and a pitch is defined that it is a moving distance of long axis(X in radius) when the spindle rotates one rev.

Pitch of taper thread is defined that it is a moving distance of long axis(X in radius). When absolute value of Z coordinate difference between B point and C point is more than that of X (in radius), Z is long axis; and vice versa.

Cycle process: straight thread as Fig. 4-2-24-1, and taper thread as Fig4-2-24-2。

① X traverses from starting point to cutting starting point;

② Thread interpolates (linear interpolation) from the cutting starting point to cutting end point;

③ X retracts the tool at the cutting feedrate (opposite direction to the above-mentioned ①), and return to the position which X absolute coordinate and the starting point are the same;

④ Z rapidly traverses to return to the starting point and the cycle is completed.

Notes:

● Length of thread run-out is specified by №019 when J, K are omitted;

● Length of thread run-out is K in the long direction and is specified by №019 when J is omitted;
● Length of thread run-out is J=K when K is omitted;
● There is no thread run-out when J=0 or J=0, K=0;
● Length of thread run-out is J=K when J≠0,K=0;
● There is no thread run-out when J=0,K≠0;
● After executing the feed hold in thread cutting, the system does not stop cutting until the thread cutting is completed with Pause on screen;
● After executing single block in thread cutting, the program run stops after the system returns to starting point(one thread cutting cycle is completed);
● They are executed as the positive values when J, K negative values are input;
● Thread cutting decelerates to stop when the system resets, emergently stops or its driver alarms

Command path: relative position between thread cutting end point and starting point with U, W, R and tool path and thread run-out direction with different U, W, R signs below:

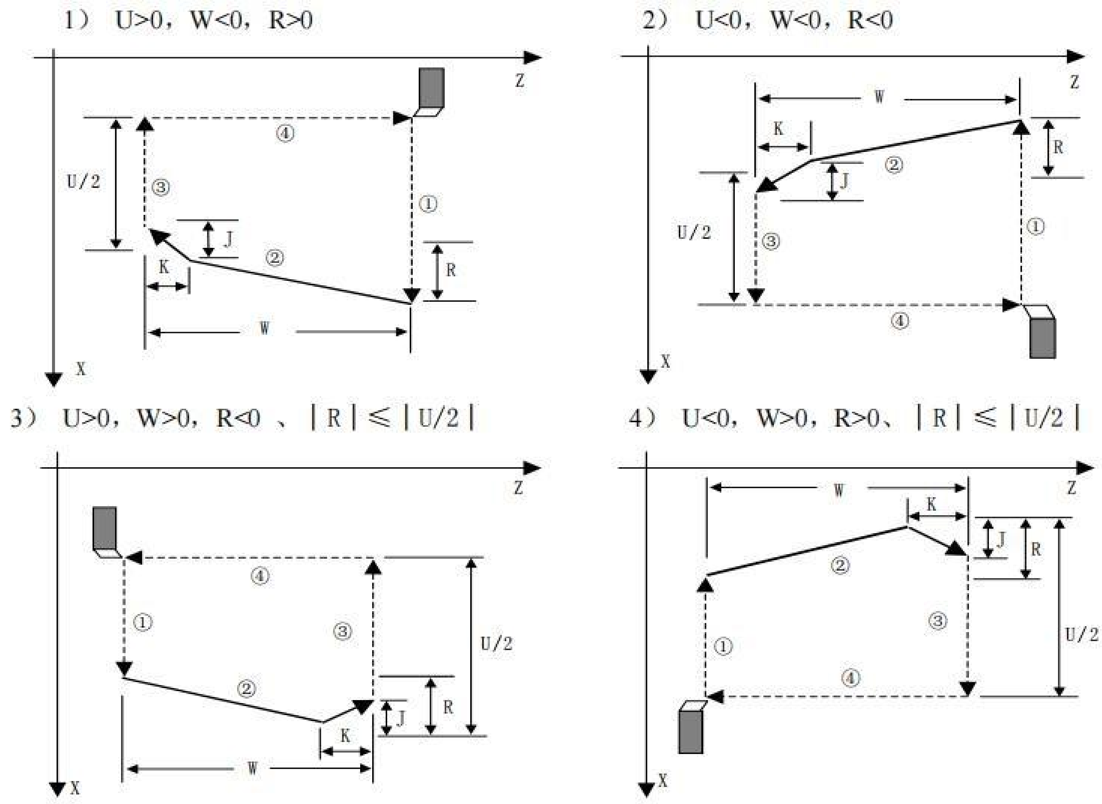
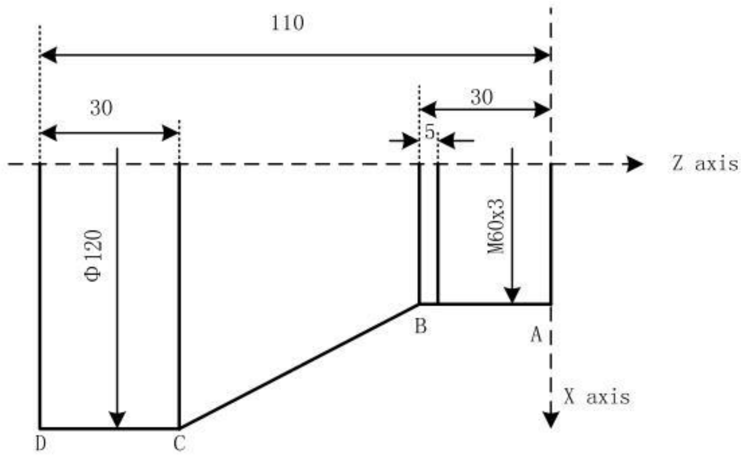


Fig. 4-2-24-3

Example:



Program: O0012;
M3 S300
G0 X150 Z50 T0101; (Thread tool)
G0 X65 Z5; (Rapid traverse)
G92 X58.7 Z-28 F3 J3 K1; (Machine thread with 4 times cutting, the first tool infeed 1.3mm)
X57.7 ; (The second tool infeed 1mm)
X57; (The third tool infeed 0.7mm)
X56.9; (The fourth tool infeed 0.1mm)
M30;

4.2.23 Multiple Thread Cutting Cycle G76

Command format: G76 P(m)(r)(a) Q(△dmin) R(d);
G76 X(U) Z(W) R(i) P(k) Q(△d) F(I) ;

Command function: Machining thread with specified depth of thread (total cutting depth) is completed by multiple roughing and finishing, if the defined angle of thread is not 0°, thread run-in path of roughing is from its top to bottom, and angle of neighboring thread teeth is the defined angle of thread. G76 can be used for machining the straight and taper thread with thread run-out path, which is contributed to thread cutting with single tool edge to reduce the wear of tool and to improve the precision of machining thread. But G76 cannot be used for machining the face thread. Machining path is shown in Fig. 4-2-25-1

Command specifications:

Starting point (endpoint): The position of the program segment before and at the end of operation, represented as point A;

Thread end point: The thread cutting end point defined by X (U) Z (W), represented as point D. If there is thread tailing, the cutting end point
The long axis direction is the end point of thread cutting, and the short axis direction is the position after retreating.

Thread starting point: The absolute coordinates of the Z-axis are the same as point A, and the difference between the absolute coordinates of the X-axis and the X-axis of point D is

i (thread taper and radius values), denoted as point C. If the defined thread angle is not 0 ° , point C cannot be reached during cutting;
Thread cutting depth reference point: The Z-axis absolute coordinate is the same as point A, and the difference between the X-axis absolute coordinate and the C-axis X-axis absolute coordinate is k (the total cutting depth and radius value of the thread), denoted as point B. The thread cutting depth of point B is 0, which is the reference point for the system to calculate each thread cutting depth;
Thread cutting depth: The cutting depth of each thread cutting cycle. The intersection point between the reverse extension line of each thread cutting trajectory and the straight line BC, and the difference (unsigned, radius value) between this point and the absolute coordinate of the X-axis of point B, is the thread cutting depth. The thread cutting depth for each rough turning is n × △ d, n is the current number of rough turning cycles, and △ d is the thread cutting depth of the first rough turning;
Thread cutting amount: The difference between the current thread cutting depth and the previous thread cutting depth ;
End point of tool retraction: The end point position of the radial (X-axis) tool retraction after the end of thread cutting in each coarse and fine thread turning cycle, denoted as point E;
Thread cutting point: The actual starting point of thread cutting in each thread rough turning cycle and precision turning cycle, denoted as Bn point (n is the number of cutting cycles), B1 is the first thread rough turning point, Bf is the last thread rough turning point, and Be is the thread precision turning point. The displacement of point Bn relative to the X and Z axes of point B follows the formula:

$$tg\frac{a}{2} = \frac{|\text{ Z axis displacement }|}{|\text{ X axis displacement }|}$$

X: X absolute coordinate (unit: mm) of thread end point;
U: Different value (unit: mm) of X absolute coordinate between thread end point and starting point;
Z: Z absolute coordinate (unit: mm) of thread end point;
W: Different value (unit: mm) of Z absolute coordinate between thread end point and starting point;
P(m): Times of thread finishing: 00～99 (unit: times). The value of system parameter №043 is regarded as finishing times when m is not input. In thread finishing, every feed cutting amount is equal to the cutting amount d in thread finishing dividing the finishing times m;
P(r): Width of thread run-out 00～99(unit: 0.1×L,L is the thread pitch). The value of system parameter №005 is the width of thread run-out when r is not input. The thread run-out function can be applied to thread machining without tool retraction groove and the width of thread run-out defined by system parameter №005 is valid for G92, G76;
P(a): Angles at taper of neighboring two tooth, range: 00～99, unit: deg(°). The system parameter №044 value is regarded as angle of thread tooth. The actual angle of thread in defined by tool ones and so a should be the same as the tool angle;
△ △Q(dmin): Minimum cutting travel of thread roughing, range: 0 ～ 999999(IS-C)/ 0 ～ 99999(IS-B),(unit: least input increment, radius value). When (n − n −1) × d △ <△dmin, dmin △ is regarded as the cutting travel of current roughing and the subsequent rough-machining cutting value and the in-feed depth

will not calculate based upon the formula any more. Setting dmin is to △ avoid the too small of roughing amount and too many roughing times caused by the cutting amount deceleration in thread roughing. When Q (dmin) △ is not input, the system data parameter NO.045 value is taken as the least cutting amount;

R(d): It is the cutting amount in thread finishing, range: 00～99.999(IS_B) /00～99.9999(IS_C) (unit: mm/inch, radius value without sign symbols), the radius value is equal to X absolute coordinates between cut-in point Be of thread finishing and Bf of thread roughing.. The value of system parameter №046 is regarded as the cutting travel of thread finishing when R(d) is not input;

R(i): It is thread taper and is the different value of X absolute coordinate between thread starting point and end point, rang: -99999.999~99999.999(IS_B) /-9999.9999~9999.9999(IS_C) (unit:: mm/inch, radius value). The system defaults R(i)=0(straight thread) when R(i) is not input;

P(k): Depth of thread tooth, the total cutting depth of thread, range: 1～99999999(unit: least input increment, radius value, without sign symbols). The system alarms when P(k) is not input;

Q(d): △ Depth of the 1st thread cutting, range: 1～99999999 unit: Least input increment, radius value, without sign symbols). The system alarms when d △ is not input; F: metric thread pitch, its range is referred to Table 1-2;

I: thread teeth per inch for inch thread, its range is referred to Table 1-2.

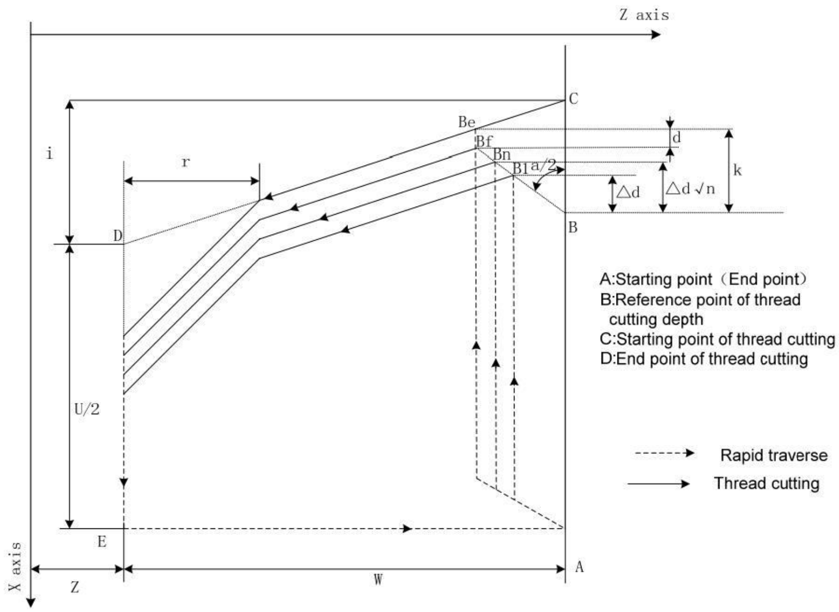


Fig. 4-2-25-1

Cut-in method as follows: Fig. 4-2-25-2

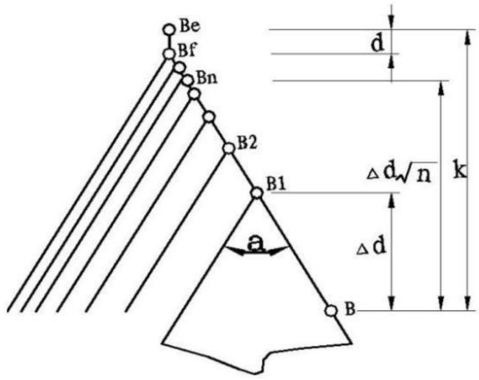


Fig. 4-2-25-2

Pitch is defined to moving distance (X radius value) of long axis when the spindle rotates one rev. Z is long when absolute value of coordinate difference between C point and D point in Z direction is more than that of X direction (radius value, be equal to absolute value of i); and vice versa

Execution process: ① The tool rapidly traverses to B1, and the thread cutting depth is △d. The tool only traverses in X direction when a=0; the tool traverses in X and Z direction and its direction is the same that of A→D when a≠0;

② The tool cuts threads paralleling with C→D to the intersection of D→E (r≠0: thread run-out);

③ The tool rapidly traverses to E point in X direction;

④ The tool rapidly traverses to A point in Z direction and the single roughing cycle is completed;

⑤ The tool rapidly traverses again to tool infeed to Bn (is the roughing times), the cutting depth is the bigger value of (n ×△d) ,(n -1 ×△d+△dmin) , and execute ② if the cutting depth is less than(k-d) ; if the cutting depth is more than or equal to(k-d) , the tool infeeds(k-d) to Bf, and then execute ⑥ to complete the last thread roughing;

⑥ The tool cuts threads paralleling with C→D to the intersection of D→E (r≠0: thread run-out);

⑦ X axis rapidly traverses to E point;

⑧ Z axis traverses to A point and the thread roughing cycle is completed to execute the finishing;

⑨ After the tool rapidly traverses to B(the cutting depth is k and the cutting travel is d), execute the thread finishing, at last the tool returns to A point and so the thread finishing cycle is completed;

⑩ If the finishing cycle time is less than m, execute ⑨ to perform the finishing cycle, the thread cutting depth is k and the cutting travel is 0; if the finishing cycle times are equal to m, G76 compound thread machining cycle is completed.

Notes:

1) In thread cutting, execute the feed hold, the system displays Pause after the thread cutting is executed completely, and then the program run pauses;

2) The thread cutting decelerates to stop when the system resets and emergently stop or the driver alarms;

3) Omit all or some of G76 P(m) (r) (a) Q(△dmin) R(d) . The omitted address runs according to setting value of parameters;

4) m, r, a used for one command address P are input one time. Program runs according to setting value of №57, 19, 58 when m, r, a are all omitted; Setting value is a when address P is input with 1 or 2 digits; setting values are r, a when address P is input with 3 or 4 digits;

5) The direction of A→C→D→E is defined by signs of U,W , and the direction of C→D is defined by the sign of R(i) . There are four kinds of sign composition of U, W corresponding to four kinds of machining path as Fig. 3-96. Example: Fig4-2-24-3。

Thread M68×6。

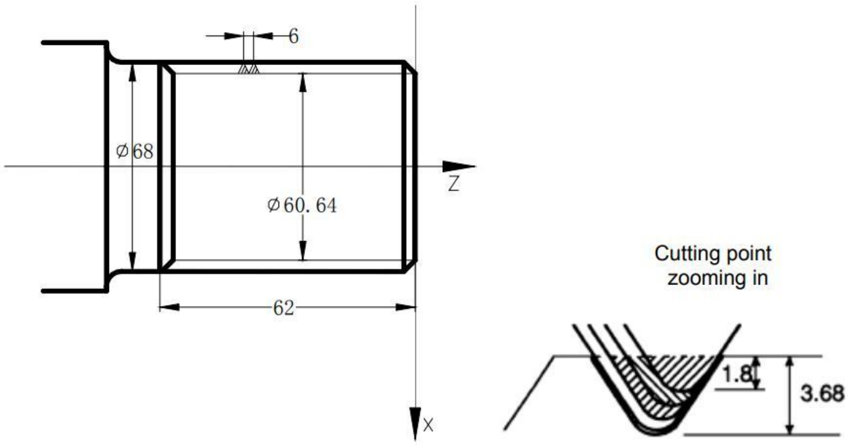


Fig. 4-2-25-3

Program:O2253;

G50 X100 Z50 M3 S300; (Set workpiece coordinate system, start spindle and specify spindle speed)

G00 X80 Z10; (Rapid traverse to starting point of machining)

G76 P020560 Q150 R0.1; ((Finishing 2 times, chamfering width 0.5mm, tool angle 60°, min. cutting depth 0.15, finishing allowance 0.1)

G76 X60.64 Z-62 P3680 Q1800 F6; (Tooth height 3.68, the first cutting depth 1.8)

G00 X100 Z50 ; (Return to starting point of program)

M30; (End of program)

4.2.24 Constant Surface Speed Control G96, Constant Rotational Speed Control G97

The detailed is referred to Section 6.3.

4.2.25 Feedrate per Minute G98, Feedrate per Rev G99

Command format: G98 F_; (its range is referred to Section 1.6.5, the leading zero can be omitted, feed rate per minute is specified, mm/min)

Command function: cutting feed rate is specified as mm/min, G98 is the modal G command. G98 cannot be input if the current command is G98 modal.

Command format: G99 F_; (its range is referred to Section 1.6.5, the leading zero can be omitted)

Command function: Cutting feed rate is specified as mm/min, G99 is the modal G command. G99 input may be omitted if current state is G99. The actual cutting feedrate is gotten by multiplying the F command value (mm/r) to the current spindle speed(r/min). If the spindle speed varies, the actual feedrate changes too. If the spindle cutting feed amount per rev is specified by G99 FXXXX , the even cutting texture on the surface of workpiece will be gotten. In G99 state, a spindle encoder should be fixed on the machine tool to machine the workpiece. 在 G99 模态进行加工, G98, G99 are the modal G commands in the same group and only one is valid. G98 is the initial state G command and the system defaults G98 is valid when the system turns on.

Reduction formula of feed between per rev and per min:

Fm = Fr×S

Fm: feed per min (mm/min) ;

Fr: feed per rev (mm/r) ;

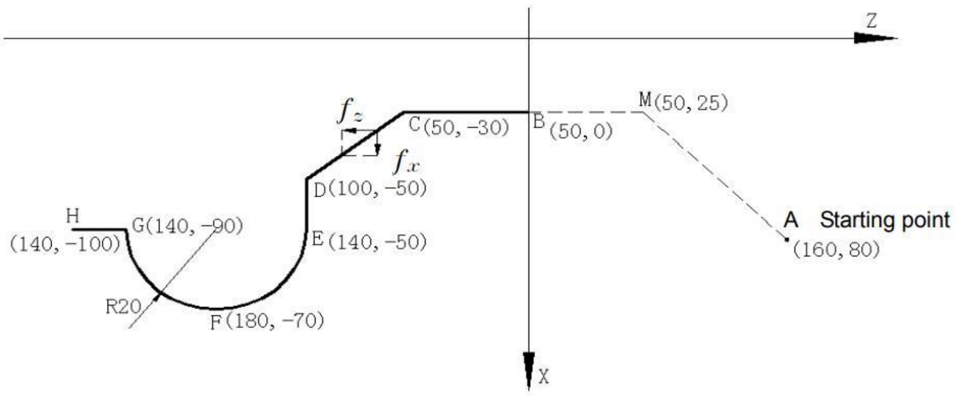
S: spindle speed (r/min) .

After the system turns on, the feedrate is ones set by №062 and F value is reserved after F is executed. The feed rate is 0 after F0 is executed. F value is reserved when the system resets and emergently stops. The feedrate override is reserved when the system is turned off.

Note: In G99 modal, there is the uneven cutting feed rate when the spindle speed is lower than 1 r/min; there is the follow error in the actual cutting feed rate when there is the swing in the spindle speed. To gain the high machining quality, it is recommended that the selected spindle speed should be not lower than min. speed of spindle servo or converter.

Cutting feed: The system can control the motions in X, Z direction contributed that the motion path of tool and the defined path by commands (line straight, arc) is consistent, and also instantaneous speed on the tangent of motion path and F word is consistent, which motion control is called cutting feed or interpolation. The cutting feedrate is specified by F, the system divides the cutting feedrate specified by F according to the programming path into vector in X, Z direction, also controls the instantaneous speed in X, Z direction to contribute that the combined speed of vector in X, Z direction is equal to F command value.

Example: The coordinates of each point are shown in parentheses (the X-axis represents the diameter value), with a fast moving speed of 3600 on the X-axis and 7200 on the Z-axis. Both the fast and feed rates are 100%.



Program as follows:

G50 X160 Z80；（Create a workpiece coordinate system）

G0 G98 X50 Z0；(Rapid traverse from A to B through M point. A→M: X-axis rapid traverse speed 7600mm/min, Z-axis 7600mm/min in Z direction, M→B: X-axis rapid traverse
speed 0mm/min, Z-axis 7600mm/min in Z direction ）

G1 W-30 F100；（B→C, X-axis rapid traverse speed 0mm/min, Z-axis 100mm/min）

X100 W-20； （C→D, X-axis rapid traverse speed 156mm/min, Z-axis 62mm/min）

X140； （D→E, X-axis rapid traverse speed 200mm/min, Z-axis 0mm/min）

G3 W-40 R20；（(EFG circular interpolation, E point: X axis instantaneous speed 200mm/min,Z axis 0mm/min
F point: X-axis instantaneous speed 0mm/min, Z axis 100mm/min）

W-10； （G→H, X axis rapid traverse speed 0 mm/min, Z axis 100mm/min）

M30；

4.2.26 Drilling cycle code

Command format: G83 X (U)_ Y (V)_ Z (W)_ R_ P_Q_ F_ K_ M_ ; End face drilling cycle

Command function:X_ Y_或 Z_ Y_：The hole position data is only valid in the specified program segment; Effective axes other than the X and Z axes can also be specified in the hole position data.

The hole position data is only valid in the specified program segment; Effective axes other than the X and Z axes can also be specified in the hole position data.

R_: The distance from the initial plane to the R-point, the radius value, and the direction.

P_: Hole bottom pause time, in 1ms.

Q_: Each cutting amount, radius value,

F_: Cutting feed rate,

K_: Number of program executions

M: M code for Y-axis clamping (when required).

G80: Cancel fixed loop.

Example:

O00002

G98； （Feed method per minute）

G0 X0 Z10； （Positioning the X-axis and Z-axis to the starting point）

M03 S500； （The spindle speed is 500 rpm ）

G83 Z-60 P1000 F100； （G83 is the end face drilling cycle, starting from X0 Z10, with hole bottom position at X0 Z-60 The bottom pause time is 1 second, and the cutting feed speed is F100）

G80； （Cancel fixed loop.）

M30； (End of program)

Parameter Description:

213.1=1 For the deep hole drilling cycle command, 241 is the retraction amount=0

G83 Z-20 Q3 R10

Fast retraction to Z10=R10, G83 R3 feed=Z7, fast retraction to Z10=R10, then fast retraction to Z7, then G83 R3 feed=Z4, fast retraction Z10=R10, then fast retraction to Z4, then G83 R3 feed=Z1,

213.1=1 For the deep hole drilling cycle command, 241 is the retraction amount=5

G83 Z-20 Q3 R10

Quickly retract to Z10=R10, G83 R3 feed=Z7, quickly retract to Z12=Z7+retraction amount 5, move G83 R3 feed=Z4, quickly retract to Z10=R10, then quickly retract to Z9=Z4+retraction amount 5, move G83 R3 feed=Z1, quickly retract to Z10=R10, and then quickly advance to Z6=Z1+retraction amount 5, move G83 R3 feed to=Z-2

213.1=0 is the drilling cycle command, 241 is the retraction amount=0

G83 Z-20 Q3 R10

Fast retraction to Z10=R10, G83 R3 feed=Z7, fast retraction to Z10=R10, then fast retraction to Z7, then G83 R3 feed=Z4, then fast retraction to Z7, then fast retraction to Z4, then G83 R3 feed=Z1, then fast retraction to Z4, fast retraction to Z1, then G83 R3 feed=Z-2

213.1=0 is the drilling cycle command, 241 is the retraction amount=5

G83 Z-20 Q3 R10

Quickly retract to Z10=R10, G83 R3 feed=Z7, quickly retract to Z12=R3+retract amount 5, then move G83 R3 feed=Z4, then quickly retract to Z9=Z4+retract amount 5, then move G83 R3 feed=Z1, then quickly retract to Z6=Z1+retract amount 5, and then move G83 R3 feed=Z-2

4.3 TOOL NOSE RADIUS COMPENSATION (G41, G42)

4.3.1 Overview

Part program is compiled generally for one point of tool according to a workpiece contour. The point is generally regarded as the tool nose A point in an imaginary state (there is no imaginary tool nose point in fact and the tool nose radius can be omitted when using the imaginary tool nose point to program) or as the center point of tool nose arc (as Fig. 4-3-1-1). Its nose of turning tool is not the imaginary point but one arc owing to the processing and other requirement in the practical machining. There is an error between the actual cutting point and the desired cutting point, which will cause the over- or under-cutting affecting the part precision. So a tool nose radius compensation is needed in machining to improve the part precision. 。

Fig. 4-3-1-1 Tool

The method of offsetting the trajectory of the part shape by one tool tip radius is the B-type tool compensation method. This method is simple, but it only processes the motion trajectory of the next program segment after executing one program segment, resulting in overfitting and other phenomena at the intersection of the two programs.

To solve the above problems and eliminate errors, it is necessary to establish a C-type tool compensation method. The C-type tool compensation method does not immediately execute when reading a program segment, but instead reads the next program segment and calculates the corresponding motion trajectory (transfer vector) based on the connection of the intersection points of the two program segments. Due to reading two program segments for preprocessing, the C-type tool compensation method can provide more accurate compensation on the contour. As shown in Figures 4-3-1-2

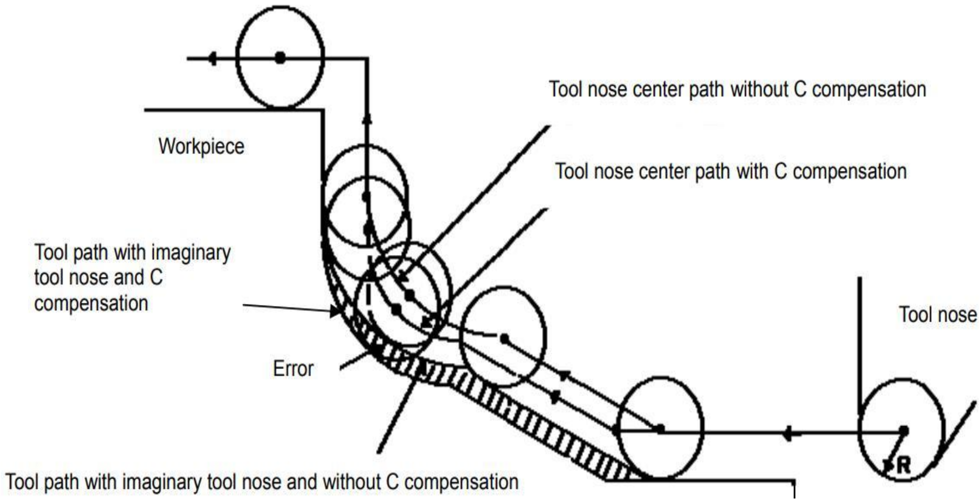
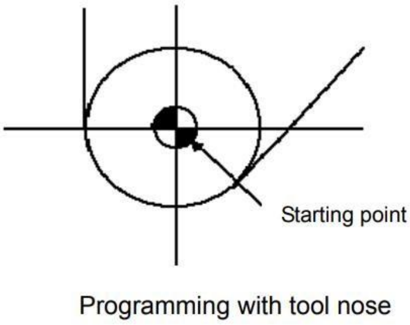


Fig. 4-3-1-2

4.3.2 Imaginary Tool Nose Direction

The setting of the hypothetical tool tip is because it is generally difficult to set the center of the tool tip radius at the starting position, as shown in Figure 4-3-2-1; It is relatively easy to assume that the blade tip is set at the starting position, as shown in Figure 4-3-2-2; The tool tip radius may not be considered during programming.



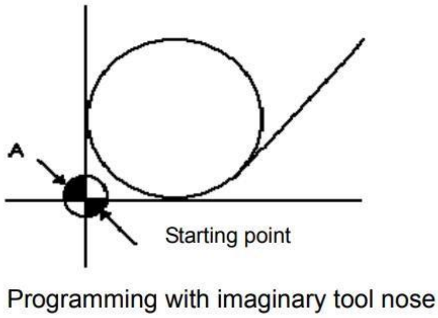


图 4-3-2-1　　　　　　　　　　　　　　图 4-3-2-2

Figures 4-3-2-3 and 4-3-2-4 show the comparison of tool trajectory maps with and without tool tip radius compensation when programming with the center of the tool tip and with an imaginary tool tip, respectively.

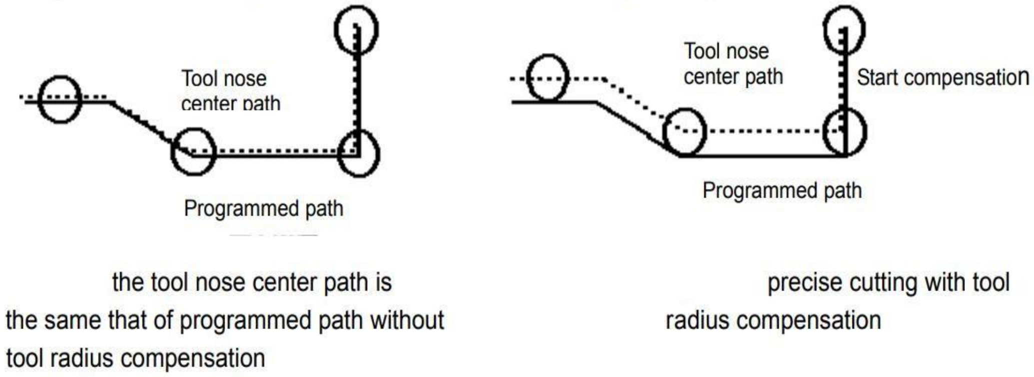


图 4-3-2-3 Tool trajectory when programming with the center of the tool tip

Using tip radius compensation will achieve precision cutting

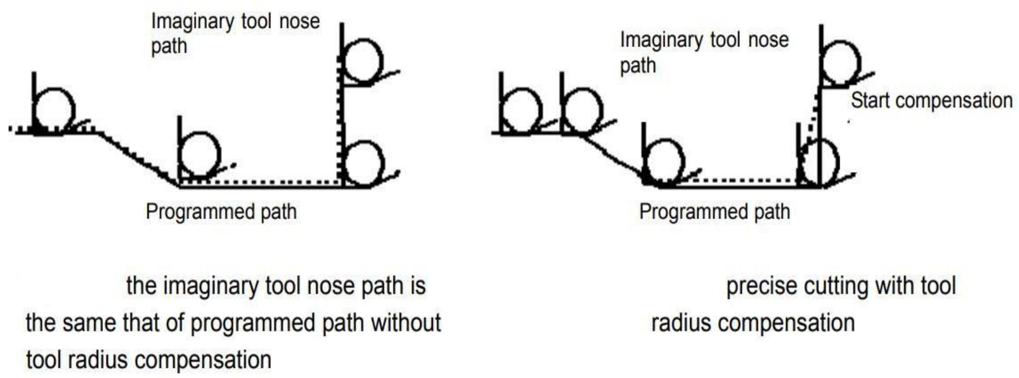


Fig. 4-3-2-4 Tool trajectory when programming with hypothetical tool tips

In the programming process, the tool is assumed to be a point, while the actual cutting edge cannot be an ideal point due to process requirements or other reasons. The machining error caused by the cutting edge being not an ideal point but a segment of an arc can be eliminated by the tool tip arc radius compensation function. In actual machining, there is a different positional relationship between the imaginary tool tip point and the center point of the tool tip arc. Therefore, it is necessary to correctly establish the direction of the imaginary tool tip (i.e., which position of the tool is the corresponding tool point).

From the center of the tool tip to the direction of the imaginary tool tip, the imaginary tool tip number is determined by the direction of the cutting tool. There are a total of 10 settings (T0-T9) for the hypothetical tool tip, which represent the positional relationships in 9 directions. It should be noted that even if the same tool tip direction number represents different tool tip directions in different coordinate systems (rear tool seat coordinate system and front tool seat coordinate system), as shown in the following figure. The figure illustrates the relationship between the tool tip and the starting point, with the arrow ending at an imaginary tool tip; The situation of the coordinate system T1~T8 of the rear cutterbed, as shown in Figure 4-3-2-5; The coordinate system T1~T8 of the front cutterbed is shown in Figure 4-3-2-6. T0 and T9 are the case when the center of the tool tip is consistent with the starting point, as shown in Figures

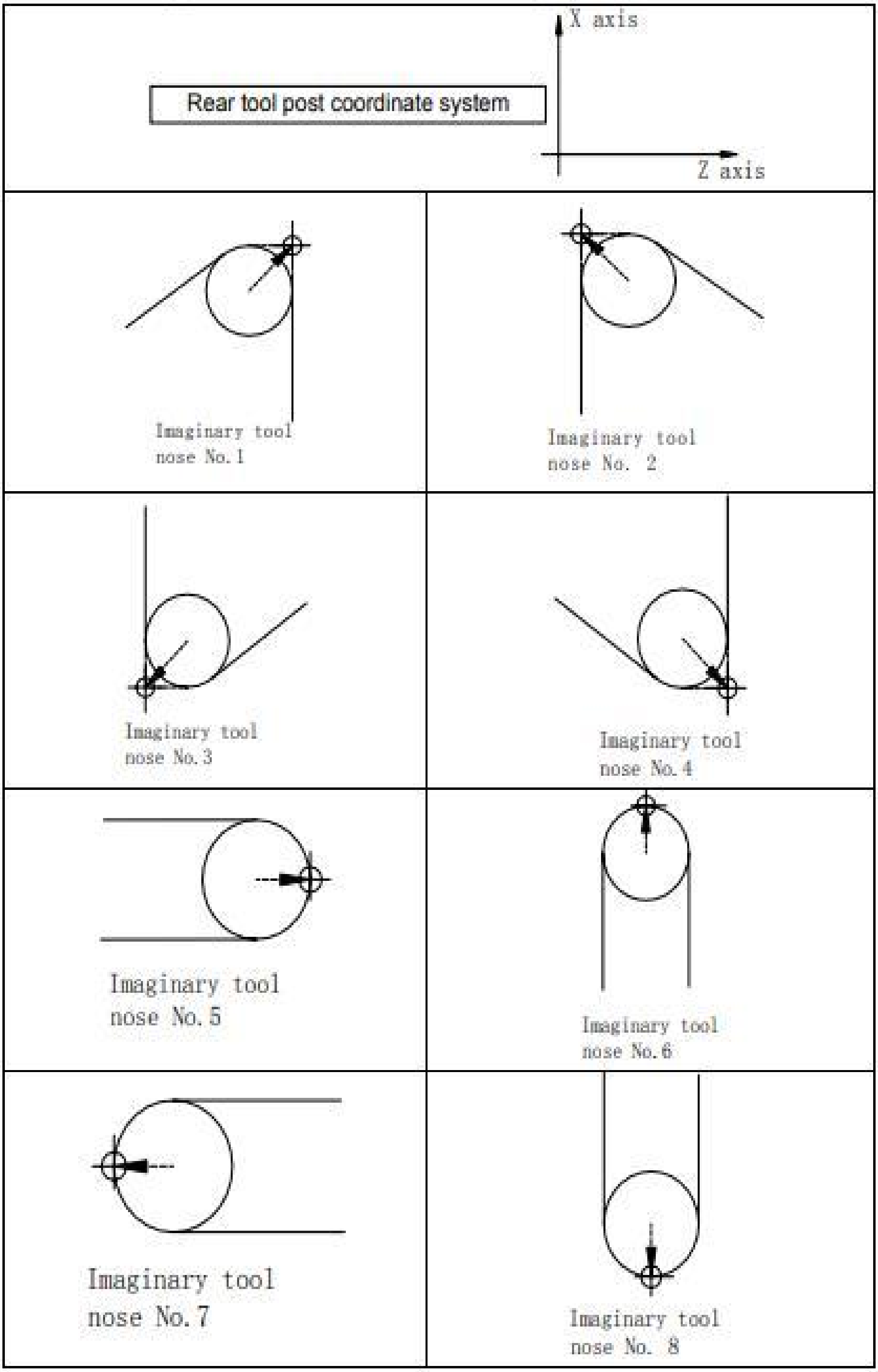

Rear tool post coordinate system
X axis
Z axis
Imaginary tool nose No.1
Imaginary tool nose No. 2
Imaginary tool nose No.3
Imaginary tool nose No.4
Imaginary tool nose No.5
Imaginary tool nose No.6
Imaginary tool nose No.7
Imaginary tool nose No. 8

Fig. 4-3-2-5

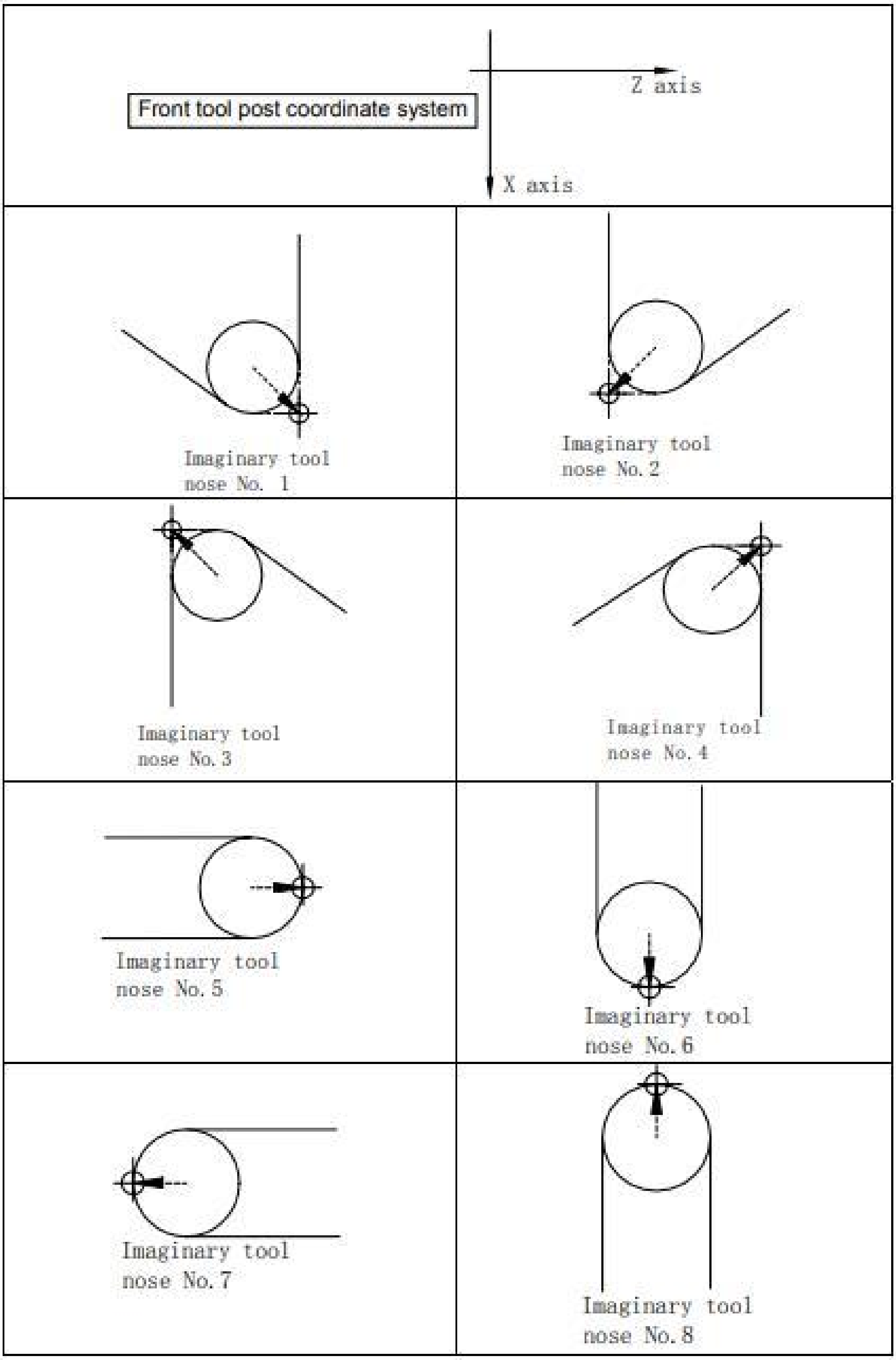


Fig. 4-3-2-6

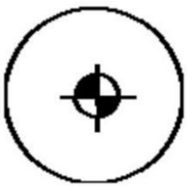


Fig. 4-3-2-7 Tool nose center on starting point

4.3.3 Compensation Value Setting

Preset imaginary tool nose number and tool nose radius value for each tool before executing tool nose radius compensation. Set the tool nose radius compensation value in OFFSET window (as Fig. 4-3-3-1), R is tool nose radius compensation value and T is imaginary tool nose number.

number	X	Z	R	T
001	5.100	0.000	0.6	8
002	170.000	10.000	0.000	4
003	-80.000	-5.000	1	5
....				
030	0.000	0.000	0.000	0

Fig. 4-3-3-1

Note: X tool offset value can be specified in diameter or radius, set by No.004 Bit4 ORC, offset value is in radius when ORC=1 and is in diameter when ORC=0.

In toolsetting, the tool nose is also imaginary tool nose point of Tn (n=0~9) when taking Tn(n=0~9) as imaginary tool nose. For the same tool, offset value from standard point to tool nose radius center (imaginary tool nose is T3) is different with that of ones from standard point to imaginary tool nose(imaginary tool nose is T3) when T0 and T3 tool nose points are selected to toolsetting in rear tool post coordinate system, taking tool post center as standard point. It is easier to measure distances from the standard point to the tool nose radius center than from the standard point to the imaginary tool nose, and so set the tool offset value by measuring distance from the standard point to the imaginary tool nose(tool nose direction of T3).

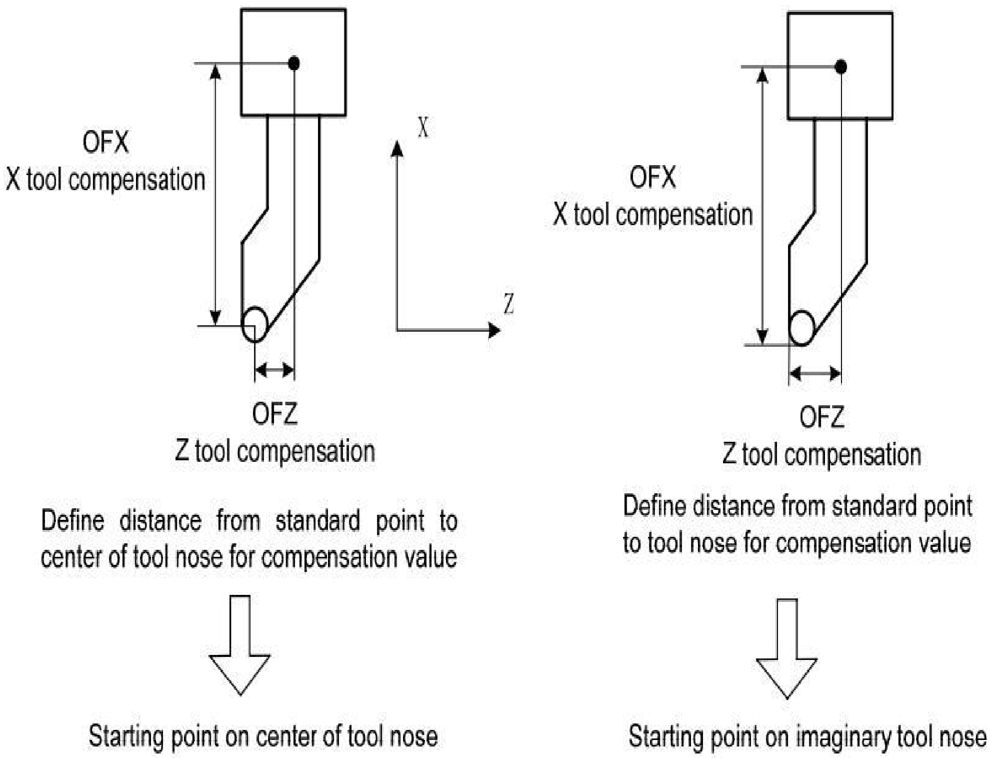


图 4-3-3-2 Tool offset value of tool post center as benchmark

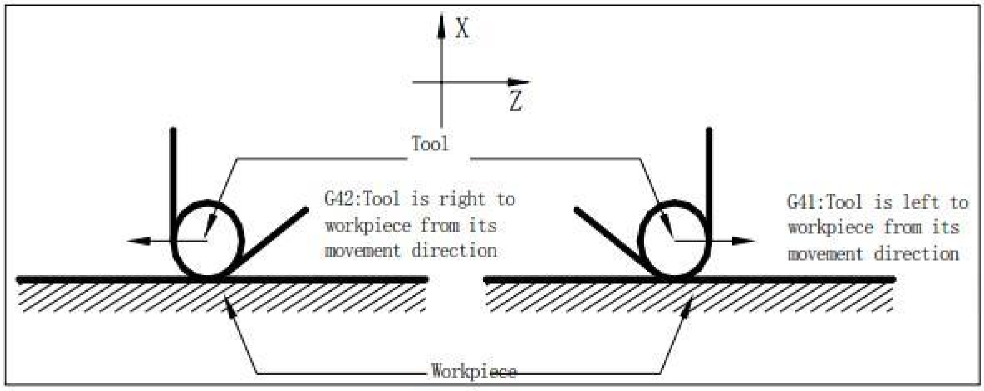
4.3.4 Command Format

$$\left\{\begin{matrix}\mathrm{G40}\\ \mathrm{G41}\\ \mathrm{G42}\end{matrix}\right\}\left\{\begin{matrix}\mathrm{G00}\\ \mathrm{G01}\end{matrix}\right\}\quad X\_\_Z\_\_T\_\_;$$

Commands	Function specifications	Remark
G40	Cancel the tool nose radius compensation	Please refer to the illustration for details
G41	Tool nose radius left compensation is specified by G41 in rear tool post coordinate system and tool nose radius right compensation is specified by G41 in front tool post coordinate system	
G42	Tool nose radius right compensation is specified by G42 in rear tool post coordinate system and tool nose radius left compensation is specified by G42 in front tool post coordinate system	

4.3.5 Compensation Direction

To apply tool tip radius compensation, the direction of compensation must be determined based on the relative position of the tool tip and the workpiece



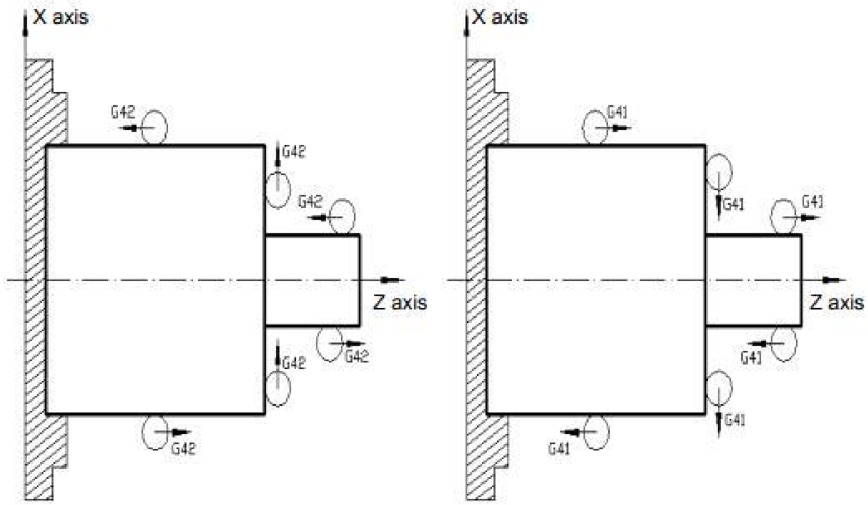
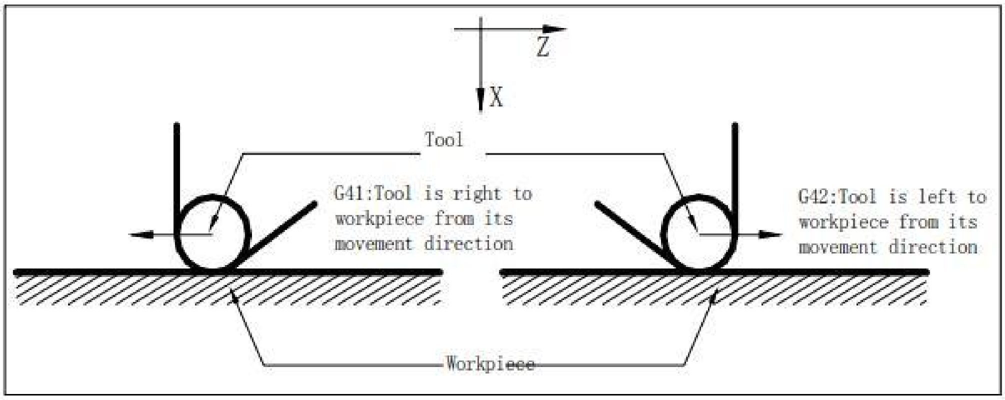


Fig. 4-3-5-1 Compensation direction of rear coordinate system



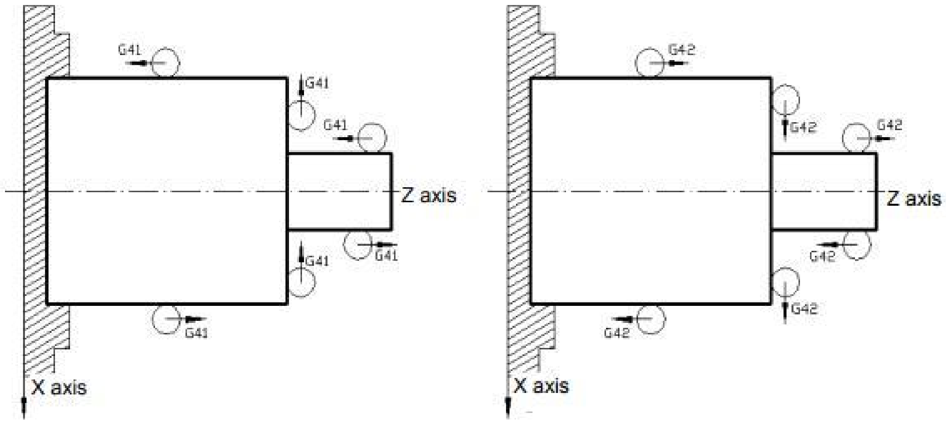


Fig. 4-3-5-2 **Compensation direction of front coordinate system**

4.3.6 Notes

- In the initial state, CNC is in the tool tip radius compensation cancellation mode. After executing G41 or G42 code, CNC begins to establish the tool tip radius compensation offset mode. At the beginning of compensation, CNC pre reads 2 program segments, and when executing one program segment, the next program segment is stored in the tool tip radius compensation buffer memory. During single segment operation, two program segments are read in, and the first program segment ends before stopping. During continuous execution, two program segments are pre read in, resulting in the program segment being executed in CNC and the subsequent two program segments.
- In tool tip radius compensation, when processing two or more program segments without moving code (such as auxiliary functions, pauses, etc.), the center of the tool tip will move to the endpoint of the previous program segment and be perpendicular to the position of the program path of the previous program segment.
- Under the input method (MDI), it is not possible to create or cancel tool compensation C.
- The R value of the tool tip radius cannot be negative, otherwise there will be an error in the running trajectory.
- The establishment and cancellation of tool tip radius compensation can only be done using G00 or G01 codes, and cannot be arc codes (G02 or G03). If specified, an alarm will be generated.

- After pressing the RESET key or executing M30, CNC will cancel the tool compensation C mode.
- G40 must be specified to cancel bias mode before the program ends. Otherwise, when executed again, the tool path deviates by one tool tip radius value.
- Using tool tip radius compensation in the main program and subroutines, before calling the subroutine (i.e. before executing M98), the CNC must establish tool compensation C again in the subroutine in compensation cancellation mode.
- The G71, G72, G73, G74, G75, and G76 codes do not perform tip radius compensation and temporarily cancel the compensation mode.
- G90 and G94 codes perform tool tip radius compensation, and both G41 and G42 are offset by one tool tip radius (according to the hypothetical tool tip number 0) for cutting

4.3.7 Application

Machine a workpiece in the front tool post coordinate system as Fig. 4-3-7-1. Tool number: T0101, tool nose radius R=2, imaginary tool nose number T=3

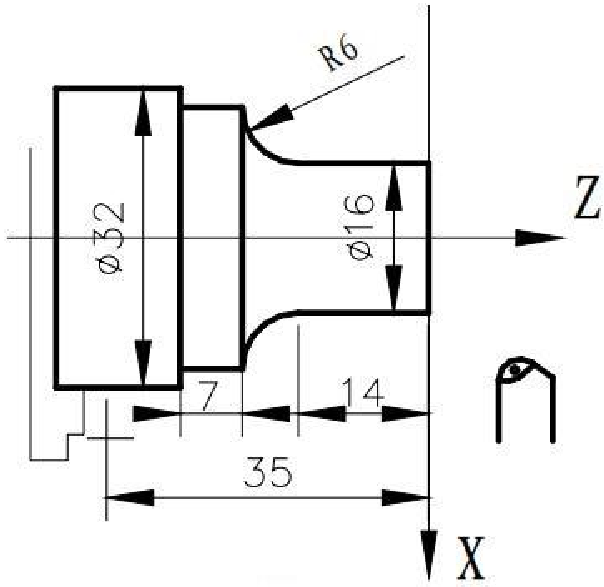


Fig. 4-3-7-1

For toolsetting in Offset Cancel mode, after toolsetting, Z axis offsets one tool nose radius and its direction is relative to that of imaginary tool nose and toolsetting point, otherwise the system excessively cuts tool nose radius when it starts to cut.

Set the tool nose radius R and imaginary tool nose direction in “TOOL OFFSET&WEAR” window as following:

序号	X	Z	R	T
001	5.100	0.000	2	3
002	170.000	10.000	0.000	4
003	-80.000	-5.000	1	5
....				
030	0.000	0.000	0.000	0

procedure: O2371

```
G00 X100 Z50 M3 T0101 S600; //Position, start spindle, tool change and execute
                                tool compensation
G42 G00 X0 Z3;  //Set tool nose radius compensation
G01 Z0 F300;  //Start cutting
X16;
Z-14 F200;
G02 X28 W-6 R6;
G01 W-7;
X32;
Z-35;
G40 G00 X90 Z40;  //Cancel tool nose radius compensation
G00 X100 Z50 T0100;
M30;
```

4.3.8 Tool radius compensation action process

According to the order of execution, the tool compensation process can be divided into three stages:

(1)Tool compensation establishment: The process of starting execution from the G40 method to the establishment of G41 or G42 code.

(2)Tool compensation process: The process from the establishment of Tool compensation to the cancellation of Tool compensation.

(3)Tool compensation cancellation: The process of switching from G41 or G42 to G40 mode.

Generally speaking, the machining trajectories that can be controlled in CNC systems are limited to straight lines and arcs. There are four connection methods between the two programming trajectories, namely:

(1)A straight line is connected to a straight line;

(2)Connect a straight line with an arc;

(3)The arc is connected to a straight line;

(4)The arc is connected to the arc.

According to the angle between the two program trajectories at the workpiece side α The tool radius compensation for linear transition can be divided into the following three transition methods:

(1)180° <= α < 360°, Shortened form;

(2)90° <= α < 180° , Elongated shape;

(3)0° <=α < 90° , Insert shape。

angle α It is called the transition angle, and its variation range is 0 ° <= α <360 ° , α The convention diagram of the corner is shown in Figure 4-3-8-1, α The angle is the angle between the two directions of motion at the workpiece side transition.

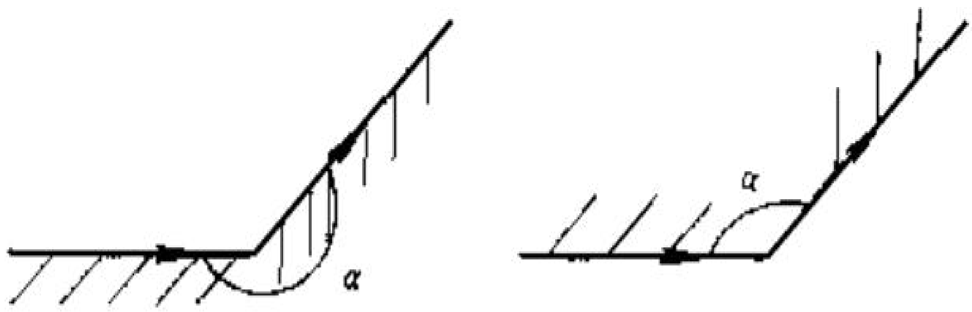


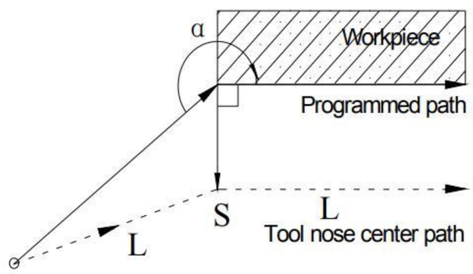
Fig. 4-3-8-1

4.3.9 Tool compensation establishment

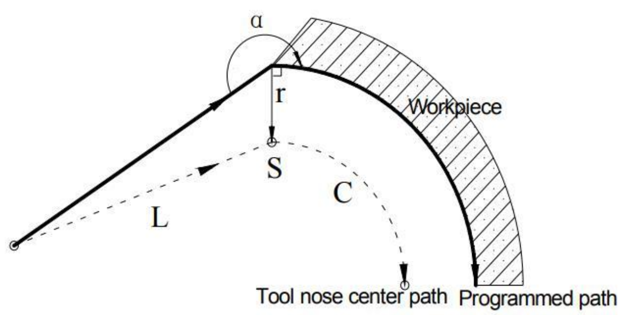
The establishment of Tool compensation includes three types: shortened type, elongated type, and inserted type. When the first trajectory is an arc, it is not allowed to perform a tool compensation establishment operation.

Shortened type(180° <= α < 360°) :

1.Linear → linear

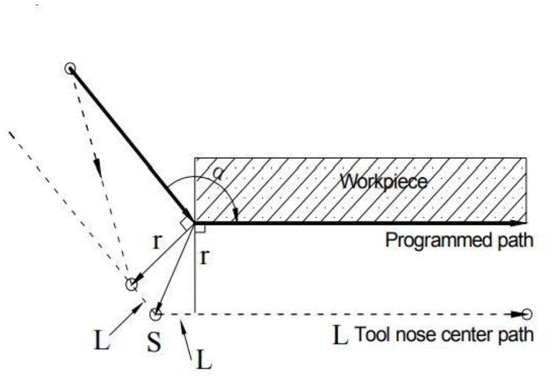


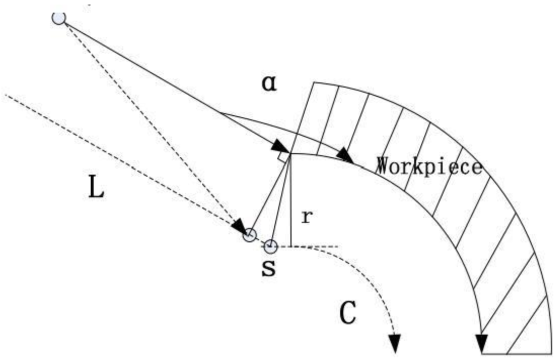
2.linear → circular



Elongate type(90° <= α < 180°):

1. Linear → linear　　　　　　　　　　　2. linear → circular

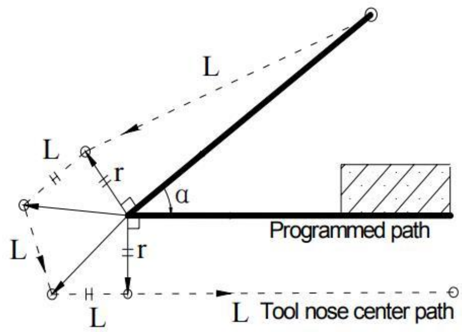


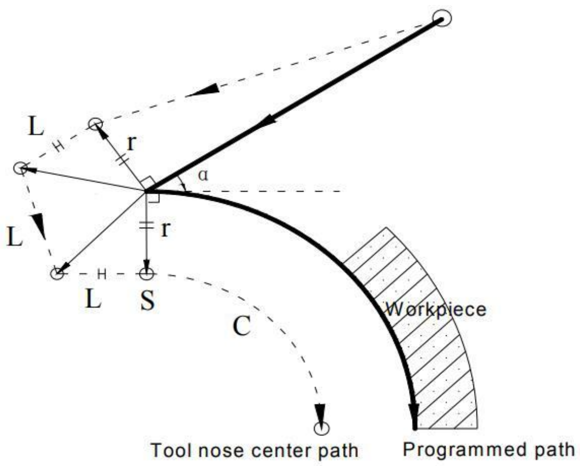


---

plug-in type（0° <= α < 90° ）:

1. Linear → linear　　　　　　　　　　　2. linear → circular



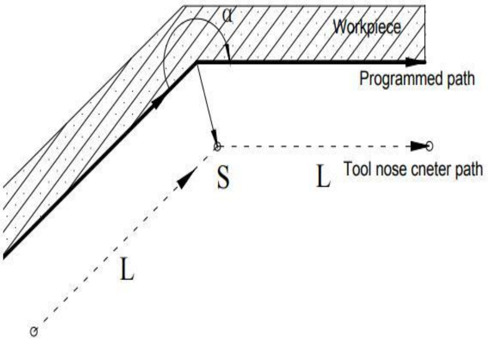


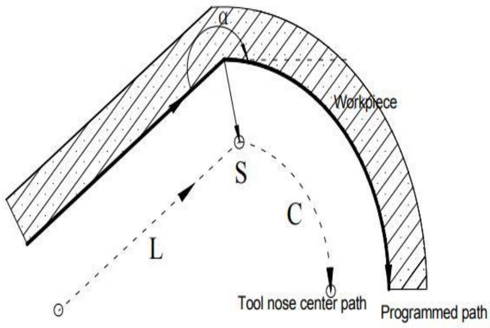
4.3.10 Tool Traversing in Offset Mode

Tool compensation can be carried out in three types: shortened type, extended type, and inserted type.

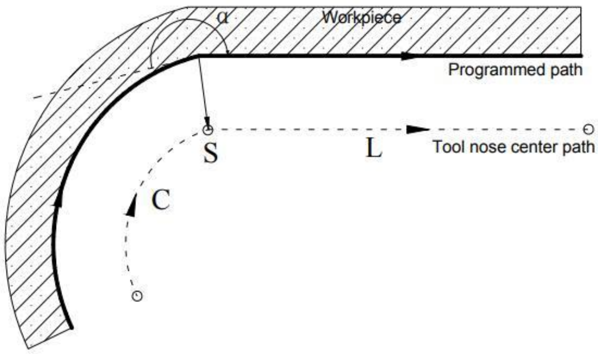
Shortened type（180° <= α < 360° ）:

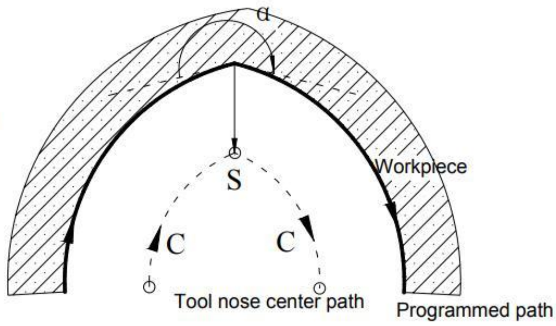
1. Linear → linear　　　　　　　　　　　2. linear → circular





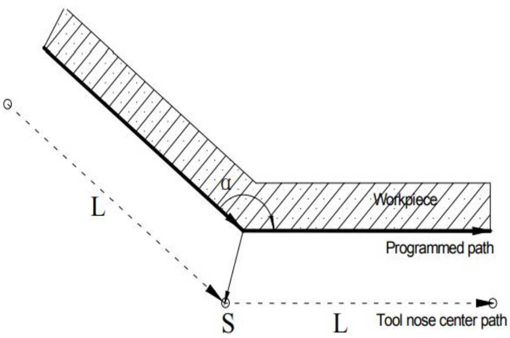
3.Circular→ linear 4.Circular→ circular

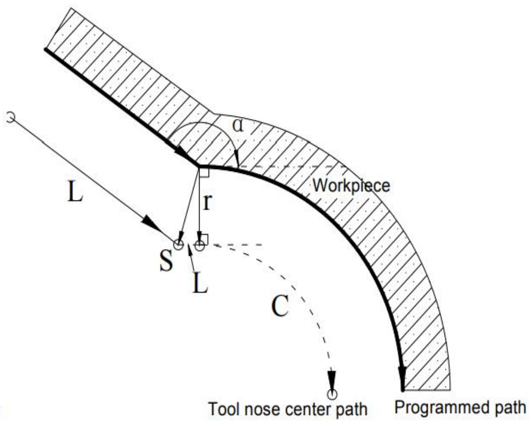




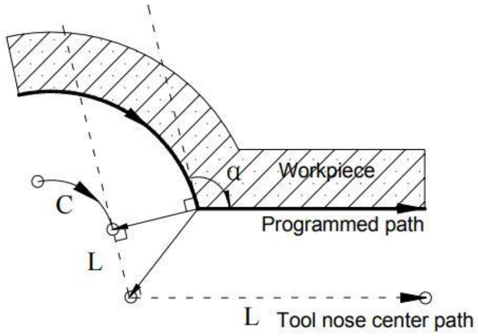
Elongate type(90° <= α < 180°):

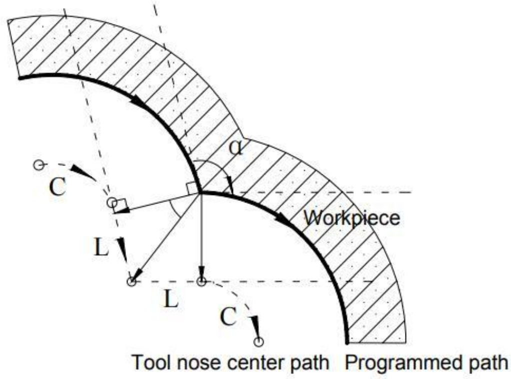
1.Linear → linear 2.linear → circular





3.Circular→ linear 4.Circular→ circular

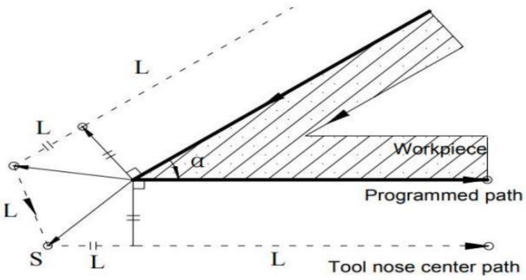




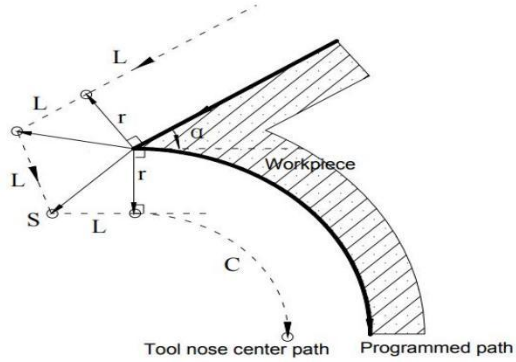
---

plug-in type（0°<= α < 90°）:

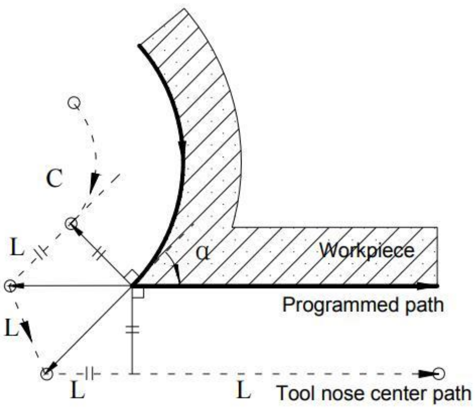
1.Linear → linear



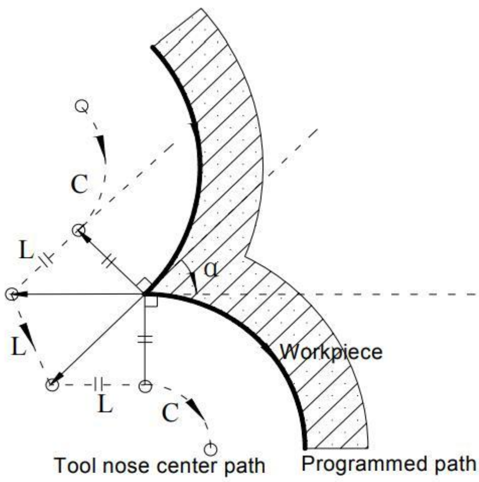
2.linear → circular



3.Circular→ linear



4.Circular→ circular

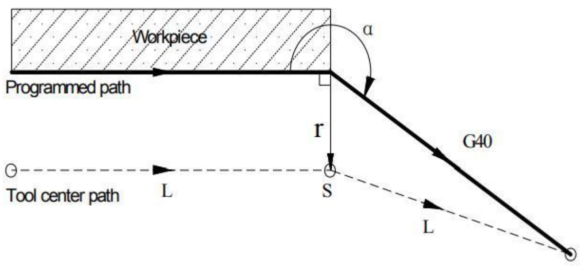


4.3.11 Tool Traversing in Offset Canceling Mode

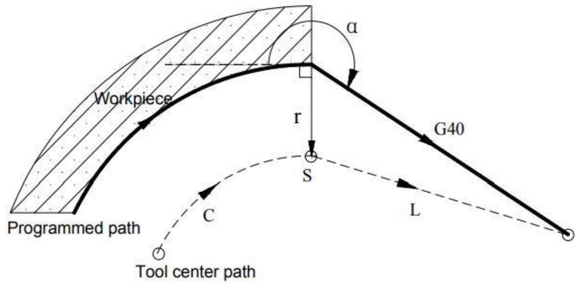
Tool compensation can be carried out in three types: shortened type, extended type, and inserted type. When the second trajectory is an arc, tool compensation cancellation is not allowed.

Shortened type(180° <= α < 360°):

1.Linear → linear

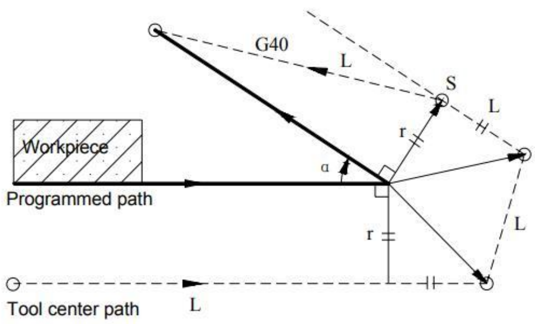


2.Circular→ linear

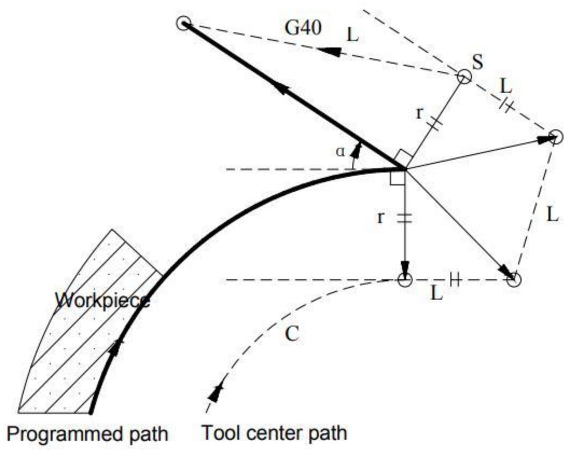


Elongate type(90° <= α < 180°):

1.Linear→linear

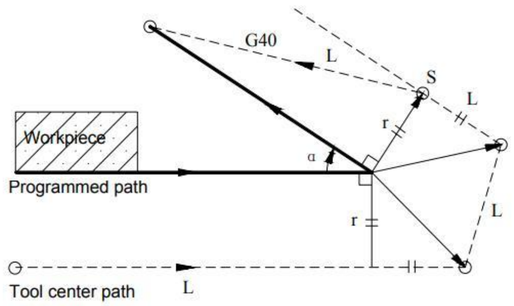


2.Circular→linear

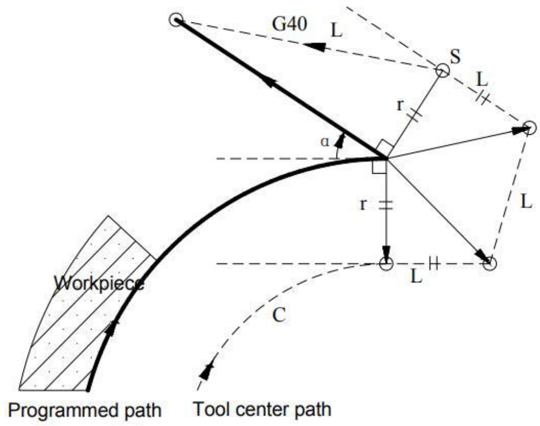


plug-in type（0° <= α < 90°）:

1. Linear→linear



2. Circular→linear



4.3.12 Tool Interference Check

Excessive cutting of the tool is called interference. The interference check function checks for tool overcutting in advance. But not all interference can be detected; Even if there is no overfitting, interference check will be conducted. When tool interference is found, the system will issue the first alarm: tool compensation overfitting error.

Criteria for detecting interference: 1. **The direction of the tool tip radius trajectory is different from the programmed trajectory direction (between 90 degrees and 270 degrees).** As shown in Figures 4-3-12-1:

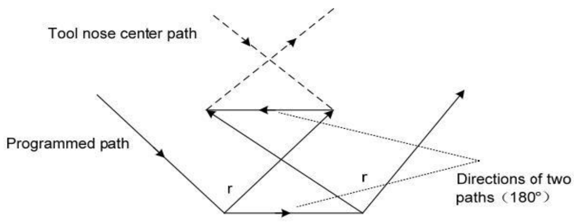


Fig. 4-16a Machining interference (1)

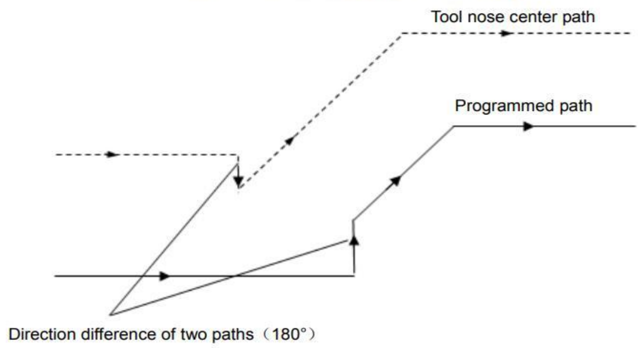
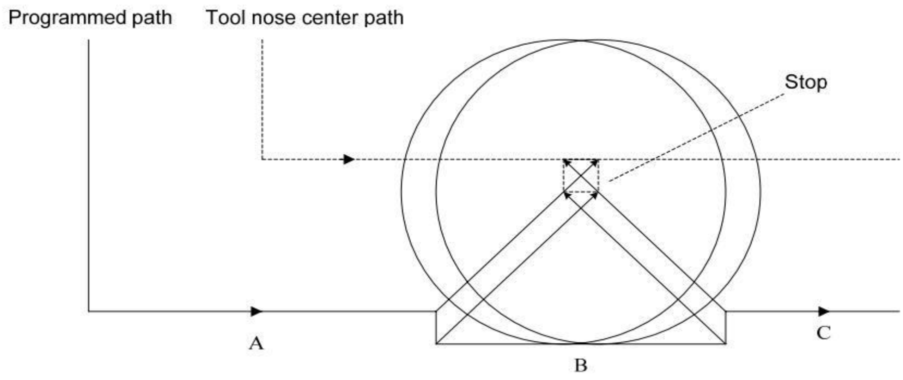


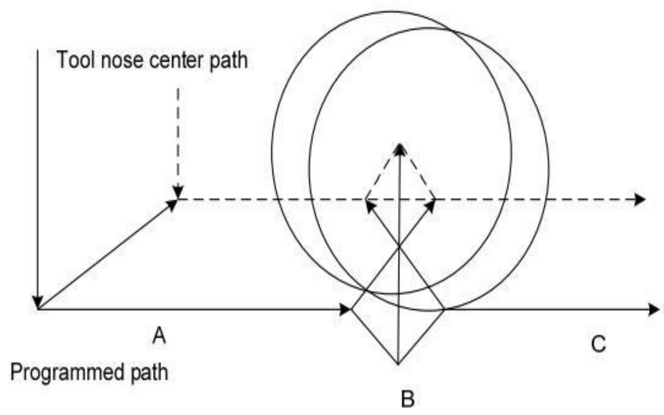
Fig. 4-3-12-1

2. **Executing it without actual interference**

(1)Concave groove less than compensation value



(2)Concave channel less than compensation value



Attention:

During the use of the tool radius compensation command, the following points need to be noted:

When the angle between the last two trajectories is less than 0.1 °, the system will issue an alarm "Tool compensation transition angle is less than the limit angle".

During the process of tool compensation, it is not allowed to directly execute the tool compensation in the opposite direction, such as directly switching from G41 mode to G42 mode, or switching from G42 mode to G41 mode. Otherwise, the system will issue an alarm.

During the tool compensation process, the T command cannot be executed to change the tool, otherwise the system will issue an alarm

4.3.13 Commands for Canceling Compensation Vector Temporarily

In compensation mode, the compensation vector is cancelled temporarily in G50, G71~G76 and is automatically resumed after executing the commands. At the moment, the compensation is cancelled temporarily and the tool directly moves from intersection to a point for canceling compensation vector. The tool directly moves again to the intersection after the compensation mode is resumed.

➢ Setting coordinate system in G50

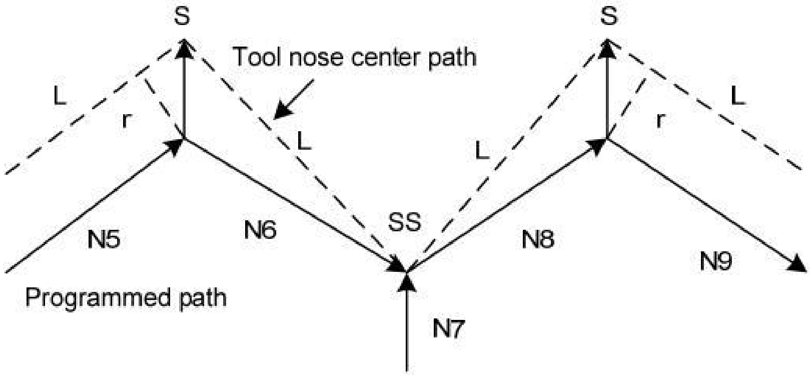


图 4-3-13-1 Temporary compensation vector in G50

Note: SS indicates a point at which the tool stops twice in Single mode.

➢ Reference point automatic return G28

In compensation mode, the compensation is cancelled in a middle point and is automatically resumed after executing the reference point return in G28.

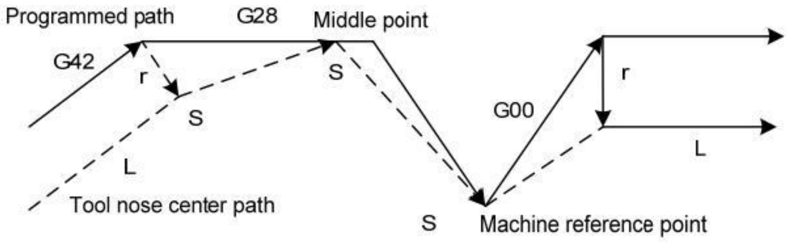


图 4-3-13-2 Cancel compensation vector temporarily in G28

➢ G71～G75 compound cycle; G76, G92 thread cutting

When executing G71～G76 , G96 thread cutting, the system does not execute the tool nose radius compensation and cancel it temporarily, and there is G00, G01,in the following blocks, and the system automatically recovers the compensation mode.

➢ G32, G33,G34 thread cutting

It cannot be operated in the mode with tip radius compensation. If it is operated, an alarm will be triggered.

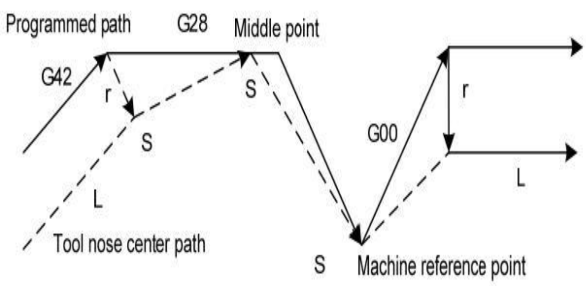


图 4-3-13-3 Cancel compensation vector temporarily in G71～G76

➢ G90, G94 (taking an example of G42)

Compensation method of tool nose radius compensation in G90 or G94:

A. Cancel the previous tool nose radius compensation;

B. Create the previous C compensation before cutting, and the path in the following figure ① creates the previous radius compensation mode;

C. The paths 2, 3 in the following figure are the radius compensation cutting;

D. The path 4 in the following figure can cancel the radius compensation, and the tool returns to the cycle starting point; there is G00,G01 in the following block, and the CNC automatically recovers the compensation mode.

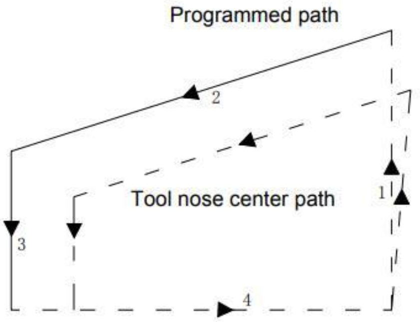


图 4-3-13-4 Offset direction of G90 tool nose radius compensation

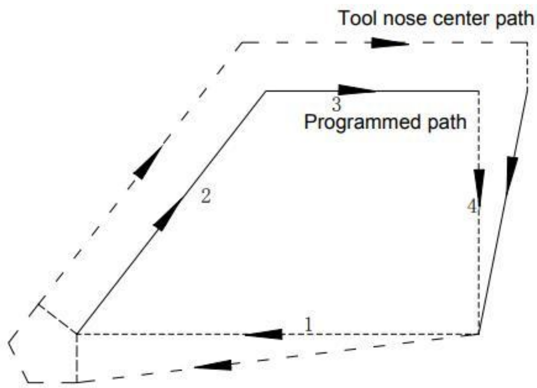


图 4-3-13-5 Offset direction of G94 tool noseradius compensation

4.3.14 Special circumstances

- When the inner corner machining is less than the tool tip radius

At this point, the inner offset of the tool can cause excessive cutting. At the beginning of the previous program segment or after the corner movement, the tool movement stops and an alarm is displayed. However, if the 'single program segment' switch is ON, the tool will stop at the end of the previous program segment.

- When machining a concave shape smaller than the diameter of the tool tip

When the compensation of the tool tip radius causes the center of the tool tip to move in the opposite direction to the program path, overcutting will occur. At this point, at the beginning of the previous program segment or after the corner movement, the tool movement stops and an alarm is displayed.

- When machining a step smaller than the tool tip radius

When the program contains a step smaller than the tool tip radius and the step is also an arc, the tool center path may form a motion direction opposite to the program path. At this point, the first vector will be automatically ignored and the line will move directly to the endpoint of the second vector. When in a single program segment, the program will stop at this point. If it is not in a single program segment mode, the loop operation will continue. If the step is a straight line, compensation will be executed correctly without generating an alarm. (However, the uncut part will still be retained)

- When G-code contains subprogram

Before calling the subroutine (i.e. before executing M98), the CNC must be in compensation cancel mode. After entering the subroutine, bias can be started, but before returning to the main program (i.e. before executing M99), it must be in compensation cancel mode. Otherwise, an alarm will appear.

- When changing the compensation amount

(a) Usually, when changing the tool in the cancel mode, the value of the compensation amount is changed. If the compensation amount is changed in the compensation mode, the new compensation amount is only effective after the tool change.

(b) If the compensation amount is negative (-), G41 and G42 exchange with each other in the program. If the center of the tool moves along the outer side of the workpiece, it will move along the inner side, and vice versa. As shown in the following example, the compensation amount is generally (+) when creating a program. When the tool path is shown in (a) production process, if the compensation amount is

If it is negative (-), the tool center moves as shown in (b), and vice versa. Additionally, please note that when the offset symbol changes, the direction of the tool tip offset also changes, but the assumed tool tip direction remains unchanged. So don't arbitrarily change the symbol of the offset.

The endpoint of the programmed arc is not on the arc

When the end point of the arc in the program is not on the arc, the tool movement stops and an alarm message "End point of the arc is not on the arc" is displayed.

4.4 macro

I5T3 I5T5 system provides macro codes similar to High-level programming language. User macro codes can realize variable assignment, arithmetic operation, logical judgment and condition transfer, which is conducive to the preparation of special parts processing programs, reduces the tedious numerical calculation during manual programming, and simplifies user programs.

4.4.1 Macro variables

- **Representation of variables**

Variables are specified with the symbol "#"+variable number;

Format: # i (i=100102103,...);

Example: # 105, # 109, # 125.

References to variables

Variables can be used to replace the numerical value after the parameter value.

(Example) When # 103=15, F # 103 is the same as the F15 instruction.

When G # 130=3, it is the same as G3.

- **The type of variable**

Variables can be divided into four types based on their numbers.

Note: 1. Parameter words O and N (program number and sequence number) cannot refer to variables. Cannot program with O # 100, N # 120.

2. If the maximum code value specified by the parameter value is exceeded, it cannot be used# When 30=120, M # 30 exceeds the maximum code value.

3. Display and setting of variable values: Variable values can be displayed on the LCD screen or set using MDI.

4.4.2 Types of variables

According to different variable numbers, variables are divided into null variables, Local variable, common variables and system variables, and their purposes and properties are different.

(1) Empty variable # 0: (This variable is always empty and no value can be assigned to it)

(2) Local variable # 1~# 50: Local variable can only be used to store data in macro programs; Bit parameter NO: 52 # 7 can be set to clear after reset or emergency stop. When the macro program is called, the argument assigns a value to the Local variable.

(3) Common variables # 100~# 199, # 500~# 999: bit parameter NO: 52 # 6 can be set to clear common variables # 100~# 199 after reset or emergency stop.

Common variables are common in the main program and various user macro programs called by the main program. The variable # i used in a user macro program is the same as the variable # i used in other macro programs. Therefore, the common variable # i of the operation result in a certain macro program can be used in other macro programs.

The purpose of public variables is not specified in the system, and users can freely use them.

Variable number	Variable type	Function
＃100～＃199	Common variables	Clear when cutting off power, reset all to "empty" when powered on
＃500～＃999		Data is saved in a file and is not lost even when power is cut off

Fig. 4-4-2-1

(4) System variables: System variables are used to read and write various data changes during CNC operation. They are as follows:

1) Interface Input Signal # 1000- # 1015 (Bitwise Read PLC Input Signal to System, i.e. G54 Signal) # 1032 (Bytewise Read PLC Input Signal to System, i.e. G Signal)

2) Interface output signals # 1100- # 1115 (write the system output signal to PLC in bits, i.e. F54 signal)

#1132 (Write system output signal to PLC in bytes, i.e. F signal)

3) Tool length compensation value # 1500- # 1799 (readable and writable)

4) Length wear compensation value # 2000- # 2299 (readable and writable)

5) Tool radius compensation values # 1800- # 1899 (readable and writable)

6) Compensation value for radius wear # 2300- # 2399 (readable and writable)

7) Imagined Blade Tip # 1900-- # 1999 (readable and writable)

8) Alarm # 3000

9) User Data Table # 3500- # 3755 (Read Only, Cannot Write)

10) Modal information # 4000- # 4030 (read-only, cannot be written)

11) Location information # 5001- # 5030 (read-only, cannot be written)

12) Workpiece zero offset # 5201- # 5235 (readable and writable)

13) Additional workpiece coordinate system # 7001- # 7250 (readable and writable)

3. Detailed Description of System Variables

1） **Modal information**

Variable number	function	Group number
#4000	G04,G28,G31,G50,G65,G70,G71,G72,G73,G74,G75,G76	00
#4001	G00,G01,G02,G03,G32,G34,G90,G92,G94	01
#4002	G96,G97	02
#4005	G98, G99	05
#4006	G20,G21	06
#4007	G40,G41,G42	07
#4014	G54,G55,G56,G57,G58,G59	14
#4022	D	
#4023	H	
#4024	F	
#4025	M	
#4026	S	
#4027	T	
#4028	N	
#4029	O	
#4030		

Note 1: The P code represents the currently selected additional workpiece coordinate system.

Note 2: When executing G # 4002, the values obtained in # 4002 are 17, 18, or 19.

Note 3: Modal information cannot be written but can only be read.2)

Current location information

Variable number	position signal	Coordinate System	Tool compensation value	Reading operations during exercise
#5001~#5005	Program segment endpoint	Workpiece Coordinate System	does not contain	possible
#5021~#5025	current location	machine coordinates	contain	impossible
#5041~#5045	current location	Workpiece Coordinate System		

4.4.3 local variable

Correspondence between address and Local variable:

Address of independent variable	Local variable number	**Address of independent variable**	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: English alphabet are added with values. Except G, L, O, N, H and P, all the other 20 English alphabet can be assigned to independent variables. Each letter can be assigned once. From A-B-C-D... to X-Y-Z, the assignment does not have to be in alphabetical order, and the address that is not assigned can be omitted.

Note 2: G65 must be specified before using any independent variable.

4.4.4. Notes on user macro program ontology

1) Method of keying in

Press the # key after the parameter words G, X, Y, Z, R, I, J, K, F, H, M, S, T, P, Q to input #.

2) In MDI state, operations can also be instructed and code can be transferred.

3) The H, P, Q, and R of operations and code transfer 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 variables cannot exceed fifteen Significant figures, the calculation result cannot exceed nine integers, and the manual input range of variables is eight Significant figures.

5) The variable value operation result can be a decimal with an accuracy of 0.0001. Except H11 (or operation), H12 (and operation), H13 (non operation), H23 (remainder operation) will ignore the decimal part of the variable in the calculation process, other operations will not round off the Decimal separator 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 operations and code transfer varies depending on the conditions, and the average value can generally be considered as 10ms.

7) When the value of a variable is undefined, such a variable becomes an "empty" variable. Variable # 0 is always an empty variable. It cannot be written, it can only be read.

a. Reference

When referencing an undefined variable, the address itself is also ignored.

For example:

When the value of variable # 1 is 0 and the value of variable # 2 is empty, the execution result of G00X # 1 Y # 2 is G00X0;

b. Operations

Except for assigning values with<null>,<null>is the same as 0 in all other cases.

When # 1=<empty>	When # 1=0
#2=#1 ↓ #2=<空>	#2=#1 ↓ #2=0
#2=#1*5 ↓ #2=0	#2=#1*5 ↓ #2=0
#2=#1+#1 ↓ #2=0	#2=#1+#1 ↓ #2=0

c. **conditional expression**

The<empty>in EQ and NE is different from 0.

When # 1=<empty>	When # 1=0
#1 EQ #0 ↓ found	#1 EQ #0 ↓ false
#1 NE #0 ↓ found	#1 NE #0 ↓ found
#1 GE #0 ↓ found	#1 GE #0 ↓ found
#1 GT #0 ↓ false	#1 GT #0 ↓ false

4.4.5 Non modal call G65

When instruction G65 is executed, the user macro program specified in address P is called, and the data is passed to the user macro program body through an independent variable.

The format is as follows:

G65 P □□□□□L□□□□ <Independent variable assignment>;

Number of calls

The program number of the macro program being called

After G65, specify the program number of the user macro program with address P, specify the number of macro program calls with L, and pass data to the macro program with an argument.

When requesting repetition, specify the number of repetitions from 1 to 9999 after address L. If L is omitted, the default number is 1.

It is specified with an argument, and its value is assigned to the corresponding Local variable.

Note 1: When the subroutine number specified with address P cannot be retrieved, an alarm is generated.

Note 2: Subprograms 90000~99999 are system reserved programs. When users call these types of subroutines, the system can execute the contents of the subroutine, but the cursor will remain in the G65 code segment, and the program interface will always display the main program content.

Note 3: Macro program calls can be nested up to five layers.

4.4.6 User Macro Program Function A

1. Command format: G65 Hm P# i Q# j R# k;

    Command significance:

    m: operation or jump command, range 01～99.

    #i: macro variables name for storing values.

    # j: macro variables name 1 for operation, can be constant.

    # k: macro variables name 2 for operation, can be constant.

    # i = #j O # k

    └──────── Operation sign specified by Hm

    Note: Macro variable name has no "#" when it is presented directly with constant. Operation sign 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 with G65 has no effect on the selection of offset.

Command format	Functions	Definitions
G65 H01 P#i Q#j	Assignment	# i = # j assign value of j to i
G65 H02 P#i Q#j R#k;	Decimal add operation	# i = # j + # k
G65 H03 P#i Q#j R#k;	Decimal subtract operation	# i = # j - # k
G65 H04 P#i Q#j R#k;	Decimal multiplication operation	# i = # j×# k
G65 H05 P#i Q#j R#k;	Decimal division operation	# i = # j÷# k
G65 H11 P#i Q#j R#k;	Binary addition	# i = # j OR # k
G65 H12 P#i Q#j R#k;	Binary multiplication(operation)	# i = # j AND # k
G65 H13 P#i Q#j R#k;	Binary exclusive or	# i = # j XOR # k
G65 H21 P#i Q#j;	Decimal square root	# i = $\sqrt{\# j}$
G65 H22 P#i Q#j;	Decimal absolute value	# i = |# j |
G65 H23 P#i Q#j R#k;	Decimal remainder	Remainder of # i = (#j÷# k)
G65 H24 P#i Q#j;	Decimal into binary	# i = BIN(# j)
G65 H25 P#i Q#j;	Binary into decimal	# i = DEC(# j)
G65 H26 P#i Q#j R#k;	Decimal multiplication/division operation	# i = # i×# j÷# k
G65 H27 P#i Q#j R#k;	Compound square root	# i = $\sqrt{\# j^2 + \# k^2}$
G65 H31 P#i Q#j R#k;	Sine	# i = # j×sin(# k)
G65 H32 P#i Q#j R#k;	Cosine	# i = # j×cos(# k)
G65 H33 P#i Q#j R#k;	Tangent	# i = # j×tan(# k)
G65 H34 P#i Q#j R#k;	Arc tangent	# i = ATAN(# j / # k)
G65 H80 Pn;	Unconditional jump	Jump to block n
G65 H81 Pn Q#j R#k;	Conditional jump 1	Jump to block n if # j = # k,otherwise the system executes in order
G65 H82 Pn Q#j R#k;	Conditional jump 2	Jump to block n if # j ≠ # k, otherwise the system executes in order
G65 H83 Pn Q#j R#k;	Conditional jump 3	Jump to block n if # j ＞ # k, otherwise the system executes in order
G65 H84 Pn Q#j R#k;	Conditional jump 4	Jump to block n if # j ＜ # k, otherwise the system executes in order
G65 H85 Pn Q#j R#k;	Conditional jump 5	Jump to block n if # j ≥ # k, otherwise the system executes in order
G65 H86 Pn Q#j R#k;	Conditional jump 6	Jump to block n if # j ≤# k, otherwise the system executes in order
G65 H99 Pn;	P/S alarm	(500+n) alarms

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;

8) P/S alarm

G65 H99 Pi; i: alarm number +500

(Example) G65 H99 P15; P/S alarm 515.

Note 1: The alarm content (/***) can be omitted. When this alarm content is omitted, only the alarm occurs, and the alarm content is null.

Note 2: Block number can be specified by variables. Such as: G65 H81 P#100 Q#101 R#102; The program jumps to block that its block number is specified by #100

4.4.7 User Macro Program Function B

1. Arithmetic and logic

The operations listed in the table below can be performed in variables. The expression to the right of the operator can contain constants and/or variables composed of functions or operators. The variables # j and # k in the expression can be replaced with constants. The variables on the left can also be assigned using expressions.

Function	Expression format	Remark
Definition or assignment	#i = #j	
addition subtraction multiplication division	#i = #j + #k #i = #j - #k #i = #j * #k #i = #j / #k	
Or And Exclusive Or	#i = #j OR #k #i = #j AND #K #i = #j XOR #K	Logic operation is executed by the binary system
Square root Absolute value Rounding-off FUP FIX Natural logarithm Exponential function	#i = SQRT[#j] #i = ABS[#j] #i = ROUND[#j] #i = FUP [#j] #i = FIX [#j] #i = LN[#j] #i = EXP[#j]	
Sine Arc sine Cosine Arc cosine Tangent Arc tangent	#i = SIN[#j] #i = ASIN[#j] #i = COS[#j] #i = ACOS[#j] #i = TAN[#j] #i = ATAN[#i]/ [#j]	Angle unit is specified by degree. For example: 90°30' is expressed by 90.5°
BCD to BIN BIN to BCD	#i = BIN[#j] #i = BCD[#j]	Used for switching with PMC

Relative explanation:

1. Angle unit

Angle units of SIN, COS, ASIN, ACOS, TAN and ATAN are degree (°). For example: 90° 30 means to be 90.5 ′ ° (degree).

2. Arc sine # i=ASIN[#j]

i. result output range:

No.180#7 NAT is set to 1: 90° ~ 270° ;

No.180#7 NAT is set to 0: -90° ~ 90° ;

ii. when #j exceeds the range from -1 to 1, the system alarms P/S.

iii. the constant replaces the variables #j.

3. Arccosine # i =ACOS[#j]

i. Result output range 180° ~ 0° .

ii. When #j exceeds the range from -1 to 1, the system alarms P/S.

iii. The constant replaces the variables #j.

4. Arc tangent #i=ATAN[#j]/[#k]

Specify the lengths of two sides and separate them with a slash “/”.

i. Result output range:

When No.180#7 NAT is set to 1: 90° ~ 270° ;

[For example] #1=ATAN[-1]/[-1]: #1=225° ;

When No.180#7 NAT is set to 0 -90° ~ 90° ;

[For example]#1=ATAN[-1]/[-1]: #1=45.0° ;

ii. The constant replaces the variables #j.

5. Natural logarithm #i=LN[#j]

The constant replaces the variables #j

6. Exponential function #i=EXP[#j]

The constant replaces the variables #j

7. ROUND function

When arithmetical operation or logic operation IF or WHILE includes ROUND, ROUND rounds in the first decimal place.

For example: #1=ROUND[#2]: #2=1.2345, the variables 1 is 1.0.

8. FUP FIX

After CNC executes the operation, the result integer absolute value is bigger the previous absolute value, which is called FUP; the result integer absolute value is less than the one, which is call FIX. Pay more attention to the negative execution.

Example: Hypothetically, #1=1.2, #2= -1.2

When #3=FUP[#1] is executed, 2.0 is assigned to #3.

When #3=FIX[#1] is executed, 1.0 is assigned to #3.

When #3=FUP[#2] is executed, -2.0 is assigned to #3.

When #3=FIX[#2] is executed, -1.0 is assigned to #3.

2. Transfer & Cycle

1) Transfer & Cycle

In the program, the system uses GOTO and IF statement to change the control flow. There are three types of transfer and cycle operation.

1. GOTO statement (unconditional transfer).
2. Condition control IF statement.
3. WHILE cycle statement.

2) Unconditional transfer (GOTO statement)

Transfer to the block which serial number is n. The system alarms when others exceeds the range from 1 to 99999, and it specifies the serial number with the statement.

Format: GOTO n; n: serial number(1~99999)

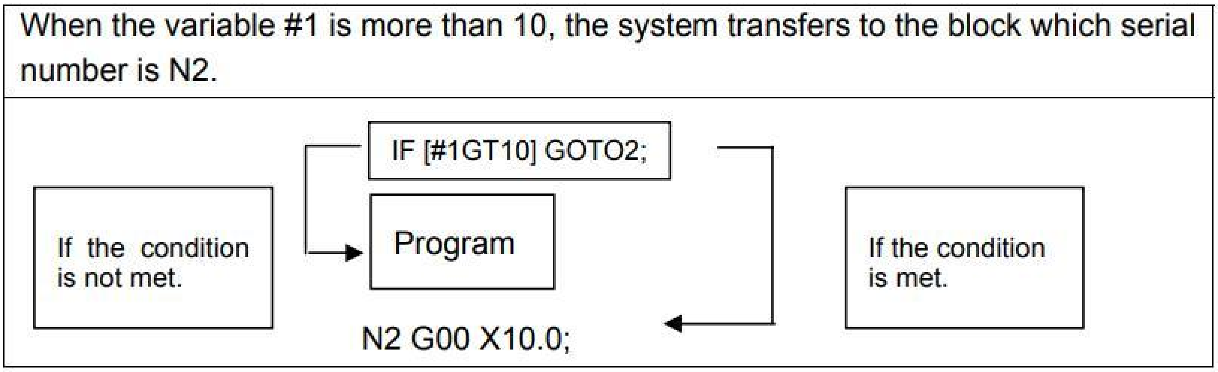
Example: GOTO 1;

GOTO #101;

3) Conditional control (IF statement)

GOTO format: IF[conditional statement]GOTO n; When the specified conditional statement is valid, the system transfers to the block which serial number is n; When the specified conditional statement is valid, the system executes the next block.

Example:



THEN format: IF[conditional expression]THEN＜macro program statement＞;

When the condition expression is valid, the system executes only one statement following THEN.

Example: IF[#1 EQ #2] THEN #3=0;

When #1 value is equal to the #2, 0 is assigned to the variable #3; when they are not equal, the system orderly executes the followings instead of the assignment statement after THEN.

Conditional expression: the conditional expression must include the conditional operator, two sides of conditional operator can be variable, constant or expression, and it must be closed with the brackets '[' ']'.

Conditional operator: the system uses the conditional operators listed in the following table.

Conditional operator	Meaning
EQ	Equal to (=)
NE	Not equal to (≠)
GT	More than (>)
GE	More than or equal to (≥)
LT	Less than (<)
LE	Less than or equal to (≤)

Example: IF[3<>2]GOTO 2; its meaning: when 3 is not equal 2, the system skips to N2 block;

IF[#101>=7.22]THEN #101=SIN30; its meaning: when #101 is more than or equal to 7.22, the system executes the assignment after THEN. i.e. the sine value of 30 degree is assigned to the variable #101.

Typical program: the following program counts the sum of the integer 1~10.

O9500

#1=0; … …the sum is initialized to be 0

#2=1; … …the summand number is initialized to be 1

N1 IF[#2 GT 10]GOTO2; … … the system skips to N2 when the summand is more than 10

#1= #1+#2; … …count the sum of two numbers

#2= #2+1; … … the summand adds 1

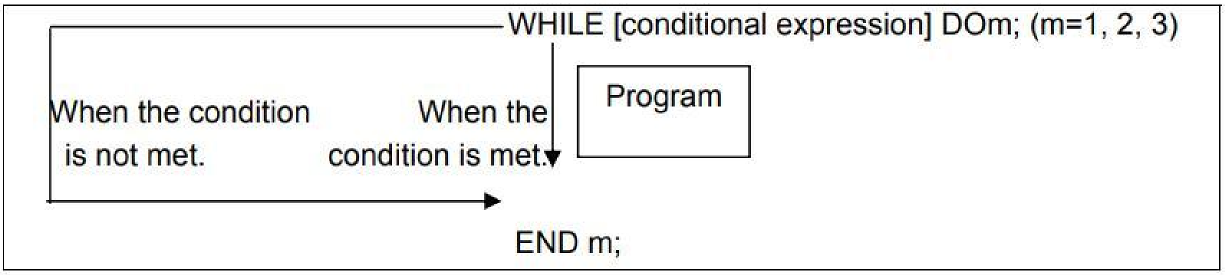
GOTO1; … … unconditionally skip to the block N1

N2 M30； … … end of program

4）Cycle (WHILE statement)

Specify one conditional expression after WHILE. When the specified conditional is valid, the system executes the blocks between DO and END; otherwise, the system skips to the block after END.

Example:



Explanation: when the specified condition is valid, the system executes the block between DO and END; otherwise, executes the block after END. The two tabs after DO and END are consistent, and the tab value can be 1, 2 or 3, otherwise, the system alarms.

Chapter 5 M (Miscellaneous Function)

The list of M codes available for users in this system is as follows:

	Commands	Functions
M code used for control programs	M30	The program ends and returns to the program header, increasing the number of processed pieces by 1
	M02	The program ends and returns to the program header, increasing the number of processed pieces by 1
	M98	Calling subroutines
	M99	Subprogram end return/repeat execution
M code controlled by PLC	M00	Program Pause
	M01	Program selection pause
	M03	Spindle clockwise (CW)
	M04	Spindle counterclockwise (CCW)
	M05	Spindle stop
	M08	Cooling ON
	M09	Cooling OFF
	M10	Tailstock forward
	M11	Tailstock backward
	M12	Chuck clamping
	M13	Chuck clamping
	M14	Spindle position control
	M15	Spindle speed control
	M18	Cancel spindle orientation
	M19	Spindle orientation
	M20	Spindle clamping
	M21	Spindle releasing
	M28	Cancel rigid tapping
	M29	Rigid tapping
	M32	Lubricating ON
	M33	Lubricating OFF
	M35	Start the chip conveyor
	M36	Close the chip conveyor
	M41	Spindle automatic gear shifting 1
	M42	Spindle automatic gear shifting 2
	M43	Spindle automatic gear shifting 3
	M44	Spindle automatic gear shifting 4
	M70	Automatic shifting feed forward output
	M71	Automatic shifting feeding forward output closed
	M80	M80 output (customizable)
	M81	Turn off M80 output
	M90	M90 waiting for input signal (customizable)

When the mobile code and auxiliary function are specified in the same program segment, the mobile code and auxiliary function code are executed simultaneously.

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 these functions. Usually, only one M code can be specified in a program segment. However, due to the limitations of mechanical operation, some M codes cannot be specified simultaneously. For the limitations of mechanical operation on specifying multiple M codes for the same program segment, please refer to the user manual of the machine tool manufacturer.

5.1 M code controlled by PLC

When the M code controlled by PLC is in the same segment as the mobile code, both the M code and the mobile code are executed simultaneously

5.1.1 Spindle CW, CCW Control （M03、M04）

Command format:M03（M04） Sx x x;

Command function: Forward rotation refers to observing from positive to negative along the Z-axis direction, while counterclockwise rotation of the main axis (CCW) indicates forward rotation; Conversely, clockwise rotation (CW) is considered reverse.

The Sx x code refers to the speed of the spindle, which is the gear in which it is located during gear control.

Unit: revolutions per minute (r/min)

When controlled by a frequency converter, Sx x x refers to the actual speed, for example, S1000 specifies that the spindle rotates at a speed of 1000r/min.

5.1.2 Spindle Stop Control M05

Command format: M05，When M05 is executed in automatic mode, the spindle will stop rotating. But the speed of the S code instruction is preserved. The deceleration method when the spindle stops is determined by the machine tool manufacturer's settings. Usually, it is energy consumption braking.

5.1.3 Cooling Control （M08、M09）

Command format:M08（M09）

Control the start and stop of the cooling water pump. If an auxiliary function lock is encountered in automatic mode, the water pump control code will not be executed.

5.1.4 Tailstock Control （M10、M11）

Command format:M10（M11），tailstock going forward （backward）

5.1.5 Chuck Control（M12、M13）

Command format:M12（M13），chuck clamping（releasing）

5.1.6 Spindle Position/Speed Control Switch（M14、M15）

Command format:M14，Spindle is in the position control mode from speed control mode; **M15** Spindle is in speed control mode from the position control mode.

5.1.7 Spindle directional start and stop（M19、M18）

Command format:M19，Spindle directional start；M18，Spindle orientation shutdown code.

5.1.8 Spindle Clamped/Released（M20、M21）

Command format:M20，Spindle clamped；M21，Spindle released

5.1.9 Rigid tapping（M29）

Command format:M29，**Rigid tapping**

5.1.10 Lubricating Control（M32、M33）

Command format:M32（M33）,Control the start and stop of the lubrication pump. If an auxiliary function lock is encountered in automatic mode, the water pump control code will not be executed.

5.1.11 Chip conveyor on/off（M35、M36）

Command format:M35（M36）, Control the start and stop codes of the spiral chip conveyor.

5.1.12 Spindle shifting from first to fourth gear（M41、M42、M43、M44）

Command format:M41、M42、M43、M44 Spindle shift code.

5.1.13 Automatic feeding forward output and shutdown（M70、M71）

Command format:M70（M71）Automatically move the feed forward output and close the code.

5.1.14 M80、M81 Custom output, close（M80、M81）

Command format:M80（M81），Customize the output and close the code.

5.1.15 M90 Waiting for signal input（M90）

Command format:M90，Wait for signal input code.

5.2 M code used for control programs

The M code used for control programs is divided into main program control class and macro program control class.

Note: 1. The M00, M01, M02, M06, M30, M98, and M99 codes cannot be specified together with other M codes, otherwise the system will alarm. When these M codes are in the same segment as other non M codes, the other non M codes in the same segment will be executed first before executing them.

2.This type of M code includes code that causes the CNC to send the M code itself to the machine tool, while also enabling the CNC to perform internal operations, such as M code that invalidates the pre read function of the program segment. Additionally, only allowing CNC to send the M code itself to the machine tool without performing internal operations can be specified within the same program segment.

5.2.1 Program ends and returns（M30、M02）

When running in automatic mode, the program stops running automatically when it reaches M30 (M02). If there is any program afterwards, it will not be executed and the spindle and cooling operation will be stopped. The number of workpiece processing will be increased by 1. M30 can use bit parameter N0:33 # 4 to control whether to return to the program header, while M02 can use bit parameter N0:33 # 2 to control whether to return to 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 subsequent program segments.

5.2.2 Program Pause（M00）

When running in automatic mode, the program pauses its automatic running state when it reaches M00, and the previous modal information will be saved at this time. After pressing the cycle start button, it will continue to run. Its function is equivalent to pressing the feed hold button.

5.2.3 Program selection pause（M01）

When running in automatic mode, there is a selected pause automatic running state when the program reaches M01. If the "Select Stop" switch is set to on, the M01 and M00 codes have the same effect. If the "Select Stop" effective switch is set to off, the M01 code has no effect. Please refer to the operation manual for operation

5.2.4 Program Call Subprogram Code Instructions（M98）

You can write M98 code in the main program to call subroutines for execution. Specific format:

M98 P○○○○□□□□

Called subprogram number（0000～9999）. The leading zero of subprogram number can be omitted when the calling times is not input; the subprogram number must be with 4 digits when the calling times is input..

Call times: 1～9999. The calling times cannot be input when it is 1.

Neto:M98 is invalid in MDI mode.

5.2.5 Return From Subprogram （M99）

1.When running in automatic 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. If there are any programs afterwards, they will not be executed and the number of workpiece processing will not be accumulated.

2. If M99 is used at the end of the subroutine, the program will return to the main program M98 segment for processing.

Chapter 6 Spindle Function

By using code S and its subsequent numerical values, the code signal is converted into an analog signal and sent to the machine tool for spindle control.

S is the modal value.

6.1 Spindle simulation control

When the SPT of bit parameter N0:1 # 2 is 0, the address S and its subsequent values are controlled by analog voltage to control the spindle speed. Please refer to the operating instructions for specific operations.

Command format:S_

explanation:

1. An S code can be instructed in a program segment.
2. The address S and its subsequent data directly specify the spindle speed, in revolutions per minute (r/min). For example: M3 S300 Indicates that the spindle is running at a speed of 300 revolutions per minute.
3. When the mobile code and S code are in the same program segment, the mobile code and S function code start executing simultaneously

The spindle speed is controlled by code S and the following Numerical control.

6.2 Spindle switch quantity control

When the SPT of bit parameter N0:1 # 2 is 1, the address S and its subsequent two digit switch value control the spindle speed.

When selecting the switch value to control the spindle speed, the system can provide 3 levels of spindle mechanical shifting. Correspondence between S code and spindle speed

The system and machine tool provide several levels of spindle speed change, please refer to the instructions of the machine tool manufacturer.

Command format: S01（S1）;

S02（S2）;

S03（S3）;

explanation:

1. At present, the software has 8 gear shifts, and the ladder diagram only performs 3 level shifts. When S code other than the above is specified in the

program,The system will display 'Auxiliary function execution in progress'

2. When S is two digits, if the instruction has S4 digits, the last two digits are valid.

6.3 Constant surface cutting speed control G96/G97

Command format: G96 S__; (S0000～S9999, the leading zero can be omitted.)

Command function: The constant surface speed control is valid, the cutting surface speed is defined (m/min) and the constant rotational speed control is cancelled. G96 is modal G code. If the current modal is G96, G96 cannot be input.

Command format: G97 S__; (S0000～S9999, the leading zero can be omitted.)

Command function: The constant surface speed control is cancelled, the constant rotational speed control is valid and the spindle speed is defined (r/min). G96 is modal G code. If the current modal is G97, G97 cannot be input.

Command format: G50 S__; (S0000～S9999, the leading zero can be omitted.)

Command function: define max. spindle speed limit (r/min) in the constant surface speed control and take the current position as the program reference point.

G96, G97 are the modal word in the same group but one of them is valid. G97 is the initial word and the system defaults G97 is valid when the system is switched on.

When the machine tool is turning it, the workpiece rotates based on the axes of spindle as the center line, the cutting point of tool cutting workpiece is a circle motion around the axes, and the instantaneous speed in the circle tangent direction is called cutting surface (for short surface speed). There are different surface speed for the different workpiece and tool with different material.

When the spindle speed controlled by the analog voltage is valid, the constant surface control is valid. The spindle speed is changed along with the absolute value of X absolute coordinates of programming path in the constant speed control. If the absolute value of X absolute coordinates adds, the spindle speed reduces, and vice verse, which make the cutting surface speed as S command value. The constant speed control to cut the workpiece makes sure all smooth finish on the surface of workpiece with diameter changing.

Surface speed=spindle speed $\times$ $|X|$ $\times$ π $\div 1000$ (m/min)

Spindle speed: r/min $|X|$: absolute value of X absolute coordinate value, mm

$\pi \approx 3.14$

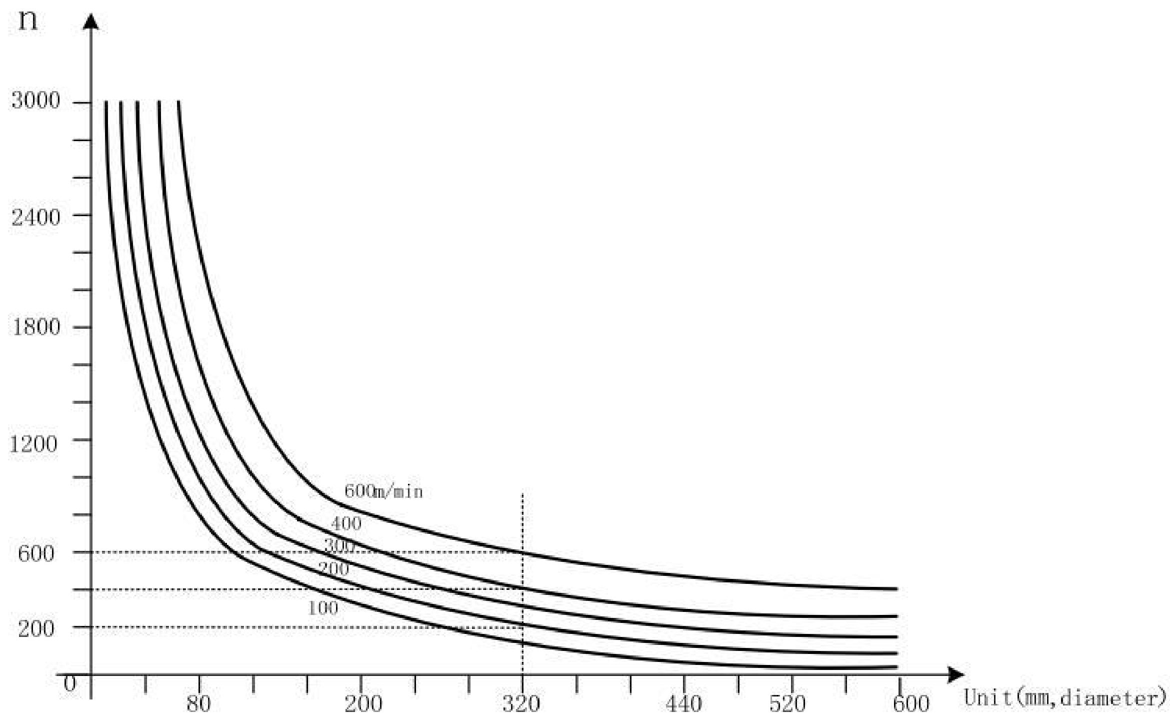


Fig. 6-3-3

In G96, the spindle speed is changed along with the absolute value of X absolute coordinates value of programming path in cutting feed (interpolation), but it is not changed in G00 because there is no actual cutting and is counted based on the surface speed of end point in the program block.

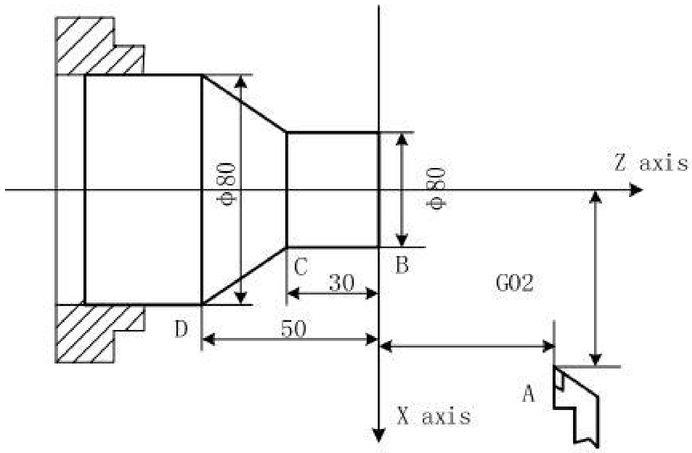
In G96 (constant surface speed control), Z coordinates axis of workpiece system must consist with the axes of spindle (rotary axis of workpiece), otherwise, there is different between the actual surface speed and the defined one.

G96 control is valid, G50 S_ can limit max. spindle speed (r/min). The spindle actual speed is the limit value of max. speed when the spindle speed counted by the surface speed and X coordinates value is more than the max. spindle speed set by G50 S_. After the system powers on, max. spindle speed limit value is not defined and its function is invalid. Max. spindle speed limit value defined by G50 S_ is reserved before it is defined again and its function is valid in G96. Max. spindle speed defined by G50 S_ is invalid in G97 but its limit value is reserved.

Note: When NO.029 (lowest spindle speed in constant surface speed control) is set to 0 and G50 S0 is executed, the spindle speed is limited to 0 r/min (the spindle does not rotate).

When the constant surface speed is controlled by the system parameter No.043, the spindle speed is lower limit, which is higher than one counted by the surface speed and X axis coordinates value.

Example:



O0001 ; (Program name)

N0010 M3 G96 S300; (Spindle rotates clockwise, the constant surface speed control is valid and the surface speed is 300 m/min)

N0020 G0 X100 Z100; (Rapid traverse to A point with spindle speed 955 r/min)

N0030 G0 X50 Z0; (Rapid traverse to B point with spindle speed 1910 r/min)

N0040 G1 W-30 F200; (Cut from B to C with spindle speed 1910 r/min)

N0050 X80 W-20 F150; (Cut from C to D with spindle speed 1910 r/min and surface speed 1194 r/min)

N0060 G0 X100 Z100; (Rapid retract to A point with spindle speed 955 r/min)

N0110 M30; (End of program, spindle stopping and cooling OFF)

Note 1: S value commanded in G96 is also reserved in G97. Its value is resumed when the system is in G96 again;

Example:

G96 S50; (Cutting surface speed 50m/min)

G97 S1000; (Spindle speed 1000 r/min)

G96 X3000; (Cutting surface speed 50m/min)

Note 2: The constant surface speed control is valid when the machine tool is locked (X, Z do not move when their motion command are executed);

Note 3: To gain the precise thread machining, it should not be adopted with the constant surface speed control but the constant rotational speed (G97) in the course of thread cutting;

Note 4: From G96 to G97, if none of S command (r/min) is commanded in the program block in G97, the last spindle speed in G96 is taken as S command in G97, namely, the spindle speed is not changed at this time;

Note 5: In G96, when the spindle speed counted by the cutting surface speed is more than max. speed of current spindle gear (system parameter No.037～No.040), at this time, the spindle speed is limited to max. one of current spindle gear.

Chapter 7 Feed function F code

The feed function controls the feed speed of the tool, and the feed function and control method are as follows:

7.1 Quick movement

Use code (G00) for quick positioning. The rapid feed speed is set by the parameter - [Feed Axis Parameter] P022-P023. The following magnification adjustments can be made using the magnification adjustment button on the operation panel:

Fig.7-1-1 Fast feed rate button

Among them, F0: is set by parameter - [Feed Axis Parameter] P031.

The acceleration of rapid positioning (G0) can be set by the parameter - [Acceleration and Deceleration Parameters] P008-P009, and can be reasonably set based on the response characteristics of the machine tool and motor.

Note: In the G00 program segment, even if the feed rate F code is specified, it is invalid, and the system positions at G0 speed.

7.2 Cutting Speed

In linear interpolation (G01) and circular interpolation (G02, G03), the numerical value after the 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 in the program. Use the feed rate button on the machine tool operation panel to adjust the cutting feed rate (with a rate adjustment range of 0% to 150%).

The cutting feed speed in automatic mode when the power is turned on is set by parameter - [Feed Axis Parameter] P032.

The cutting speed can be specified in two ways:

A) Minute feed (G98): After F, specify the tool feed rate per minute.

B) Feed per revolution (G99): After F, specify the tool feed rate per revolution of the spindle。

Note: When F specifies the cutting speed, the system displays it as an integer value. When the entered value is not an integer, the value after the Decimal separator will be rounded and displayed. The system still processes the actual input values internally. When F specifies the pitch, one decimal place can be displayed. The system processes the actual input values internally.

Code format: G98 F_

Function: Tool feed rate per minute. Unit: mm/min or inch/min.

Explanation:

1. After specifying G98 (feed rate per minute), the feed rate of the tool per minute is directly specified by the value after F.
2. G98 is a modal code that, once specified, remains valid until G99 is specified. When starting up, the default feed rate is per minute, and the default cutting feed rate is set by the data parameter P032.
3. You can adjust the feed rate per minute through the magnification adjustment button or band switch on the panel, with a magnification of 0% to 150%.

7.2.1 Feedrate per Rev G99

Command format: G99 F_;

Command function: Cutting feed rate is specified as mm/min, G99 is the modal G command

Explanation:

1. The machine tool must be equipped with a spindle encoder to use this function.
2. After specifying G99 (feed rate per revolution), the feed rate of the tool per revolution is directly specified by the value after F.
3. G99 is a modal code that, once specified, remains valid until G98 is specified. The default feed rate per revolution during initialization is zero.
4. You can adjust the feed speed per revolution through the magnification adjustment button or band switch on the panel, with a magnification of 0% to 150%

Note 1: When the spindle speed is low, feed rate fluctuations may occur. The lower the spindle speed, the more frequent the feed rate fluctuations occur.

Note 2: For the G99 per revolution feed method, the maximum speed that the system can handle per revolution feed is F500. If it exceeds F500, an alarm will appear.

7.3 Feed rate multiplier button

The feed rate in manual mode and automatic mode can be adjusted through the rate adjustment button on the operation panel, and a rate of 0-150% (16 gears in total for each gear of 10%) can be used. In automatic mode, when the rate adjustment button is set to zero, the system will stop feeding and display a cutting rate of 0%. Adjust the rate adjustment button and the program will continue to run.

7.4 Automatic acceleration and deceleration

The system drives the motor to automatically perform acceleration and deceleration control at the beginning and end of movement, so it can start and stop smoothly. And when the movement speed changes, it also automatically accelerates and decelerates, so the speed change can be carried out smoothly. Therefore, there is no need to consider acceleration or deceleration when programming.

Rapid feed: front acceleration and deceleration (0: linear type; 1: S-type)
Rear acceleration and deceleration (0: linear type; 1: Exponential type type)
Cutting feed: front acceleration and deceleration (0: linear type; 1: S-type)
Rear acceleration and deceleration (0: linear type; 1: Exponential type type)
Manual feeding: post acceleration/deceleration (0: linear type; 1: Exponential type type)
(Set universal time constants for each axis using parameters)

7.5 Acceleration and deceleration processing at the corner of the program section

For example, the previous program segment only moves X, while the next program segment only moves Z. When X decelerates, the Z line accelerates. At this time, the tool trajectory is as follows:

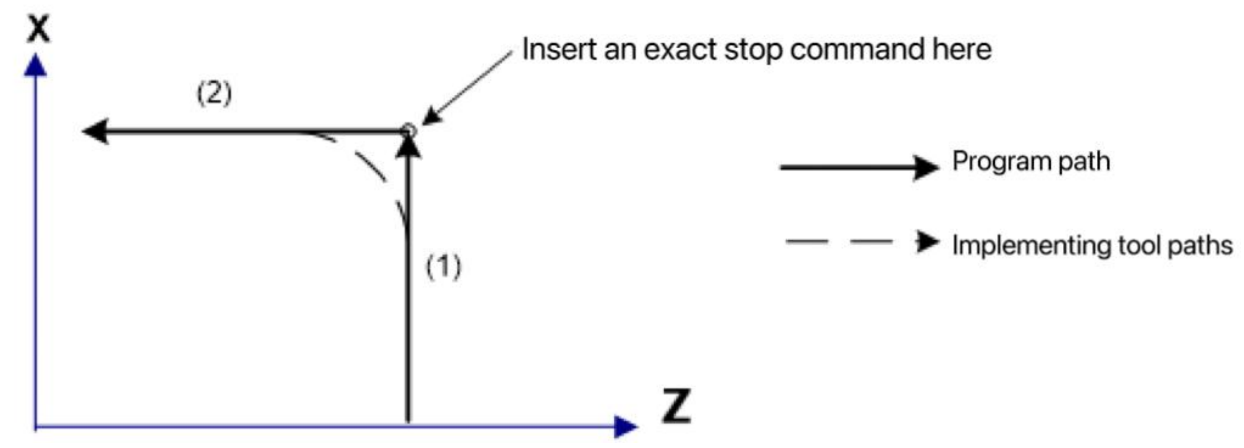


Fig. 7-5-1

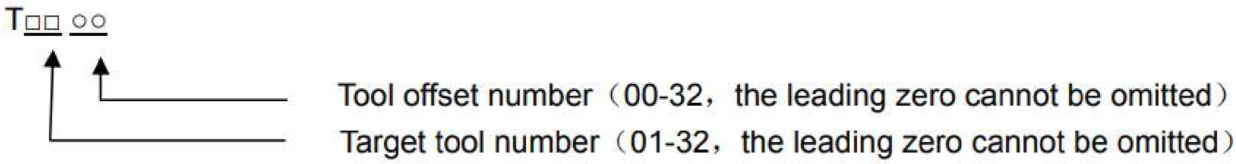
If a quasi stop code is added, the tool will move according to the program instructions as shown in the solid line above. Otherwise, the higher the cutting feed speed or the longer the acceleration and deceleration time constant, the greater the curvature at the corner. When using the arc command, the actual arc radius of the tool path is smaller than the arc radius given by the program. To reduce the error at the corners, the acceleration and deceleration time constant should be minimized as much as possible, as allowed by the mechanical system.

Chapter 8 Tool Function

8.1 Tool Control

T functions of I5T3 I5T5: automatic tool change and executing tool offset. Control logic of automatic tool change is executed by PLC and tool offset is executed by NC.

Command format:



Command function: The automatic tool post rotates to the target tool number and the tool offset of tool offset number commanded is executed. The tool offset number can be the same as the tool number, and also cannot be the same as it, namely, one tool can corresponds to many tool offset numbers. After executing tool offset and then T□□00, the system reversely offset the current tool offset and the system its operation mode from the executed tool length compensation into the non-compensation, which course is called the canceling tool offset, called canceling tool compensation. When the system is switched on, the tool offset number and the tool offset number displayed by T command is the state before the system is switched off.

Only one T command is in a block, otherwise the system alarms.

Toolsetting is executed to gain the position offset data before machining (called tool offset), and the system automatically executes the tool offset after executing T command when programs are running. Only edit programs for each tool according to part drawing instead of relative position of each tool in the machine coordinate system. If there is error caused by the wearing of tool, directly modify the tool offset according to the dimension offset.

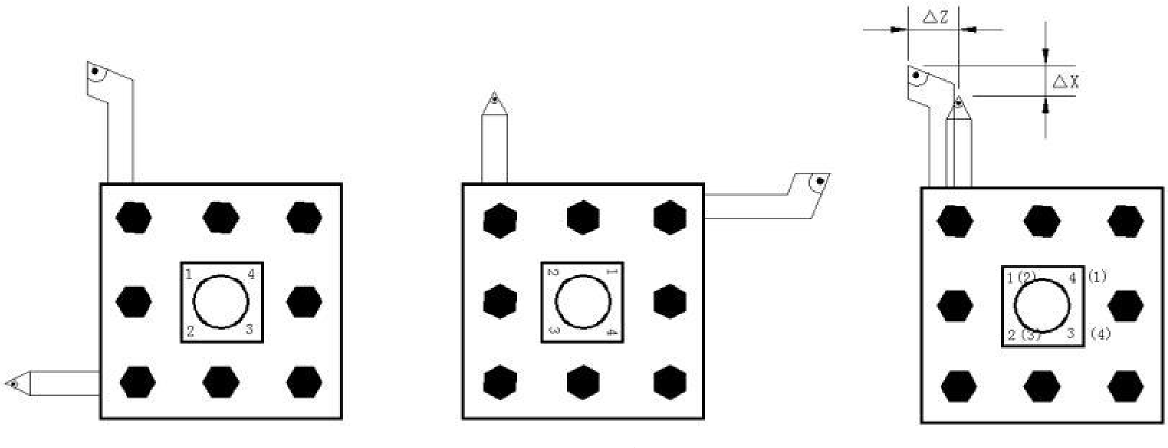


Fig.8-1-1 tool offset

The tool offset is used for the programming. The offset corresponding to the tool offset number in T command is added or subtracted on the end point of each block. Tool offset in X direction in diameter or radius is set by No.004 Bit4. For tool offset in diameter or radius in X direction, the external diameter is changed along with diameter or radius when the tool length compensation is changed.

Example: When the state parameter No.004 Bit4 is set to 0 and X tool length compensation value is 10mm, the external diameter of workpiece is 10mm; when No. 004 is set to 1 and X tool length compensation value is 10mm, the external diameter of workpiec is 20mm. Fig.2-5 is the course of creating, executing and canceling tool offset in traverse mode.

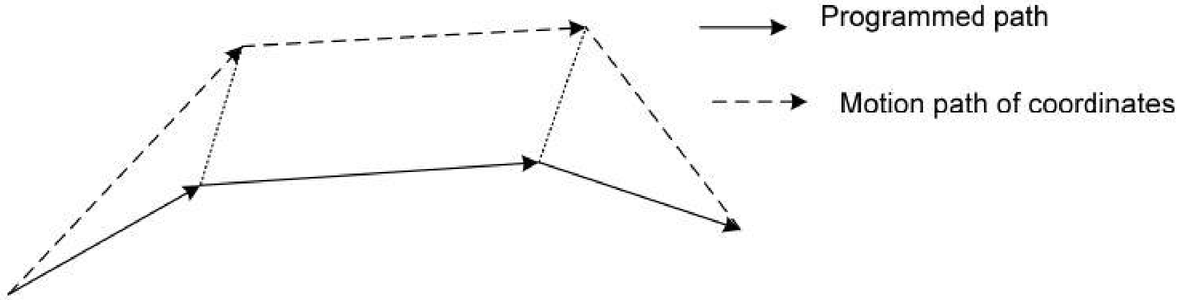


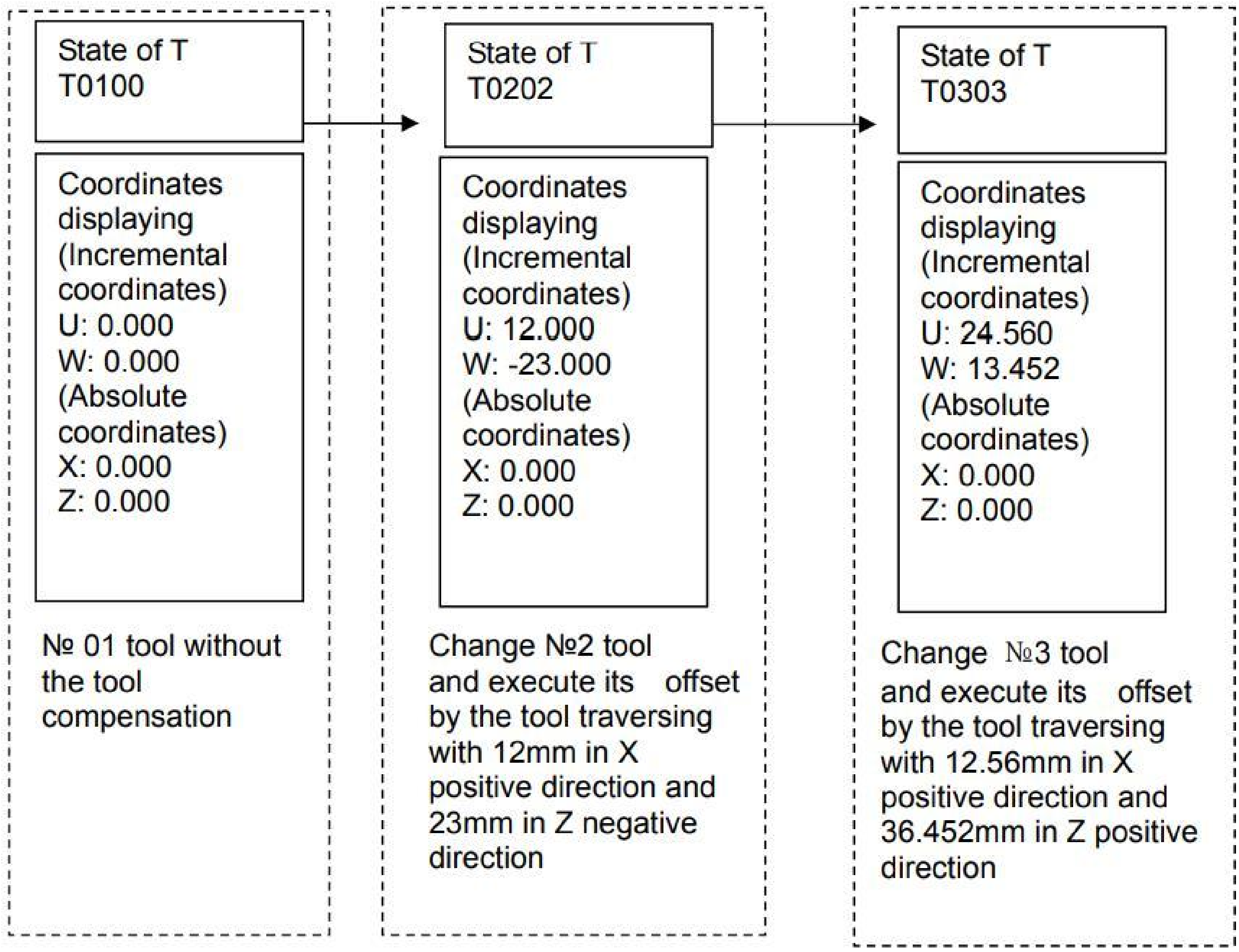
Fig.8-1-2 Creation, execution and cancellation of tool length compensation

G01 X100 Z100 T0101; (Block 1, start to execute the tool offset)
G01 W150; (Block 2, tool offset Block 2, tool offset)
G01 U150 W100 T0100; (Block 3, canceling tool offset)

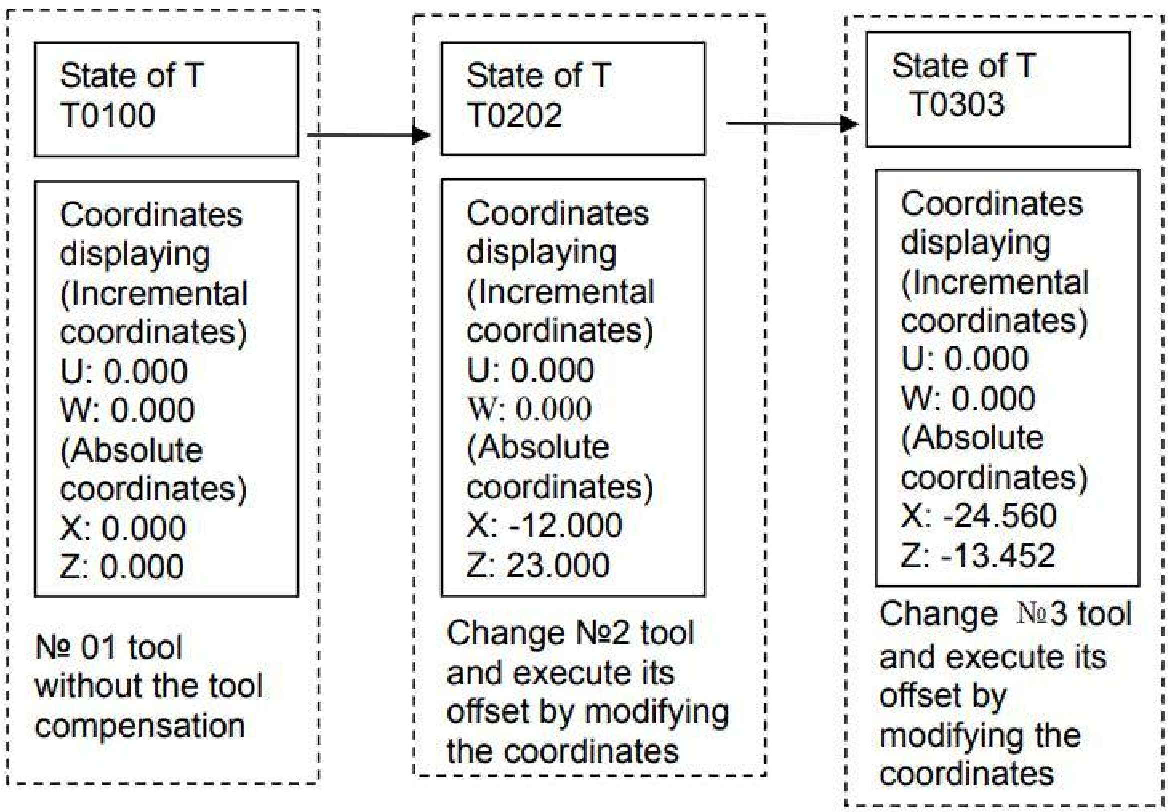
There are two methods defined by No.003 Bit4 to execute the tool length compensation: Bit4=0: The tool length compensation is executed by the tool traversing; Bit4=1: The tool length compensation is executed by modifying the coordinates;

Example:

Tool offset number	X	Z
00	0.000	0.000
01	0.000	0.000
02	12.000	-23.000
03	24.560	13.452



8-1-3 Tool traversing mode



8-1-4 Modifying the coordinates mode

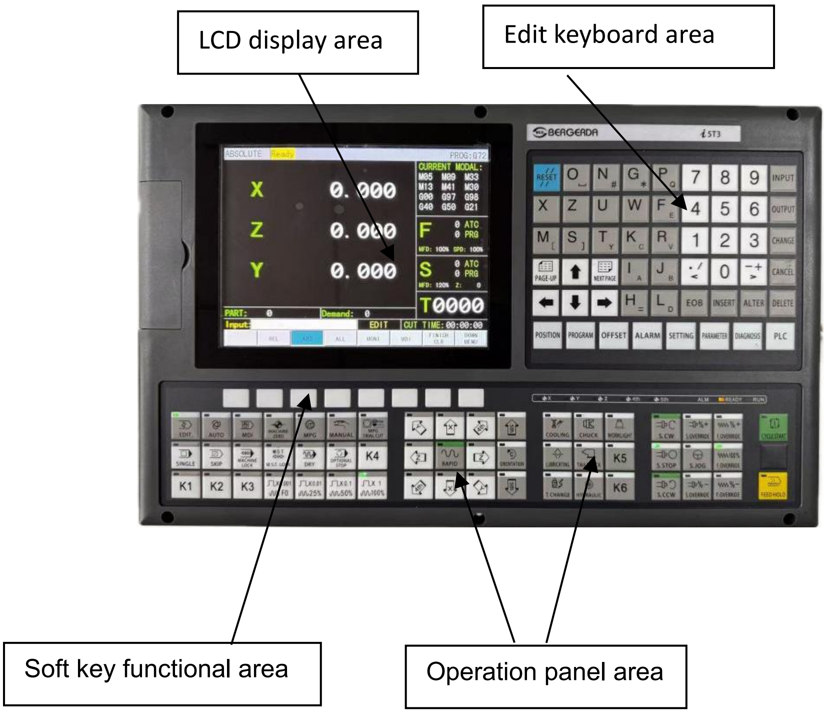
In Edit and Auto mode, a sole T word in executing tool offset (it is not with the motion command in the same block) is relative to No.004 BIT3 setting (as Fig.8-1-3 and Fig.8-1-4). When No.003 Bit4=1 and a sole T command is executed, the tool offset number is displayed in poor, which is cleared out(tool offset number is still displayed in poor when tool offset is not executed for one axis, the previous bit of tool offset number is for X axis tool compensation and the next one is for Z axis tool compensation) after executing tool offset.

VOLUME II OPERATING AND CONNECTION

Chapter 1 Operation Panel

1.1 Panel division

The 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

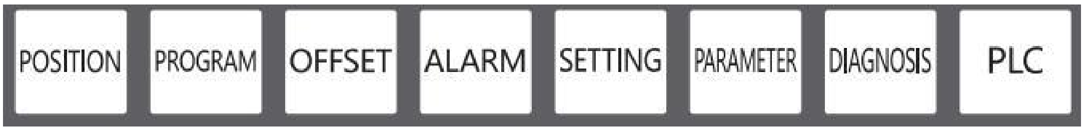
Fig. 1-2-2-1

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 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 PageDown	Used for page conversion and program flipping in the same display mode.
9	←↑→ ↓	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.

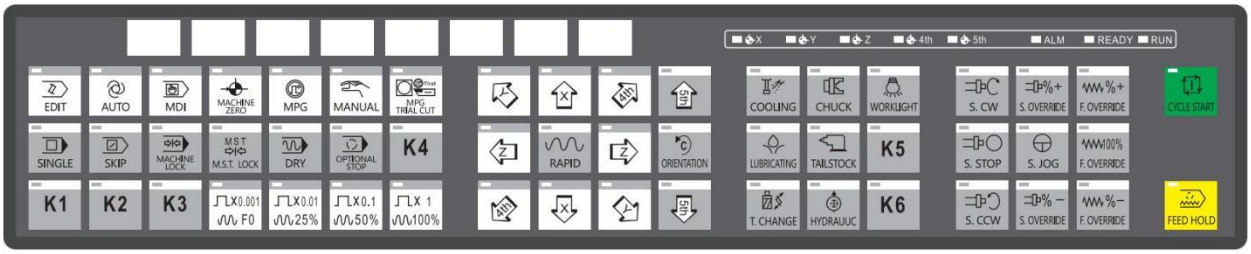
1.2.3 Introduction to screen operation keys

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.
PARAMETE R	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 。
S.CCW S.STOP S.CW	Spindle control keys	For spindle CCW For spindle stop For spindle CW	MPG, Manual, MPG trial-cut mode
Spindle override keys	Spindle magnificatio n knob	spindle speed adjustment (spindle analog	Auto, Edit, MDI, Machine zero return, Manual, Step,

		control active)	MPG, Program zero return mode
Hydraulic key	Hydraulic action key	Hydraulic output ON/OFF	Auto, Edit, MDI, Machine zero return, Manual, Step, MPG, Program zero return mode
Chuck key	Chuck switch	Chuck clamping/releasing	Auto, Edit, MDI, Machine zero return, Manual, Step, MPG, Program zero return mode
Manual tool change key	Manual tool change	manual tool change	Machine zero return, Manual, Step, MPG, Program zero return mode
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 Interface Display and Data Modification and Setting

2.1 Position display

2.1.1 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 2-1-1-1**）

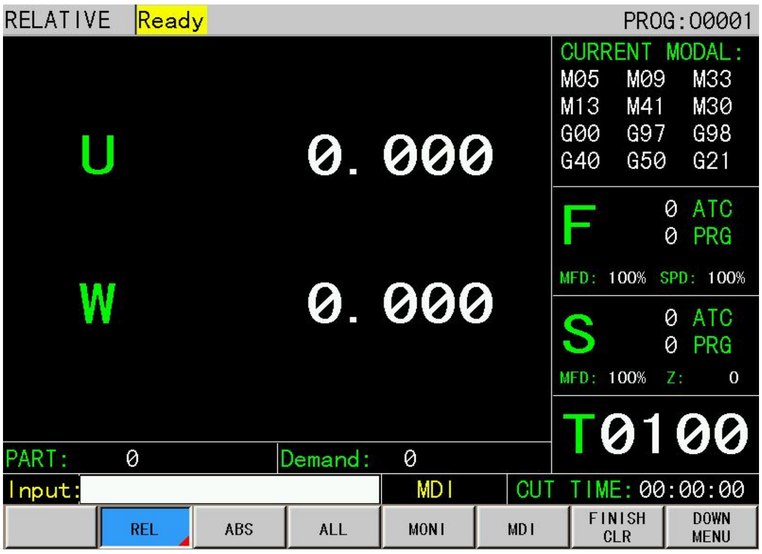


Figure **2-1-1-1**

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 2-1-1-2**）

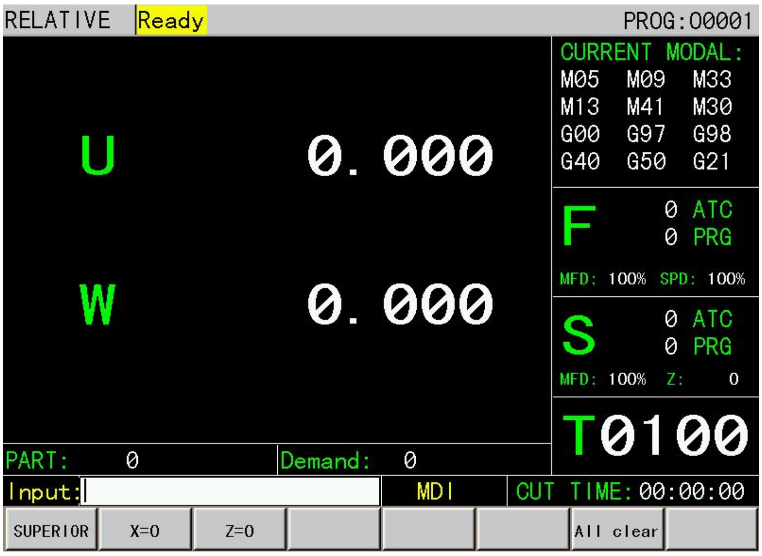


Figure **2-1-1-2**

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 2-1-1-3**）

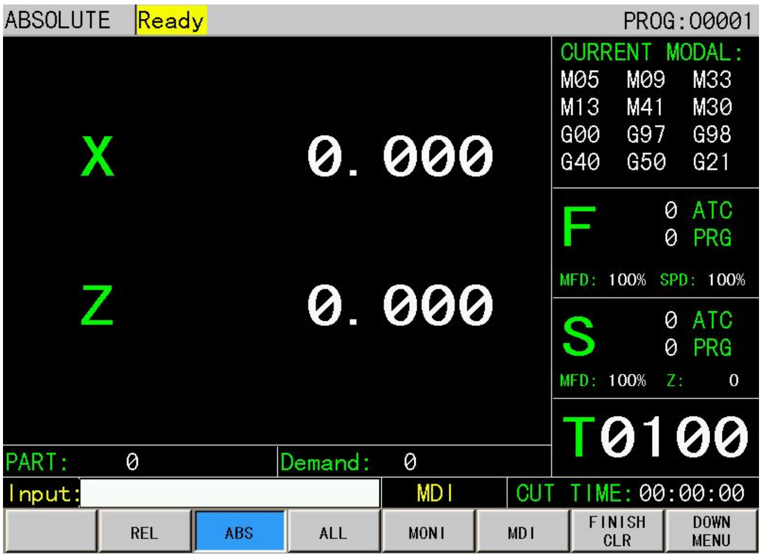


Figure **2-1-1-3**

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 2-1-1-4

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 2-1-1-5**）

Figure 2-1-1-5

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 2-1-1-6**）

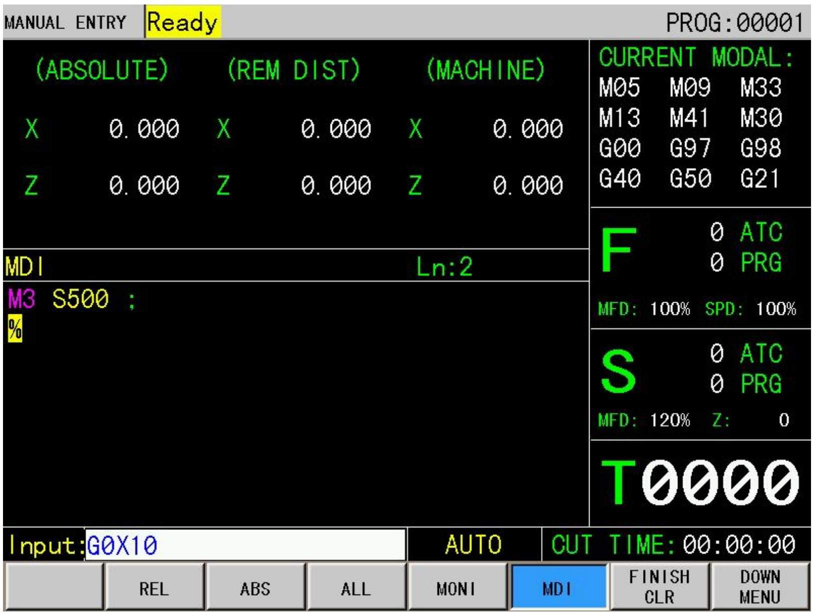


Figure 2-1-1-6

2.1.2 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

2.1.3 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 2-1-1-7**）

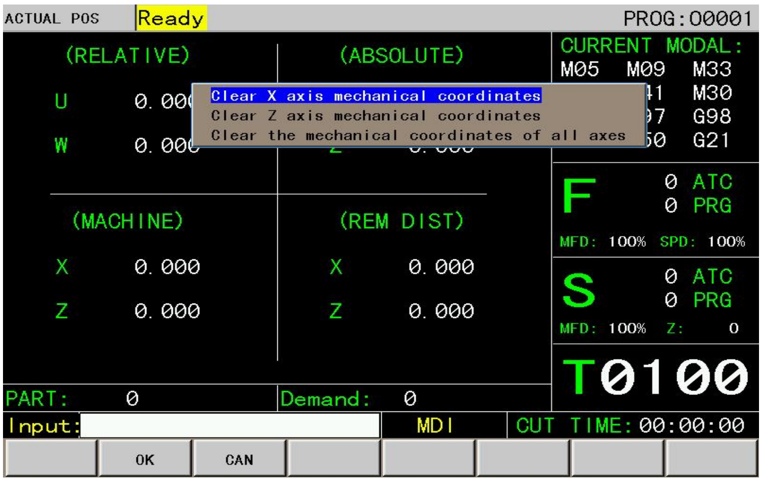


Figure 2-1-1-7

2.2 program display

Press the keys PROGRAM 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 2-2-1**。Specifically, as follows:

2.2.1program 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 2-2-1-1**）。

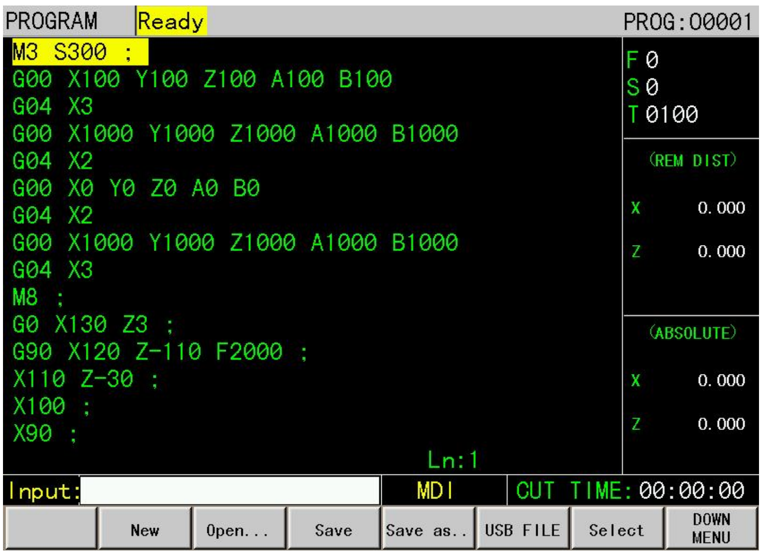


Figure 2-2-1-1

After pressing the soft key (New Program) again, the interface enters the editing and modification page of the program（**See figure 2-2--1-2**）：

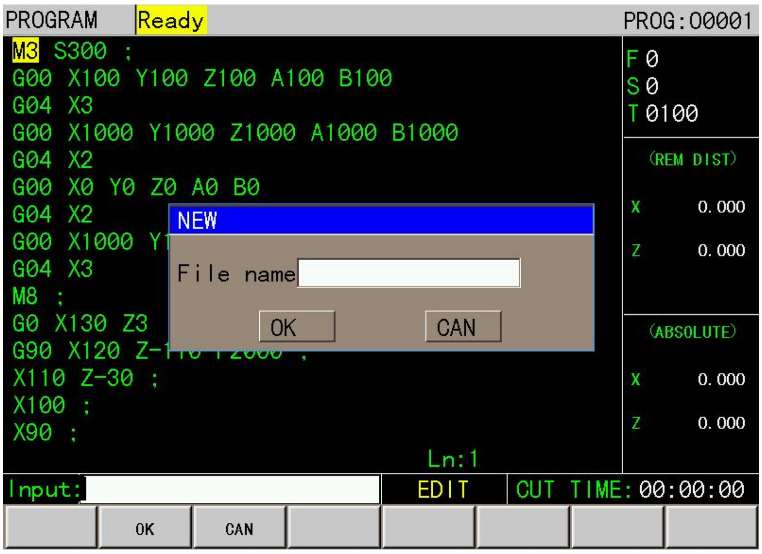


Figure 2-2-1-2

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 2-2-1-3**）

Figure 2-2-1-3

2.2.2 Program (directory) display

Press the (Program Catalog) soft key to enter the directory display interface, and the displayed content is as follows (see Figure 2-2-2-1):

(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.

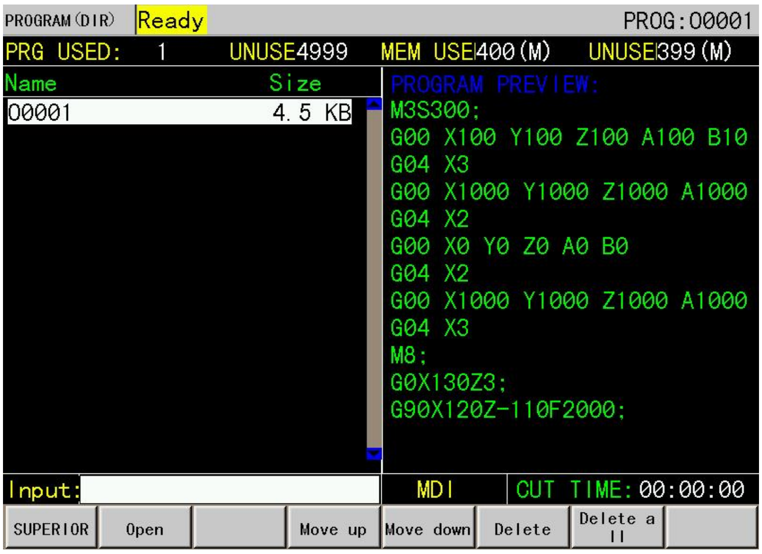


Figure 2-2-2-1

Description: Display all program numbers in the memory through the page key.

2.2.3Program (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.

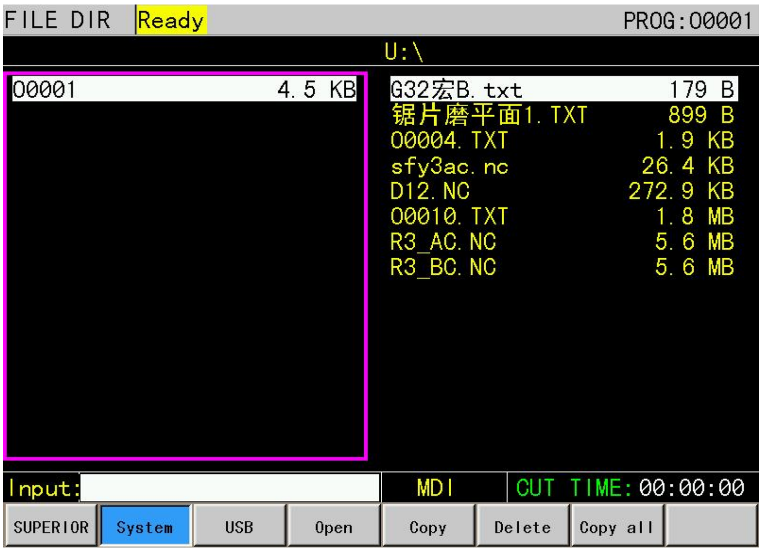


Figure 2-2-3-1

2.3 Tool compensation display, modification, and settings

2.3.1 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 2**–3–1–1）：

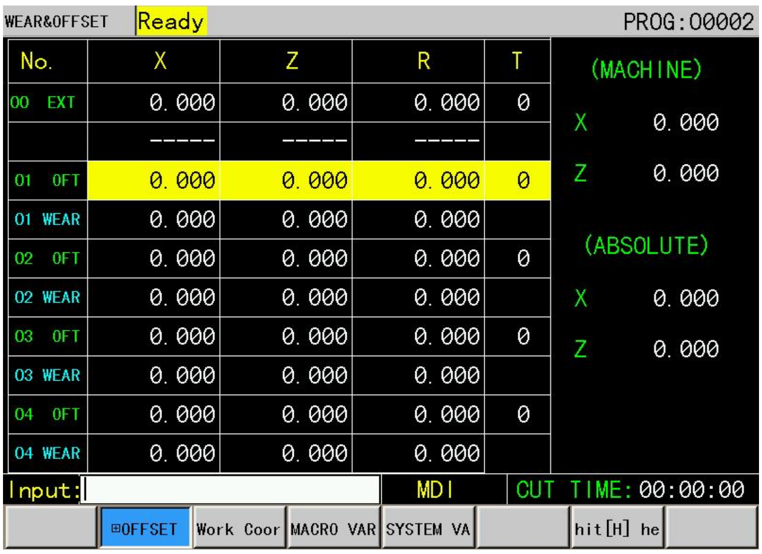


Figure 2-3-1-1

1) Press the function key OFFSET 。**See figure 2**-3-1-1

2) Press the soft key [Offset] to display the tool compensation screen. The variation of the screen depends on the type of tool offset memory.
3) Use the page and cursor keys to move the cursor to the location where you want to set and change the compensation value, or enter the P letter key and a number to find the compensation number.
4) Set compensation value: 1>Press [X], then [Enter], and both the X Tool compensation value and compensation value will be reset to zero

2> Press [Z], then [Enter], Z Tool compensation value and compensation value will be reset simultaneously

3> Repeatedly press the [CHG Conversion] key to lock (or unlock) the blade compensation value

4> After pressing the [X]+value, press [Enter] to establish the X-direction tool compensation value

5> After pressing the [Z]+value, press [Enter] to establish the Z-direction tool compensation value

6> After pressing the [Y]+value, press [Enter] to establish the Y-direction tool compensation value

7> After pressing the [U]+value, press [Enter] to modify the X-direction tool compensation value

8> After pressing the [W]+value, press [Enter] to modify the Z-direction tool compensation value

9> After pressing the [V]+value, press [Enter] to modify the Y-axis tool compensation value

2. 3. 2 [Coordinate System] Interface Operation Instructions

Press the soft key button 【 Coordinate System 】 to enter the workpiece system interface（**See figure 2**-3-2-1）:

Figure 2-3-2-1

Perform the following operations:

(a) Enter the<Enter>/<Edit>operation mode;

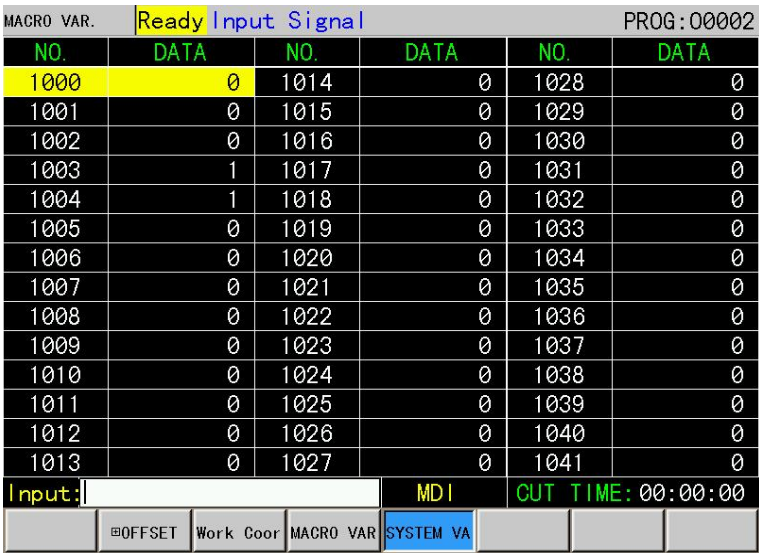
(b) Press the up and down keys to move the cursor to the item to be changed;

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.

2-3-3 Press the (Common Variables) soft key on the common page to enter the common variable interface.See figure 2-3-3-1：

Figure 2-3-3-1

2-3-4 Press the (System Variables) soft key on the System Variables page to enter the System Variables interface（See figure2-3-4-1）：



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

⊞OFFSET | Work Coor | MACRO VAR | SYSTEM VA

Figure 2-3-4-1

2.4 Parameter switch, modification and setting

2.4.1 **Set Display**

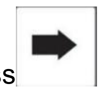
Press the SETTING key to enter the bias information display page, which has five interfaces: [Switch], [Password], [Data Restoration], [Data Backup], and [Graphics]. You can view or modify them through the corresponding soft keys, as follows:

Press the 【 Switch 】 soft key on the switch interface to enter the bias interface（See figure 2-4-1-1）:

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.

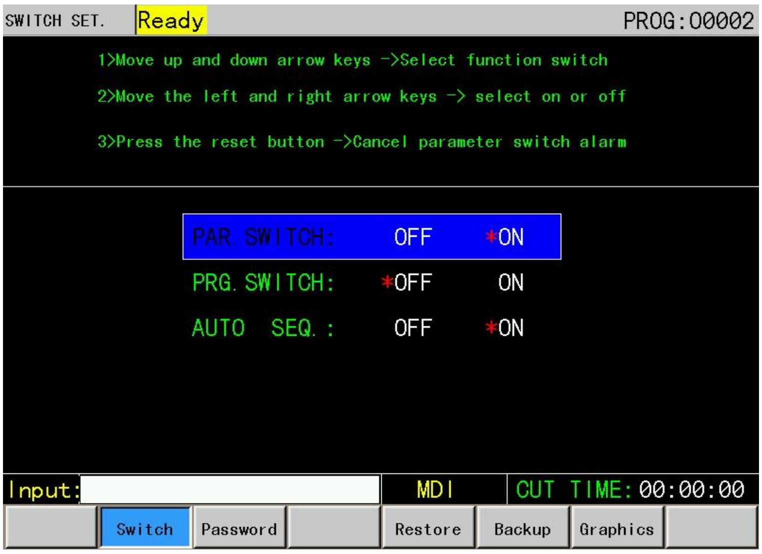


Figure 2-4-1-1

2.4.2 Password modification and setting

Operation of the login interface（**See figure 2-4-2-1**）

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 2-4-2-1**）。

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.

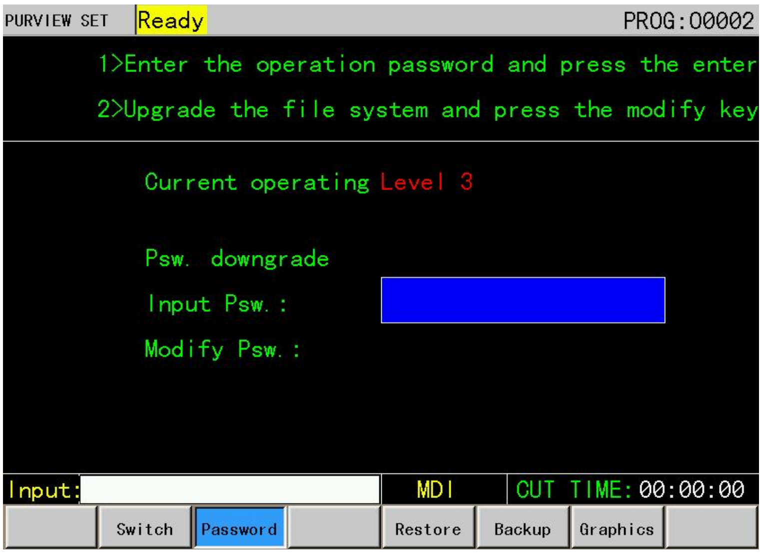


Figure 2-4-2-1

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.

2.4.3 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 2-4-3-1**）

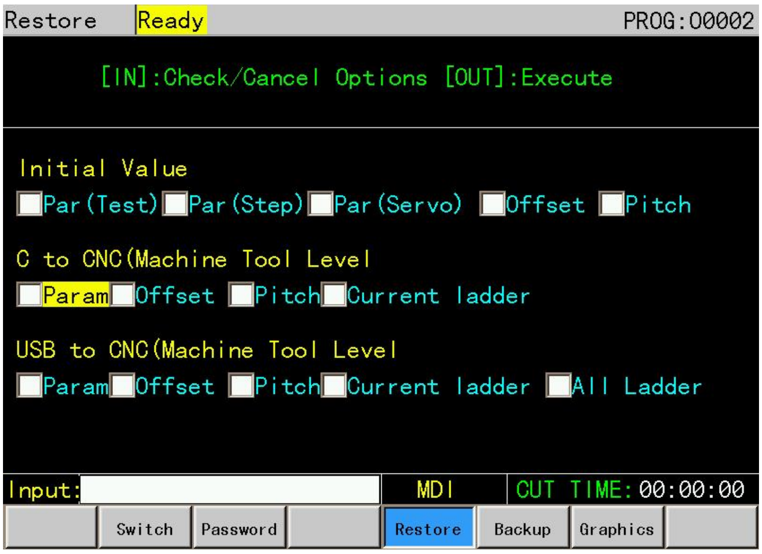


Figure 2-4-3-1

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 INPUT, display a check mark, and press the key OUTPUT to complete the data recovery.

2.4.4 Data backup（See figure 2-4-4-1）

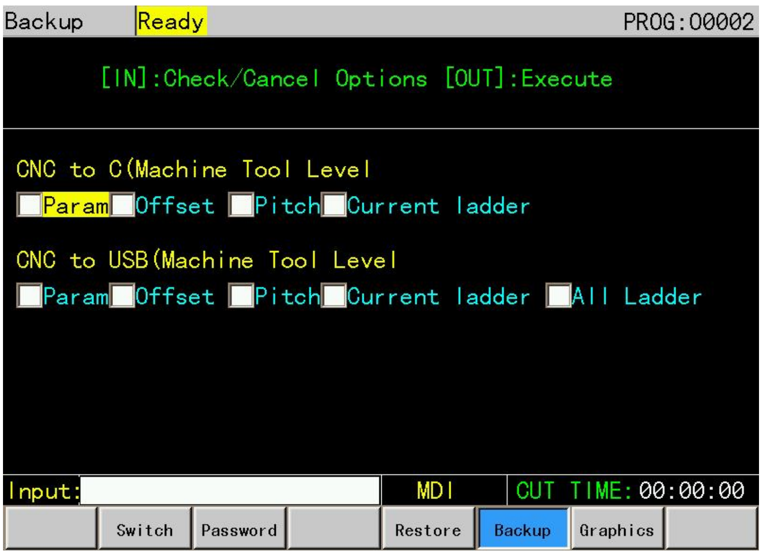


Figure 2-4-4-1

Note: The operation is the same as 2-4-3

2.4.5 **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 2-4-5-1**）

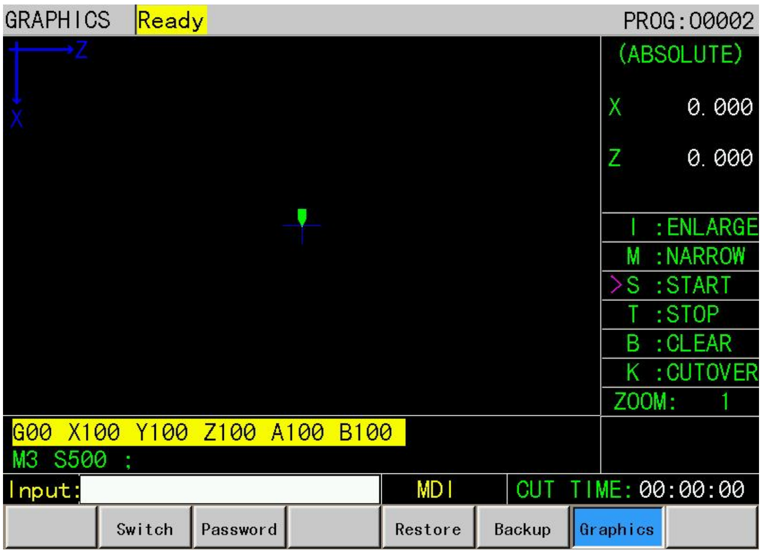
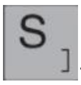


Figure 2-4-5-1

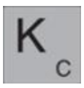
In the graphic page, the machining trajectory of the running program can be monitored.

A、Press 

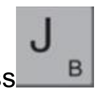
the button to enter the start drawing state, and the '*' sign will move to S: before starting drawing;

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.

2.5 PARAMETER DISPLAY

2.5.1 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**）：

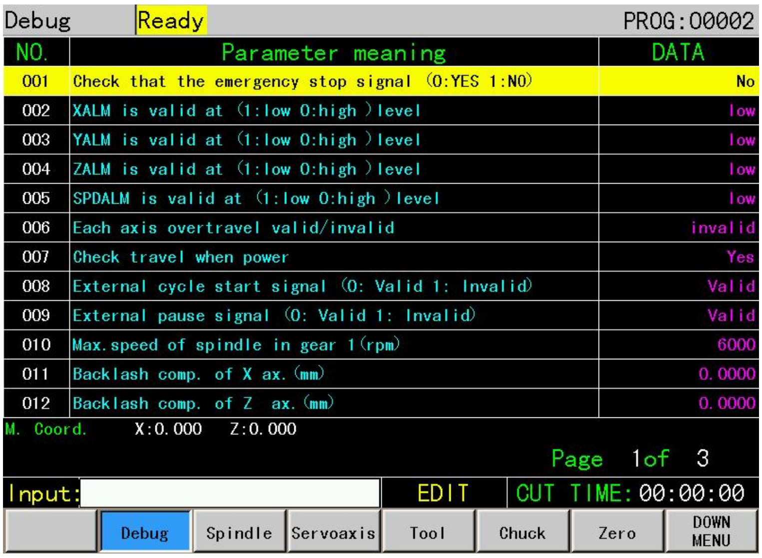


Figure 2–5–1–1

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.

2.6 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:

2.6.1 Diagnostic data display

1、Press the 【 CNC Diagnosis 】 soft key in the<Diagnosis>interface to enter the CNC diagnosis interface.**See figure 2-6-1-1**：

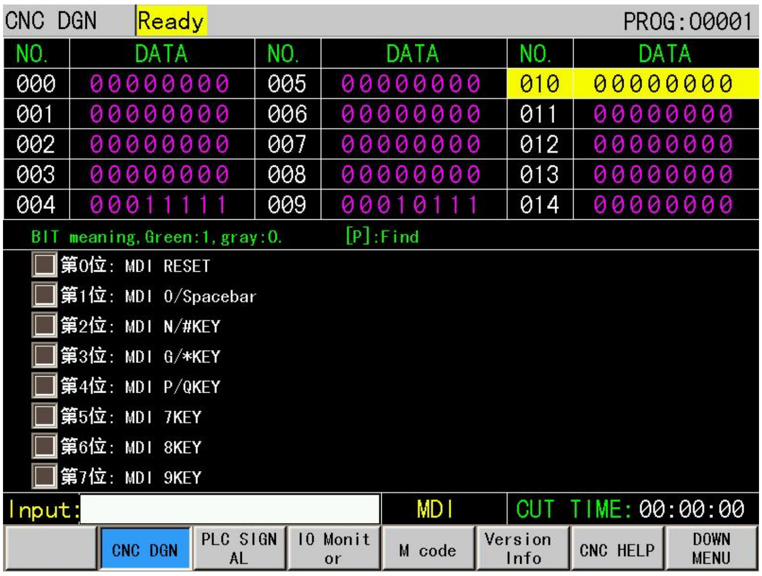


Figure 2-6-1-1

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 ,The 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.

2.6.2 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 2-6-2-1**：

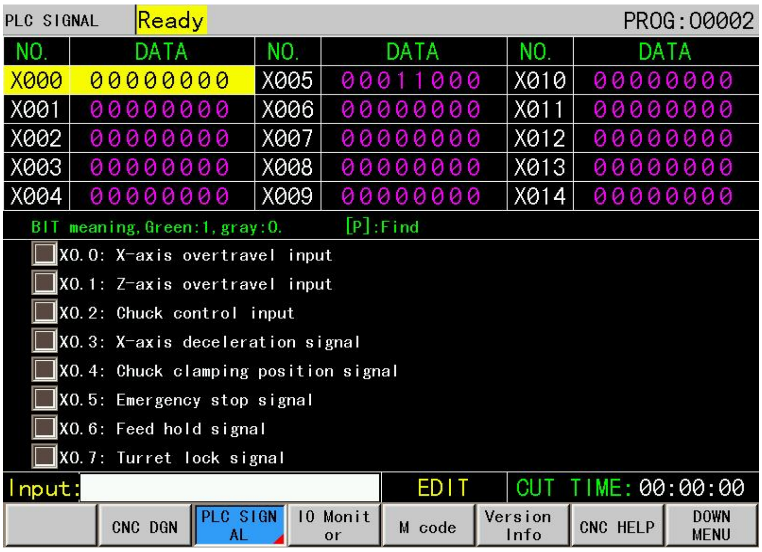


Figure 2-6-2-1

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.

2.6.3 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 2-6-3-1：

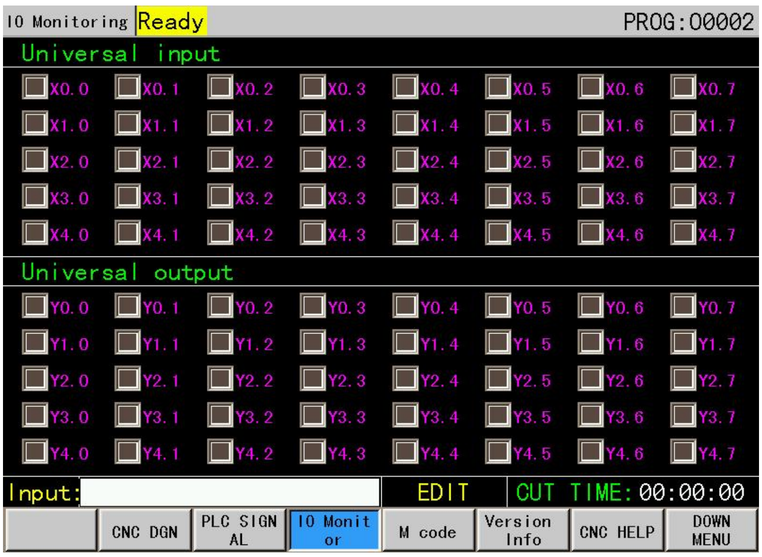


Figure 2-6-3-1

2.6.4 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 2-6-4-1**：

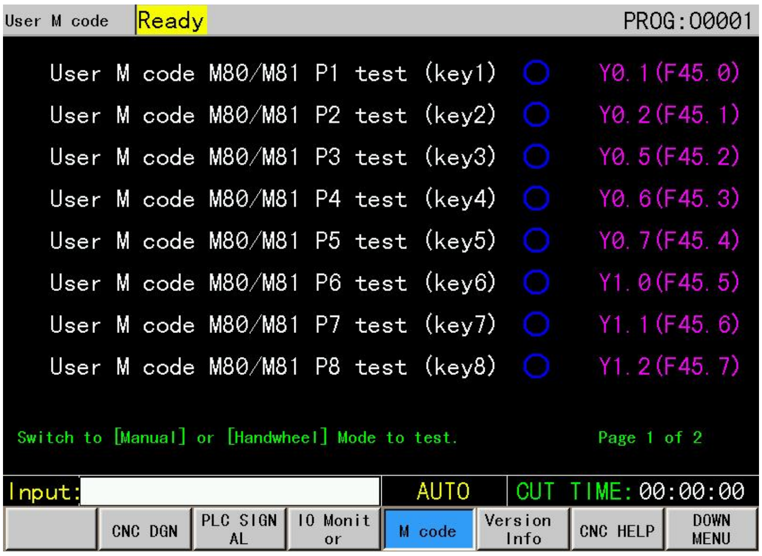


Figure 2-6-4-1

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.

2.6.5 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 2-6-5-1**：

2. 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 2-6-6-2

Figure 2-6-6-2

G CODE 2-6-6-3

Figure 2-6-6-3

Note: Use the up and down arrow keys to move the cursor to the specified G-code. Press the INPUT 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 2**-6-6-4**

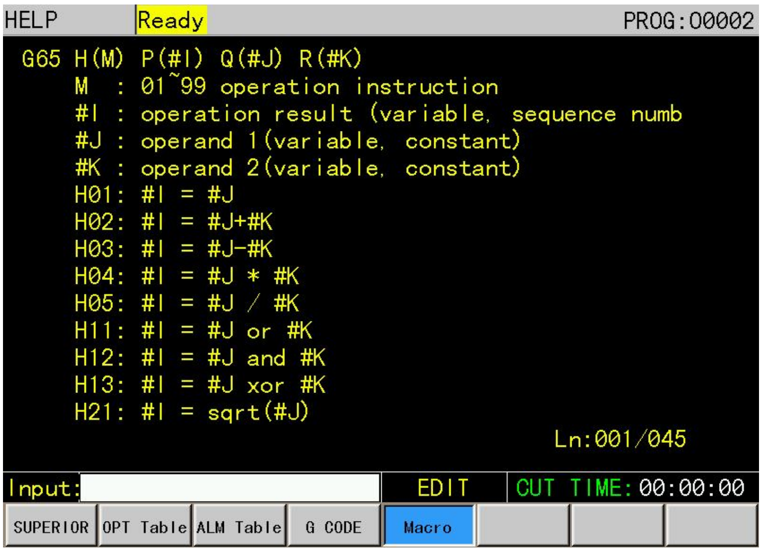


Figure 2-6-6-4

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.

2.7 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:

2.7.1 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

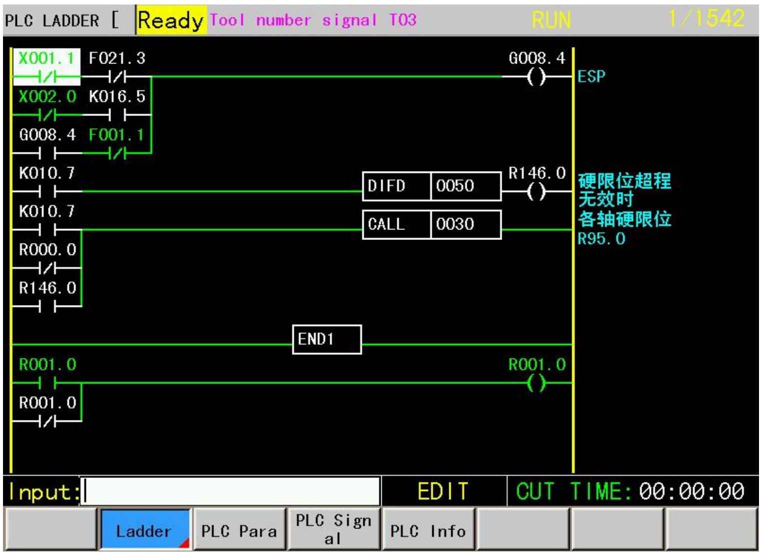


Figure 2-7-1-1

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 2-7-1-2

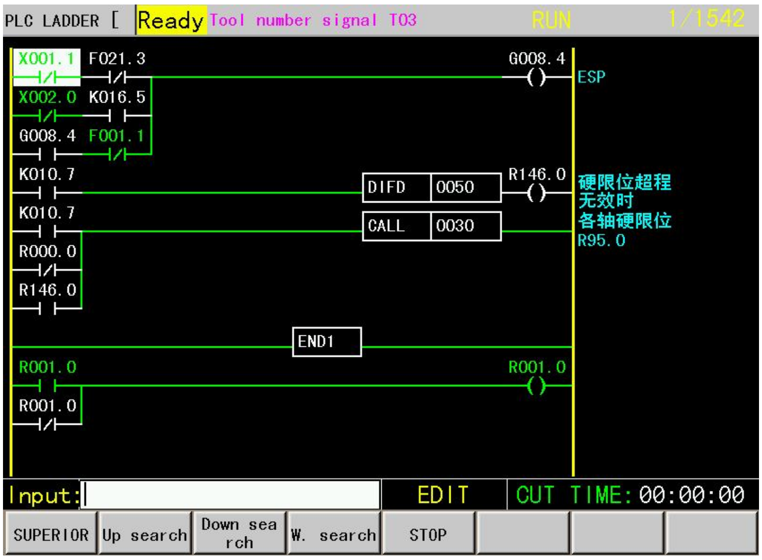


Figure 2-7-1-2

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 2**-7-1-3

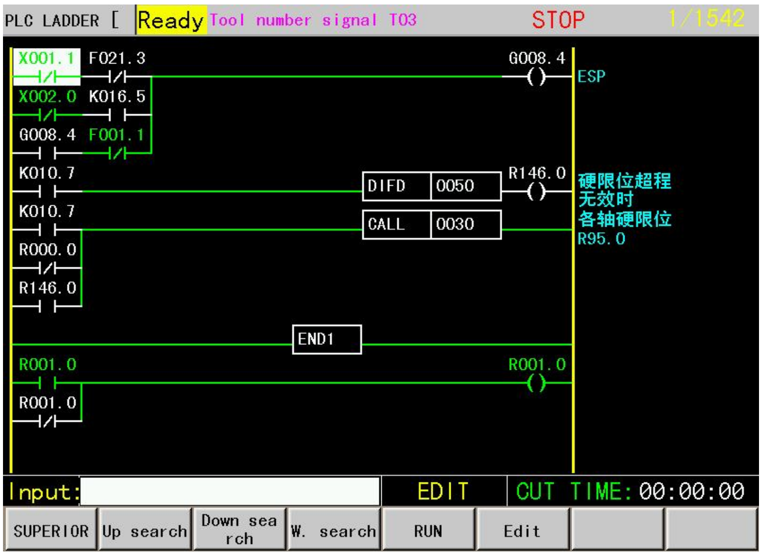


Figure 2-7-1-3

After entering the stop interface, press the 【 Edit 】 soft key to select the touch function,**See figure 2**-7-1-4, 2-7-1-5

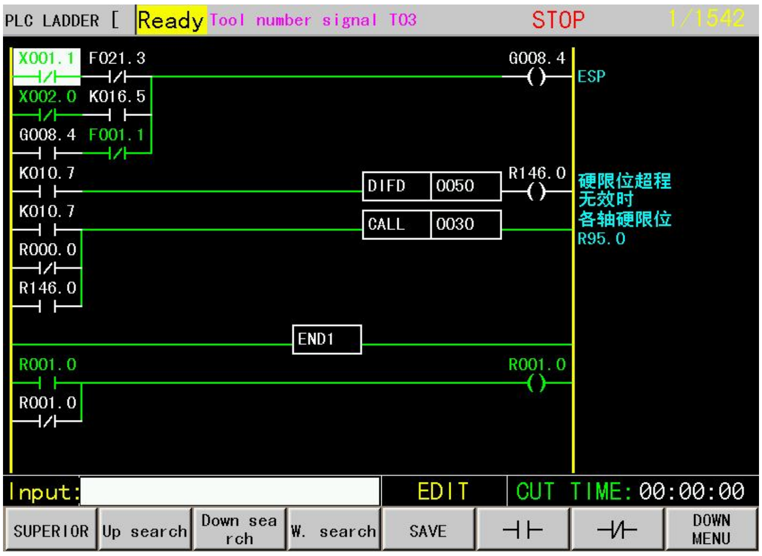


Figure 2-7-1-4

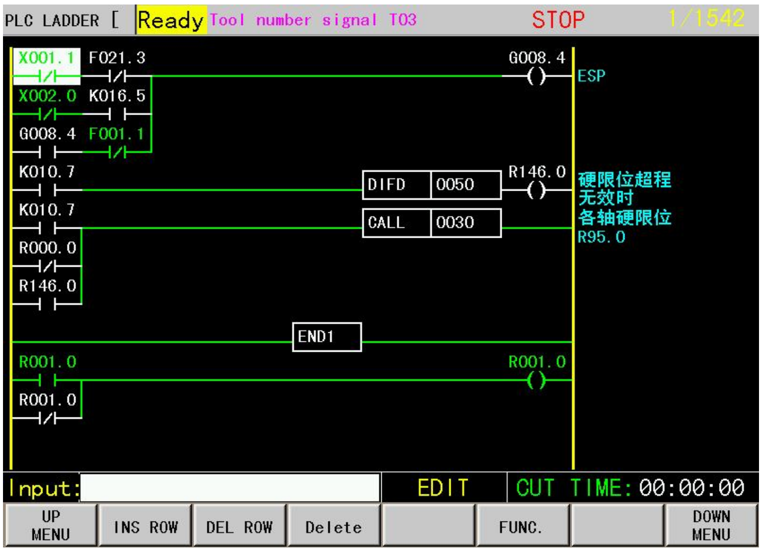


Figure 2-7-1-5

Note: After editing the ladder diagram, you must first press the 'Save' soft key, and then press the 'Run' software

2.7.2 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 2**-7-2-1

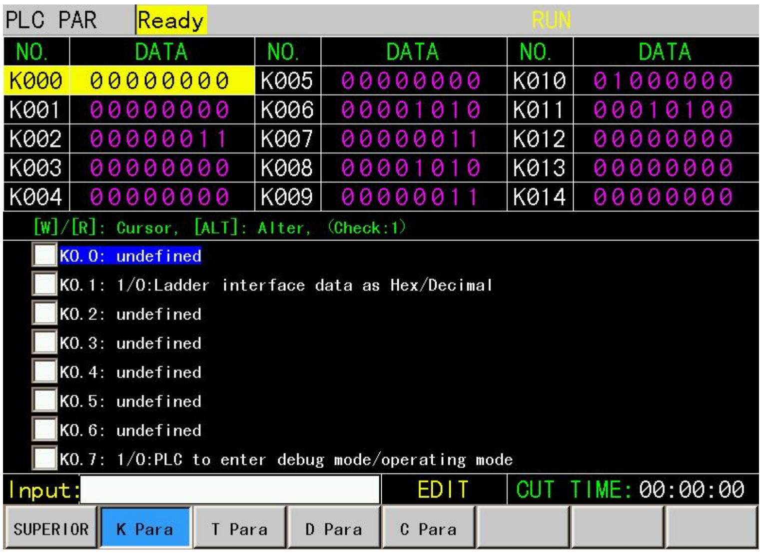


Figure 2-7-2-1

2.7.3 PLC signal

PLC signal interface**See figure 2**-7-3-1

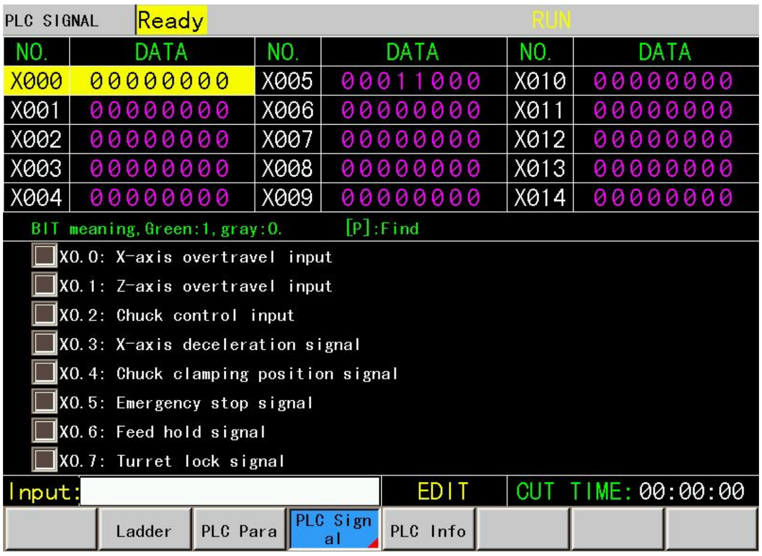


Figure 2-7-3-1

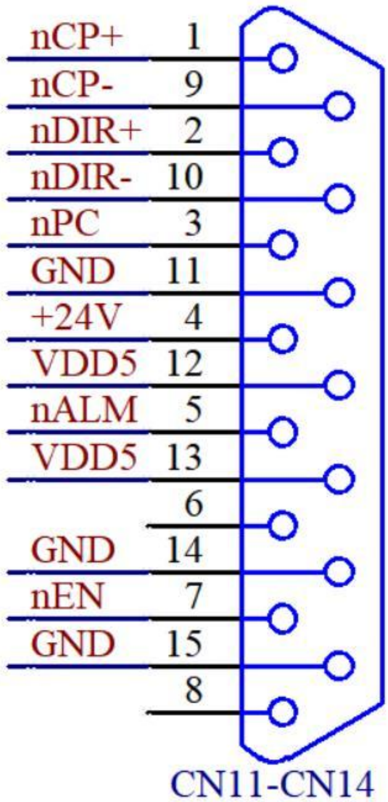
2.7.4 PLC information

PLC Information Interface**See figure 2**-7-4-1

Chapter 3 Debugging and Connection

3.1 X. Connection of Y-axis and Z-axis and A-axis interfaces

3.1.1 Driver interface definition

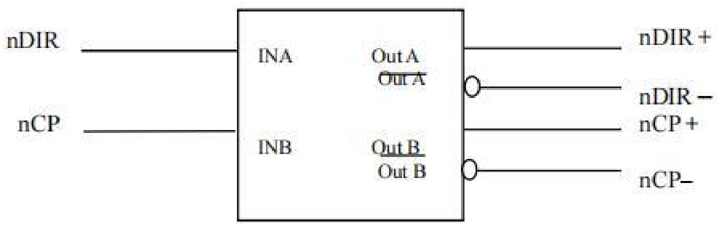


Feed shaft (hole seat)

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	

3.1.2 Command pulse signal and command direction signal

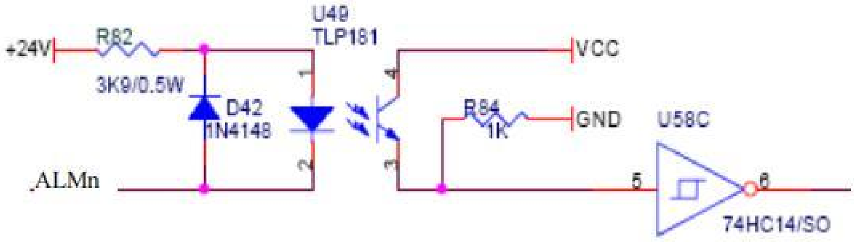
CPn+and CPn - are instruction pulse signals, and DIRn+and DIRn - are instruction direction signals. Both sets of signals are differential (AM26LS31) outputs, and the internal circuit is shown in the following figure



Internal circuit diagram of instruction pulse signal and instruction direction signal

3.1.3 Drive unit alarm signal nALM

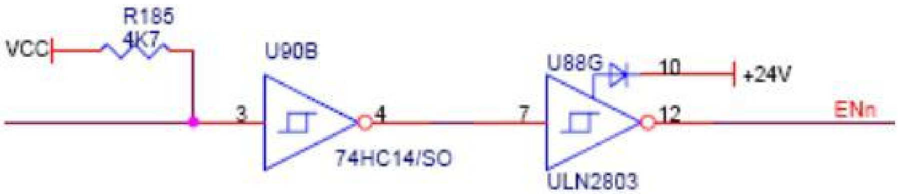
Set the alarm level of the X, Y, Z, 4th, and 5th axis drive units to low or high based on the CNC status parameter №. 009 Bit0 to Bit4 or in the classification parameter [feed axis]. Internal circuit diagram



Internal circuit diagram of drive unit alarm signal

3.1.4 Axis enable signal ENn

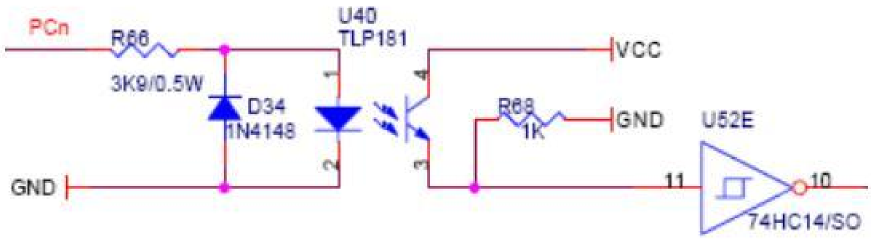
When the CNC is working normally, the ENn signal output is valid (ENn signal is connected to 0V). When the driver alarms, the CNC turns off the ENn signal output (ENn signal is disconnected from 0V). The internal interface circuit is shown in the following figure



Internal interface circuit diagram of axis enable signal

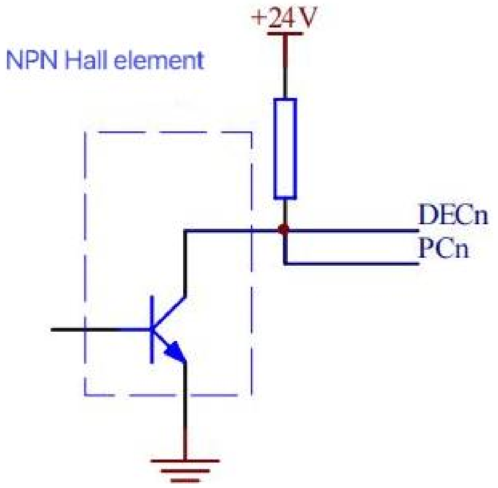
3.1.5 Zero point signal PCn

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



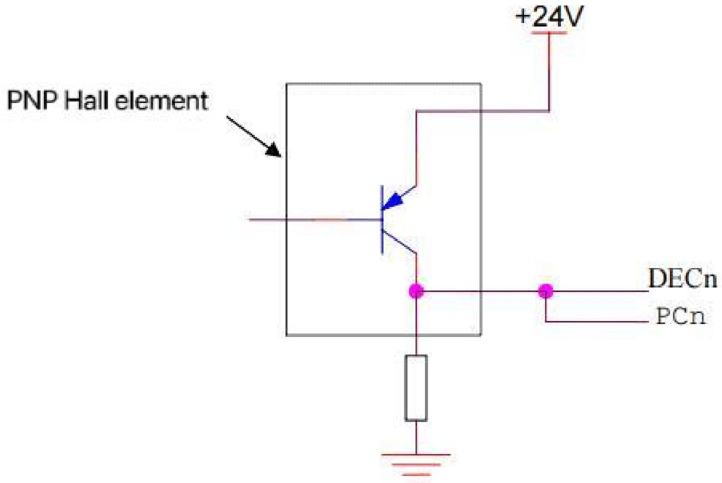
Zero signal circuit diagram note: PCn signal adopts+24V level

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



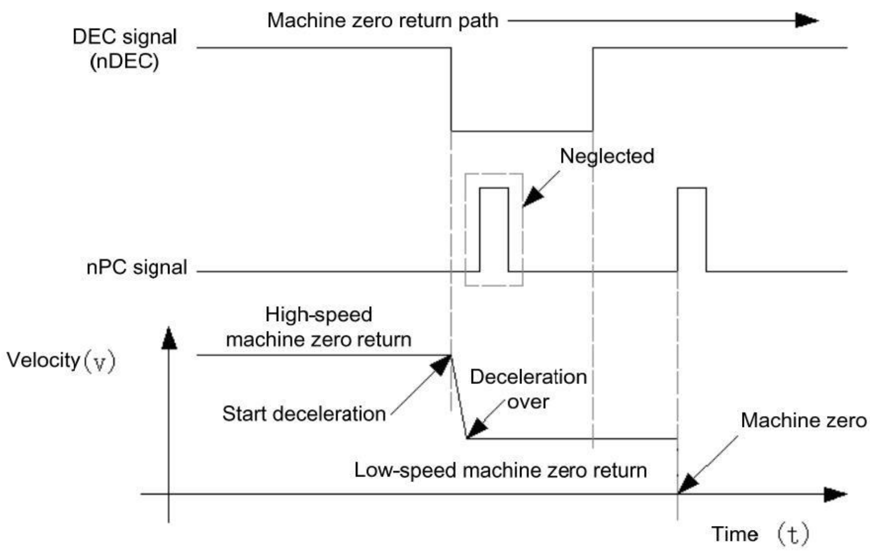
Connection diagram of NPN Hall element: Pull up resistor connected to 2K

b）The connection method for using a PNP Hall element for both deceleration signal and zero signal is shown in the following figure



Connection diagram using PNP type Hall element

The waveform of the system's PC signal is shown in the following figure:

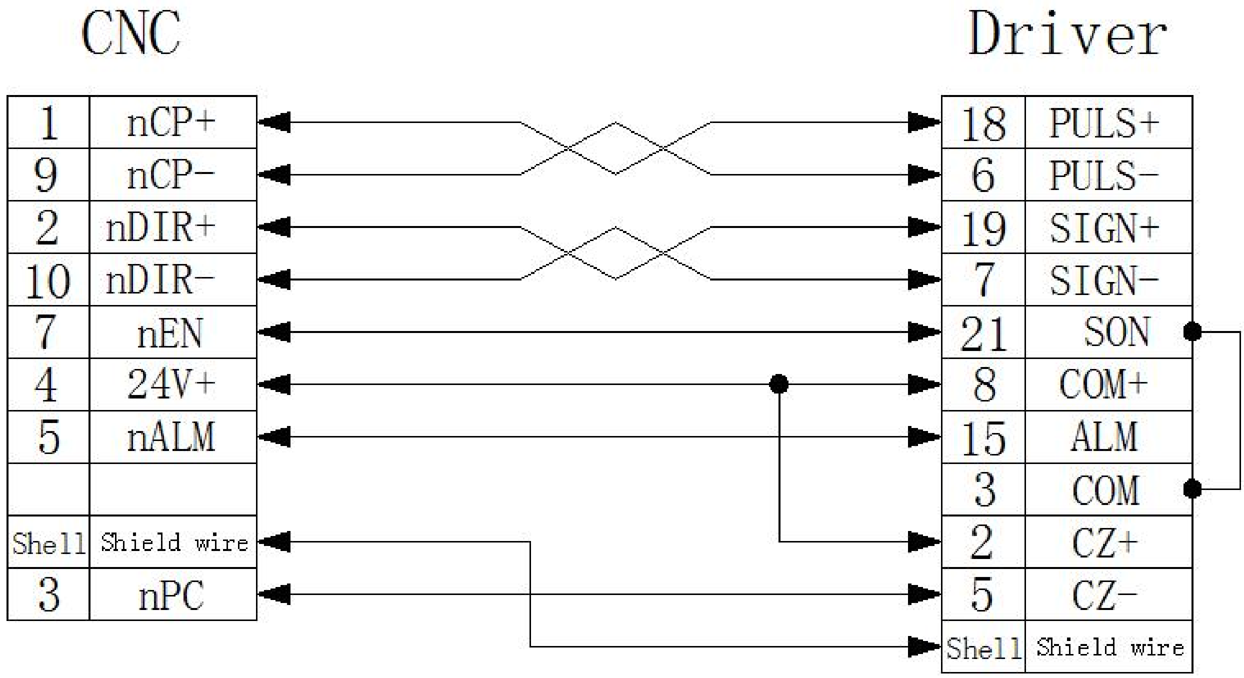


PC signal waveform diagram

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.

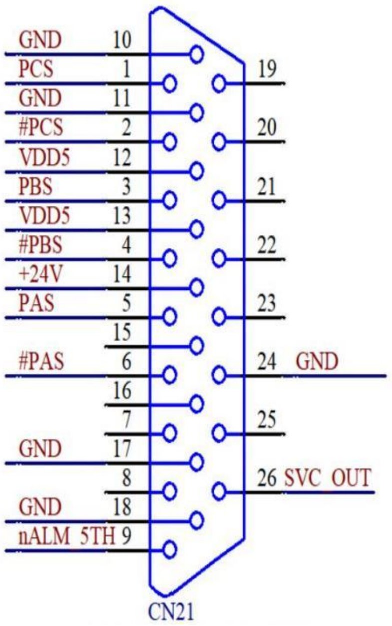
3.1.6 Connection with servo drive

The connection between the system and our company's driver unit is shown in the following figure:



3.2 Connection of spindle encoder interface

3.2.1 Definition of spindle encoder interface

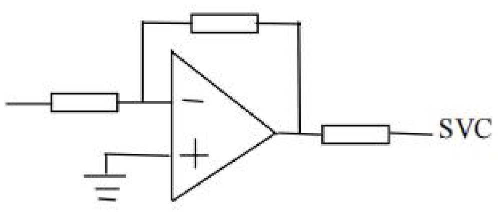


Spindle encoder/analog voltage (hole seat)

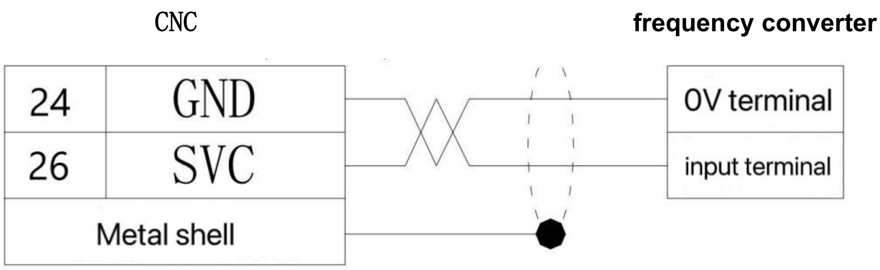
Pin	signal	description	Wire Color
1	PCS	Encoder Z phase pulse	blue
2	#PCS	Encoder Z- phase pulse	Blue black
3	PBS	Encoder B phase pulse	black
4	#PBS	Encoder B- phase pulse	Black white
5	PAS	Encoder A phase pulse	green
6	#PAS	Encoder A- phase pulse	Green black
12.13	VDD5	5V	red
11	GND	0V	Red white
26	SVC OUT	0~10V Analog voltage	Red (four core)
10.24	GND	0V	Red white (four core)
9	ALM	Spindle Alarm	Blue (four core)
17.18	GND	0V	Blue black (four core)

3.2.2 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:



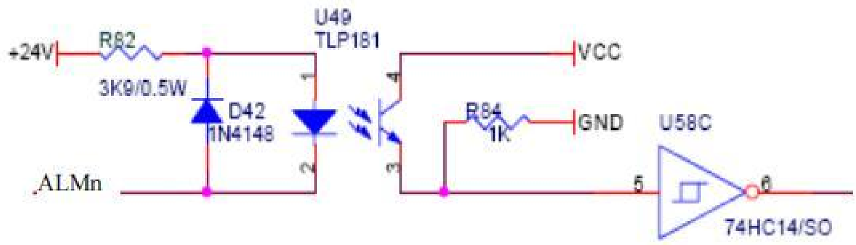
SVC signal internal circuit diagram



Connection between the system and the frequency converter

3.2.3 ALM Description

The spindle alarm signal is used as an alarm signal for ordinary frequency converters or gear spindles

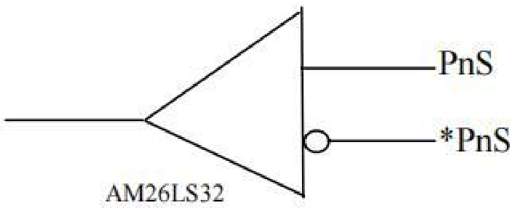


Internal structure diagram of spindle alarm note: spindle alarm input 0 volts is valid

3.2.4 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.

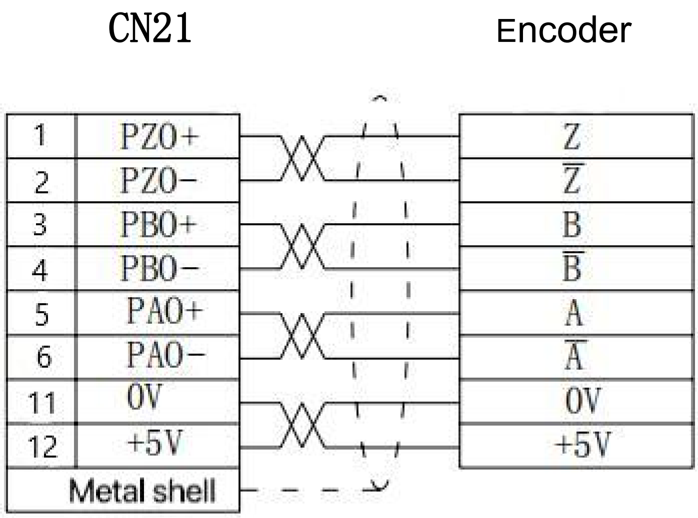
Its interior circuit is shown in Fig



Encoder signal circuit

3.2.5 Spindle encoder interface connection

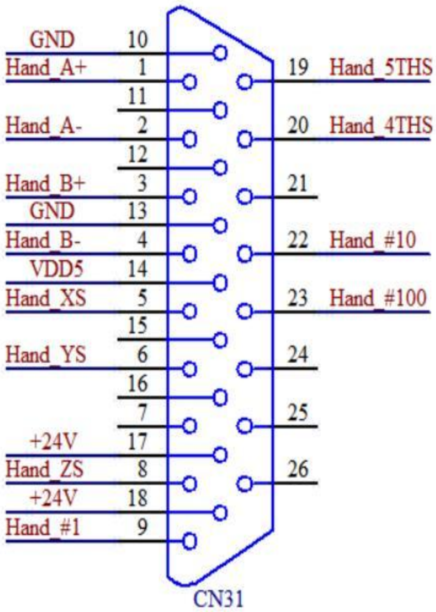
The connection between the system and the spindle encoder is shown in the following figure, using twisted pair cables.



Connection between system and encoder

3.3 Connection of MPG interface

3.3.1 Hand pulse interface definition

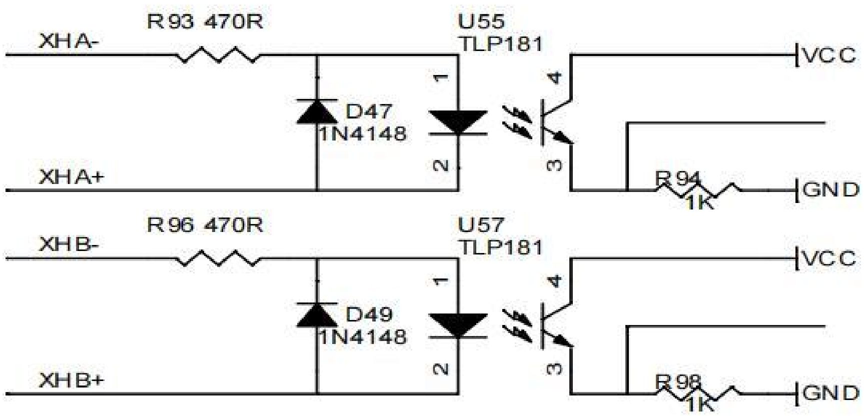


Hand pulse (needle socket)

Pin	signal	description	Wire Color
1	Hand A+	MPG A+ phase signal	black
2	Hand A-	MPG A- phase signal	Black white
3	Hand B+	MPG B+ phase signal	blue
4	Hand B-	MPG B- phase signal	Blue black
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	red
10.13	GND	0V	Red white
17.18	24V	24V axis selection common terminal	

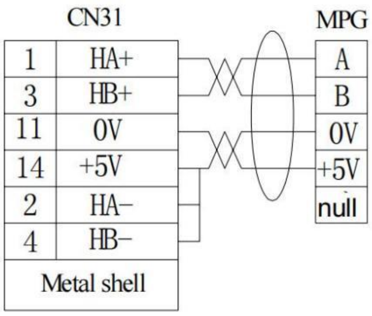
3.3.2 **signal description**

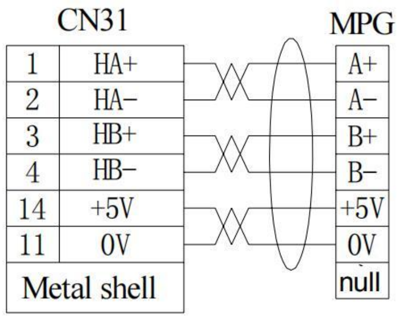
HA+、HA-、HB+、HB- The input signals for hand pulse phases A and B. The internal connection circuit is shown in the following diagram



MPG signal circuit

The connection between the system and the hand pulse is shown in the following figure

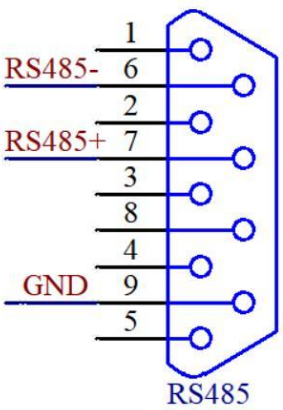




CN31 connects with the single-port MPG input difference MPG input

Note: The external handheld box handwheel axis selection and magnification wiring common terminal COM is 24 volts.

3.4 communication interface：



Communication (hole socket)

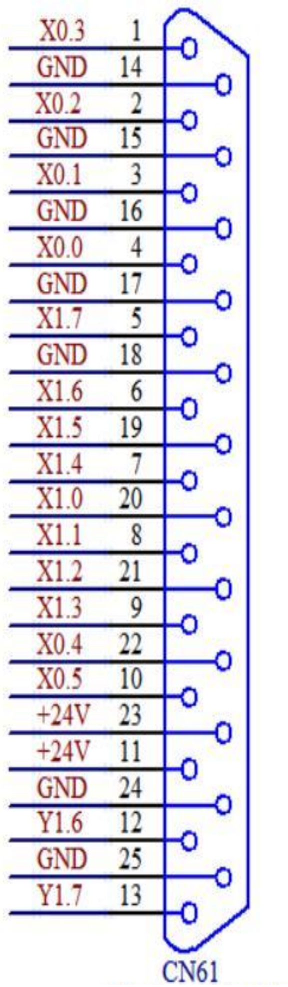
Pin	signal	description
6	RS485-	**signaling**
7	RS485+	**signal reception**
9	GND	0V

485 communication usage: The system reads the motor position through 485 communication and driver connection. No other functions

3.5 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!**

3.5.1 Input signal



Input 1 (needle socket)

Pin	Address	Explanation	Wire Color
1	X0.3	No. 6 Tool position signal/external feed allowance signal	orange
2	X0.2	No. 5 Tool position signal/spindle rotation permission signal	null
3	X0.1	Z-axis deceleration signal back to reference point	Orange black
4	X0.0	Chuck control input	brown
5	X1.7	X-axis deceleration signal back to reference point	Brown black
6	X1.6	X-axis limit switch	blue
9	X1.3	Z-axis limit switch	Blue black
12	Y1.6	Tool holder forward rotation output	yellow
13	Y1.7	Tool holder reverse output	Yellow black
19	X1.5	No. 4 Tool position signal	Green black
20	X1.0	No. 1 Tool position signal	Red white
21	X1.2	No. 2 Tool position signal	Black white
22	X0.4	No. 3 Tool position signal	black
23	+24V	24V	red
14～18 24.25	0V	0V	green
7	X1.4	External start signal ST	Blue black (four core)
8	X1.1	Emergency stop signal ESP	Blue(four core)
10	X0.5	External pause signal SP	Red white (four core)
11	+24V	24V	Red(four core)

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

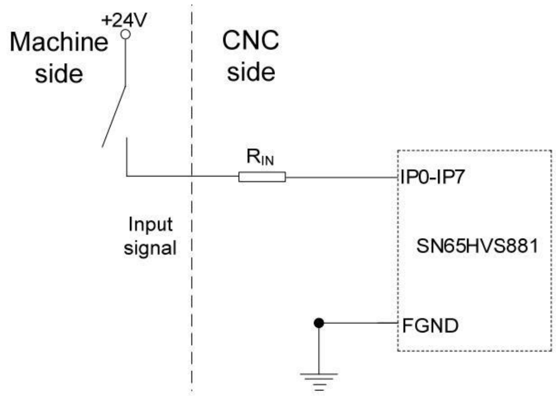
The input signal refers to the signal from the machine tool to the CNC, which is valid when connected to+24V; When the input signal is disconnected from+24V, the input is invalid. The contact points of the input signal on the machine tool side should meet the following conditions:

Contact capacity: DC30V, 16mA or above

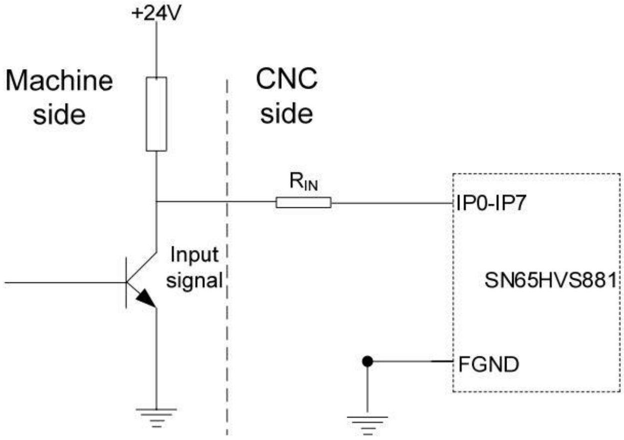
Leakage current between contacts during open circuit: below 1mA

Voltage drop between contacts during closed circuit: below 2V (current 8.5mA, including cable voltage drop)

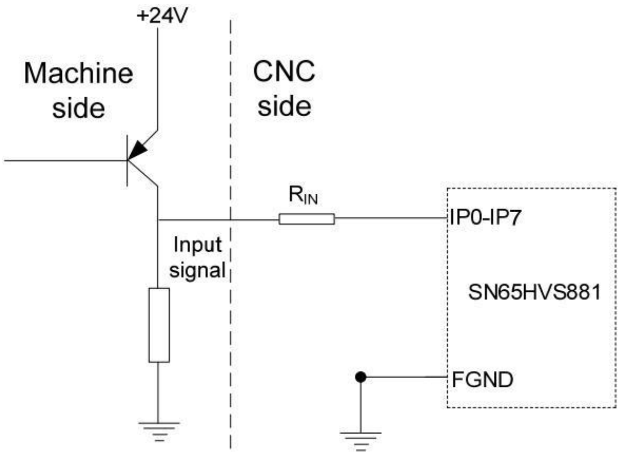
There are two ways to input external signals: one is to use a contact switch input, and the signals using this method come from the machine's side buttons, limit switches, and relay contacts, etc., connected as shown in the diagram



The other type is input by switch with no contacts (transistor) as shown in Fig:

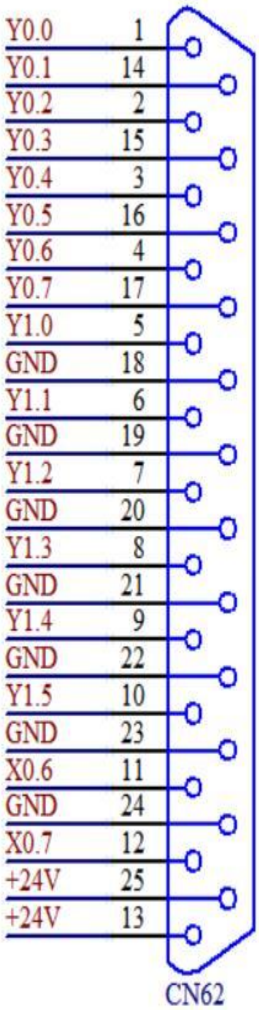


Connection of NPN type



Connection of PNP type

3.5.2 Output Signal



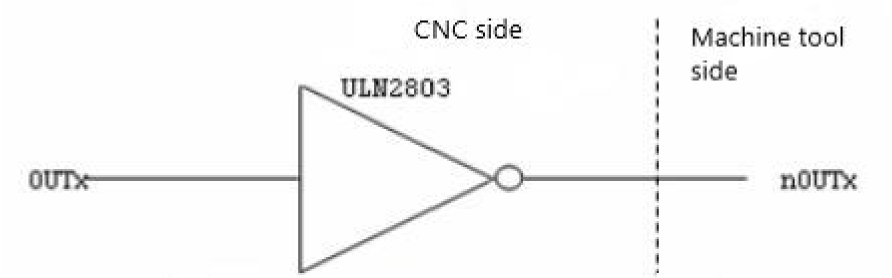
Output 1 (hole socket)

Pin	Address	Explanation	Wire Color
1	Y0.0	cool	Purple black
2	Y0.2	Tail seat input/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	Brown black
8	Y1.3	Gear spindle 4/red light	brown
9	Y1.4	Chuck clamping output	Orange black
10	Y1.5	Chuck release output	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	Tail seat retraction output	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

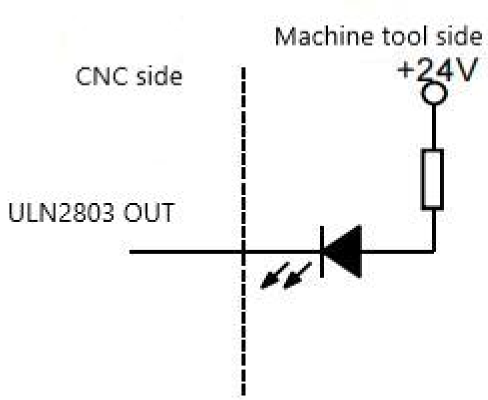


Circuit structure diagram of digital output module

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:

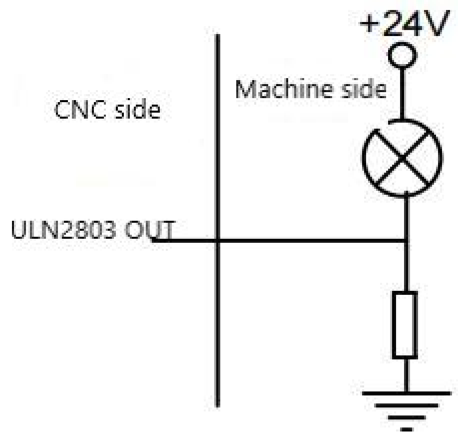
Drive LED

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



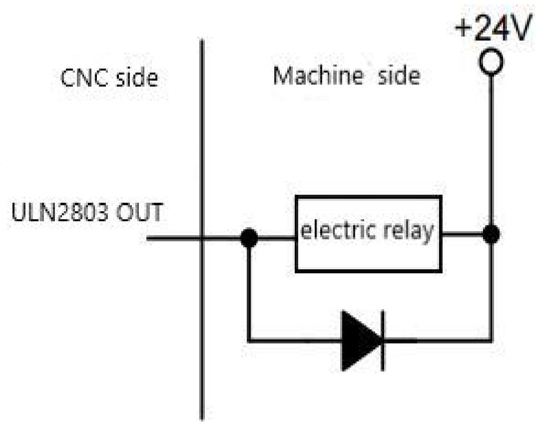
Drive filament type indicator light

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)

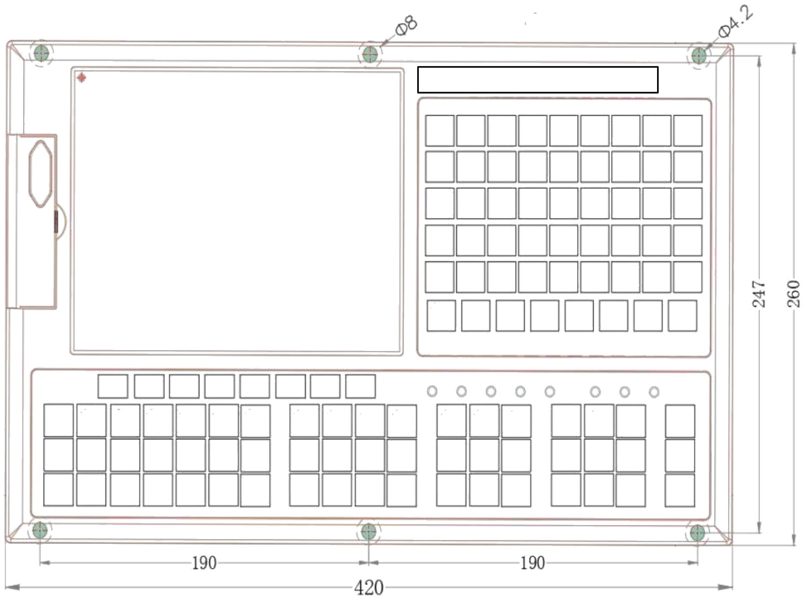
When using ULN2803 output to drive inductive loads, it is necessary to connect a freewheeling diode near the coil to protect the output circuit and reduce interference. As shown in the following figure

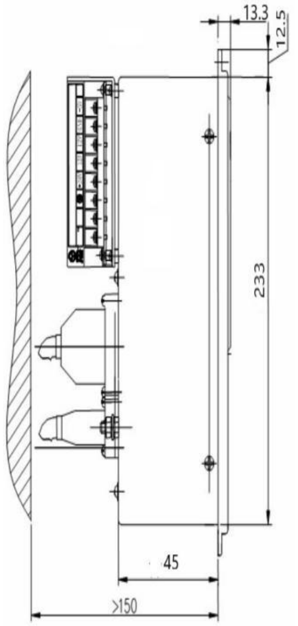


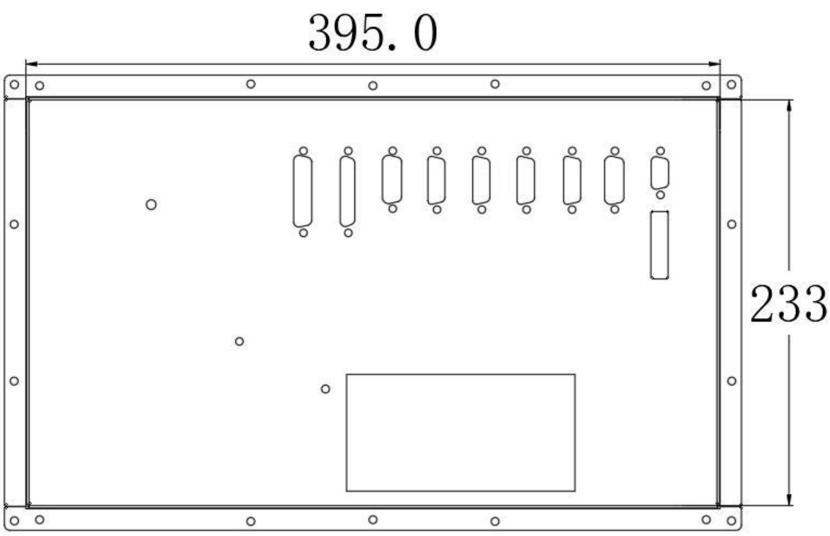
3.6 MPG Box connection

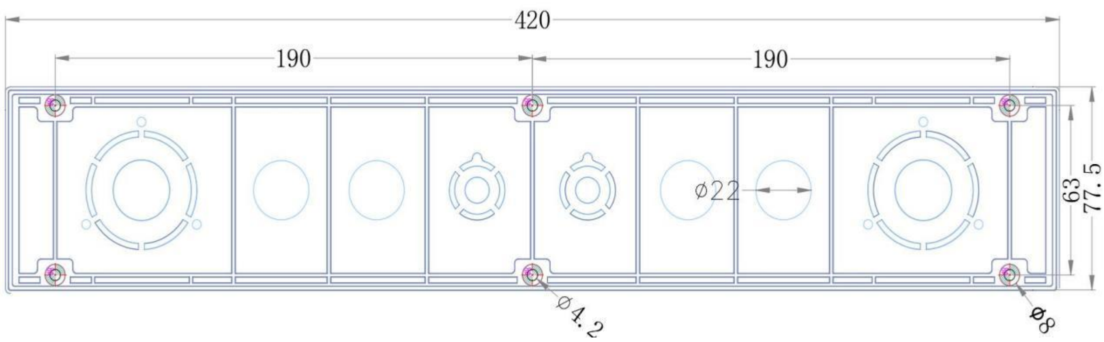
description	Wire Color
+V Pulse Generator	red
0V Pulse Generator	black
A Pulse Generator	green
B Pulse Generator	white
/A Pulse Generator	purple
/B Pulse Generator	Purple black
X MPG axis selection	yellow
Y MPG axis selection	Yellow black
Z MPG axis selection	brown
4TH_axis selection signal	Brown black
*1 Select magnification	grey
*10 Select magnification	Grey black
*100 Select magnification	orange
COM Control switch	Orange black
C Emergency stop switch	blue
CN Emergency stop switch	Blue black
N.Cshield	

3.7 CNC Installation dimensions









Additional panels

Chapter 4 I/O Function and Connection

4.1.1 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

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:

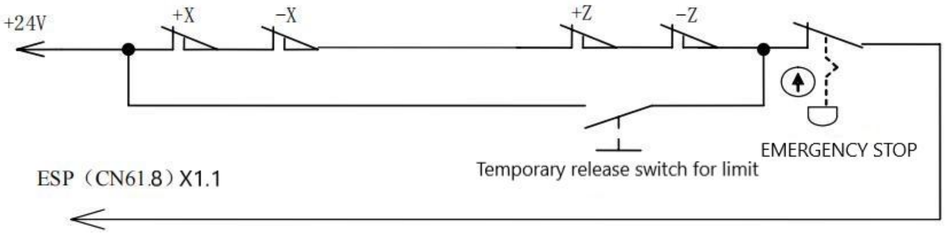


Fig.2-27 Series connection between emergency stop and travel switch

② The separate connection between the emergency stop and travel switch is shown in Fig. 2-34B

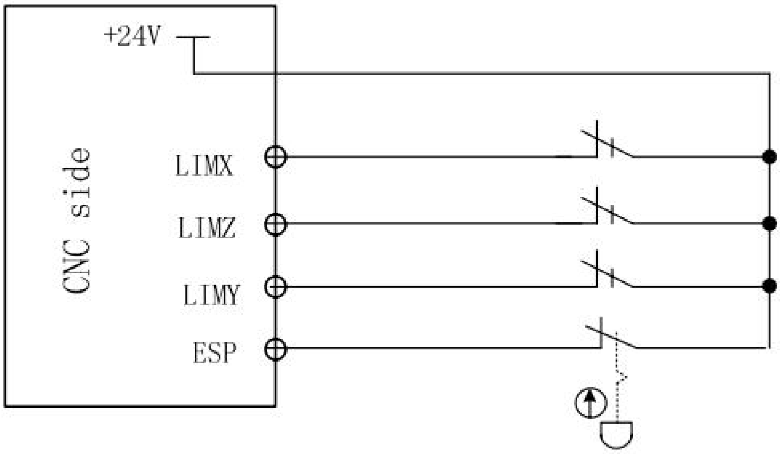


Fig. 2-28 Separate connection between the emergency stop and travel switch

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

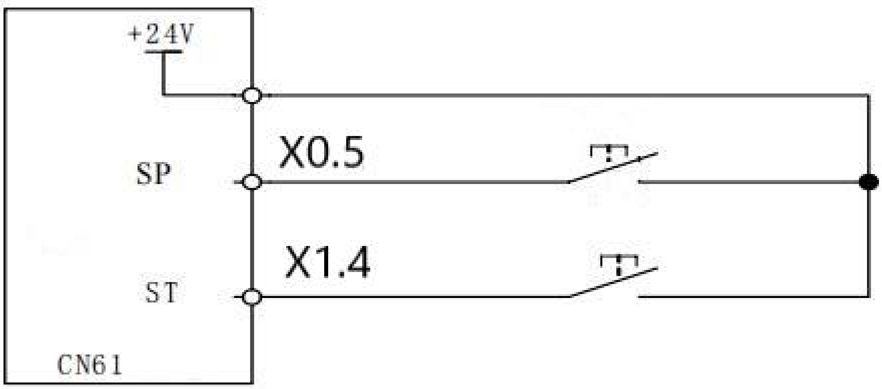
4.1.2 External cycle start and feed hold

- 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**

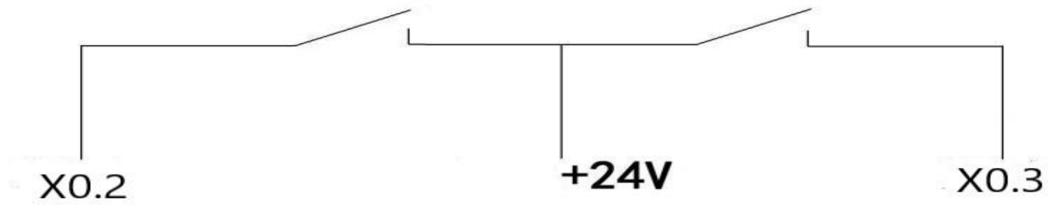
Parameters - [PLC Parameters]

001	External loop start signal (0: high level 1: low level)	high level
002	External input to hold signal (0: high level 1: low level)	high level

4.1.3 Three position start pause switch

- External connection circuit

The external connection of the three temporary start signals is shown in the following figure:



Main axis allowable signal　　Feed allowance signal

➢ **Control parameters**

Parameters - [PLC Parameters]

003	Three position switch function (0: invalid 1: valid) K13.2	valid
004	Three position switch function cycle start (0: invalid 1: valid) K13.4	valid
005	Three position switch spindle rotation permission signal switch type (0: normally open 1: normally closed) K13.3	normally open
006	Three position switch external feed signal switch type (0: normally open 1: normally closed) K13.7	normally open

➢ Related warnings:

A21.0:The three position switch is not turned to the (normal) position, unable to start the program

A21.1:The three position switch is not turned to the (normal) position, and the spindle cannot be started

4.1.4 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]

010	Lubrication and cooling of the spindle during reset (0: off 1: hold)	off

Parameters - [Spindle Parameters]

014	After the spindle stops (M05), the spindle braking delay output time is measured in milliseconds	0
015	Spindle brake output time (in milliseconds)	0

➢ **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.

4.1.5 SPINDLE JOG

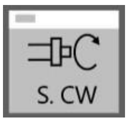
➢ **Control parameters**

Parameters - [Spindle Parameters]

011	Effective range of spindle jog (0: manual, handwheel, return to zero 1: any method)	any method
012	Spindle jog time T012	0
013	Rotation speed during spindle jog	300

➢ **control logic**

Press the key in incremental, manual, and pulse 

modes to enter the spindle jog mode. Press 

the button and turn the spindle counterclockwise to jog; Press 

the button and turn the spindle clockwise. After releasing the forward and reverse buttons, the spindle jog time is set by PLC parameter T12.

4.1.6 Spindle orientation and position switching function

➢ **Address Definition**

Y1.0： Spindle orientation output signal (M19)

Y1.1： Output signal for spindle position switching (M14)

X0.6: Spindle orientation detection completion signal

X0.7: Spindle position detection completion signal

➢ **Control parameters**

Parameters - [Spindle Parameters]

030	Main axis orientation M19 and CS axis M14 function (0: invalid 1: valid)	invalid
032	Spindle orientation completion signal level selection (0: high level 1: low level)	high level
033	Whether M19 has detected the signal in place (0: detect 1: not detect)	detect
034	M14 position switch completed signal level selection (0: high level 1: low level)	high level
035	Whether M14 has detected the signal in place (0: detect 1: not detect)	detect
036	Main spindle position switch detection failure signal delay time alarm (unit: milliseconds)	2000
037	Axis orientation detection not in place signal delay time alarm (unit: milliseconds)	2000

Note: When the in place signal is not detected, the completion signal is used as the in place signal through delay.

➢ 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 M14, 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 M15, and the spindle will switch to speed control mode; If the servo drive triggers an alarm or a switch timeout occurs, the CNC will switch the spindle back to speed control mode

4.1.7 Chuck control

- **Address Definition**

CN62 output pin number	PLC address	function	describe
9	Y1.4	DOQPJ	External card clamp output signal/internal card release output (M12)
10	Y1.5	DOQPS	External card release output signal/internal card clamping output (M13)
11	X0.6	WQPJ	External card plate clamping in place signal/internal card plate loosening in place
12	X0.7	NQPJ	External card disc release in place signal/internal card disc clamped in place

CN61 input pin number	PLC address	function	describe
4	X0.0	DIQP	Chuck control input signal

- **Control parameters**

Parameter - [Chuck Tail Seat]

001	Chuck control (0: invalid 1: effective) K12.0	invalid
002	Chuck control mode (0: internal card 1: external card) K12.2	internal card
003	Spindle rotation and chuck (0: interlock 1: non interlock) K12.1	interlock
004	Chuck in place signal (0: not checked 1: checked) K12.3	not checked
005	Does it remember the chuck status (0: no memory 1: memory) after power failure K13.5	no memory
006	The spindle stops and the chuck can be released for clamping delay operation time T022	0
007	The spindle zero speed output range (r/min) allows the chuck to release D100	5

➢ **control logic**

Signal under external card disk mode	Chuck clamping	DOQPJ (Y1.4): Chuck clamping output signal
		WQPJ(X0.6):Chuck clamping in place signal
	Chuck release	DOQPS (Y1.5): Chuck release output signal
		NQPJ(X0.7):Chuck release in place signal
Signal in external card disk mode	Chuck clamping	DOQPS (Y1.5): Chuck clamping output signal
		NQPJ(X0.7):Chuck clamping in place signal
	Chuck release	DOQPJ (Y1.4): Chuck release output signal
		WQPJ(X0.6):Chuck release in place signal

①When SLQP=1, SLSP=0, NYQP=0, CCHU=1, CNC chooses inner chuck mode, and chuck in-position signal detecting is active:

DOQPS: chuck releasing output; WQPJ: releasing in-position signal;

DOQPJ: chuck clamping output; NQPJ: clamping in-position signal.

DOQPJ and DOQPS output high resistance at power on, when CNC detects that the chuck input signal DIQP is active for the 1st time, DOQPJ is connected to 0V and chuck is clamped.

After M12 is executed, DOQPS outputs high resistance, DOQPJ outputs 0V, chuck is clamped and CNC waits for NQPJ signal to be in-position.

After M13 is executed, DOQPJ outputs high resistance, DOQPS outputs 0V, chuck is released and CNC waits for WQPJ signal to be in-position.

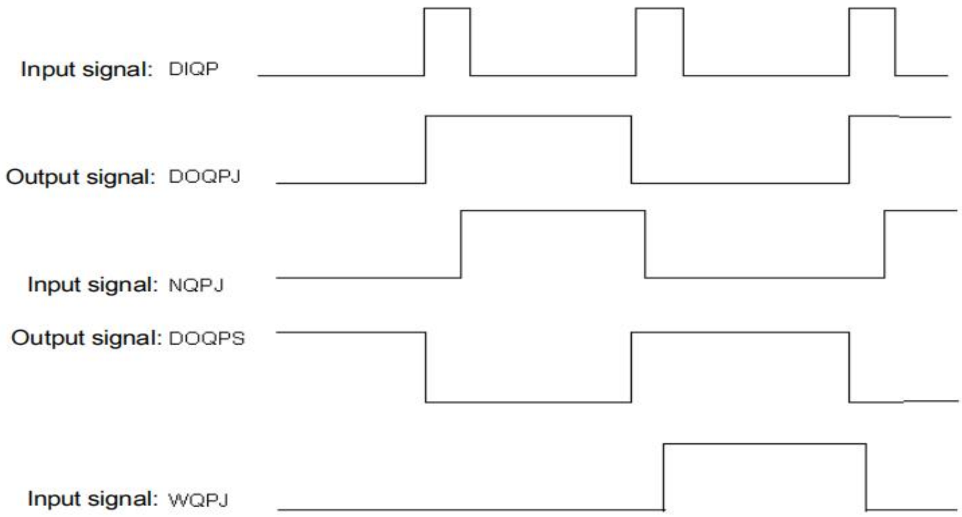


Fig. 2-53 (Chuck clamping, releasing signals are level output)

②When SLQP=1, SLSP=0, NYQP=1, CCHU=1, CNC chooses outer chuck mode, and chuck in-position signal detecting is active:

DOQPS: chuck clamping output; WQPJ: clamping in-position signal;

DOQPJ: chuck releasing output; NQPJ: releasing in-position signal.

DOQPJ and DOQPS output high resistance at power on, when CNC detects that the chuck input signal DIQP is active for the 1st time, DOQPS is connected to 0V and chuck is clamped.

After M12 is executed, DOQPS outputs 0V, DOQPJ outputs high

resistance, chuck is clamped and CNC waits for WQPJ signal to be in-position.
After M13 is executed, DOQPJ outputs 0V, DOQPS outputs high
resistance, chuck is released and CNC waits for NQPJ signal to be in-position.

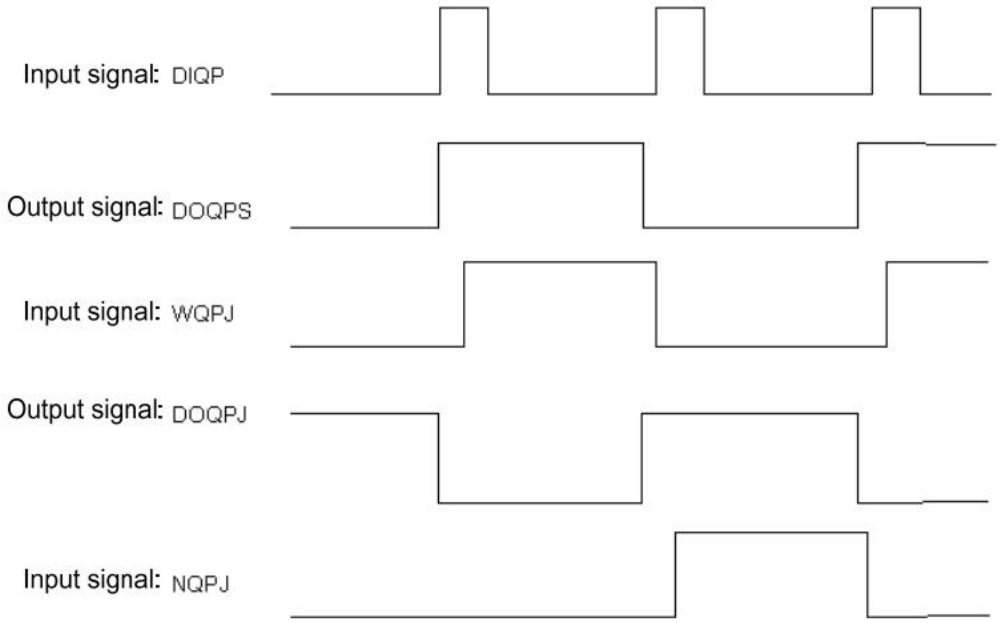


Fig. 2-54 Chuck clamping, releasing signals are level output

As the 2nd chuck input is active, DOQPS outputs 0V, chuck is released. The chuck clamping/releasing signal is output alternatively, i.e. the output is changed each chuck input signal is active.

③The interlock between the chuck and the spindle

When SLQP=1, SLSP=0, M3 or M4 is active, the alarm is issued if M13 is executed and the output is unchanged.

When SLQP=1, SLSP=0, CCHU=1, if M12 is executed in MDI or Auto mode, CNC does not execute next code till it detects the chuck clamping in-position signal is active. When the chuck input signal DIQP is active in Manual mode, the panel spindle CW, CCW key are inactive till it detects the chuck clamping in-position signal is active. In spindle running or auto cycle processing, DIQP input signal is inactive. And DOQPS, DOQPJ is held on at CNC reset and emergency stop.

4.1.8 Tail seat control

- **Address Definition**

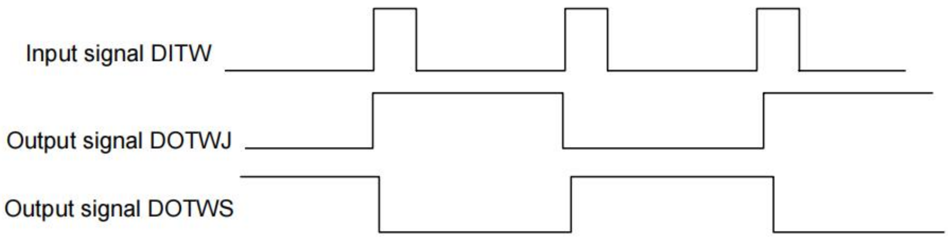
CN62 output pin number	PLC address	function	describe
2	Y0.2	DOTWJ	Tail seat input/output signal(M10)
16	Y0.5	DOTWS	Tail seat retreat output signal(M11)

- **Control parameters**

Parameter – [Chuck Tail Seat]

008	Tail Seat Control (0: Valid 1: Invalid) K13.0	Invalid
009	Spindle rotation and tailstock advance and retreat (0: interlock 1: non interlock) K13.1	interlock

➢ **Action sequence diagram**



Tailstock advancing (DOTWJ) and retracting(DOTWS) are both inactive when power on; when the tailstock input (DITW) is active for the 1st time, tailstock advancing is active; when it is active for the 2nd time, tailstock retracting is active, so the DOTWJ/ DOTWS signal interlock is output alternatively, i.e. The output changes each time the DITW signal is active. If M10 is executed, DOTWJ outputs 0V and tailstock advances; if M11 is executed, DOTWS outputs 0V and tailstock retracts.

DITW signal is inactive as spindle is running. If M11 is executed, alarm will be issued, and its output are held on. And DOTWS, DOTWJ outputs are held on at CNC reset or emergency stop

4.1.9 Lubrication control

➢ Parameter - [Quick Debugging]

19	When automatic lubrication is effective, start up and output lubrication (0: No 1: Yes)	NO
20	Lubrication start output time (in milliseconds) T014	0
21	Automatic lubrication interval time (in seconds) T015	7200

Lubrication start output time (unlimited lubrication time)

➢ **Functional Description**

There are two types of lubrication functions, non automatic lubrication and automatic lubrication. When T14=0 or T15=0, the automatic lubrication function is invalid.

a) Non automatic lubrication

When T14>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 T15, 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 T14=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 T14>0, When T15>0, the system starts timing the time set by T15 after power on, and then lubricates the output. After the time set by T14, 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.

4.2.0 Tri color lamp control

➢ Address Definition

CN62 output pin number	PLC address	function	describe
7	Y1.2		Tri color light - green light, operating status
8	Y1.3		Tri color light - red light, alarm status

➢ **Control parameters**

Parameters - [PLC Parameters]

12	Three color light function output signal (0: invalid 1: valid)	invalid
25	Three color light function yellow light output signal Y1.1 (0: invalid 1: valid)	invalid

4.2.1 Tool change control

The standard ladder diagram supports four types of knife holder control logic; Select which tool holder to adapt to by setting the control position corresponding to the parameters of the tool holder ladder diagram in parameter - [tool holder parameters] 1.

Parameters - [Tool holder parameters]

1	Knife holder type (0: Row 1: Four station 2: AK31 3:	1

	Liuxin Knife Tower	

Parameter 1=0, no electric tool holder used, fixed blade holder used

Parameter 1=1, standard tool changing method (select tool changing method A or tool changing method B based on the parameter)

Parameter 1=2: Suitable for Yantai knife holder AK31 series (8, 10, 12 stations).

Parameter 1=3: Suitable for Taiwan Liuxin hydraulic tool holder

Parameter 1=4: Suitable for Deou servo tool holder (8, 10, 12 stations).

Note: To modify this parameter, the system needs to be powered off and restarted

- Parameter 1=1, standard tool changing method (select tool changing method A or tool changing method B based on the parameter)

Control parameters

001	Tool holder type selection (0: Row 1: Four station 2: AK31 3: Liuxin Knife Tower	Four station
002	Number of cutting tools	4
003	Knife holder position signal (0: high level 1: low level) K11.2	low level
004	Is the tool position signal checked at the end of the tool change (0: not checked 1: checked) K11.5	not checked
005	Tool holder reverse locking time (in milliseconds) 1 second equals 1000 milliseconds T009	1000
006	The maximum tool change time (in milliseconds) during tool change is 1 second, which is equal to 1000 milliseconds T004	15000
007	Delay time (in milliseconds) from the stop of forward rotation of the knife holder to the start of reverse rotation T007	0
008	Alarm time (in milliseconds) T008 for not receiving tool holder locking * TCP signal	5000
010	Is the locking signal of the knife holder detected (0: not detected 1: detected) K11.3	not detected
011	Tool holder locking signal level selection (0: low level 1: high level) K11.4	low level
012	Selection of workstation tool changing method (0: Method B 1: Method A) K11.0	Method B

Standard tool holder changing method A

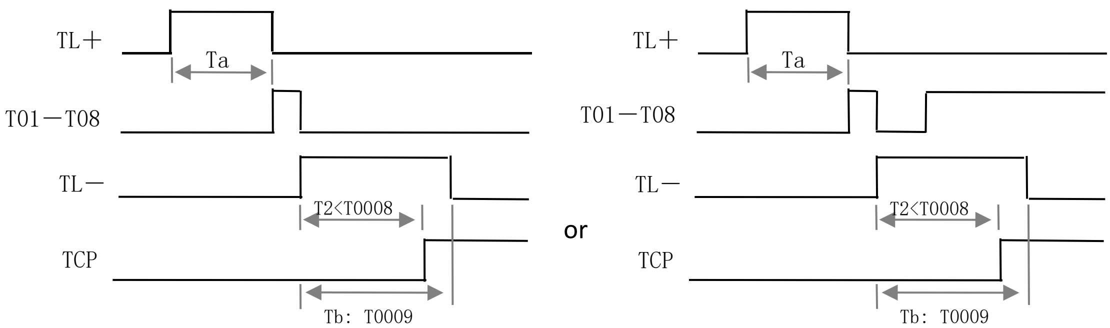
Knife changing process

① In manual, input, or automatic mode, tool replacement is performed, and the CNC outputs a forward rotation signal (TL+) for the tool holder and begins detecting the tool position signal. After detecting the tool position signal, turn off the forward rotation signal (TL+) of the tool holder and start detecting whether the tool position signal jumps. If a jump occurs, output a tool holder reversal signal (TL -). After delaying the time set by PLC parameter T009, turn off the tool holder reverse signal (TL -).

② If K0011.4 is set to 1 (detecting lock signal), the system will start detecting the lock signal of the tool holder. If the system does not receive a TCP signal within the time set by PLC parameter T008, the system will issue an alarm.

③ If K0011.5 is set to 1 (end of tool change check tool position signal), after the tool holder reversal time ends, confirm whether the current tool position input signal is consistent with the current tool number. If they are inconsistent, the system will issue an alert.

④ The process of changing the knife has ended.



Timing diagram of tool change method A

Standard tool holder changing method B

1）Knife changing process

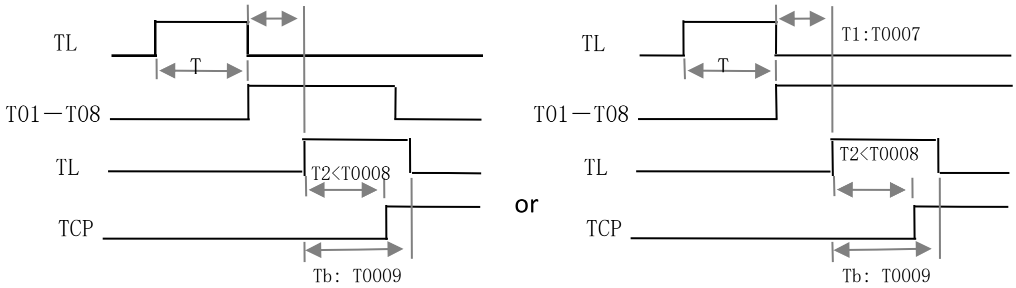
① After performing the tool change operation, the system outputs the tool holder forward rotation signal TL+and starts detecting the tool position signal. After detecting the tool position signal, the TL+output is turned off. After delaying the time set by PLC parameter T007, the tool holder reverse signal TL - is output. After delaying the time set by PLC parameter T009, the tool holder reverse signal (TL -) is turned off.

② If K0011.4 is set to 1 (detecting locking signal), the system will start detecting the locking signal of the tool holder. If the system does not receive the TCP signal within the time set by PLC parameter T008, the system will generate an alarm.

③ If K0011.5 is set to 1 (end of tool change check tool position signal), after the tool holder reversal time ends, confirm whether the current tool position input signal is consistent with the current tool number. If they are not consistent, the system will generate an alarm.

④ The process of changing knives has ended.

T1:T0007



Timing diagram of tool change

4.2.2 User M code control

➢ **Control parameters**

Parameters - [PLC Parameters]

013	User M Function (M80/M81/M90) (0: Invalid 1: Valid)	Invalid
014	M90 timeout detection (0: valid 1: invalid)	valid
015	M90 Px user instruction delay alarm time (milliseconds)	12000

➢ **Instruction Description**

① M80 P1 controls Y0.0 output
M81 P1 turns off Y0.0 output
② M80 P2 controls Y0.1 output
M81 P2 closes Y0.1 output
③ M80 P3 controls Y0.2 output
M81 P3 turns off Y0.2 output
④ M80 P4 controls Y0.3 output
M81 P4 turns off Y0.3 output
⑤ M80 P5 controls Y0.4 output
M81 P5 turns off Y0.4 output
⑥ M80 P6 controls Y0.5 output
M81 P6 turns off Y0.5 output
⑦ M80 P7 controls Y0.6 output
M81 P7 turns off Y0.6 output
⑧ M80 P8 controls Y0.7 output
M81 P8 turns off Y0.7 output
⑨ M80 P9 controls Y1.0 output
M81 P9 turns off Y1.0 output
⑩ M80 P10 controls Y1.1 output
M81 P10 turns off Y1.1 output
⑪ M80 P11 controls Y1.2 output
M81 P11 closes Y1.2 output
⑫ M80 P12 controls Y1.3 output
M81 P12 turns off Y1.3 output
⑬ M80 P13 controls Y1.4 output
M81 P13 turns off Y1.4 output
⑭ M80 P14 controls Y1.5 output

M81 P14 turns off Y1.5 output

⑮ M80 P15 controls Y1.6 output

M81 P15 turns off Y1.6 output

⑯ M80 P16 controls Y1.7 output

M81 P16 turns off Y1.7 output

M90 P1 (X0.0 in place detection)

M90 P2 (X0.1 in place detection)

M90 P3 (X0.2 in place detection)

M90 P4 (X0.3 in place detection)

M90 P5 (X0.4 in place detection)

M90 P6 (X0.5 in place detection)

M90 P7 (X0.6 in place detection)

M90 P8 (X0.7 in place detection)

M90 P9 (X1.0 in place detection)

M90 P10 (X1.1 in place detection)

M90 P11 (X1.2 in place detection)

M90 P12 (X1.3 in place detection)

M90 P13 (X1.4 in place detection)

M90 P14 (X1.5 in place detection)

M90 P15 (X1.6 in place detection)

M90 P16 (X1.7 in place detection)

- **instructions**

Diagnostic Interface - [User M Code]

In the handwheel mode, press the corresponding 1-9 numeric keys to output the corresponding Y signal for the M code, and press again to cancel the Y signal output.

example:G00 X50 Z5

M80 P1 //Output Y0.0 signal, such as controlling a forward cylinder

M90 P3 //Waiting for X0.2 to detect the signal that the forward cylinder is in place

M81 P1 //Turn off the Y0.0 signal output

M80 P2 //Output Y0.1 signal, cylinder retreats

M03 S500

G00 X30 Z0

4.2.3 M100、M101—User output and detection

- M100 and M101 are instructions for users to flexibly control output and detection.
- Instruction field I is used to specify the detection input interface Xn. m and its voltage level;
- The instruction field Q is used to specify the output signal Yn.m and its level;**For example, Q0.0 corresponds to Y0.0=0, and Q0.1 corresponds to Y0.0=1**
- The instruction field R is used to specify the detection output signal Yn.m and its level;
- The integer part of the I value corresponding to input interface Xn. m is defined as I (n * 8+m), for example, the I value corresponding to X3.1 is I25;
- The integer part of the Q/R value corresponding to the output interface Yn.m is defined as Q (n * 8+m)/R (n * 8+m), for example, the Q value corresponding to Y1.7 is Q15; Reverse and check the input interface Yn.m corresponding to Q15: N=(15/8) rounded to 1, m=15-8*n=7, Q15 corresponds to Y1.7
- The value of. 0 after the decimal point in instruction fields I, Q, and R indicates a low level (can be omitted,

the system defaults to. 0) 1 represents high level.

Y signal corresponding to Q value							
Q0	Q1	Q2	Q3	Q4	Q5	Q6	Q7
Y0.0	Y0.1	Y0.2	Y0.3	Y0.4	Y0.5	Y0.6	Y0.7
Q8	Q9	Q10	Q11	Q12	Q13	Q14	Q15
Y1.0	Y1.1	Y1.2	Y1.3	Y1.4	Y1.5	Y1.6	Y1.7
The X signal corresponding to the I value							
I0	I1	I2	I3	I4	I5	I6	I7
X0.0	X0.1	X0.2	X0.3	X0.4	X0.5	X0.6	X0.7
I8	I9	I10	I11	I12	I13	I14	I15
X1.0	X1.1	X1.2	X1.3	X1.4	X1.5	X1.6	X1.7

M100- Post output detection

【instruction format】

M100 Q_ [D_] [I_/R_] [P_/E_]

Instruction function: Output a specified level signal at the Q value specified interface, hold for D time, and cancel the Q level output. During this process, if the interface specified by the I/R value is detected as the specified input level, the Q level output will be cancelled, and the instruction execution will be completed. If the signal holding time D is delayed and the I specified input level is not detected, the instruction will continue to wait

Q_: Specify the output interface and output level, such as Q15.0 indicating Y1.7 output low level;

D_: Output signal holding time (unit: seconds), default D or D0, the output signal will not be canceled after the instruction is executed;

I_: Detect the input interface level, such as I25.1 indicating waiting to detect X3.1 input as high level

R_: Detect the output interface level, such as R10.0 indicating whether Y1.2 is at a low level I and R cannot be detected simultaneously

If no valid signal is detected after time D, the program will jump to the line number specified by the P value for execution P is invalid at D0

If no valid signal is detected after time D, the alarm number specified by E will be triggered, and E will be invalid at D0

For example: M100 Q0.1 // Y0.0 Output High Level

M100 Q0.0 // Y0.0 Turn off output to low level

M100 Q15.0 I20.0 D1//Y1.7 outputs low level for 1 second and then outputs high level.

If X1.4 is detected as low level, the instruction is completed

M100Q0.1Q2.1Q3.1 can simultaneously output three signals

M100Q0.0Q2.0Q3.0 can simultaneously turn off three signals

M100 and T or G instructions are on the same line

M100Q1.1G00X100Z100

M100Q10.1T0101

M100Q10.0G01X10Z10F100

M100 currently does not support other M codes on the same line

M101—Output after detection

【instruction format】

M101 I_/R_ [D_] [Q_] [P_/E_]

Command function:

After detecting that the I/R value specified interface is at the specified level signal within time D, the specified level is output based on the Q value

When D0 or D is default, the P/E field is invalid;

When there is no P/E field, if no I/R level is detected within time D, the next instruction is executed

When there is a P field, if no I/R level is detected within D time, it will jump to the P program line for execution

When there is an E field, if no I/R level is detected within D time, the alarm number specified by E will be triggered

When M101 I2/R_ [D_] [P_] detects that the I/R value specified interface is at the specified level signal within time D, it will jump to the program line specified by P for execution. Otherwise, the next instruction will be executed

Q_: Specify the output interface and output level, such as Q15.0 indicating Y1.7 output low level

D_: Signal detection time, default or continuous detection at D0

I_: Specify the detection input interface and level, such as I25.1 indicating waiting to detect X3.1 input as high level

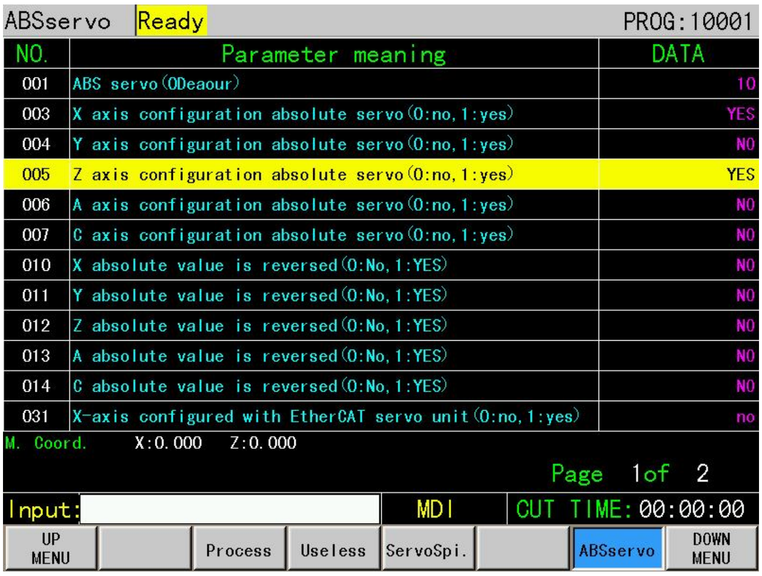
P_: Jump to the program line specified by the P value after timeout detection to continue execution

For example: M101 I25.0 Q15.1 D5 P1000//If the X3.1 signal is detected as low level within 5 seconds, Y1.7 outputs a high level, otherwise it jumps to N1000 line for execution

4.2.4 Bus servo debugging

- **Control parameters**

Parameters - [Bus Parameters] -485 Communication



- Driver setting requirements

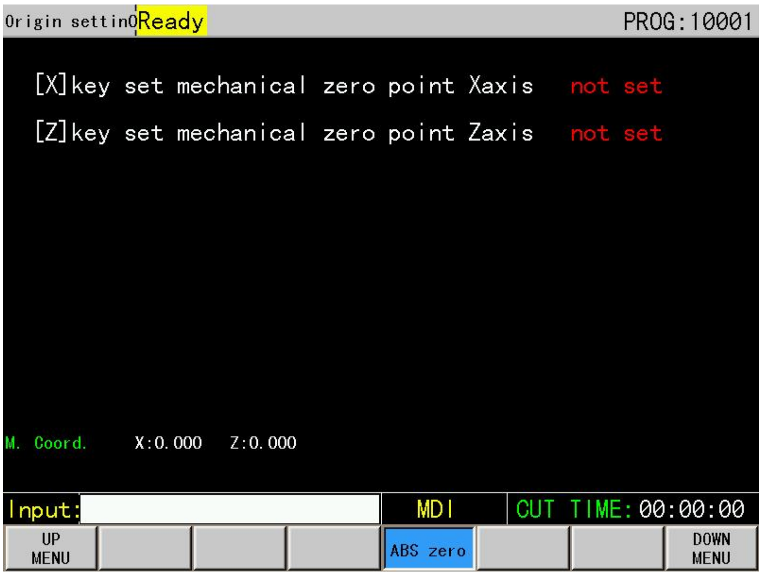
Axis name	Address Station Number	**Baud rate**	Parity check	**describe**
X	1	9600	Ineffective verification	The station number parameter of the driver connected to the X-axis must be

				set to: 1
Z	2	9600	Ineffective verification	The station number parameter of the driver connected to the Z-axis must be set to: 2
Y	3	9600	Ineffective verification	The station number parameter of the driver connected to the Y-axis must be set to: 3

Note: 485 communication requires the manufacturer to specify the driver model in order to be compatible

➢ Absolute mechanical zero point setting

Switch to parameter - [Absolute value zero] as shown in the following figure:



Note: After setting the zero point for the first time, you must first power off and restart.

➢ Absolute value setting success and verification

① After the zero point is successfully set, the system will power off first.

② In manual mode, move the X-axis and move the machine coordinate to around+28.000, and the system will power off again.

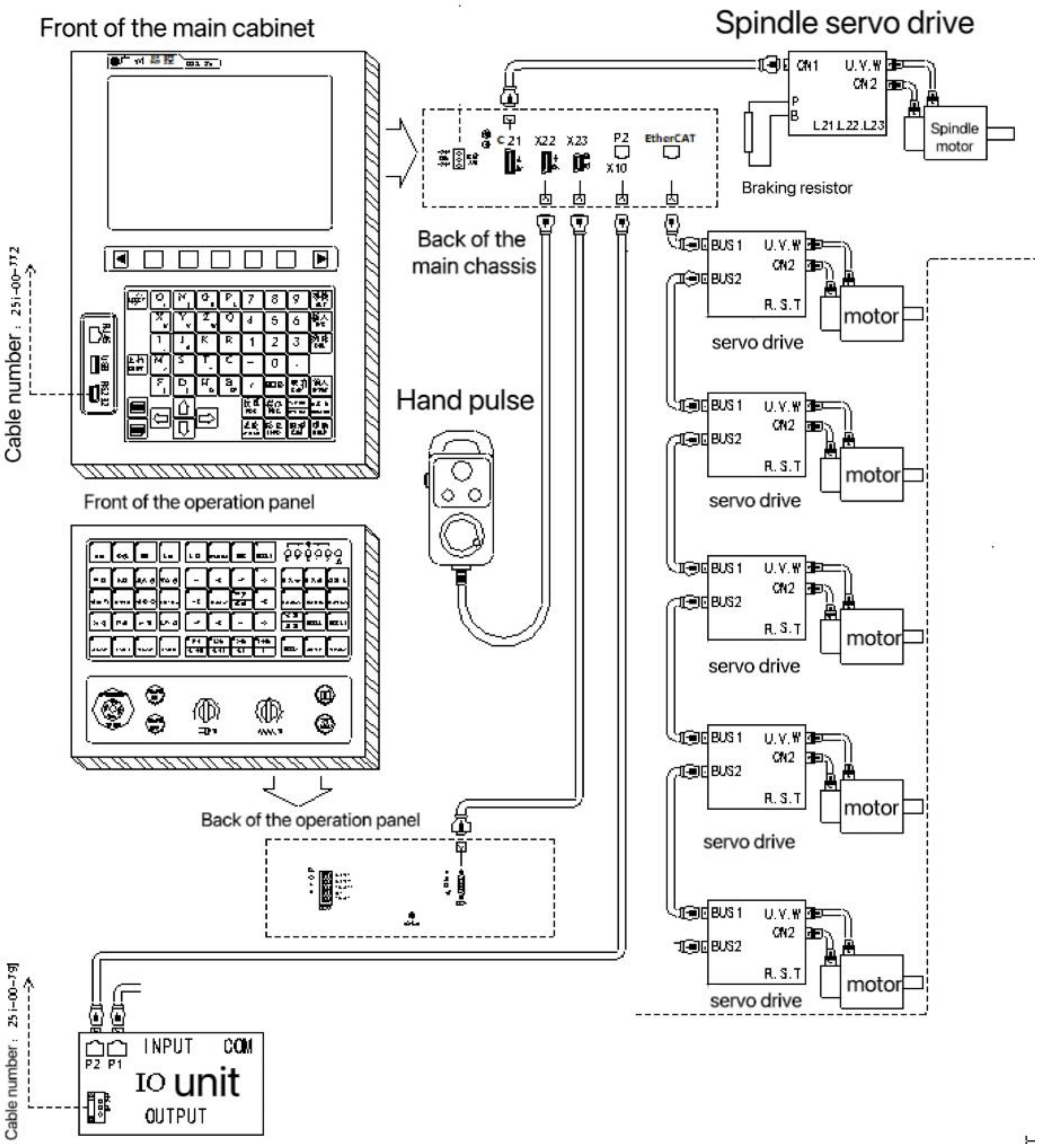
③ After powering on the system, check the coordinate numbers of the machine tool to ensure that the values are around+28.000 before the power outage. If the coordinates change to around -28.000, it is necessary to modify the parameter - [Absolute Value Parameter]. Parameter 6, X-axis communication, obtains the direction of the absolute value in reverse, select reverse. If there are numerical values in the Z-axis coordinate, the station number of the drive needs to be modified.

④ Follow the above steps for other axes.

Note: The number of changes in the system's electrical coordinates is within the normal range.

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.

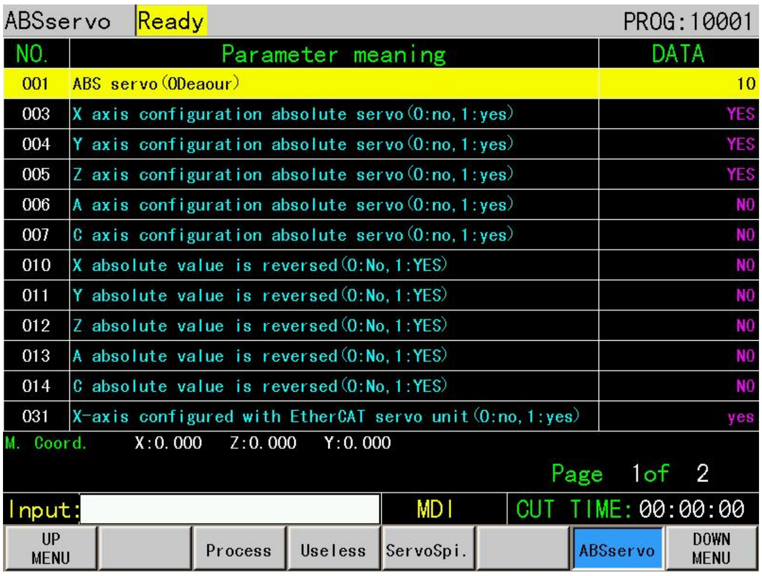
Milling machine system:

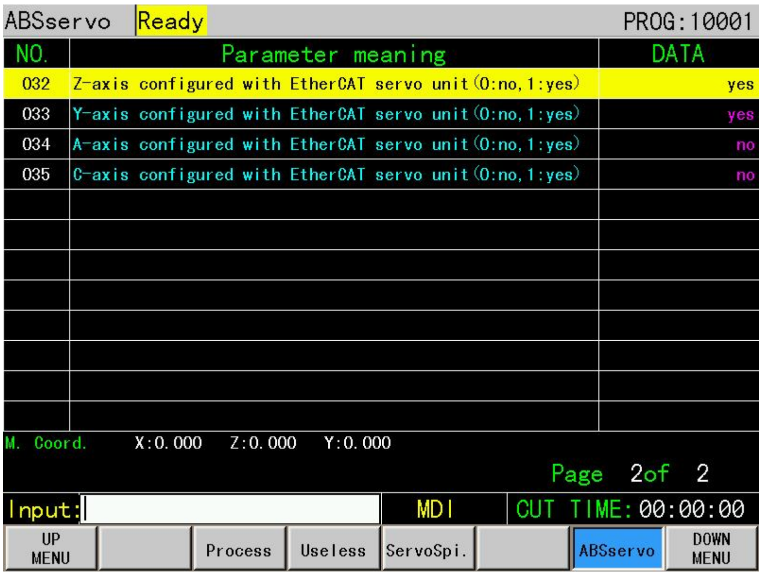
System connection driver timing: The system network cable is connected to the X-axis driver, the X-axis driver is connected to the Y-axis driver, the Y-axis driver is connected to the Z-axis driver, the Z-axis driver is connected to the A-axis driver, and the A-axis driver is connected to the B-axis 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 - [Bus 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

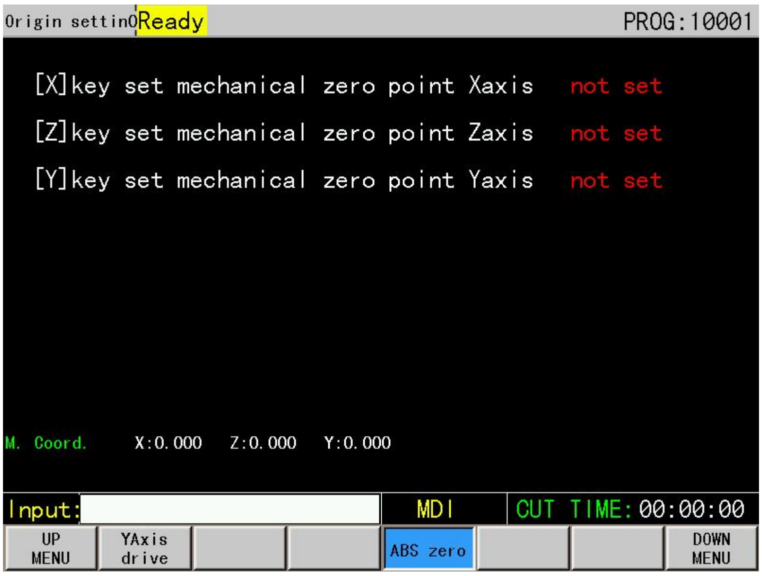
rotation axis

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**

4.2.5 Automatic feeding M70, M71 functions

➢ **Control parameters**

Parameters - [PLC Parameters]

050	Automatic feeding function (0: invalid 1: effective) K30.7	invalid
051	Feed forward time (in milliseconds) T28	2000
052	Return time (in milliseconds) T25	1000
053	Number of feeding and returning cycles C15	3
054	Delay completion time for material detection in place (unit: milliseconds) T27	50
055	Whether the automatic feeding has been detected properly and the signal timeout alarm (0: detect 1: not detect) K30.6	detect
056	System alarm or reset to cancel the forward feeding signal (0: not cancelled 1: cancelled) K30.5	not cancelled

➢ Address Definition

Y0.7 is the forward signal output for material feeding

X0.1 is the signal for the advance of material feeding

➢ **Function Description**

① Execute the M70 instruction, output the material advance signal Y0.7, and output if X0.1 (material arrival signal) is not detected after the T28 set time.

② Turn off the output Y0.7, execute the backoff time T25, and when the backoff time is reached, output Y0.7 again.

③ If X0.1 is detected to be in place, M70 is complete. If X0.1 is still not detected, cycle according to the number of times C15 is set for feeding until the feeding frequency is reached and an alarm is triggered: "The feeder has no material or there is a problem with the feeding in place signal.

④ M71 is to turn off the feeding forward signal.

4.2.6 Pulse spindle selection parameters (this function is only available for 5-axis systems)

It can achieve dual pulse spindle or dual analog spindle, choose freely.

- The first spindle pulse is fixed on the fifth axis port and needs to be wired according to the instructions.

For example, the system needs to display the X-axis, Z-axis, and C-axis

Parameters that need to be modified:

Feed axis parameters	**Parameter Description**	**price**
P100	Maximum number of axes in CNC	5
P060	Prohibit Y-axis display	YES
P061	Disable 4th axis display	YES

- C-axis gear ratio setting

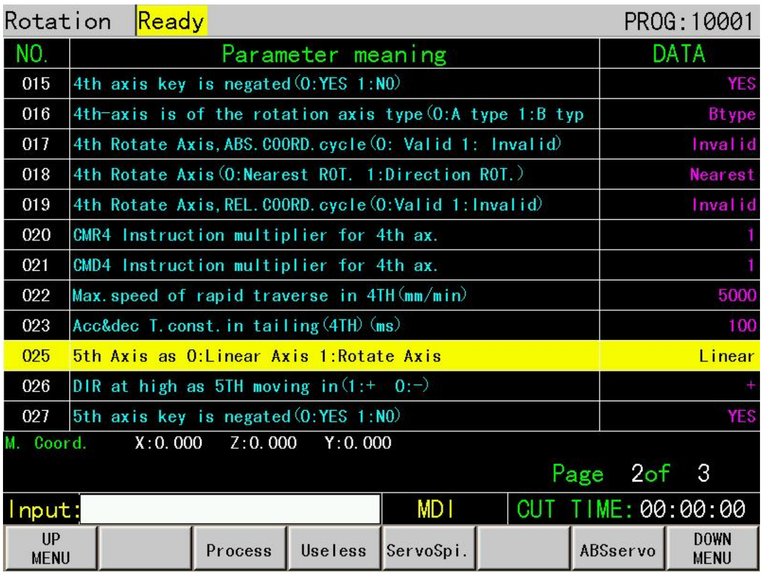
C-axis spindle servo motor:

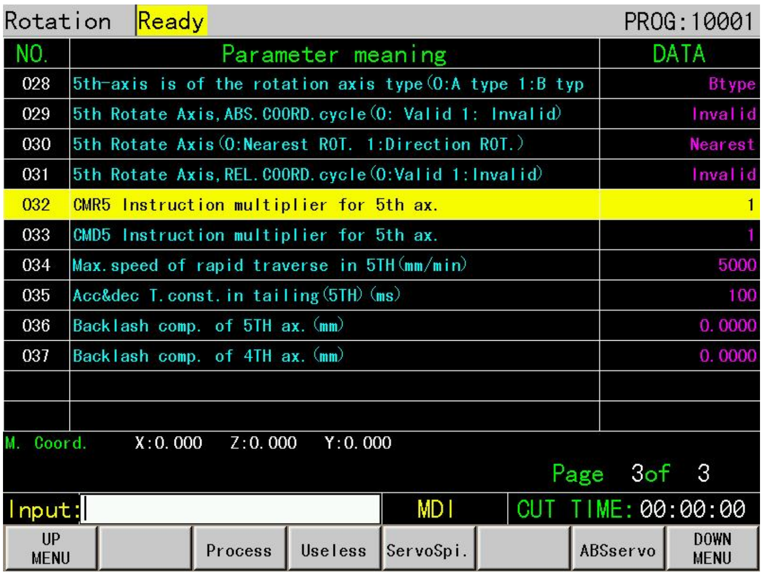
1024 line encoder gear ratio 1024X4/1000X360=64:5625

2500 line encoder gear ratio 2500X4: 1000X360=1:36

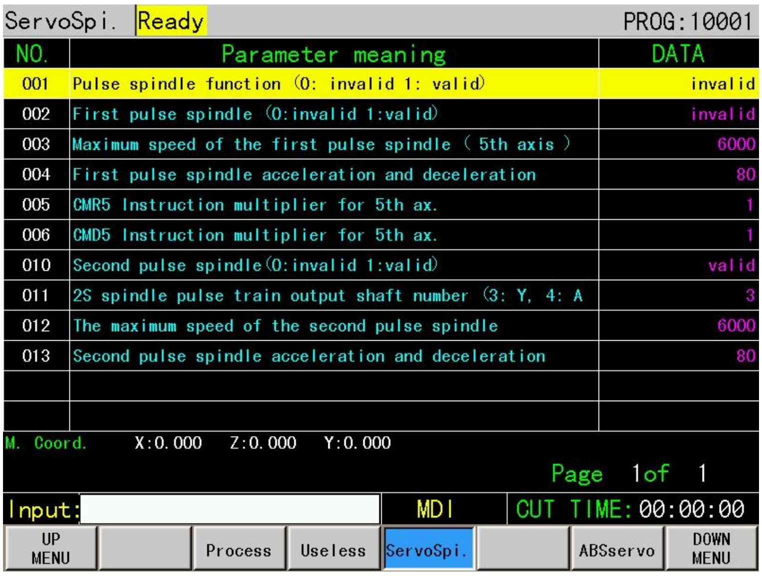
The following image is set according to the 2500 line encoder

Rotation axis parameters





Servo spindle parameters

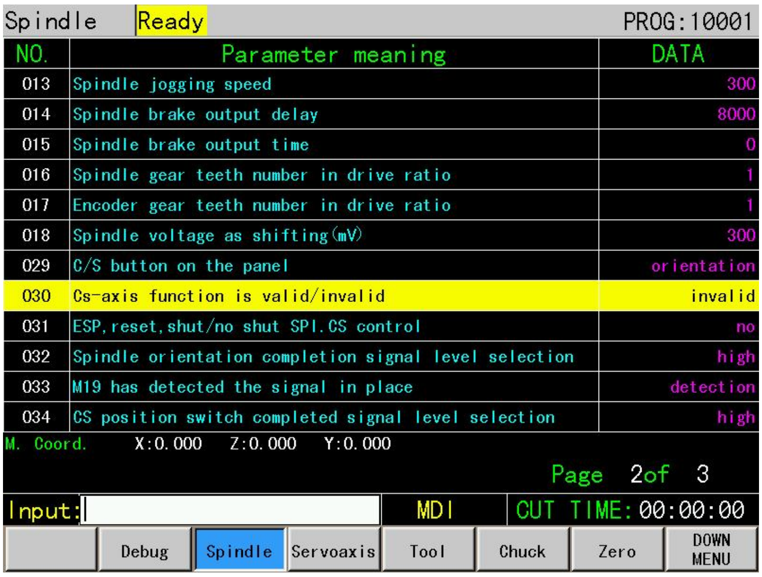


ServoSpi. Ready PROG:10001

NO.	Parameter meaning	DATA
001	Pulse spindle function (0: invalid 1: valid)	invalid
002	First pulse spindle (0:invalid 1:valid)	invalid
003	Maximum speed of the first pulse spindle (5th axis)	6000
004	First pulse spindle acceleration and deceleration	80
005	CMR5 Instruction multiplier for 5th ax.	1
006	CMD5 Instruction multiplier for 5th ax.	1
010	Second pulse spindle(0:invalid 1:valid)	valid
011	2S spindle pulse train output shaft number (3: Y, 4: A	3
012	The maximum speed of the second pulse spindle	6000
013	Second pulse spindle acceleration and deceleration	80

Change parameter 001 to valid and parameter 002 to valid

Parameter 030 that needs to be modified for spindle parameters



Spindle Ready PROG:10001

NO.	Parameter meaning	DATA
013	Spindle jogging speed	300
014	Spindle brake output delay	8000
015	Spindle brake output time	0
016	Spindle gear teeth number in drive ratio	1
017	Encoder gear teeth number in drive ratio	1
018	Spindle voltage as shifting(mV)	300
029	C/S button on the panel	orientation
030	Cs-axis function is valid/invalid	invalid
031	ESP,reset,shut/no shut SPI.CS control	no
032	Spindle orientation completion signal level selection	high
033	M19 has detected the signal in place	detection
034	CS position switch completed signal level selection	high

Programming:

M3 S600//Start the first axis pulse spindle

T0101//Knife number

G98 G00 X30 Z5//Quick positioning

G01 Z-20 F150//Machining and Cutting

G00 X50 Z10//Quick tool return

M05//Stop spindle

M14//spindle switched to C-axis

G50 C0//C-axis absolute coordinates reset to zero

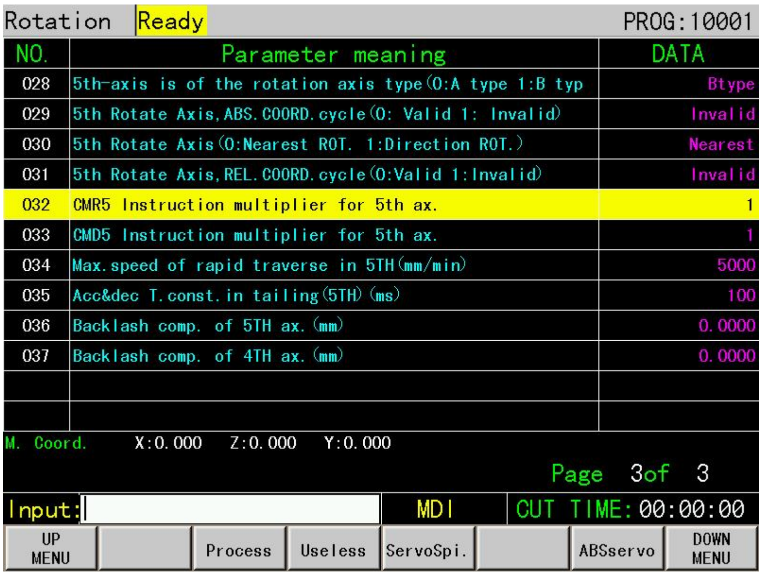
G00 C90//C axis moved to 90 degrees

M15//Cancel spindle switch to C-axisM30

Attention: To switch the M14 spindle to the C-axis, if you need to perform spindle rotation M3 S300, you must execute the M15 command to switch to pulse spindle control.

Second pulse spindle setting

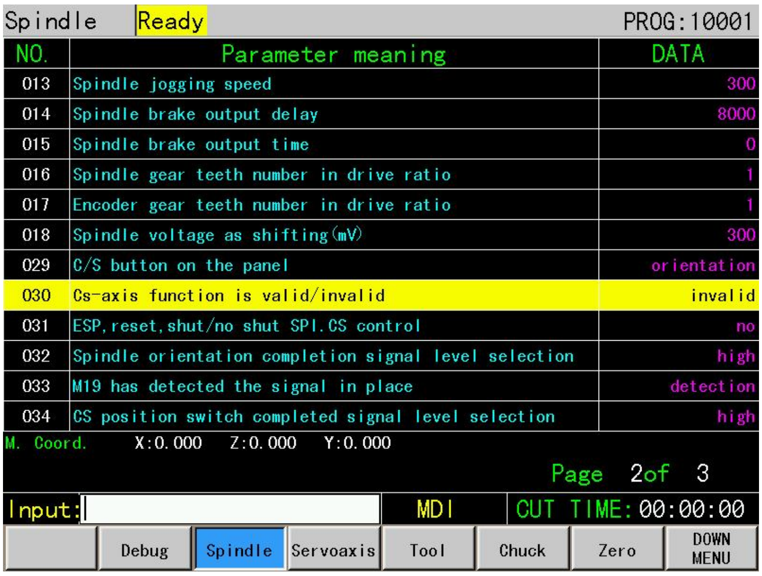
- Servo spindle parameters


Rotation Ready PROG:10001

NO.	Parameter meaning	DATA
028	5th-axis is of the rotation axis type(0:A type 1:B typ	Btype
029	5th Rotate Axis,ABS.COORD.cycle(0: Valid 1: Invalid)	Invalid
030	5th Rotate Axis(0:Nearest ROT. 1:Direction ROT.)	Nearest
031	5th Rotate Axis,REL.COORD.cycle(0:Valid 1:Invalid)	Invalid
032	CMR5 Instruction multiplier for 5th ax.	1
033	CMD5 Instruction multiplier for 5th ax.	1
034	Max.speed of rapid traverse in 5TH(mm/min)	5000
035	Acc&dec T.const.in tailing(5TH)(ms)	100
036	Backlash comp. of 5TH ax.(mm)	0.0000
037	Backlash comp. of 4TH ax.(mm)	0.0000

Change the second pulse spindle to effective

- Rotation axis parameters


Spindle Ready PROG:10001

NO.	Parameter meaning	DATA
013	Spindle jogging speed	300
014	Spindle brake output delay	8000
015	Spindle brake output time	0
016	Spindle gear teeth number in drive ratio	1
017	Encoder gear teeth number in drive ratio	1
018	Spindle voltage as shifting(mV)	300
029	C/S button on the panel	orientation
030	Cs-axis function is valid/invalid	invalid
031	ESP,reset,shut/no shut SPI.CS control	no
032	Spindle orientation completion signal level selection	high
033	M19 has detected the signal in place	detection
034	CS position switch completed signal level selection	high

Change parameter 002 to rotation axis

Gear ratio setting:

The motor has 2500 wires

2500 line encoder gear ratio 2500X4: 1000X360=1:36

The motor has an absolute value of 17 digits

17 positions 131072:1000X360 (pitch)=131072:360000=2048: 5625

The motor has an absolute value of 23 digits

23 bits 8388608:1000X360 (pitch)=8388608:360000=131072:5625

Enter according to the actual situation

- **instruction**

M63 represents the forward rotation of the second spindle

M64 represents the second spindle reversal

M65 represents the second spindle stopping

Programming example:

```
M3 S600//Start the first axis pulse spindle
T0101//Knife number
G98 G00 X30 Z5//Quick positioning
G01 Z-20 F150//Machining and Cutting
G00 X50 Z10//Quick tool return
M05//Stop spindle
M14//spindle switched to C-axis
G50 C0//Absolute axis coordinates reset to zero
G00 C90//axis moved to 90 degrees
M63 S600//Start the second spindle rotation
G00 X60//X-axis positioning
G00 Z-20//Z-axis positioning
G87 X0 Q2 F200/Side Punching Cycle
C180//C-axis indexing 180 degrees
C270//C-axis division 270 degrees
G80//End fixed loop instruction
M65//Stop the second spindle
M30//Program End
```

Hangzhou Bergerda Automation Technology Co., LTD.

Address: Building No.8 Sitai Technology Park ,No.493 Linping Avenue, Yuhang District

Hangzhou, China

Sales hotline: 0571-88326782

Service hotline: 0571-89719501

Website:www.bergerda.com

I5T series Manual V1.21