Key-cutting Machine Control System

KY260A/KY360A/KY460A

Basic Information

This Manual is written by Adtech (Shenzhen) Technology Co., Ltd.

This Manual is mainly written by: Qu Chong

This Manual was first released on March 27, 2013 with version No. V1.1 and is applicable for control systems with item number BZ001B098A, BZ001B099A and BZ001B100A.

Copyright

Precautions and Explanations

※Transport and storage:

- ☞ Do not stack product package more than six layers;
- ☞ Do not climb, stand on or place heavy stuff on the product package;
- ☞ Do not pull the cable still connecting with machine to move product.
- ☞ Forbid impact and scratch on the panel and display;
- ☞ Prevent the product package from humidity, sun exposure, and rain.

※Open-box inspection:

- ☞ Open the package to confirm the product to be purchased by you.
- ☞ Check damages situation after transportation;
- ☞ Confirm the integrity of parts comparing with the parts list or damages situation;
- ☞ Contact our company promptly for discrepant models, shortage accessories, or transport damages.

※Wiring

- ☞ Ensure the persons involved into wiring and inspecting are specialized staff;
- ☞ Guarantee the product is grounded with less than 4Ω grounding resistance. Do not use neutral line (N) to substitute earth wire.
- ☞ Ensure grounding to be correct and solid, in order to avoid product failures or unexpected consequences;
- ☞ Connect the surge absorption diodes to the product in the required direction, otherwise, the product will be damaged;
- ☞ Ensure the power switch is OFF before inserting or removing plug, or disassembling chassis.

※Overhauling

- ☞ Ensure the power is OFF before overhauling or components replacement;
- ☞ Make sure to check failures after short circuit or overloading, and then restart the machine after troubleshooting
- ☞ Do not allow to frequently connect and disconnect the power, and at least one minute interval between power-on and power-off.

※Miscellaneous

- ☞ Do not open housing without permission;
- ☞ Keep power OFF if not in use for a long time;
- ☞ Pay close attention to keep dust and ferrous powder away from control;
- ☞ Fix freewheel diode on relay coil in parallel if non-solid state relay is used as output relay. Check whether power supply meets the requirement to ensure not burning the control.
- ☞ Install cooling fan if processing field is in high temperature, due to close relationship between service life of the control and environmental temperature. Keep proper operative temperature range for the control: 0℃ ~ 60℃.
- ☞ Avoid using the product in the overheating, humid, dusty, or corrosive environments;
- ☞ Add rubber rails as cushion on the place with strong vibration.

※Maintenance

Please implement routine inspection and regular check upon the following items, under the general usage conditions (i.e. environmental condition: daily average 30℃, load rate: 80%, and operating rate: 12 hours/ day)

Routine Inspection	Routine	● Confirm environmental temperature, humidity, dust, or foreign objects. ● Confirm abnormal vibration and noise; ● Check whether vents are blocked by yarn etc.
Regular Check	One year	● Check whether solid components are loose ● Confirm whether terminal block is damaged

Contents

Chapter 1 Overview

1.1 Product figures

KY260A, KY360A and KY460A have same appearance. KY460A appearance is shown below:

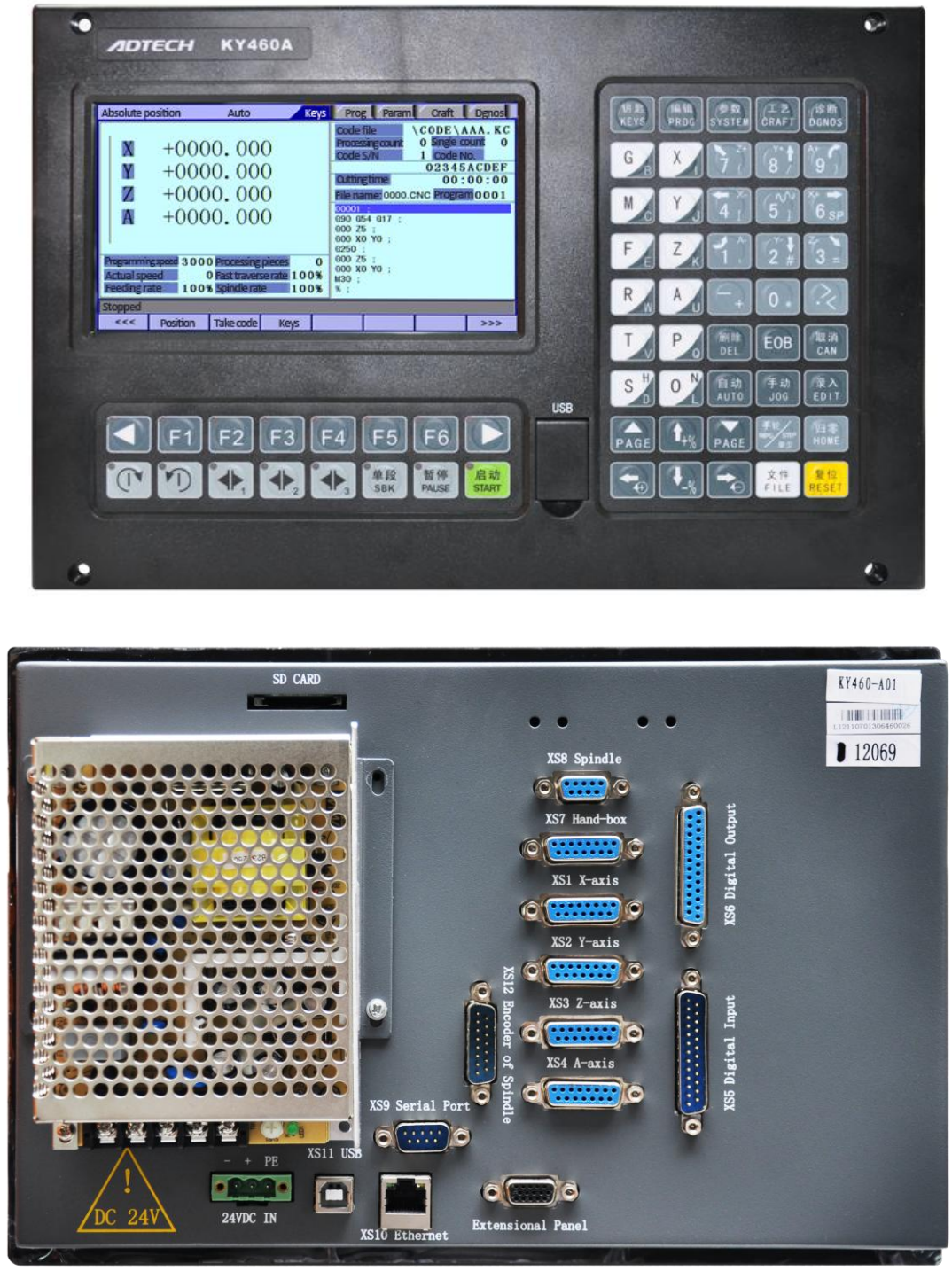


Fig. 1.1 Front and Rear View

1.2 Function overview

KY260A, KY360A and KY460A are 2-axis, 3-axis and 4-axis key control systems designed by Adtech for the key processing industry on the basis of milling machine. The system is mainly used for processing of keys, lock shells, lock cylinders and other related workpieces. It uses standard and industry G-code programming, and is widely used in a variety of CNC key machining equipment. Because the system is a standard milling system featuring low cost, fully functional and small size, so it has powerful function, high flexibility, broad applicability and extensibility in the same teeth milling machine system. You can select 2-axis, 3-axis and 4-axis system as needed.

1.2.1 Self-diagnosis

Diagnose CPU, memory, LCD, I/O interface, parameter state, coordinates and processing program comprehensively every time the system is started or reset; diagnose power supply, spindle, limit and I/O ports in real-time during operating.

1.2.2 Compensation

Automatic reverse clearance compensation

1.2.3 Abundant Instruction System

Program loop, program jump, program transfer, program calling, different end modes, macro definition and other program management instructions

Capable of processing a variety of mainstream keys, new keys can be easily customized

Fixed-point instructions: starting point, setting point, etc.

Linear interpolation, arc interpolation

Six workpiece coordinate systems, nine extension coordinate systems and one reference point

1.2.4 Full Chinese Menu Operation & Full Screen Edit

KYX60A CNC system uses cascading menu structure and full Chinese operation to ensure simple operation and visibility.

1.2.5 Abundant Error-correction Functions

Point out the nature and correct the errors in operation.

1.2.6 Program Exchange between CNC System and PC

Perform CAD/CAM/CAPP auxiliary programming with abundant software in PC, and then transmit CNC program to the system through communication interface (USB disk, RS232 interface), or transmit the programs from the system to PC.

1.3 Product configuration

- CPU: ARM industrial motherboard;
- Communication: USB interface;
- Capacity: 64MB RAM, 60MB Flash ROM;
- Feedback: AB phase pulse feedback;
- Control: FPGA motion controller;
- MPG: Incremental hand encoder;
- Display: 800×480 dot matrix 7 inches LCD;
- Input / output full optocoupler isolation;
- Touch, film operation panel;
- High noise immunity switching power supply;
- RS232 interface

1.4 Basic specifications

Climatic conditions		
Temperature	Operating	-5℃~50℃
	Storage temperature	-30℃~70℃
Relative humidity	Operating	20%~90% (℃)
	Storage and transportation	10%~95% (℃)
Atmospheric pressure		86Kpa~106Kpa

Chapter 2 Electrical Wiring

2.1 External interfaces

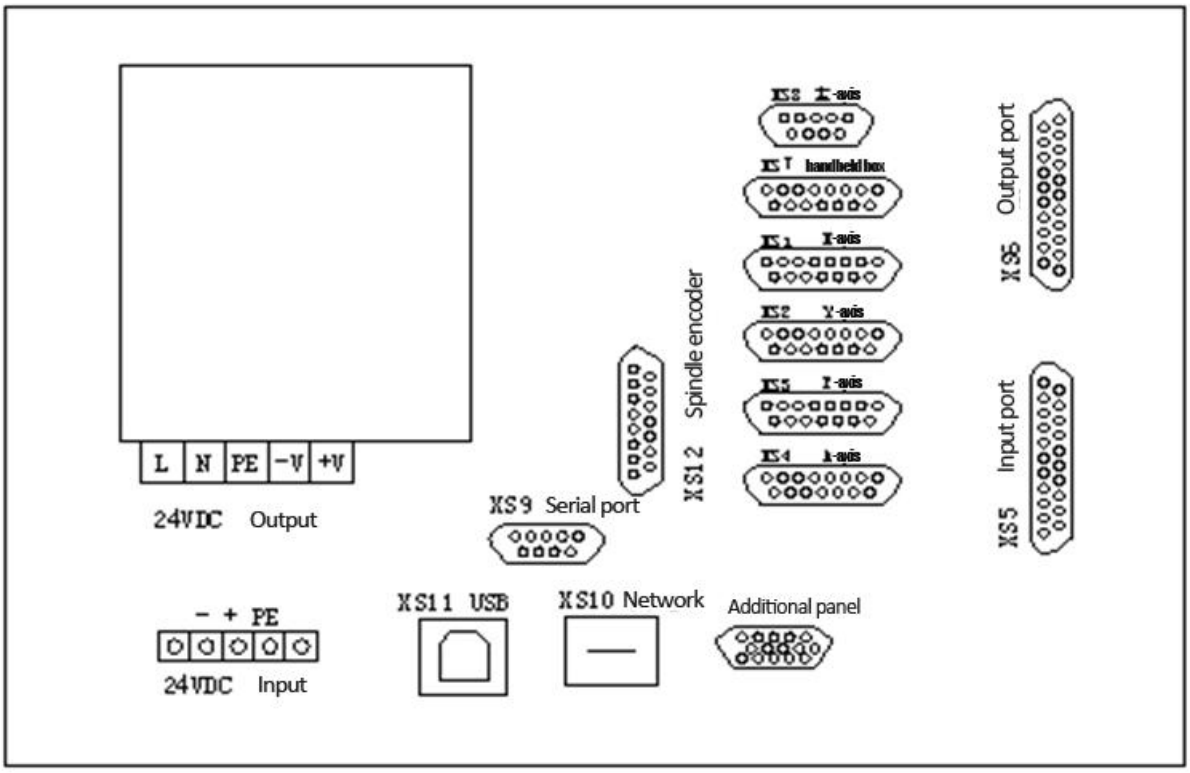


Fig. 2.1 Interfaces of KYX60A Controller

(1) XS1 ~ XS4:

15-core D-pin socket, XS1 indicates X-axis, XS2 indicates Y-axis, XS3 indicates Z-axis, XS4 indicates A-axis, connecting to stepper motor drive or digital AC servo drive

(2) XS5:

25-core D-pin socket inputs signals for every axis limit and other switching quantity

(3) XS6:

25-core D-pin socket, switch output signal interface

(4) XS7:

15-core D-pin socket, connecting to handheld box

(5) XS8:

9-core D-pin socket, connecting to main axis inverter

(6) XS9, XS11:

Serial port and USB are used to exchange files between KYX60A controller and computers and achieve other functions.

(5) KYX60A controller uses 24V DC power supply, and internal power consumption is about 5W.

(6) Additional panel isn't used by the key cutting machine temporarily

(7) Network interface is used to communicate

(8) Controller 24VDC power supply is only used to power the control system and can't be used for other loads

2.2 Overall connection diagram

2.2.1 Overall system wiring diagram

The wiring of every part of the system is shown below:

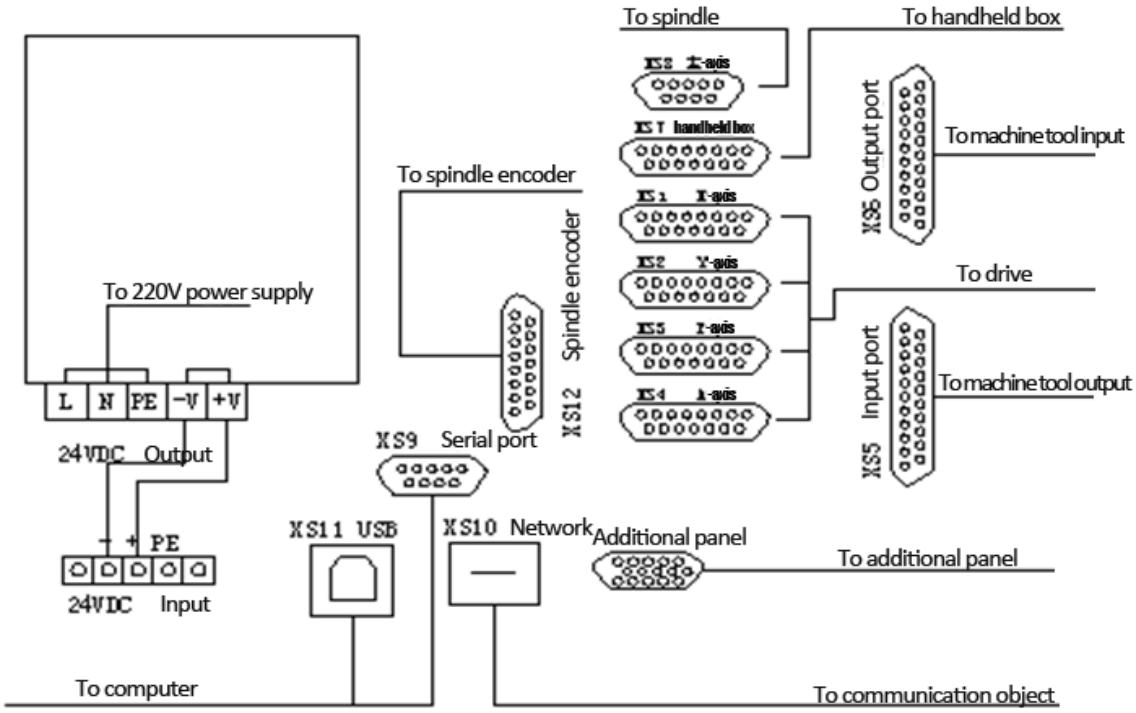
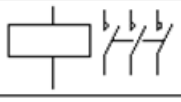
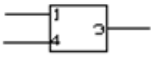
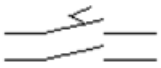
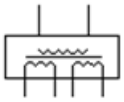
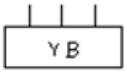
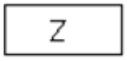
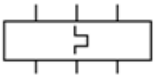
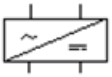
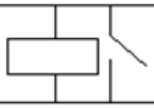
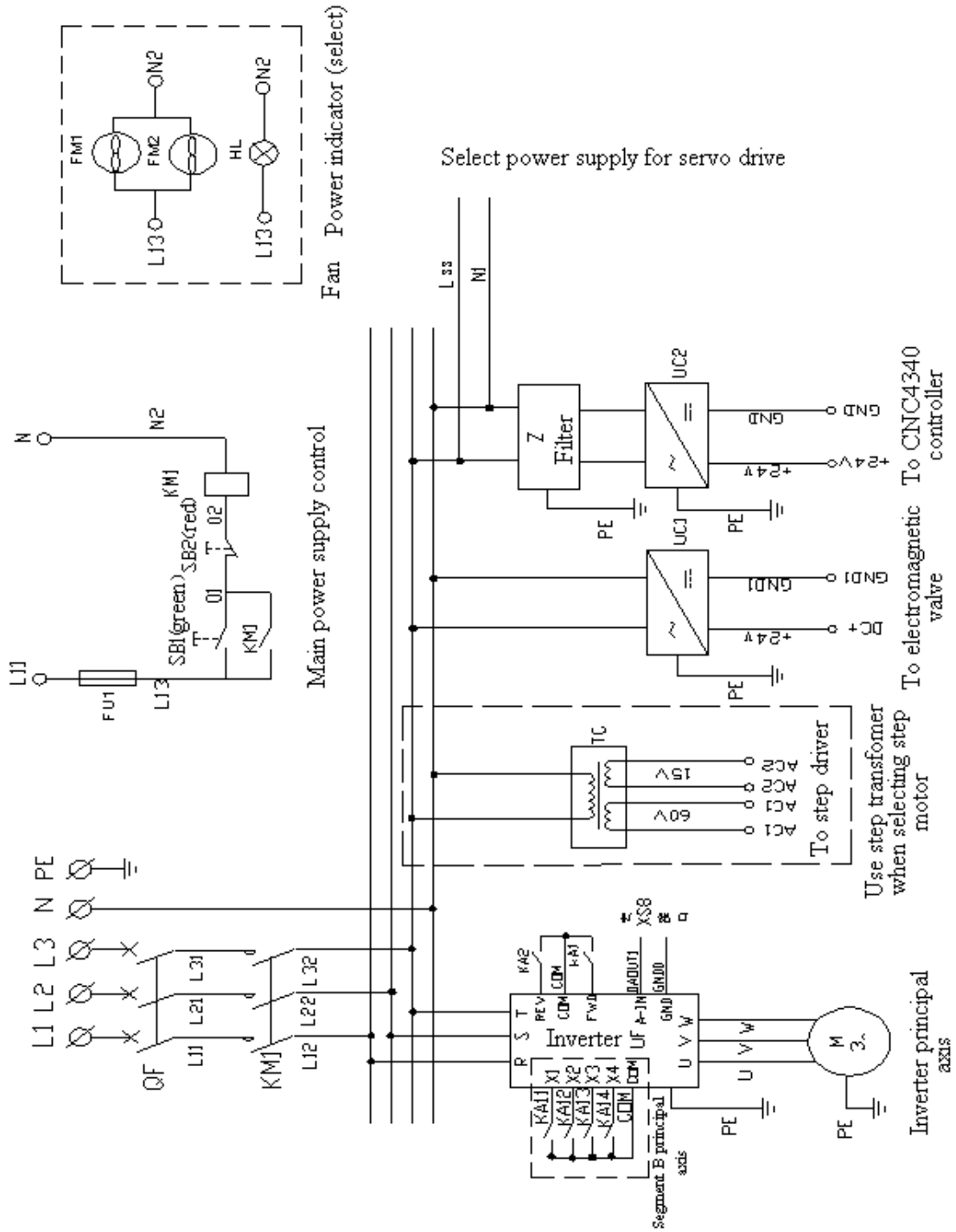


Fig. 2.2.1 Overall System Wiring Diagram

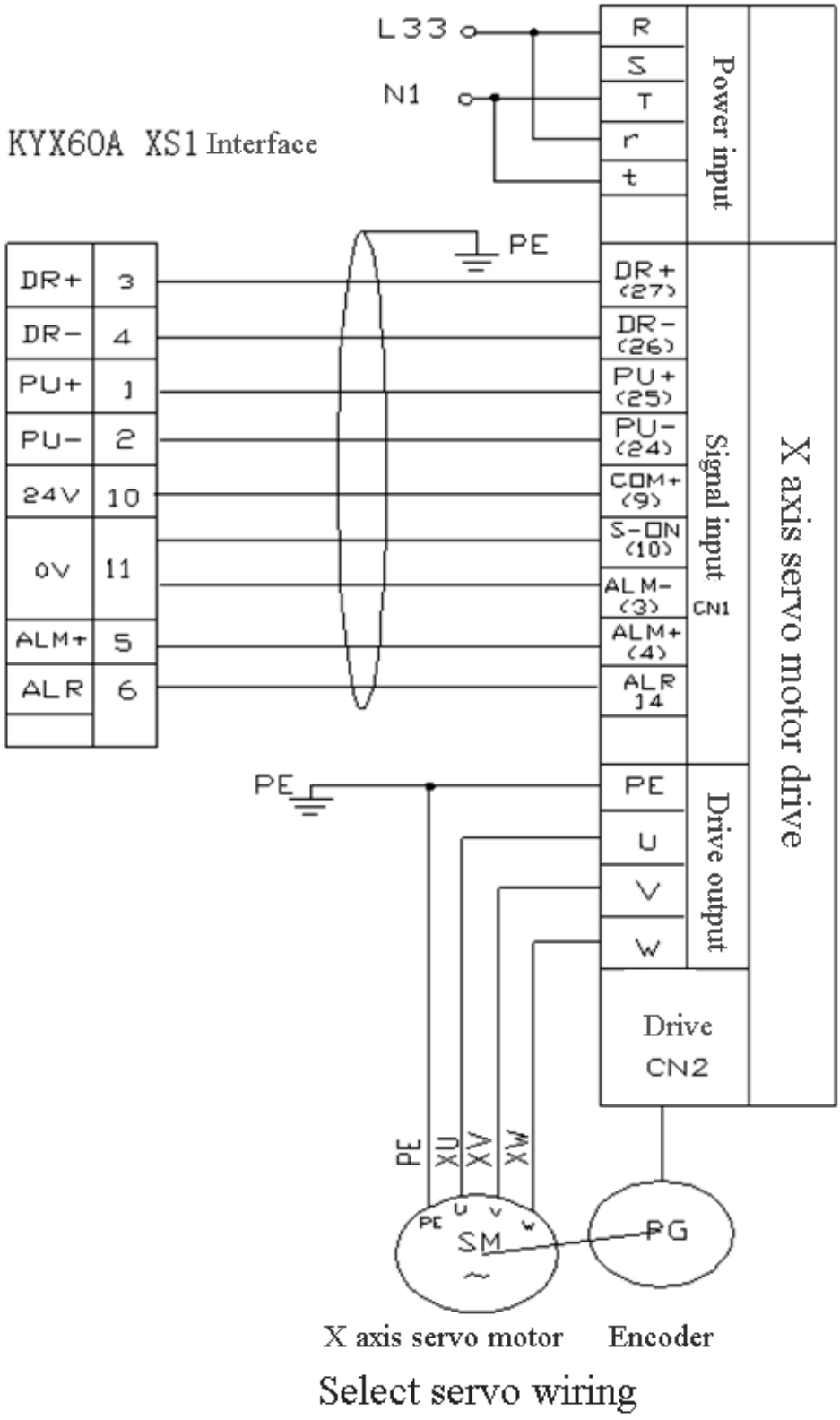
2.2.2 Symbol Schematic Diagram

Symbol	Name	Figure	Symbol	Name	Figure
QF	Breaker		SM	Servo motor	
KM	Contactor		M	Step motor	
UF	Inverter		SQ	Approach switch	
M	Motor		SA	Foot switch	
TC	Transfomer		YB	Motorbrake	
Z	Filter		FR	Thermalrelay	
FU	Fuse		UC	Switching power supply	
SB	Button		YV	Electromagnetic valve	
FM	Fan		C	Capacitor	
HL	Indicator		R	Resistor	
QS	Touch switch		QS	Travelswitch	
PG	Encoder		KA	Relay	

2.2.3 Power Connection Diagram

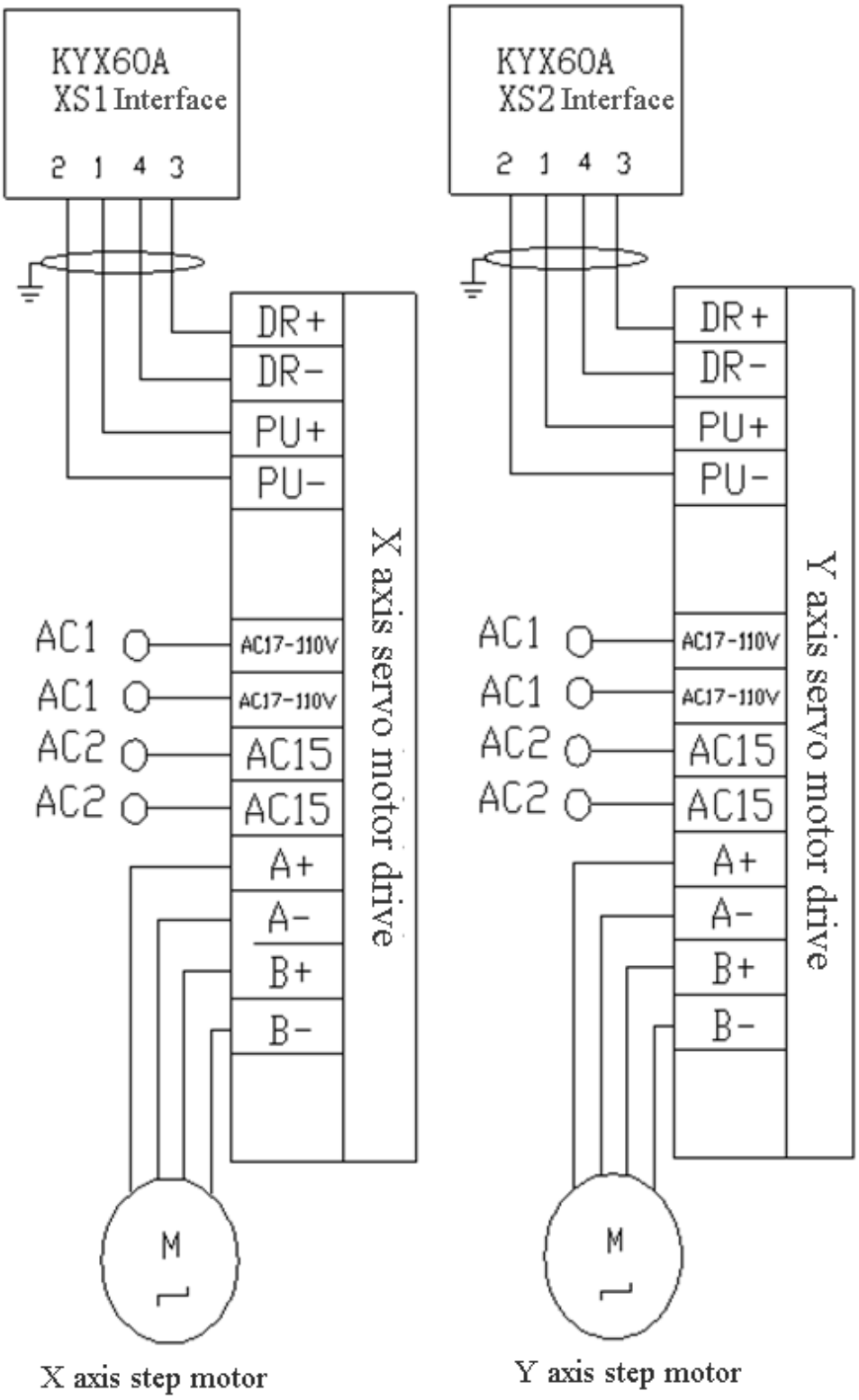


2.2.4 Servo Drive Connection Diagram



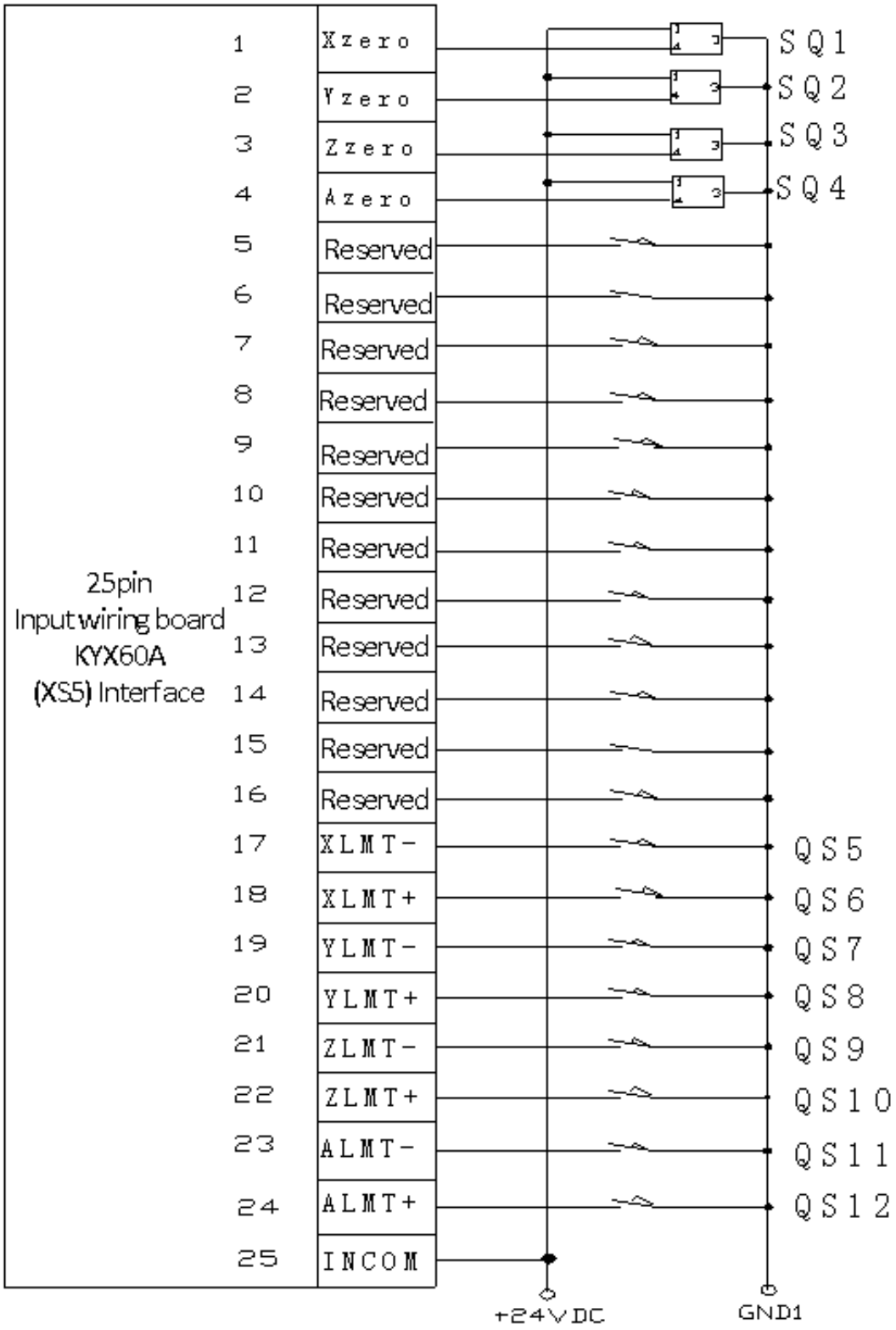
Select servo wiring

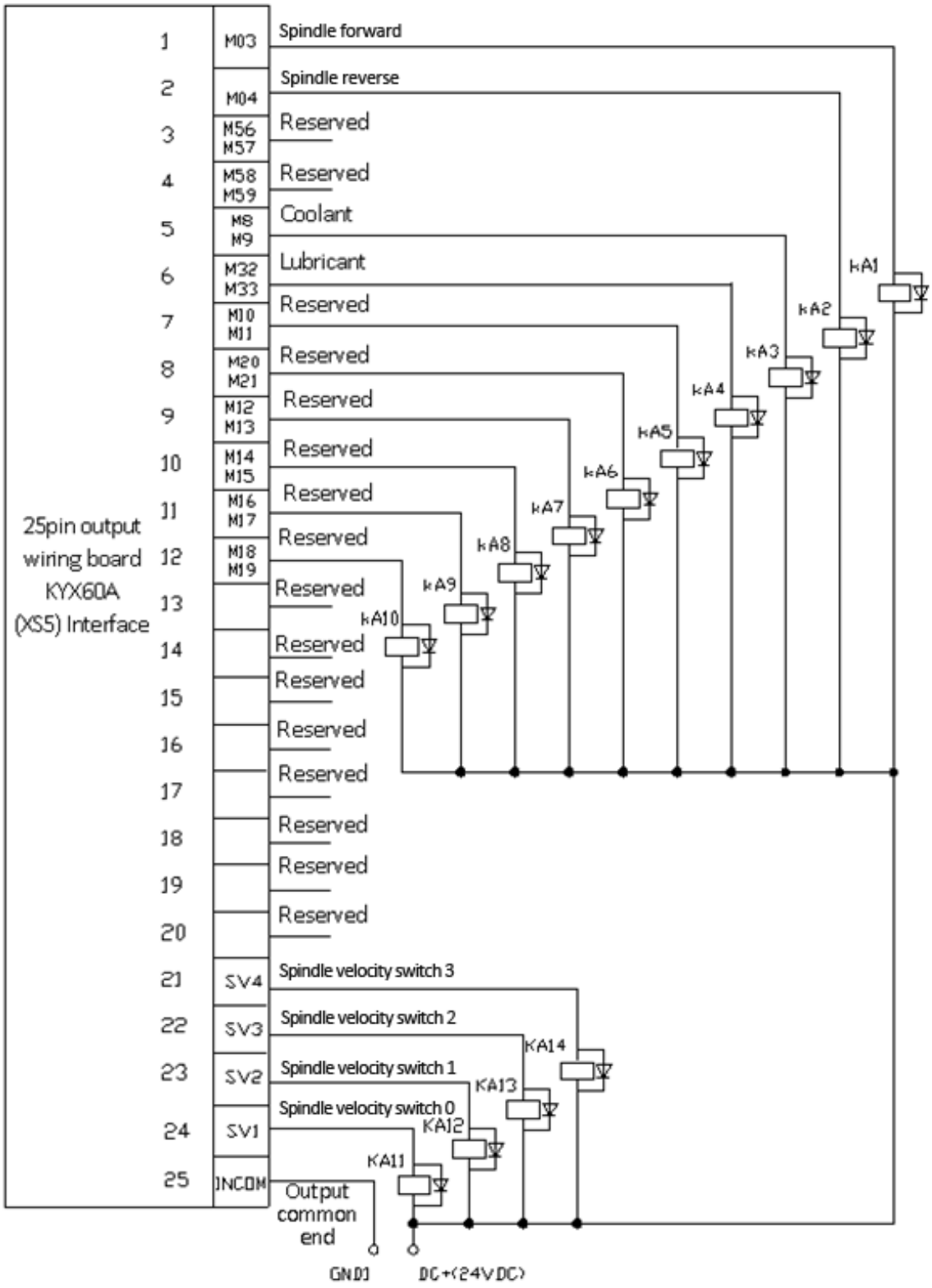
2.2.5 Step Connection Diagram



Q2BYG1106M Step System

2.2.6 IO electrical connection diagram

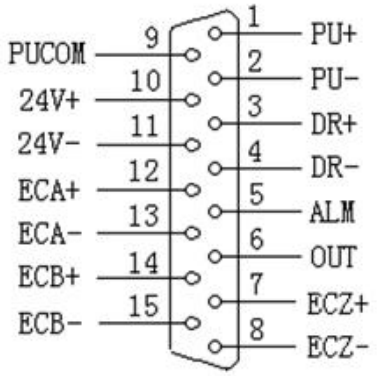




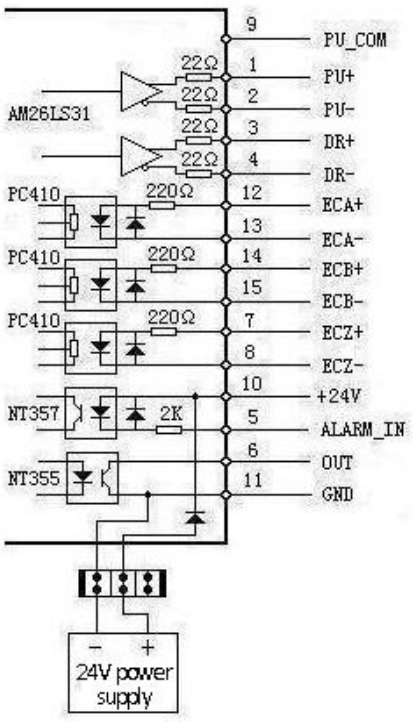
2.3 Interface definition and wiring diagram

2.3.1 Motor Drive Control Interface (XS1…XS4)

Four drive interfaces are available (XS1 X axis, XS2 Y axis, XS3 Z axis, XS4 A axis), and they have the same definition, as shown below:



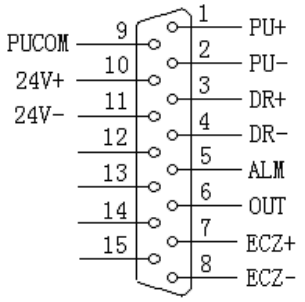
KYX60A pulse connection



Simple Internal Circuit Diagram for Pulse Output

Wire No.	Definition	Function
1	PU+	Pulse signal+
2	PU-	Pulse signal -
3	DR+	Direction signal +
4	DR-	Direction signal -
5	ALM	Servo alarm signal input X axis: IN34, Y axis: IN35, Z axis: IN36, A axis: IN37
6	OUT	Axis alarm reset output signal X axis: OUT24, Y axis: OUT25, Z axis: OUT26 A, axis: OUT27
7	ECZ+	Encoder phase Z input +
8	ECZ-	Encoder phase Z input -
9	PUCOM	Controller for single end input
10	24V+	Internally provided 24V power supply, directly connected to 24V power supply of the controller
11	24V-	
12	ECA+	Encoder phase A input +
13	ECA-	Encoder phase A input -
14	ECB+	Encoder phase B input +
15	ECB-	Encoder phase B input -

Standard pulse wiring diagram

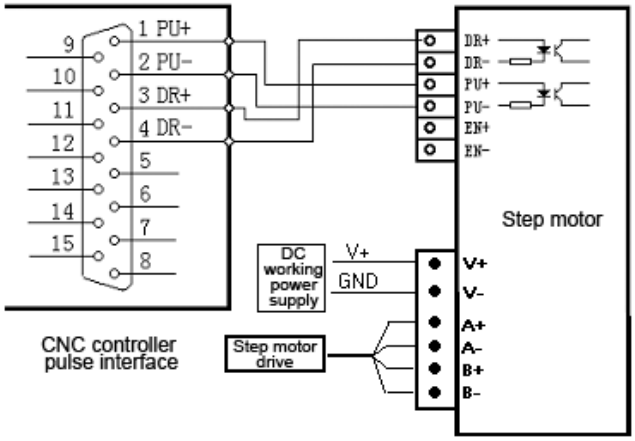


XS1 … XS4 Pulse Interfaces Standard Wiring

This wiring is suitable for NCT-XX controller;

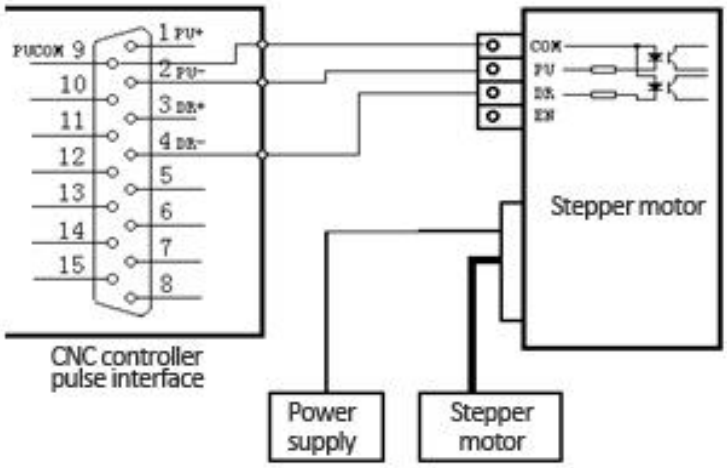
Step motor drive cable to differential input

Adtech CNC drive is for reference, all of which use differential input mode. This mode has strong anti-interference and is recommended. Please refer to the figure below for the connection of KYX60A with step motor drive and step motor



Wiring diagram of single end input stepper motor drive

Certain companies connect together the optocoupler input cathodes of step drives, i.e. common cathode connection, which isn't suitable for KYX60A controller. Common anode connection connects together the anodes of optocoupler input. The wiring shall follow the figure below, and do not connect PU+ and DR+ together, or else the pulse interface may be damaged.



Wiring diagram of common anode input stepper motor drive

Servo motor drive wiring diagram

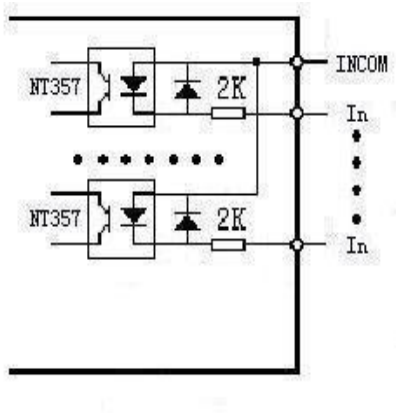
Since differential connection is used in most cases, please refer to differential mode for the pulse connection. Most servo drives require 12-24V power supply, and the 24V power provided by pin 10, 11 is available. The specific connection depends on servo drive. Please contact us if you have any question.

- **Caution:**

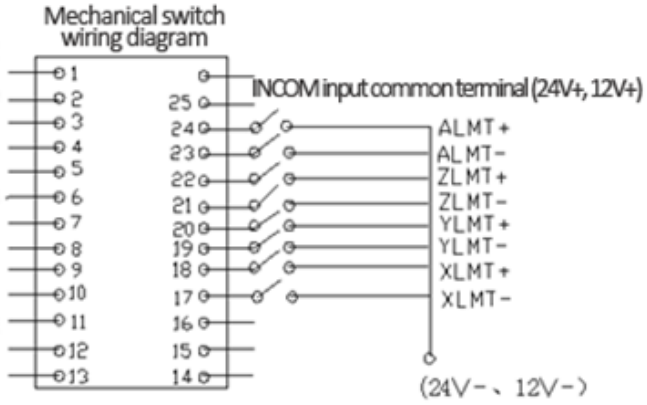
Either two of PU+, PU-, DR+ and DR- shouldn't be connected, or else the pulse interface may be damaged.

2.3.2 Digital input interface (XS5)

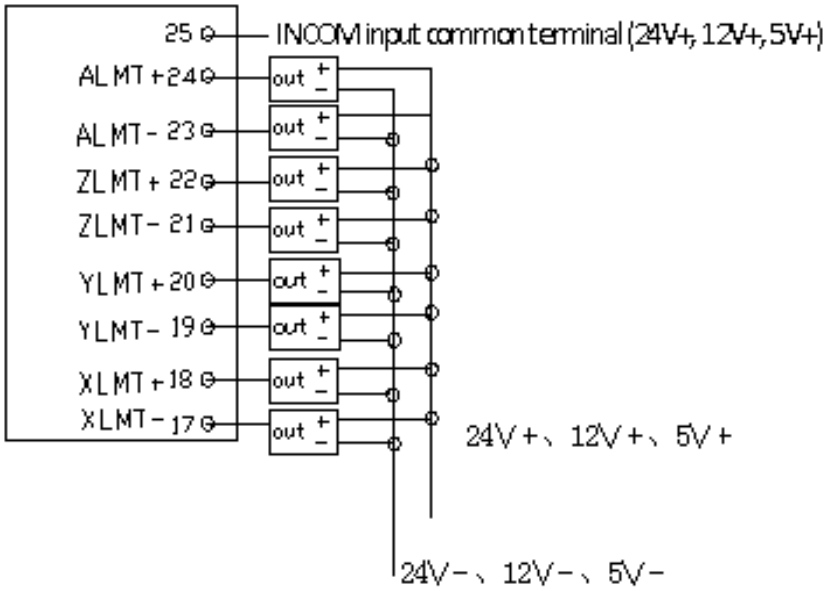
Digital input interface contains the hardware limit signal of each axis of the system and is defined in the following figure



Digital Input Brief Internal Output Diagram



Photoelectric Switch Wiring Diagram



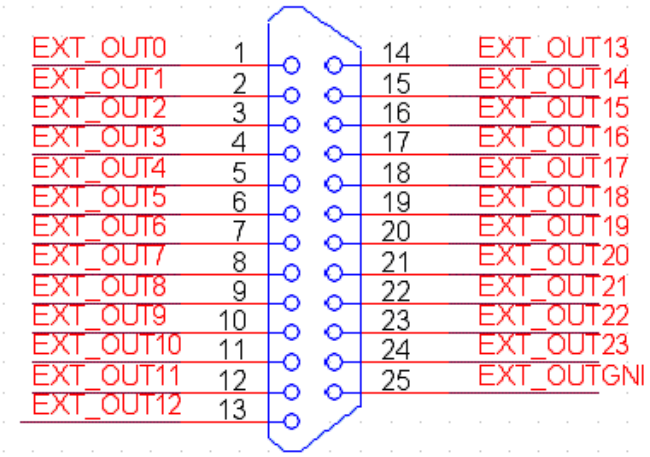
The + terminal of external module is the positive pole of the power supply of proximity switch, - terminal is the ground wire of the proximity switch, and OUT terminal is the output signal. The power supply for general proximity switch should be 10-30V, and NPN output. Photoelectric switch is similar.

Refer to the following allocation table for the usage of input ports:

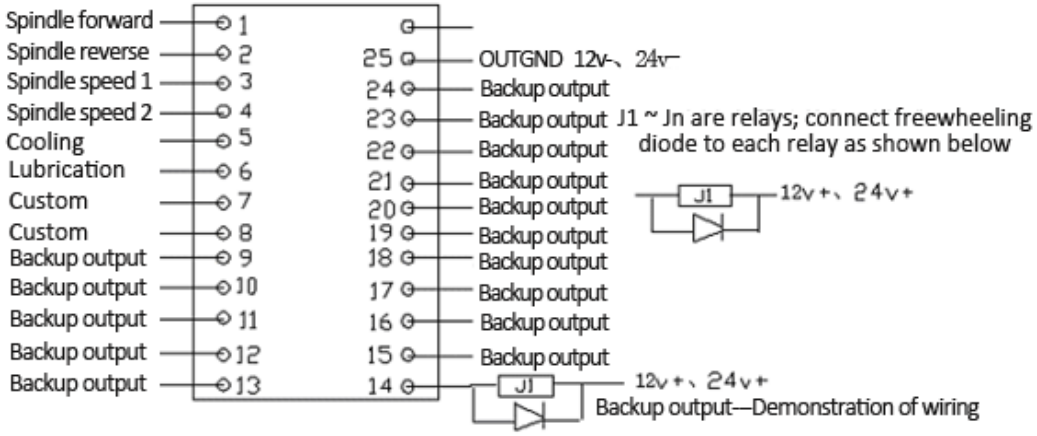
Wire No.	Port definition	Function
1	IN0	X axis zero point
2	IN1	Y axis zero point
3	IN2	Z axis zero point
4	IN3	A axis zero point
5	IN4	Reserved input
6	IN5	Reserved input
7	IN6	Reserved input
8	IN7	Reserved input
9	IN8	Reserved input
10	IN9	Reserved input
11	IN10	Reserved input
12	IN11	Reserved input
13	IN12	Reserved input
14	IN13	Reserved input
15	IN14	Reserved input
16	IN15	Reserved input
17	IN16(XLMT-)	X-axis negative limit (Reserved IN32)
18	IN17(XLMT+)	X-axis positive limit (Reserved IN33)
19	IN18(YLMT-)	Y-axis negative limit (Reserved IN34)
20	IN19(YLMT+)	Y-axis positive limit (Reserved IN35)
21	IN20(ZLMT-)	Z-axis negative limit (Reserved IN36)
22	IN21(ZLMT+)	Z-axis positive limit (Reserved IN37)
23	IN22(ALMT-)	A-axis negative limit (Reserved IN38)
24	IN23(ALMT+)	A-axis positive limit (Reserved IN39)
25	INCOM	Input common terminal INCOM (24V+, 12V+) connected to internally or externally supplied power

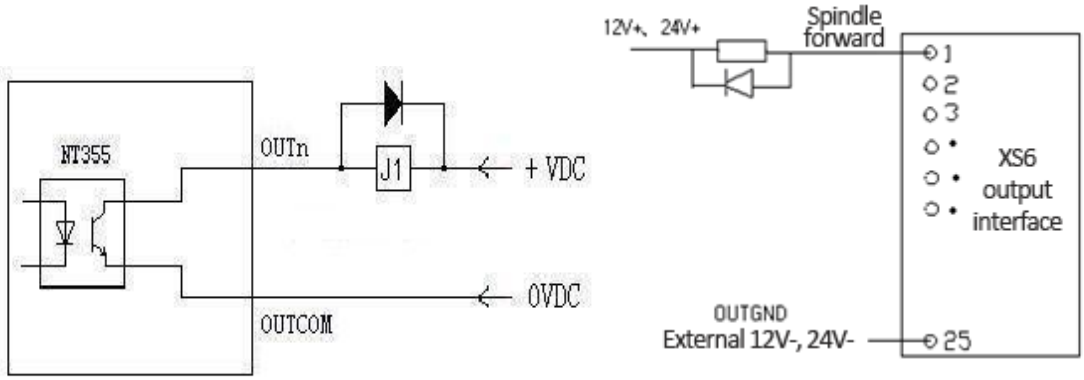
2.3.3 Digital Output Interface (XS6)

The wiring of digital output interface follows:



XS6 Output Interface Wiring Diagram

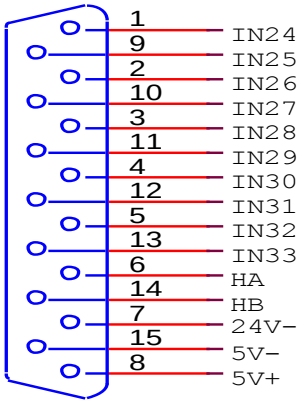


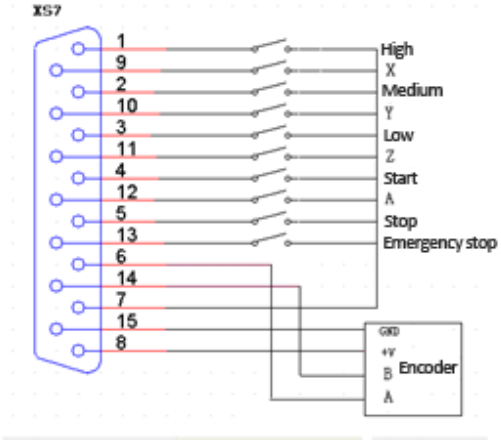


Digital output brief internal circuit (left) See figure at right for machine tool wiring (main axis CW as an example)

Wire No.	Port Definition	Function
1	OUT0	Main axis CW (M03)
2	OUT1	Main axis CCW (M04)
3	OUT2	Reserved output(M56, M57)
4	OUT3	Reserved output(M58, M59)
5	OUT4	Reserved output(M08, M09)
6	OUT5	Reserved output(M32, M33)
7	OUT6	Reserved output(M10, M11)
8	OUT7	Reserved output(M20, M21)
9	OUT8	Reserved output(M12, M13)
10	OUT9	Reserved output(M14, M15)
11	OUT10	Reserved output(M16, M17)
12	OUT11	Reserved output(M18, M19)
13	OUT12	Reserved output(M40, M41)
14	OUT13	Reserved output(M42, M43)
15	OUT14	Reserved output(M44, M45)
16	OUT15	Reserved output(M46, M47)
17	OUT16	Reserved output(M48, M49)
18	OUT17	Reserved output(M50, M51)
19	OUT18	Alarm indicator
20	OUT19	Running indicator
21	OUT20	Variable frequency speed switch 3 (M66, M67)
22	OUT21	Variable frequency speed switch 2 (M64, M65)
23	OUT22	Variable frequency speed switch 1 (M62, M63)
24	OUT23	Variable frequency speed switch 0 (M60, M61)
25		OUTGND12V-, 24V- connects to output port common power supply

2.3.4 Handheld box interface (XS7)

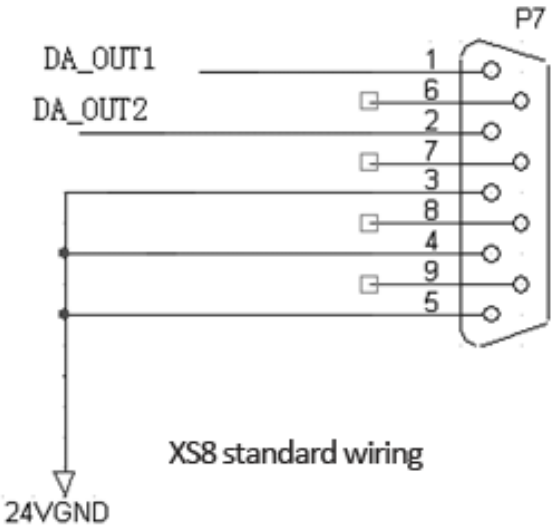


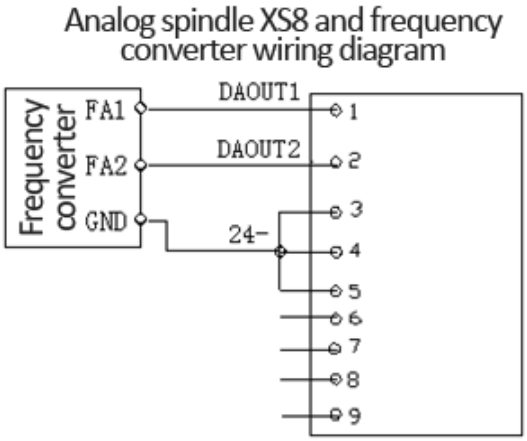


Wire No.	Definition	Function
1	(IN24) position switch	0.1---High speed
2	(IN26) position switch	0.01---Medium speed
3	(IN28) position switch	0.001---Low speed
4	(IN30) button	Cycle start
5	(IN32) button	Pause
7	24V-	Cathode of internal 24V power supply
9	(IN25) axis selection	X axis
10	(IN27) axis selection	Y axis
11	(IN29) axis selection	Z axis
12	(IN31) axis selection	A axis
13	(IN33) button	Emergency stop
6	HA	Hand encoder A phase input signal
14	HB	Hand encoder B phase input signal
15	5V-	Cathode of internal 5V power supply
8	+5V	Anode of internal 5V power supply
7	24V-	Cathode of internal 24V power supply

2.3.5 Analog output interface (XS8)

The standard wiring diagram of analog output interface:

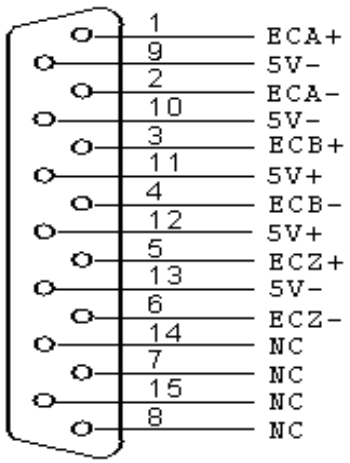




Wire No.	Definition	Function
1	DAOUT1	Analog voltage output (0~10)V
2	DAOUT2	Analog voltage output (0~10)V
3	GND	Internal 24V power grounding
4	GND	Internal 24V power grounding
5	GND	Internal 24V power grounding

2.3.6 Main axis encoder interface (XS12)

The standard wiring diagram of main axis encoder:



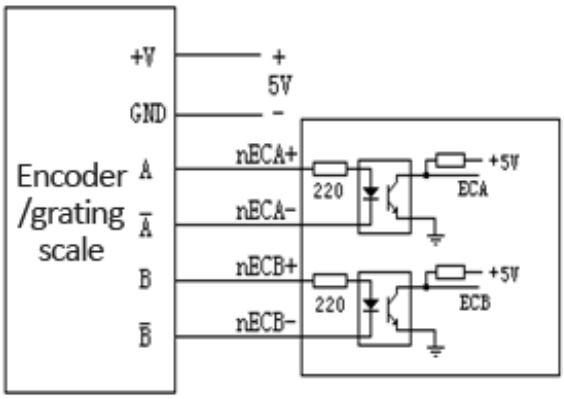
The standard wiring of main axis encoder applies to KYX60A controller;

Wire No.	Definition	Function
1	ECA+	Encoder phase A input +
2	ECA-	Encoder phase A input -
3	ECB+	Encoder phase B input +
4	ECB-	Encoder phase B input -
5	ECZ+	Encoder phase Z input + (Reserved)
6	ECZ-	Encoder phase Z input - (Reserved)
7	NC	Null
8	NC	Null
9	5V-	Internal 5V power cathode, can't be connected to external power supply
10	5V-	Internal 5V power cathode, can't be connected to external power supply
11	5V+	Internal 5V power anode, can't be connected to external power supply
12	5V+	Internal 5V power anode, can't be connected to external power supply
13	5V-	Internal 5V power cathode, can't be connected to external power supply
14	NC	Null
15	NC	Null

- AB phase decoding input includes differential connection and common anode connection, which is determined by the encoder type.

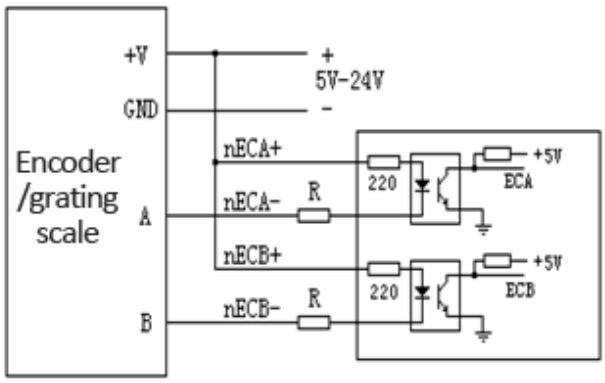
- Encoder output generally includes open collector output, complementary output, voltage output and long drive output, of which open collector output, complementary output and voltage output use common anode connection, and long drive output uses differential connection.
- As shown below, AB phase decoding input signal is differential input wiring; if it is common anode connection, the positive terminal of phase A should be connected to positive terminal of phase B; if it is common cathode connection, the negative terminal of phase A should be connected to negative terminal of phase B.

Differential connection is shown below:



- The 5V power supply is provided externally.

Common anode connection is shown below:



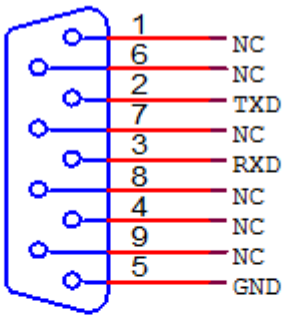
- The supply voltage is determined by the encoder. When 5V power supply is used, the resistor R is unavailable; when 12V power supply is used, R uses 1K-2K resistor; when 24V power supply is used, R uses 2K-5K resistor.

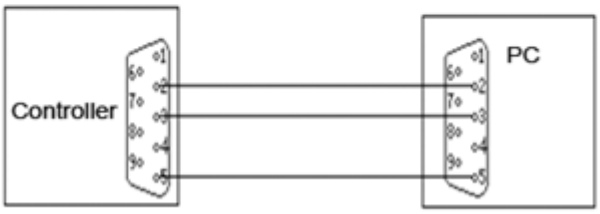
Suggestion:

It is recommended to use differential output encoder, because differential mode has better noise immunity if the line is long.

2.3.7 RS232 Transmission Interface (XS9)

Serial communication interface ---9-core signal socket (male)





RS-232 communication mode

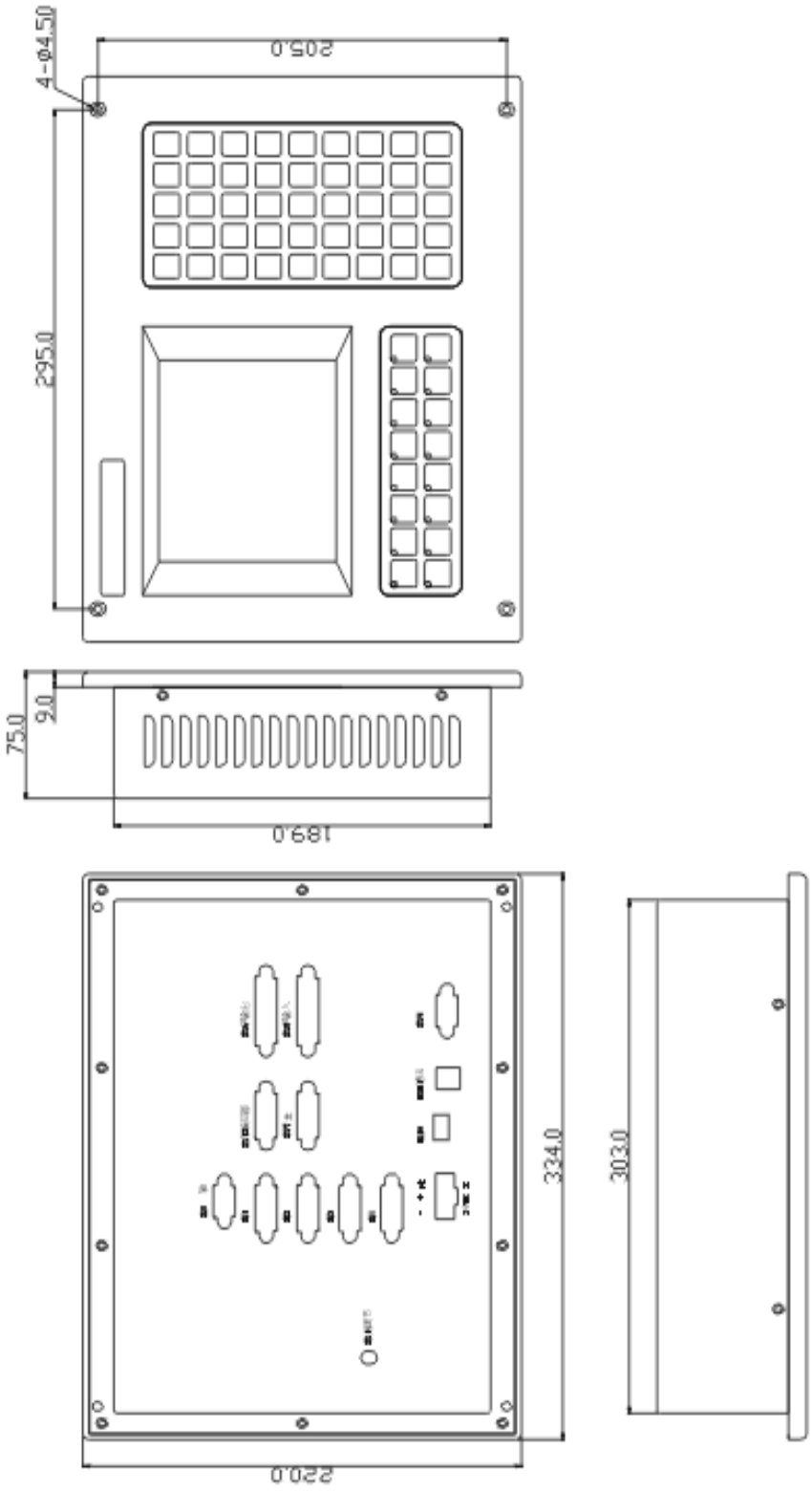
2.3.8 USB Memory Connection Interface (XS10)

Standard USB memory (e.g. USB disk) interface;

2.3.9 PC USB Communication Interface (XS11)

Standard USB communication interface;

2.4 Mounting Dimensions



Installation precautions:

Installation condition for electric cabinet

(1) The cabinet must be able to effectively prevent dust, coolant and organic solution entering;

(2) When design electric cabinet, the distance between rear cover and case should be at least 20CM; considering the temperature rises in the cabinet, the temperature difference between interior and exterior of the cabinet shouldn't exceed 10℃;

(3) The cabinet should be installed with fan to ensure interior ventilation;

(4) The display panel should be installed at the position can't be sprayed by the coolant;

(5) When design electric cabinet, the external electrical interference should be reduced to lowest to prevent interfering with the system;

To prevent interference

The system is designed with anti-interference measures such as shielding space electromagnetic radiation, absorbing impact current and filtering power clutter, which can prevent interference with the system in certain degree. To ensure system stability, please take the following measures to install and connect the system:

(1) KYX60A must be kept away from the equipment with interference (e.g. inverter, AC contactor, electrostatic generator, high voltage generator, and sub-unit of power lines), and the switching power supply should be connected to a filter to improve the anti-interference of KYX60A (as in Fig.1-4);

(2) To supply power to the system through isolation transformer, the machine tool must be grounded, KYX60A and drive must be connected to separate earth wire.

To suppress interference

Connect RC circuit (0.01μF, 100~200Ω, as in Fig. 1-5) to both sides of AC coil in parallel. RC circuit should be installed close to inductive load; connect freewheeling diode reversely on both sides of DC coil in parallel (as in Fig.1-6); connect surge absorber to the winding of AC motor in parallel (as in Fig. 1-7).

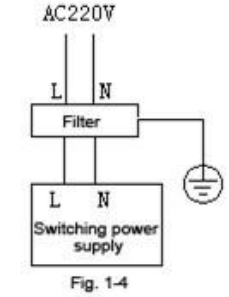


Fig. 1-4

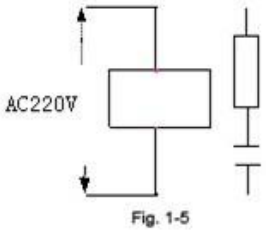


Fig. 1-5

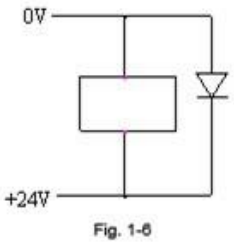


Fig. 1-6

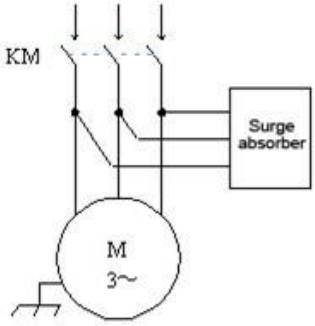


Fig. 1-7

To reduce the interference between KYX60A signal cables and strong current cables, the wiring shall follow the principles below:

Group	Cable type	Wiring Requirement
A	AC power cord	Bundle the cables of group A separately from group B and C, keep at least 10cm clearance, or make electromagnetic shielding for group A
	AC coil	
	AC contactor	
B	AC coil (24VDC)	Bundle the cables of group B separately from group A or shield group B; group B and group C should be as far as possible
	DC relay (24VDC)	
	Cable between system and strong current cabinet	
	Cable between system and machine tool	
C	Cable between system and servo drive	Bundle the cables of group C separately from group A, or shield group C; keep at least 10cm clearance between group C and group B and use twisted pair
	Position feedback cable	
	Position encoder cable	
	Handwheel cable	
	Other cables for shielding	

Chapter 3 Function operation

3.1 Full operation

3.1.1 LCD/ Keypad

Keypad

Fig. 3.1.1.1 Diagram of KYX60A series operation panel

Press the submenu button to operate the sprite corresponding content of the sub-menu.

Manual axis motion and edit input buttons are multiplex and have different definitions in different modes.

Working mode switching area is used to switch the working mode of the system. Different modes are used to enhance the safety and performance of the system operation. Hand wheel and single-step modes are switched through repeat button.

LCD display unit

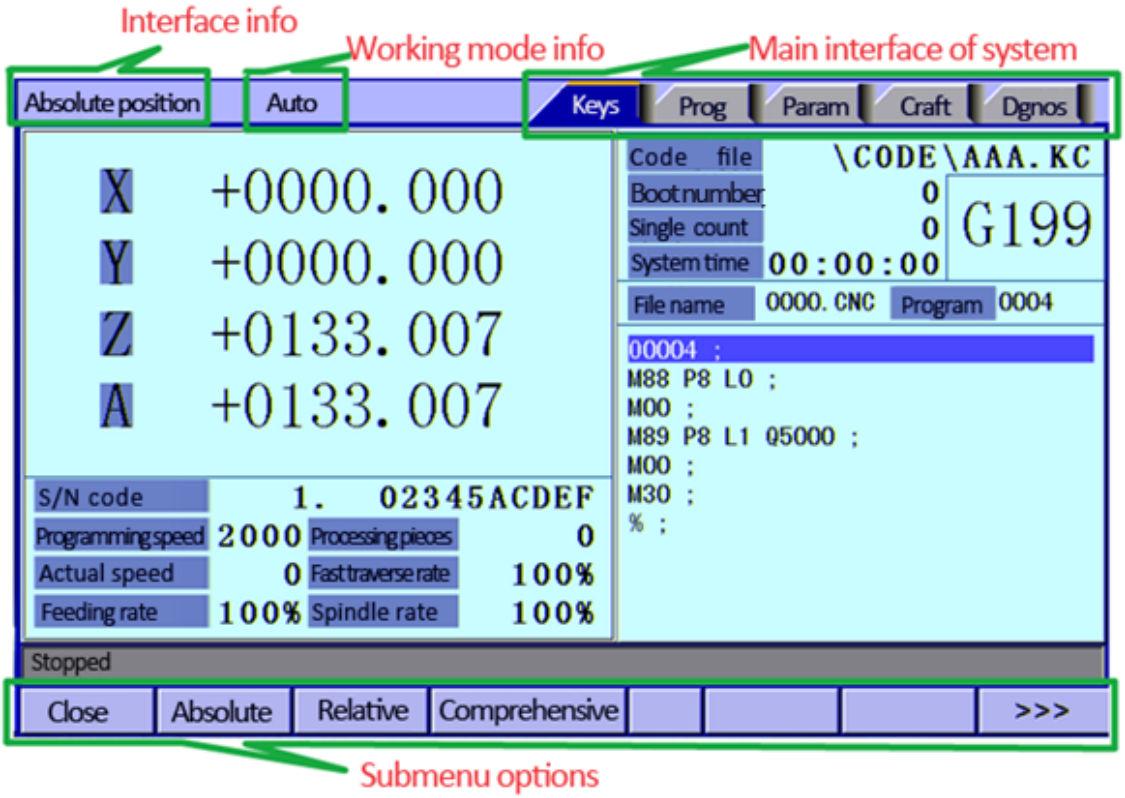


Fig. 3.1.1.2 KYX60ALCD interface Diagram

Information on the interface shows the current window information

Working mode information shows the current selected working mode

Main interface information displays the current contents of the main interface.

Sub-interface menu options are used to switch corresponding sub-screen with left triangle, F1 ~ F6 and right triangle button. The right arrow is used to turn pages, the left arrow is used to close submenu and return to previous menu.

- **Note:**

  KYX60A doesn't support interface brightness adjustment.

3.1.2 System Menus

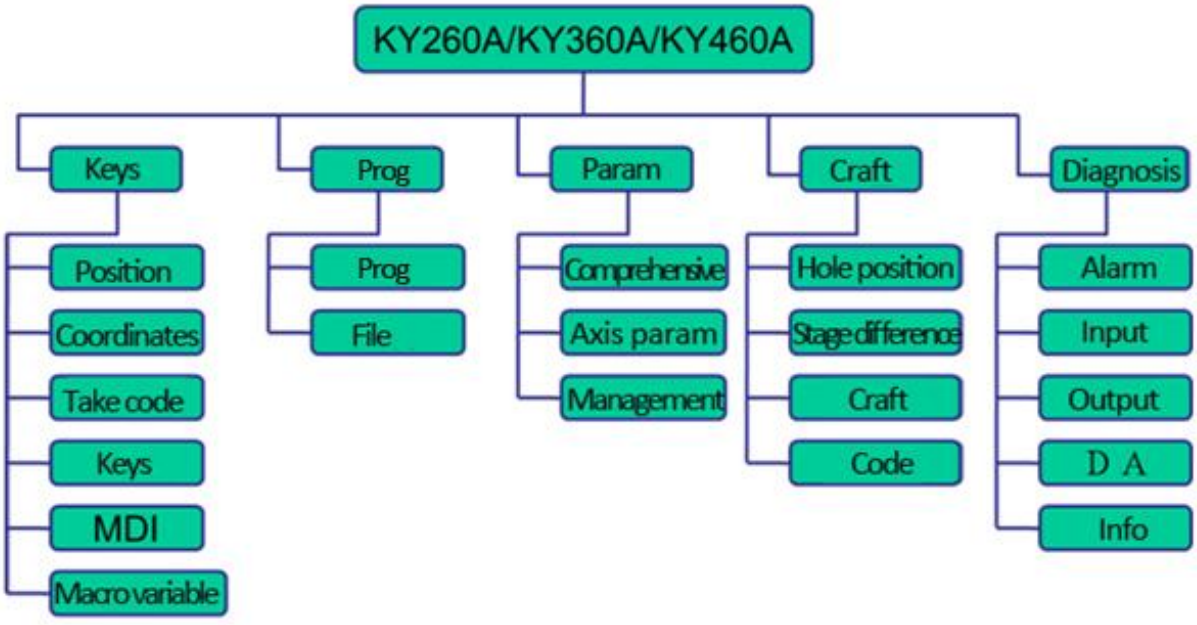
KYX60A system uses cascading menu structure. You can press the following keys to operate the menus

Press a key to show the corresponding content in the bottom of the LCD.

Key in the left: Return to previous menu

Key in the right: Turn pages to show other menus of same level

The main menus of the system include [Keys], [Program edit] , [Parameter], [Craft] and [Diagnosis]. Each main menu contains several submenus, which are shown below:



3.1.3 Operating Keys

The keys of NCT-XX system are defined below:

Keys	Purpose
[RESET]	Cancel alarm, reset CNC
Address/number key	Enter letters, numbers, etc.
[EOB], [CAN]	Confirm or cancel operation
[EOB], [CAN], [DEL]	Program edit (insert, delete, modify)
Mode switch key	Select operating mode
Cursor moving key	Four keys are available: Up/Down: adjust ration, move cursor between subsections; Left/Right: move cursor to left/right
Page key	Up/Down: Turn pages
Menu key	Select the menus
Main axis CW	Press once and the main axis rotates clockwise, press

Keys	Purpose
	it again and the main axis stops
Spindle CCW	Press once and the spindle rotates counterclockwise, press it again and the spindle stops
Cylinder button 1	Define cylinder output flexibly
Cylinder button 2	Define cylinder output flexibly
Cylinder button 1	Define cylinder output flexibly
[SBK]	SBK function ON/OFF
[PAUSE]	Automatic operation is temporarily stopped
[START]	Start automatic running

3.2 Manual Operation

3.2.1 Returning to reference point manually

KYX60A machine tool has specific mechanical position, which is called as reference point and for tool exchange and coordinates setting. Generally, when the power supply is connected, the tool should be moved to the reference point. This operation is also called as home operation, which will make the CNC system confirm the origin of machine tool.

The home operation includes program and mechanical mode:

- **Program home**

  For program home, the action completes when the coordinates of machine tool are 0, and won't check whether origin switch is in position;

- **Mechanical home**

  For mechanical home, the external home sensor switch is used to locate the origin of the machine tool; three checking modes are available:

  (1) With the external sensor switch, the home operation completes when the sensing is successfully repeatedly.

  (2) The external sensor switch is used as deceleration switch, the servo home is enabled as home signal after sensing and then the sensing stops.

  (3) Homing relying on servo Z-phase zero point.

For specific selection, set the 'Homing mode' to "Program" or "Mechanical" in [Parameters] [Comprehensive parameters]. You can also press [SBK] key in home mode to switch among "Mechanical – Program – Mechanical…" quickly. This method doesn't conflict with parameter setting. You can select accordingly. To use servo home as the home signal, you need to set "Axis phase Z home enable" to "1" in [Parameters][Axis

Configuration] in mechanical home mode, and the setting will take effect in next home checking.

Several methods are available for tool returning to reference point and the steps follow:

➢ **Single axis homing**

Press the mode switch key [Home] to select home operation;

Press the composite key [X-], [Y-], [Z-], [A-] in the numbers section to return the corresponding axis to reference point.

➢ **Multi-axis homing**

Press the mode switch key [Home] to select home operation;

Press the [Start] key to return Z axis to reference point, and other axes return to reference point simultaneously. The automatic home sequence can be configured in the parameters.

➢ **Reset machine tool position**

Press the mode switch key [Home] to select home operation;

In [Absolute Position] and [Coordinate System] screen, press [X], [Y], [Z], [A] respectively to show the value of corresponding axis position, and then press the [Cancel] key to reset the machine tool position of current axis, i.e. current point is used as machine tool origin. After this operation, the system considers it as a home action. Therefore, when the program is running, the alarm of not home won't occur. If you press by mistake, it will switch the screen and cancel selection automatically.

➢ **Reset relative position manually**

Press the mode switch key [Manual] to select manual operation;

In [Relative Position] and [Coordinate System] screen, press [X-], [Y-], [Z-], [A-] respectively to show the value of corresponding axis, and then press the [Cancel] key to reset the relative position of current axis.

✧ **Note**

The tool also can return to reference point according to program instruction, i.e. returning to reference point automatically.

● **Caution:**

Generally, the system will perform home operation after connecting the power supply. If the power fails while the machine tool is moving, the system also will return to reference point when the power supply is connected again. First, return to Z-axis to avoid collision of tool and workpiece and damage the tool, workpiece and fixture.

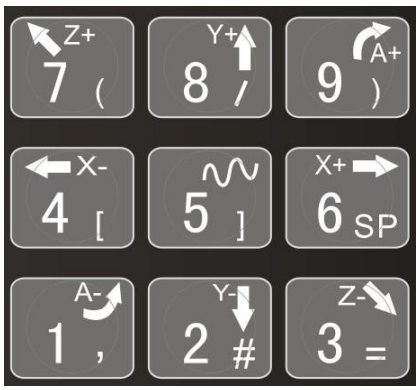
3.2.2 Continuous Feeding Manually

Press the keys on the operation panel or handwheel to move the tool along every axis.

The operation follows:

Press the mode switch key [Manual] to select manual operation;

Press composite keys [X+], [X-]; [Y+], [Y-]; [Z+], [Z-]; [A+], [A-] in numbers area to move the tool along selected axis. The keypad follows:



In manual mode, 5# key can be used to switch the manual speed and rapid traverse speed. The rapid traverse speed of every axis depends on comprehensive parameter 009-012 (rapid traverse speed setting). After switching to rapid traverse speed, the manual speed of the position interface will be highlighted, while the actual speed of the position interface is sampled from the moving speed of current axis. This value can truly reflect the moving speed of current axis (unit: mm/min);

✧ **Note:**

Only single axis motion is available in manual mode.

3.2.3 Single Step Feeding

Single step mode is similar to manual mode, the operations are same, but only moves a specified pulse increment every time press the key.

The specific operation follows:

Press the mode switch key [Handwheel /Single step] (this key is composite, and you can press it repeatedly to switch the modes) to select the single step operation;

Press composite keys [X+], [X-]; [Y+], [Y-]; [Z+], [Z-]; [A+], [A-] in numbers area to move the tool for a fixed distance along the selected axis. This distance is controlled by four rates (1.000, 0.100, 0.010, 0.001) (unit: mm). To select pulse increment, press Up (+) and Down (-) key in the [Position] interface.

3.2.4 Handwheel feeding

In Handwheel mode, rotate the handwheel to make the machine to perform single step or continuous motion. The feeding amount is determined by detecting the MPG signal of the handheld box. In Handwheel mode, feeding axis and feeding unit are determined by axis selection signal of the handheld box.

Handwheel feeding steps:

(1) Press the operating mode switch key [Handwheel / Single step] to select the handwheel operation;

(2) Rotate the axis selection DIP switch on the MPG to select the Handwheel axis (X, Y, Z, A);

(3) Rotate the incremental DIP switch on the Handwheel to select the amount of movement (0.1, 0.01, 0.001);

(4) Rotate the Handwheel to make the machine tool move; the tool moves a certain distance every time the Handwheel rotates a scale. (For example, when the X-axis is selected in step 2, and 0.01 is selected in step 3, the tool moves 0.01mm when rotate a scale). Rotate the handle continuously to move the machine tool continuously in the axis.

✧ **Note:**

Handwheel feeding mode can only controls one axis; the faster the Handwheel speed, the faster the machine tool moves.

3.2.5 Manual auxiliary function operation

Spindle CW/stop

 In Handwheel / Single step / manual mode, press this button to start clockwise rotation of main axis. Press the button again to stop rotation of the main axis. Button LED: In any mode, the LED turns on when the main axis rotates clockwise, or else it turns off.

Spindle CCW/stop

 In Handwheel / Single step / manual mode, press this button to start counterclockwise rotation of main axis. Press the button again to stop rotation of the main axis. Button LED: In any mode, the LED turns on when the main axis rotates counterclockwise, or else it turns off.

Cylinder 1 ON / OFF

 In Handwheel / Single step / manual mode, press this button and cylinder 1 switch between on and off.

Button LED: In any mode, the LED turns on when cylinder 1 opens, or else it turns off.

Cylinder 2 ON / OFF

In Handwheel / Single step / manual mode, press this button and cylinder 2 switch between on and off.

Button LED: In any mode, the LED turns on when cylinder 2 opens, or else it turns off.

Cylinder 3 ON / OFF

In Handwheel / Single step / manual mode, press this button and cylinder 3 switch between on and off.

Button LED: In any mode, the LED turns on when cylinder 3 opens, or else it turns off.

General description of manual operation buttons

Cooling, lubrication and CW/CCW rotation of main axis are valid in Handwheel, Single step and manual mode;

When the main axis rotates, press the rotation button of opposite direction, the main axis will first stop, press again and it will rotate in the opposite direction;

When the auxiliary output turns on, if the system is switched to another mode, the output remains unchanged; press 'Reset' button to turn it off, or execute the corresponding M code to turn off corresponding output in automatic operation mode, or execute the corresponding M code to turn off corresponding output in MDI interface in edit mode;

When execute M04/M03 directly during main axis CW/CCW rotation, the system first stops rotation, and then executes M04/M03 command; during emergency stop, main axis CW/CCW rotation stops, and other output can be set according to the system parameters.

3.2.6 Tool setting operation

Tool setting is a main operation and critical skill in CNC processing. Under certain conditions, the tool setting precision determines machining precision of parts, and tool setting also has a direct impact on CNC machining efficiency. Handheld box can be used for tool setting of KYX60. The operating mode of KYX60A should be switched to Handwheel mode, use Handwheel to move each axis of the system to the tool setting position, press [X-], [Y-], [Z-], [A-] in the main screen respectively, and then press the [Cancel] button to set the home of [X-], [Y-], [Z-], [A-] respectively. Tool setting can be performed for X, Y, Z, A axis respectively or

simultaneously according to the specific circumstances. After setting the home with the hand-held box, switch the system to the [Manual] mode in [Coordinate System], and then press the [EOB] button to set the current coordinate position in the coordinate system. Similarly, it can be switched to [Edit] mode, and set the coordinates in the coordinate system manually.

3.2.7 Data setting

System parameters can be modified as follows:

Select the edit mode

Press [Parameters] in the main menu to enter the parameters setting s interface;

Press the submenu selection button to select the parameters type (comprehensive, management...);

Move the cursor to select the parameters, enter the new parameters and press [EOB], and the parameters at the cursor is modified.

3.2.8 System shortcuts

Numeric Input dialog box has two input methods: direct assignment and incremental assignment. Direct assignment refers that the input number is assigned to the specified parameter directly; incremental assignment refers that the input number plus current value of the specified parameter are assigned to the parameter; incremental input and direct input box have note symbols on the left side, '=' indicates direct input, '+' indicates incremental input, and the default mode is direct assignment; to change to incremental assignment mode, press the "UP/DOWN arrow".

In [Position] menu in [Keys] interface, press the 'O' button to bring up the G-code O program number box and quickly switch to O block. Press "EOB" to confirm or press "Cancel" to return.

In [Position] menu in [Keys] interface, press '←'/'→' in "manual" mode to trim main axis speed quickly; if you press and hold the button, the speed value will increase quickly. Note that this feature is available only when the main axis has been turned on and the current speed can’t be zero; if the speed is 0, first set a non-zero number in MDI mode, and then perform the quick action.

3.3 Automatic Operation

In [Position] menu in [Keys] interface, press '↑'/'↓' in "manual" mode to adjust the magnification. In "automatic" mode, magnification is adjusted automatically. Note that if additional panel is enabled, this operation will be overwritten by the magnification of additional panel automatically, that is, the operation will be invalid. The machine tool moving according to prepared program is called as automatic operation. The automatic operation modes of KYX60A system follow:

Memory operation, MDI operation, USB disk DNC operation

3.3.1 Memory Operation

The machine tool can operate according to the program in KYX60A memory, which is called as memory operation.

The program is pre-stored in the memory. Select and load a program with the operation panel and press the “Start” key to start the automatic operation. Then, press “Pause” key to pause, press “Start” key again to resume the operation, and press “Reset” during operation to stop the program immediately.

The step of memory operation follows:

(1) Save the program in the memory (see 3.2.2 for details);

(2) Select [Prog], [File] in the menu or press [File] on the panel to enter file operation interface;

(3) Press the direction keys to move the cursor, press [EOB] to select a program and load the file into the work area;

(4) Press mode selection key [Auto] to switch to automatic mode;

(5) Press the [Start] key to run the program, and the indicator is on.

3.3.2 MDI Operation

In [Keys] interface, switch to [MDI], enter the program with keypad and make the machine tool operate according to the program. The program block isn’t saved in system memory, and can’t be preserved upon power failure. This is called as MDI operation and the step follows:

(1) Press mode selection key [Prog];

(2) Select [Keys], [MDI] in the menu to enter MDI interface;

(3) Enter program block instruction manually;

(4) Press [Start], [EOB] to start executing the program block.

3.3.3 USB disk DNC

The program read from external USB disk can operate the machine tool without saving in KYX60A memory. This operation is called as USB disk DNC operation.

The step of USB disk DNC operation follows:

Insert the USB disk;

Select [Keys], [File] in the menu to enter file operation interface;

Select USB disk and press [EOB] to enter;

Move cursor to select a file in the disk;

Press [EOB] to load the file into work area (system buffer);

Press mode selection key [Auto];

Press the [Start] key to run the program, and the indicator is on.

- **Caution**

The system won't record the USB disk path. If power failure occurs during DNC processing, the program info will be lost when the power supply is connected again.

3.3.4 Speed rate adjustment

Feeding rate

In automatic mode, press Up/Down key in [Position] interface to adjust the feeding rate; Press the key once to increase or decrease by 10% (10%-150%).

Manual rate

In manual mode, press Up/Down key in [Position] interface to adjust the manual rate; Press the key to increase or decrease by 10% (10%-150%). If you press the FF key and Up/Down key, you can adjust the fast forward rate by 10% (10%-150%).

Spindle rotation

In automatic or manual mode, press the Left/Right key to adjust the spindle rotation by 100r/min. The maximum rotation is set by the spindle parameters in the system and the minimum rotation is 16r/min. If you press and hold the key for three seconds, the value will be increased or decreased quickly.

3.3.5 Run idle

(Reserved)

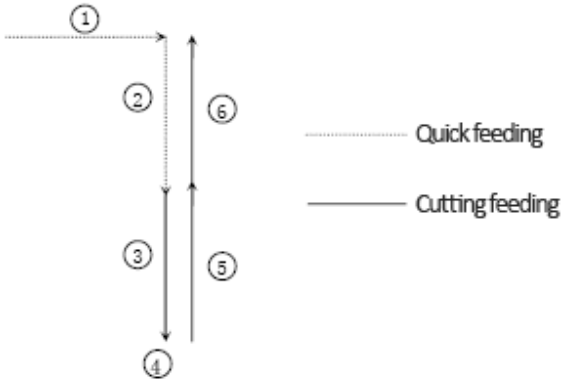
3.3.6 SBK function

In automatic mode, press [SBK] to start the SBK function. Current program block stops after executing; press [Start] again and next block stops after executing. The SBK mode allows checking the program block by block.

- **Caution:**

① In G28-G30, single block also can be stopped at the center point;

② The stop points of single block in fixed circle are ①, ②, ⑥ in the figure below; when the single blocks of ①, ② stops, the feeding pauses and the pause indicator is on.



3.3.7 BDT function

In automatic mode, press [BDT] to start the BDT function, which will make the block instructions in the line after '/' in the program invalid.

3.3.8 Stopping Automatic Operating

Two methods are available to stop automatic operating, i.e. enter stop command where the program will stop (M00, M01) and press the key on the operation panel to stop the machine tool.

Program stops

After executing the block with M00 or M01, the automatic operating stops, which is same to single block stop, and all mode information is saved. Start with KYX60A and the automatic operation can be started again.

After processing a part, the automatic operation stops.

Program ends

After executing the block with M30, the automatic operating stops, changes into reset state, and returns to program start.

Feeding pause

During automatic operation, press the [Pause] key on the operation panel, the automatic operation pauses and the indicator is on; press [Start] again to continue operating the machine tool and the pause indicator is on.

Reset

During automatic operation, press the [Reset] key on the operation panel and the system stops immediately. Here, [Reset] has the same function as emergency stop button.

3.4 Safe Operation

3.4.1 Emergency Stop

Press the emergency stop button on the machine tool, which will stop immediately, and all outputs are turned off eg main axis's runing. Rotate the button clockwise to cancel emergency stop, but all outputs must be restarted.

- **Caution:**

The power supply isn't always cut off upon emergency stop. Please refer to the electrical configuration description of the machine tool manufacturer for details;
Before releasing emergency stop, please eliminate the problems of the machine tool.

3.4.2 Hard Limit Over Travel

The system alarms if the tool touches travel switch during operation. The axis in corresponding direction can't move, and only moves in reverse direction. Before the alarm is released, the system can't enter automatic operation normally. After investigating the alarm reason, press [Reset] to clear the alarm information.

3.4.3 Soft Limit Over Travel

If the tool enters the restriction area regulated by the parameter (travel limit), the system alarms over travel, and the tool decelerates and stops. At this moment, you can move the tool to safe direction in manual mode, and then press [Reset] to release the alarm.
During automatic operation, when the tool touches an axial travel switch, the tool decelerates and stops all axial motions, and only displays one over travel alarm.
During manual operation, when the tool touches an axial travel switch, the tool only decelerates and stops motion on current axis, and still moves along other axes.
When the tool is in safe position, press [Reset] to clear the alarm. Please refer to the manual of the machine tool for details.

Both limit alarm and soft limit alarm have a deceleration stop, and therefore the sensing range of the limit should have sufficient space, or else the limit protection will be disabled due to over travel.

3.5 Fault Alarm and Self-diagnosis Function

The system has several levels, and the alarm numbers also have different type, as follows:

0~1023: G code program running alarm info

1024~2048: System environment alarm info

3.5.1 NC Program Execution Alarm

0000	:	Please reset
0001	:	Program ends
0004	:	Motion functions not specified
0005	:	G code doesn’t get line code function
0006	:	Changing tool fails
0007	:	Tool is invalid
0008	:	G block repeat error
0009	:	G block program No. error
0010	:	G7x8x complex instruction code can’t run normally
0011	:	Specified port number error
0012	:	Abnormal program termination error
0013	:	Specify M01 code program pause
0014	:	Specified program number does not exist
0015	:	M98 format error
0016	:	Motion transfer fails
0017	:	This block doesn’t need compensation
0018	:	G block invalid format
0019	:	M99 instruction transfer abnormal; M99 transfer is prohibited in current position
0020	:	Abnormal motion alarm
0021	:	Note symbol format error or no symmetric note symbol
0022	:	Illegal G code
0023	:	G code compensation number error
0024	:	Undefined G code radius compensation error
0025	:	Arc programming error
0026	:	Specify illegal plane, exceed G17, G18, G19
0027	:	M98 transfer error, may exceed the maximum value
0028	:	Spindle specifying hardware axis No. error
0029	:	M code execution error
0030	:	Specifying spindle fails
0031	:	Motion repeats request
0032	:	Specified arc doesn’t exist
0033	:	X instruction missing error

0034	:	Y instruction missing error
0035	:	Z instruction missing error
0036	:	A instruction missing error
0037	:	B instruction missing error
0038	:	C instruction missing error
0039	:	D instruction missing error
0040	:	R instruction missing error
0041	:	F instruction missing error
0042	:	T instruction missing error
0043	:	S instruction missing error
0044	:	P instruction missing error
0045	:	M instruction missing error
0046	:	G instruction missing error
0047	:	I instruction missing error
0048	:	J instruction missing error
0049	:	K instruction missing error
0050	:	Q instruction missing error
0051	:	Screw distance repeat designation error
0052	:	System alarm occurs and exits abnormally
0053		Exit through human intervention
0054		G code parameter source isn’t specified
0055		G code program No. table storage address isn’t specified
0056		Macro function calling error
0057		Macro expression writing abnormal
0058		Macro variable address error
0059		Illegal variable value
0060		Jump statement execution error
0061		Macro loop statement pairing error
0062		Macro loop statement nesting error
0063		Subroutine calling nested beyond the maximum number of layers
0064		No macro defined variable is available to get the address function
0065		User-defined information
0066		User-defined alarm

0067	Constant calling error
0068	i_gcode property value error of previous track
0069	i_gcode property value error of next track
0070	End compensation point arc enters
0071	End compensation point arc exits
0072	Start point and end point of previous track of radius compensation overlap
0073	Start point and end point of next track of radius compensation overlap
0074	Value of radius compensation is greater than R value of G0203 program
0075	Code set not supported in radius compensation
0076	Excessive NURBS node
0077	NURBS parameter error
0078	Overflow error of excessive complex program memory
0079	Complex program has expression error
0080	No U instruction error
0081	No W instruction error
0082	Multiple G code expression error

3.5.2 System Environment Alarms

1024	:	The controller isn’t reset

1. The system doesn’t perform home action after started

1025	:	A axis negative soft limit
1026	:	A axis positive soft limit
1027	:	Z axis negative soft limit
1028	:	Z axis positive soft limit
1029	:	Y axis negative soft limit
1030	:	Y axis positive soft limit
1031	:	X axis negative soft limit
1032	:	X axis positive soft limit
1033	:	A axis negative hard limit
1034	:	A axis positive hard limit
1035	:	Z axis negative hard limit
1036	:	Z axis positive hard limit
1037	:	Y axis negative hard limit

1038	:	Y axis positive hard limit
1039	:	X axis negative hard limit
1040	:	X axis positive hard limit

The system has corresponding limit alarm. Please check corresponding limit sensor point or parameters.

If hard limit occurs, and the appearance of the sensor point doesn't has any problem, enter the diagnosis mode in manual mode and check the state of the input port in diagnosis mode. If the state is valid, please eliminate in sequence. Pull out the input IO cable and check whether the sense disappears. If yes, please check the circuit. If the problem still exists, the internal optocoupler is broken. Please contact the supplier.

1041	:	Emergency stop

Emergency stop button of the handheld box interface is valid.

External emergency stop 2 input is valid; check whether IO assignment has conflict or interference.

Search for corresponding function ports in IO configuration, and then check in input diagnosis.

1042	:	Servo X drive alarm
1043	:	Servo Y drive alarm
1044	:	Servo Z drive alarm
1045	:	Servo A drive alarm

Servo alarm; if the servo doesn't alarm, parameters P2.001~004 setting and actual servo alarm level may be reverse. Please modify the parameters.

The corresponding function ports are IN34~37, which can be checked in input diagnosis.

1046	:	Axis No. definition internal repeat error

Interface axis No. set by parameters P2.45~P2.49 is specified repeatedly

1047	:	Spindle isn't reset

1048	:	Mold isn't clamped
1049	:	System safety signal not in position error

1050	:	Code relocation
1051	:	System air pressure insufficient
1052	:	System material clamping signal invalid alarm
1053		Pressure alarm of oiling machine of the system
1054		Spindle alarm
1055		Frequency converter alarm
1056		Tool release failure
1057		Tool gripping failure
1058		Tool magazine door detection error
1059		Bayonet pin detection error
1060		Tool release detection error
1061		Reserve
1062		Reserve
1063		Reserve
1064		Reserve
1065		Reserve
1066		Reserve
1067		Reserve
1068		Reserve
1069		Reserve
1070		Reserve
1071		Reserve
1072		Tool presetter limit alarm
1073		Additional panel not working
1074		Tooth number of the key isn’t set correctly
1075		Encoded string is null
1076		Encoded file can’t be called directly from the USB device
1077		Encoded file can’t be opened
1078		Encoded file is larger than 4MB and can’t be opened
1079		Unable to load the complete code file
1080		Please select the system that matches number of processing axes
….		……..
1088		Set code number does not match the tooth number

1089	Code string is not available
1090	Hole coordinates not set
1091	String number of code is up to 4
1092	Selected code number setting error
1093	Tooth number of the key isn't set correctly
1094	Invalid code symbol
1095	Left/right tooth encoding setting error
1096	X, Y-axis variable isn't set
1097	Selected tool inappropriate

3.5.3 Alarm processing

If alarm occurs, please refer to the alarm code to confirm the failure reason.

When alarm occurs, if the system isn't reset, the alarm will constantly prompt no matter whether the alarm still exists, so as to avoid the conditions that false alarm causes system suspended, but can't find the reason.

If the error is caused by data setting, modify the data, and then press [Reset] to clear the alarm info.

When alarm occurs, please remove the alarm reason. Please note that several alarms may occur at the same time. Please refer to the alarm info in the Diagnosis menu for details. When the alarms are eliminated, please press [Reset] to clear the alarm ring.

3.5.4 Self-diagnosis

CNC system sometimes stops even there is no alarm. The reason may be that the system is running processes, and the self-diagnostic function is available for checking.

The self-diagnostic steps of the system are as follows:

In the main menu, press [Diagnosis] to enter the diagnosis screen;

Select [Input] to enter the input diagnosis screen, or select [Output] to enter the output diagnosis screen;

Output diagnosis: Press the direction keys select the output port, and press [EOB] to switch the output level corresponding to the output port;

Input diagnosis: When certain input signal is valid, the corresponding screen flashes.

3.6 Program Saving & Editing

3.6.1 Saving the Program in the Memory

- **Keypad Input (New Program)**

  Create new program in the memory with the keypad, and the step follows:

  In the main menu, press [Prog] to enter program edit interface;

  Press [File] to enter file operation interface;

  Select [New] to create a new file;

  Enter the file name and press [EOB] to confirm and create a new program in current directory in the memory, and load into the system by default;

  Select [Close] to exit [prog] interface;

  In edit mode, enter the program content;

  After editing all programs, press [Reset] to save the edited programs into the system memory.

- **PC Serial Port Input**

  The step of transmitting files to controller through PC follows:

  Set system baud rate and ID No.;

  Connect to PC and run Adtech serial communication software;

  Set the baud rate same as controller, and scan ID device;

  Select the [Upload file to NC] button in the communication software;

  Select CNC file in the popup dialog box and press [Open] button.

- **Copying Processing Files from USB Disk**

  The step of copying CNC processing file to system memory through USB disk follows:

  In the main menu, press [Prog] to enter program edit interface;

  Select [File] to enter file operation interface;

  Select USB disk and press [EOB] to enter;

  Move the cursor to select a CNC file and then select [Copy];

  Return to the root directory, locate the PROG directory in disk D, enter the directory, and select [Paste] to complete copying.

3.6.2 Reading Programs into Work Area

- **Reading Programs from Controller into Work Area**

  The step of loading files from system memory into work area follows:

  Press [File] to enter file operation interface;

Select desired program, which is in PROG directory in disk D by default, press [EOB] to enter subdirectory, or press [Cancel] to exit;

Move cursor to select desired program, press [EOB] to confirm and load the program.

- **Reading Programs from USB Disk into Work Area**

The step of loading files from USB disk into work area follows:

Insert the USB disk;

Press [File] to enter file operation interface;

Select USB disk, move cursor to select a file in the disk, and press [EOB] to load the file.

3.6.3 Editing & Modifying Programs

The program in KYX60A memory can be edited with NC keypad. In the main menu, press [Prog] to enter program edit interface and edit the program in current work area (for loading program into work area, refer to 8.2). The edit mode is similar to notepad in Windows. Move the cursor directly to locate, press keys to enter, press [EOB] to change line, and press [Delete] to delete the character where the cursor locates. [Cancel] is used to delete the previous character.

- **Caution**

After all operations, press Reset to save the files, and the functions base on edit mode;
KYX60A uses new file mapping technology, and allows loading processing files that exceed its memory. Therefore, to ensure the system efficiency, you can only search and process, but can't edit the processing files that exceed 2MB.

3.6.4 Deleting Programs

Deleting Programs in Memory

Follow the step below to delete the programs in system memory:

Press [File] to enter file operation interface;

Follow the prompt on the screen, select the file and press [Delete] to confirm and delete the file.

- **Caution**

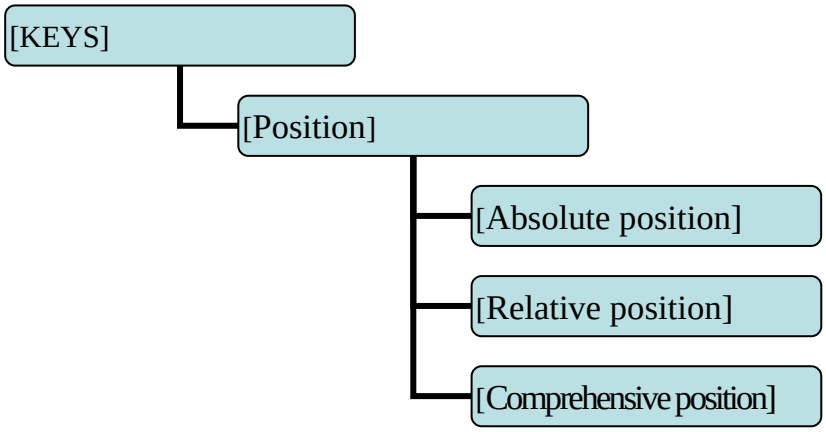
If the program has been loaded into work area, you need to restart the system to delete the program, or else the system will report error.
The programs loaded into the work area can't be deleted, or else the system will report error.

3.7 Main Interfaces of the System

3.7.1 Position Interface

The position interface shows current machine tool coordinates, including absolute position, relative position and comprehensive position. In the main interface, press [KEYS]enter the position interface.

To enter position interface:



Absolute position

The position of current machine tool coordinates relative to the origin of workpiece coordinate system

The absolute position interface follows:

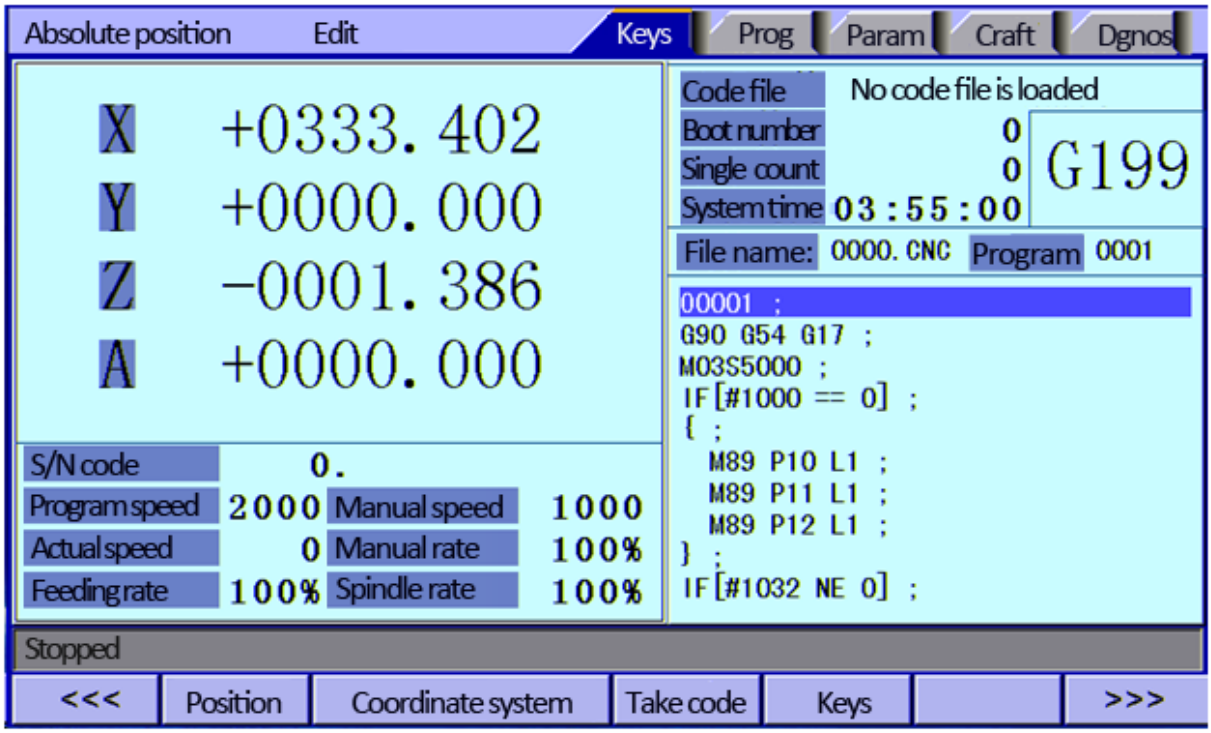


Fig. 3.7.1.1 Absolute Position Interface

Relative position

In manual mode, reset current coordinates to check the relative motion distance of any displacement, and thus it is called as relative position.

This interface is usually used for early tool setting. Considering that some operators have been used to manual calculation, this function is preserved. With the more and more powerful of automatic centered function, it is used less.

The operation follows:

Enter [Position] interface;

Switch to [Relative] interface;

Then, enter manual mode;

Press a coordinate axis No., e.g., 'X', and the X coordinate flashes;

Press "Cancel" to reset X coordinate to 0;

The relative position interface follows:

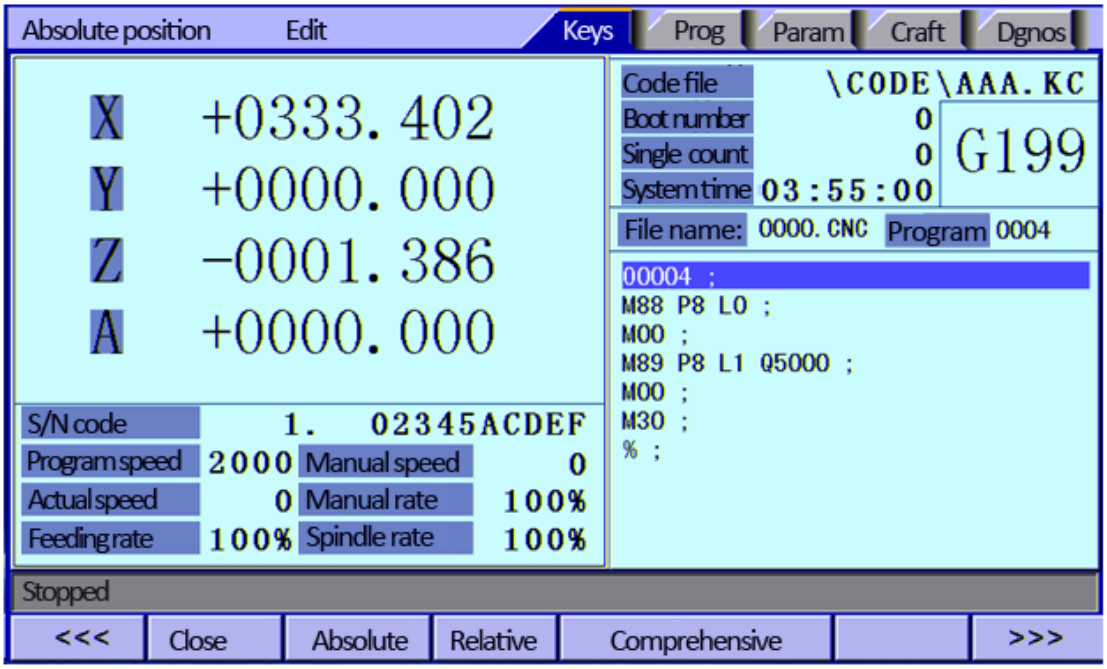


Fig. 3.7.1.2 Relative Position Interface

Comprehensive coordinates

The interface displayed by absolute coordinates and machine tool coordinates

Comprehensive position interface is shown below:

Fig. 3.7.1.3 Comprehensive Position Interface

Workpiece coordinate system

Display workpiece coordinate system, i.e. the offset of workpiece home position and machine tool home position, Totally six basic workpiece coordinate systems (G54~G59) and nine extension coordinate systems (G591~G599) are available.

The workpiece coordinate system interface is shown below:

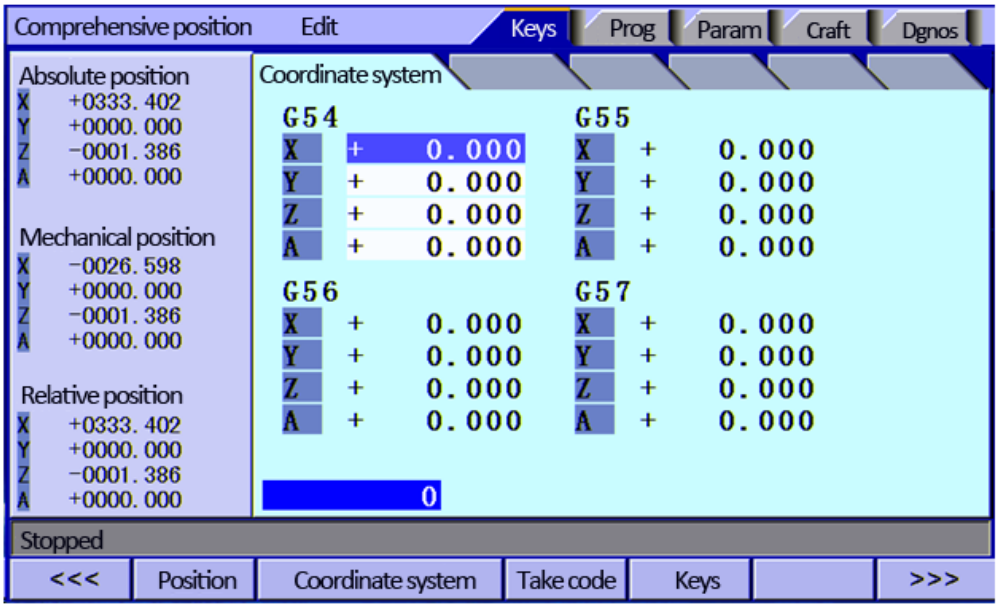
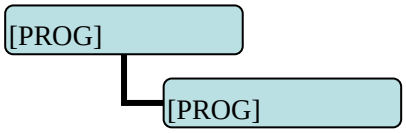


Fig. 3.7.1.4 Workpiece Coordinate System Setting Interface

3.7.2 Programming Interface

The program edit interface shows the program info in current workpiece. In the main interface, press [PROG] to enter the program interface.

To enter program edit interface:



Program edit

The program edit interface shows the NC program currently processed; in edit mode, you can edit the NC program.

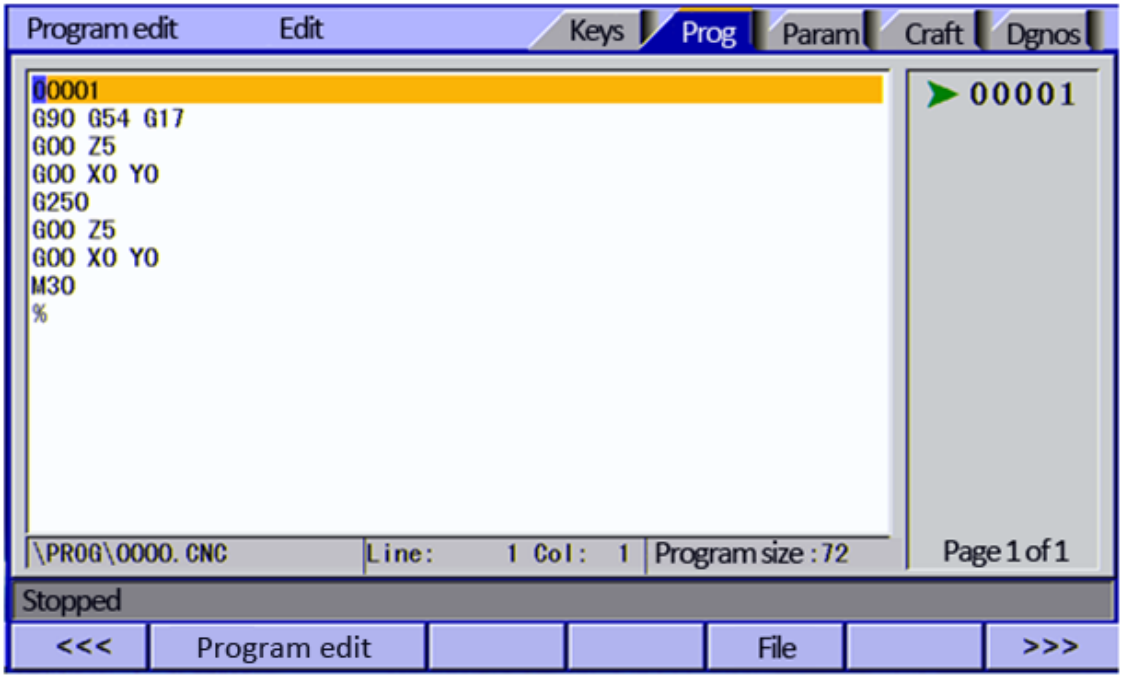


Fig. 3.7.2.1 Program Edit Interface

System info interface

The system info is a summary of the program blocks in current processing area, and calculates the resource usage in current work area. The upper right of the program directory interface shows the version info of current controller software. If our engineering personnel ask to confirm the software version of the controller on site, please provide this version info.

To enter system info interface:

[Diagnosis]

[System Info]

System info interface is shown below:

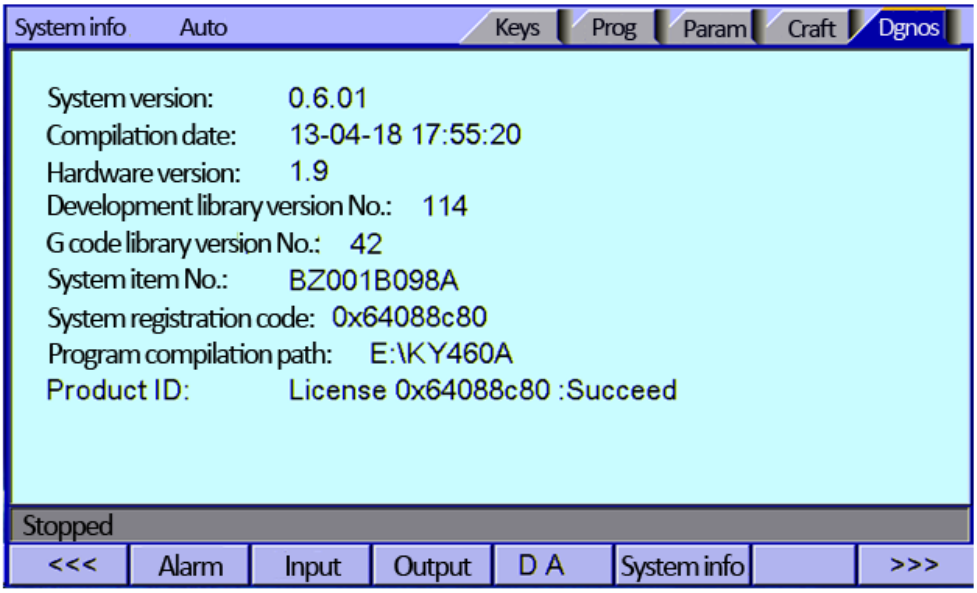


Fig. 3.7.2.2 System Info Interface

3.7.3 MDI interface

MDI mode is mainly used for the execution of single G code in certain occasions.

To enter MDI interface:

[KEYS]

[MDI]

In MDI interface, enter complete NC code instruction in edit mode, press the [Start] key in the edit mode and confirm to execute directly.

To restore the default settings quickly, press and hold the [Reset] key for three seconds and choose to reset or not. MDI interaction interface is shown below:

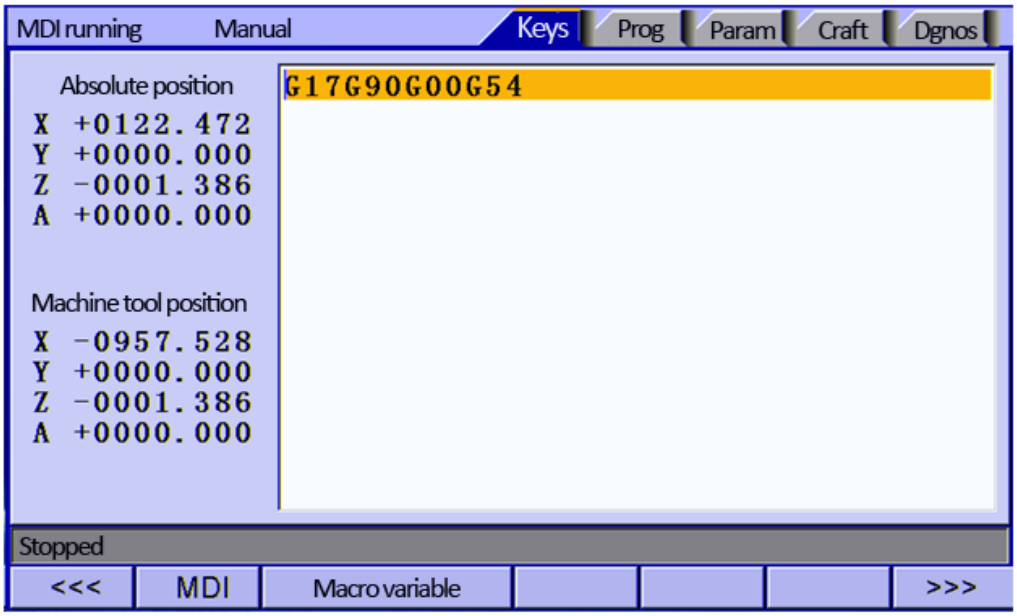
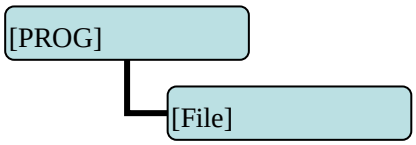


Fig. 3.7.3 MDI Interface

3.7.4 File Management

In the file management interface, you can manage the system files.

To enter file management interface:



File management mainly has the following functions:

Connect the UBS disk, and copy the files between USB disk and electronic disk;

Upgrade system software: Copy the upgrade file to system memory in either method above to upgrade the software;

Restart the controller. In [File Management] interface, press the Reset key to restart the controller. This method is different from restarting due to power failure. In certain occasions, you can restart the controller quickly in this method to make certain function take effect.

Connect to PC with the USB cable, and exchange the data between USB disk and PC.

File operation interface is shown below:

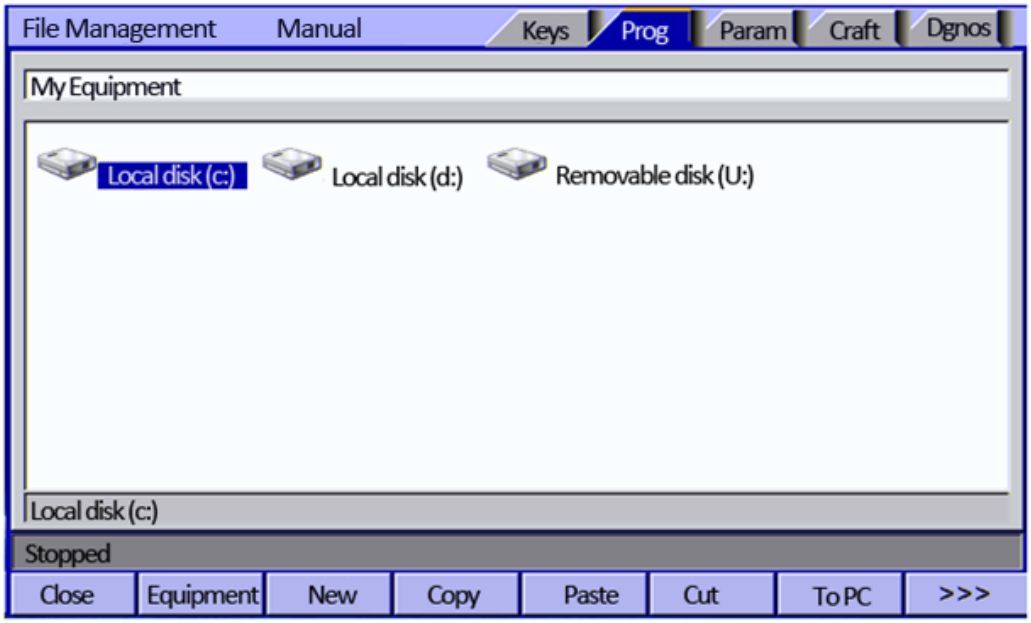
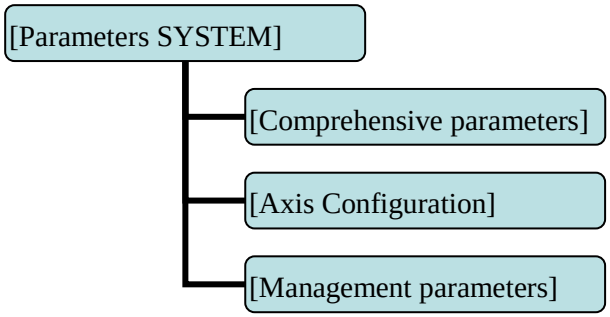


Fig. 3.7.4 File Operation Interface

3.7.5 Parameters Interface

The parameter interface shows system parameter info, including comprehensive, axis parameter, management, tool magazine, spindle, port, etc. In the main interface, press [parameter] to enter the interface.

Parameter has the following menus:



Comprehensive parameters

Comprehensive parameters are a set of functions that aren't classified in details, e.g. home mode, manual speed, etc.

Comprehensive parameters interface is shown below:

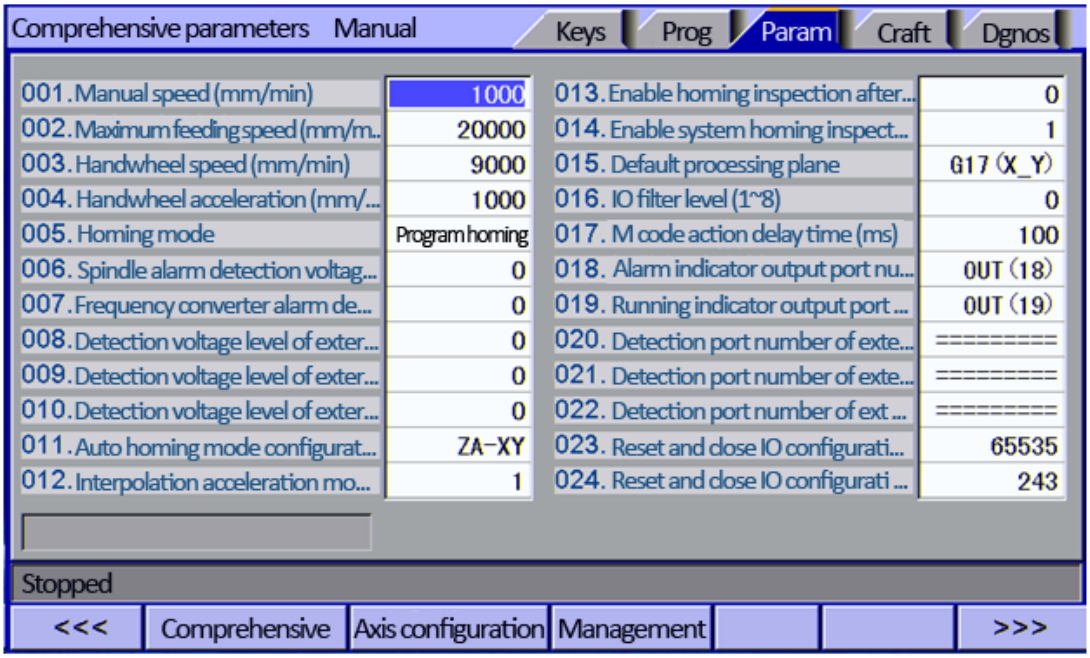


Fig. 3.7.5.1 Comprehensive parameters Interface

Axis parameters

Axis parameters are parameter set of interface characteristics of control position axis. Please refer to the parameters description for details.

Axis parameters interface is shown below:

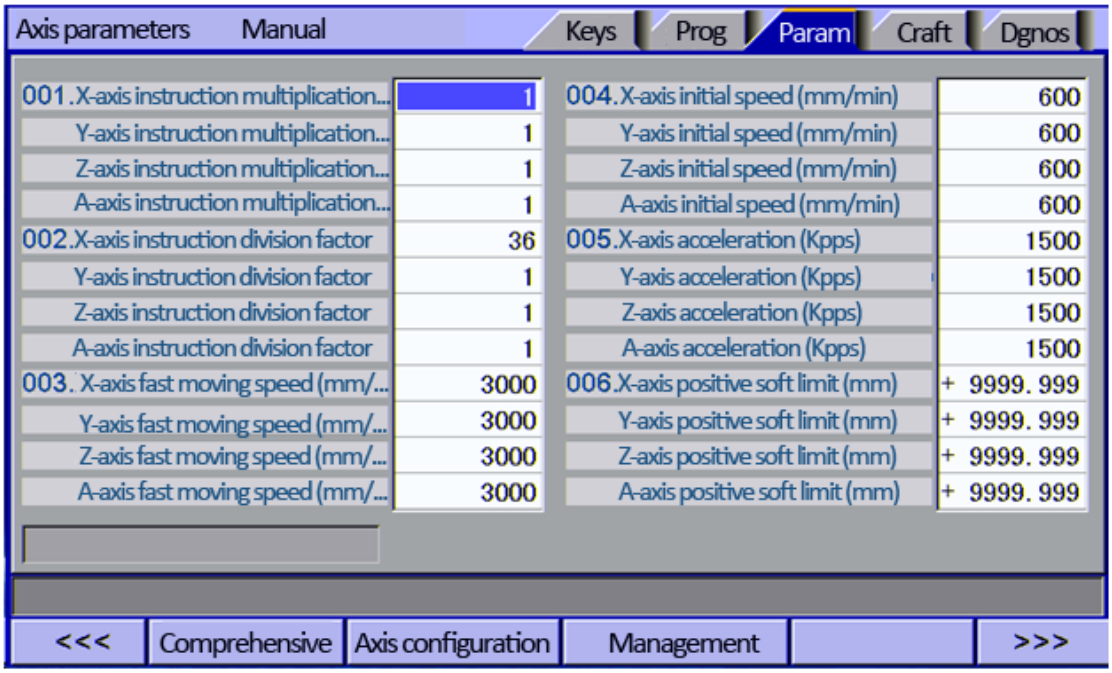


Fig. 3.7.5.2 Axis parameters Interface

Management parameters

This is a function set that confirms identity and initialize the system.

Management parameters r interface is shown below:

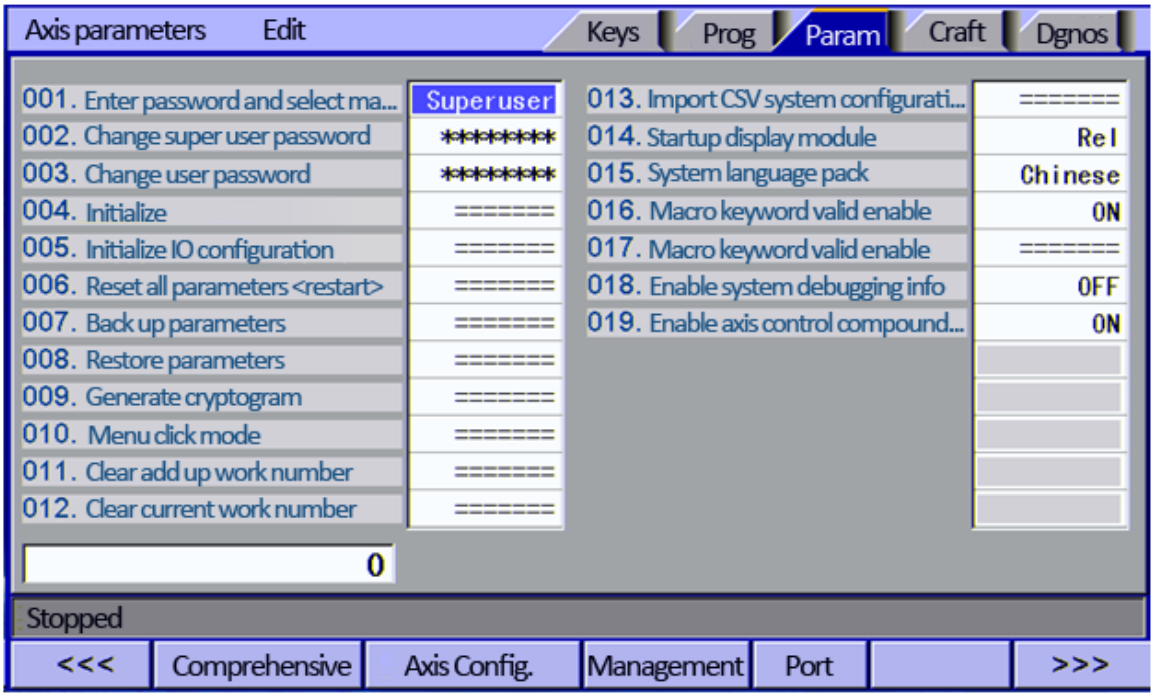
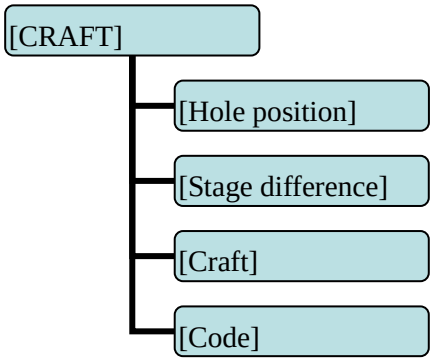


Fig. 3.7.5.3 Management Parameters Interface

3.7.6 Craft interface

The coordinates interface displays the coordinate information of the system, including hole position, stage difference, craft and code; in the main interface, press the [Craft] button to enter the interface.

To enter craft interface:



Hole position interface

Hole position is mainly used input coordinate data of the key, and customize other parameters of the key. The hole position interface follows:

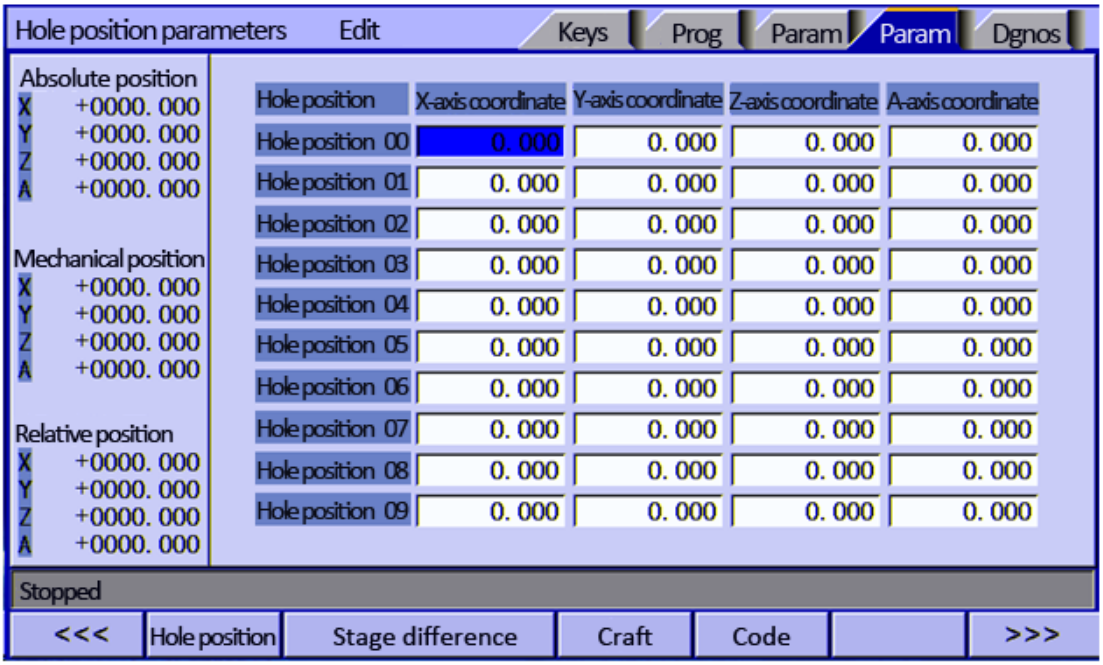


Fig. 3.7.6.1 Hole position Interface

Stage difference interface

Stage difference is mainly used to set the stage difference size (or tooth size) of keys. The types of size increase from 9 (0~9) to 36 (0~9 & A~Z), which can meet different customer requirements on size. The hole position screen is shown below:

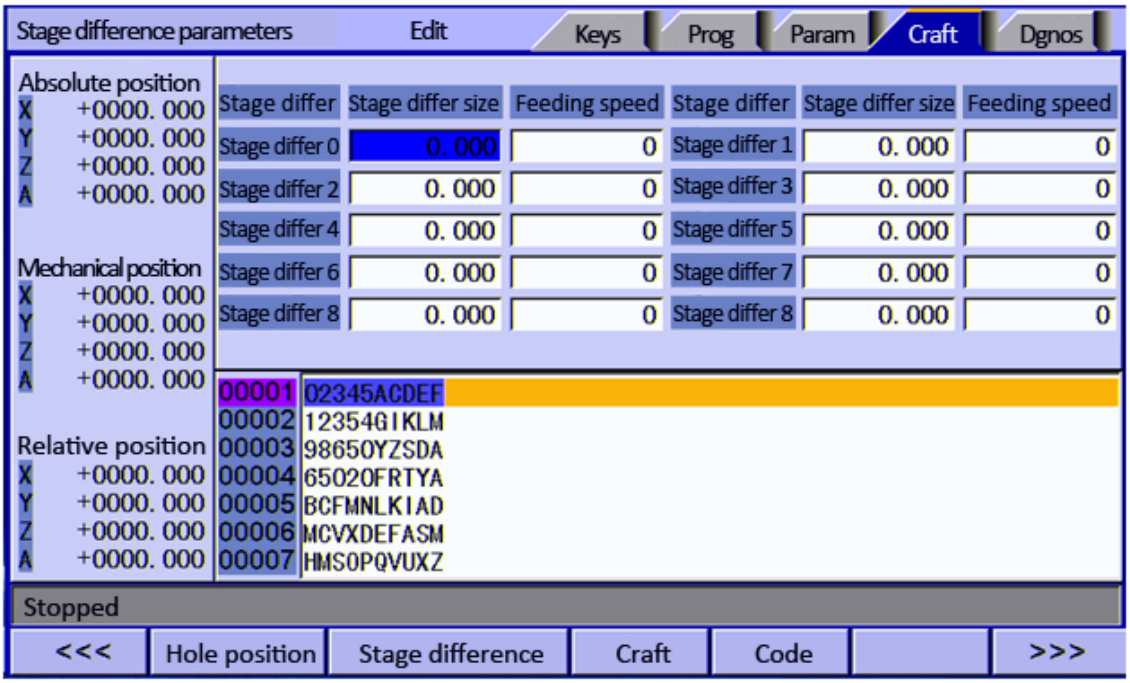


Fig. 3.7.6.2 Stage Difference Interface

Craft interface

The craft interface is mainly used to set up the parameters related to key processing, such as number of key codes, safe height of Z-axis and A-axis, processing key instructions, etc. The hole position interface is shown below:

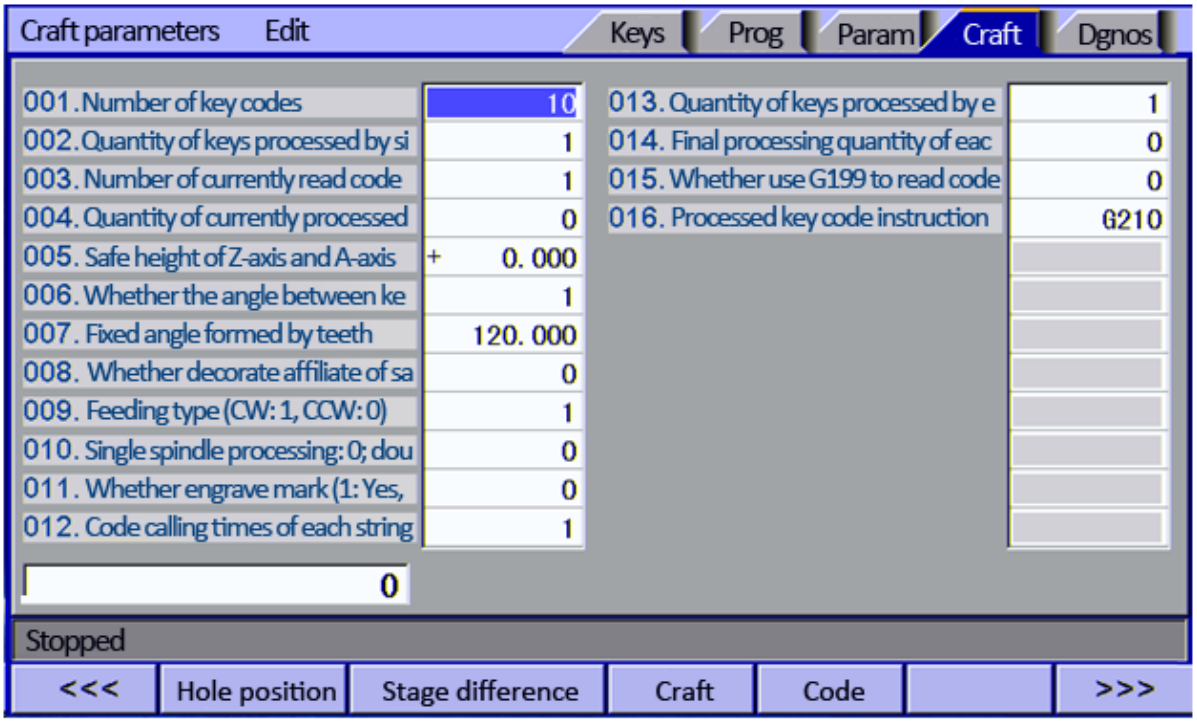


Fig. 3.7.6.3 Hole Position interface

Code interface

The code section mainly sets up parameters related to the key code generation. Below are the detailed description and steps of code generation:

Before generating code, check if the digits corresponding to the tooth codes to be generated have requirements for coding sets; if yes, please enter the code set of relevant hole position in [Craft] -> [Stage difference]; if not, clear all the code sets corresponding to each hole position, and then enter [Craft] -> [Code] to generate tooth code through the wizard.

Code sets refer to those appear in fixed hole position. For example: to generate a key teeth code file with 10 tooth positions, there are five tooth numbers already and tooth position 1 must be 2#, 3#, 4# or 5# tooth, and tooth position 10 must be 1# tooth, then hole position 1 and 10 should be set, and other holes are set to 0. Hole 1 should be set to "2345" or "3452" (as long as it is the set and there is no requirement on the order); hole 2 should be set to "1"; other holes should be set to "0" (0 can be any tooth number).

Enter code generation interface, and the system will first prompt "Confirm to generate tooth code through the wizard"; press [Cancel] to cancel, or press [EOB] to confirm and enter the wizard interface as follows:

Step 1: Set the category of key: 1: unilateral; 2: bilateral

Set whether the generated code is used for bilateral keys or unilateral keys. This option mustn't be null. Enter the corresponding value in the dialog box, and press [EOB] to enter the next step or press [Cancel] to return to the previous step.

Step 2: Set the total coding digits

Enter the corresponding code digits. The value ranges from 0 to 40 and mustn't be null. Enter 10 because there are 10 teeth totally in above example.

Step 3: Set the maximum limit of same tooth number on single side

Set each side of the key to the maximum limit of the same teeth. The value ranges from 0 to 40. In above example, there are 10 teeth, which mean that the code has 10 digits. If you add a condition: "the maximum limit of same teeth on single side shouldn't exceed 3", then the value should be set to 2. Whether it is a unilateral or bilateral key, every individual side should meet such condition. Note: Why the number shouldn't exceed 3 but the value is set to 2? Because the key must have code in a fixed position, so the adjacent number is to use the condition 3 minus 1, i.e. 3-1 = 2. This option isn't required, in which case the condition won't be judge.

Step 4: Set the matching grade between tooth numbers

Set the upper limit of the grade to be matched between tooth numbers (position) of code. This option isn't required, in which case the condition won't be judge. This function relates to the selection of key category (unilateral or bilateral) in the first step. If unilateral is selected, this function indicates that the adjacent teeth must meet the current level of matching. If there are five tooth numbers in above sample, unilateral key is selected and this option is set to 2, the matching level between adjacent teeth can't exceed 2. For example: the first tooth can be adjacent to the first, second or third type only; the second tooth can be adjacent to the first, second, third or fourth type only. If bilateral key is selected, this function indicates that the same teeth on both sides of the code must meet the current level of matching. If there are five tooth numbers known and 10 codes, bilateral key is selected and this value is set to 2, the matching level between adjacent teeth can't exceed 2, that is, the matching level between 1st tooth and 6th tooth, 2nd tooth and 7th tooth... 5^{th} tooth and 10th tooth can't exceed 2. If the tooth number of position 1 is 2#, the tooth number of position 6 only can be 1#, 2#, 3# and 4#.

Step 5: Set the least different digits between any two strings of codes

Set the least different digits between any two strings of codes in generated code file. This option should be determined according to current key type and production amount. The setting will affect the number of ultimately generated code categories and the default is 1.

Step 6: Set the types of tooth numbers

Set the total number of teeth. It is known from the above example that there are five types of tooth numbers, thus enter 5 for this option; this option can't be null.

Step 7: Set the total code strings to be generated

Set the total number of strings to be generated; this option can't be null.

Step 8: Set the absolute value of minimum stage difference of unilateral teeth number

Set the minimum stage difference that should be met by unilateral tooth number of generated code file (unilateral key) or unilateral tooth number of bilateral tooth number (bilateral key). If this value is 2, the codes of above example such as "1212121212" and "3233233232" do not meet the requirements, while the codes such as "1212121213", "3433233232" and "2541232142" meet the requirements. This option isn't required, in which case this condition does not take effect.

Step 9: Selectable code groups are **; please enter the code group**

This option is used to select the code group to be generated; the range shouldn't be greater than the total groups.

Step 10: Enter the file name

Enter the file name to be saved, then press [EOB] to complete code generation wizard and enter code generation interface or press [Cancel] to return to the previous step.

After code generating, the system will automatically go to the "File Operation" interface, and you can read the generated ******.KC code file into the work area through the file operation function.

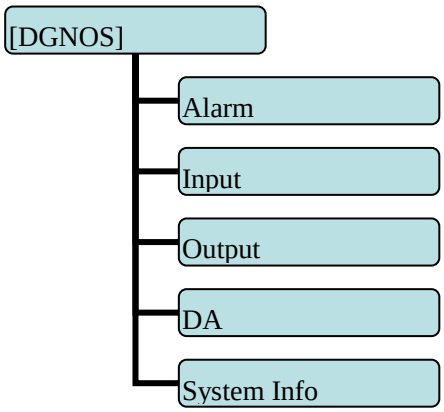
Note: The enhanced version of tooth code generation function is constantly in development. The code generation method here is not necessarily the latest.

Fig. 3.7.6.4 Code Generation interface

3.7.7 Controller Diagnosis Interface (Diagnosis)

The diagnosis interface is used to display the hardware interfaces and system info, including alarm, input, output, DA diagnosis; press [Diagnosis] to enter the diagnosis interface.

The diagnosis interface follows:



Alarm interface

Display the alarm of the system after power on, including 15 alarm records.

IO diagnosis interface

IO diagnosis allows entering at any moment. You can check current IO state of the system. In manual mode, press the direction keys to select corresponding IO, and press EOB to control the output manually.

DA diagnosis interface

Correct the output voltage of two lines DA voltage module for testing; press the direction keys to output corresponding voltage directly, input the actually measured voltage to corresponding gear position; when transferring control instructions of spindle, the system will correct according to correction value.

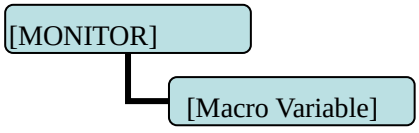
System info

The system info shows basic information of current system, and is used to mark current software version, hardware version, upgrade info, etc. When the system upgrades, the corresponding item number can be checked in this part to prevent mismatch between program update and the system, which will cause each axis of the machine tool cannot move.

3.7.8 Macro Variable View Interface (Macro Variable)

This is the variable register view menu of macro function. In this menu, you can turn pages to view the macro variables, or enter values to variable register directly in edit mode.

To enter macro variable view interface:



The macro variable menu has eight levels, as below:

Local variable

#100~#199

#500~#599

#600~#699

#700~#799

#800~#899

#900~#999

User variable

In the variable interfaces of different levels, you can check the corresponding variable number. Local variable has five levels totally, and shows the variables of current working layer by default. To view a specific layer, please enter local variable submenu, and then select according to layers.

Process variables are to customize the names of 20 variables (#100~#999) according to CSV configuration table, so that the variable names have visual meanings. In programs, the user customized variables are transferred with variable number.

3.8 System Maintenance

3.8.1 Restart

- **Enter restart from program edit interface**

(1) In the main menu, press [Prog] to enter the program interface

(2) Press [File] to enter the file interface

(3) Press [Reset] and the system asks whether restart or not

(4) Press [OK] to restart the system.

- **Press [File] shortcut to enter restart**

(1) Press [File] to enter the file interface

(2) Press [Reset] and the system asks whether restart or not

(3) Press [OK] to restart the system.

- **Turn off the power and restart**

3.8.2 System Upgrade

- **Copying upgrade program with USB disk**

(1) In the main menu, press [Prog] to enter the program interface

(2) Press [File] to enter the file interface

(3) Insert the USB disk, open the disk in the root directory and enter directly

(4) Move cursor to the upgrade file ADTROM.BIN, select [Copy], enter ADT directory in disk D and paste;

(5) Select the second upgrade file NC_RES.NC; skip this step if the file doesn't exist. Also select Copy, enter disk D, and paste in directory ADT.

(6) If the ADT directory has KEY directory and LOGO.BMP, it is not necessary to operate these two files. If not, select and copy KEY directory and LOGO.BMP, enter D disk, and paste in the ADT directory.

(7) If the CODE directory in D disk has files associated with key processing, it is not necessary to operate this part. If not, select CODE directory in USB disk, enter D disk, and paste.

(8) Restart the system after upgrade, press and hold the [Cancel] button for three seconds when the system is starting, the system pops up the interface asking the user to enter the password: "Input Pass:", enter the password "26722719", and the system enter the BIOS screen, press [4] to select "4 --- Start-up mode"; if USB disk is selected, press the [Cancel] button, and then press [.] and [9] successively to restart the system and make the program in disk D take effect. If normal start is selected, press the [Cancel] button, and then press [1] and select "1 --- BIOS Settings", then press [1] and select "1 --- Program area", press [Y] according to the prompt, then press [Cancel], and then press [.] and [9] successively to restart the system and make the program in disk D take effect.

(9) Enter System Info in Diagnosis menu to view the system version and compilation date, and check whether the upgrade is successful.

- **Enter BIOS and upgrade system via USB disk**

This method is not commonly used

- **Connect to the control system via PC and upgrade the system**

(1) Connect the equipped data cable to "XS11 USB" interface of the control system, and insert the USB interface of data cable into the USB port of the computer

(2) In the main menu, press [Prog] to enter the program interface;

(3) Press [File] to enter the file management interface;

(4) Insert the USB disk, open the disk in the root directory and enter directly

(5) Move cursor to the upgrade file ADTROM.BIN, select [Copy], enter ADT directory in disk D and paste;

(6) Select the second upgrade file NC_RES.NC; skip this step if the file doesn't exist. Also select Copy, enter disk D, and paste in directory ADT.

(7) If the ADT directory has KEY directory and LOGO.BMP, it is not necessary to operate these two files. If not, select and copy KEY directory and LOGO.BMP, enter D disk, and paste in the ADT directory.

(8) If the CODE directory in D disk has files associated with key processing, it is not necessary to operate this part. If not, select CODE directory in USB disk, enter D disk, and paste.

(9) After upgrade, press [Cancel] to disconnect the communication between the control system and PC; restart the system, press and hold the [Cancel] button for three seconds when the system is starting, the system pops up the interface asking the user to enter the password: “Input Pass:”, enter the password "26722719", and the system enter the BIOS screen, press [4] to select "4 --- Start-up mode"; if USB disk is selected, press the [Cancel] button, and then press [.] and [9] successively to restart the system and make the program in disk D take effect. If normal start is selected, press the [Cancel] button, and then press [1] and select "1 --- BIOS Settings", then press [1] and select "1 --- Program area", press [Y] according to the prompt, then press [Cancel], and then press [.] and [9] successively to restart the system and make the program in disk D take effect.

(10) Enter System Info in Diagnosis menu to view the system version and compilation date, and check whether the upgrade is successful.

- **Caution:**

The update of KYX60A system is different from KYX00 system. KYX60A system should be updated according to corresponding item number, or else the registration number of the system will not match, and the interface display is proper, but X-axis, Y-axis, Z-axis, A-axis of the system do not move on the machine tool. The item number and registration number of KYX60A are as follows:

Item No.	Model	Number of axes	Hardware platform	Registration No.
BZ001B098A	KY260A	2	CNC4620	0X64088c40
BZ001B099A	KY360A	3	CNC4640	0X64088c60
BZ001B100A	KY460A	4	CNC4640	0X64088c40

After the update, check the last line in [System Info] of [Diagnosis] if ID is registered successfully; if yes, the registration is successful; if not, the registration failed.

3.8.3 Reset

(1) Select the edit mode;

(2) In the main menu, press [Parameters] to enter the parameter interface;

(3) Press [Management] key to enter management parameters interface;

(4) Move cursor to “006 Reset all parameters”;

(5) Press [EOB], the system confirms, restores the default parameters and restarts automatically.

3.8.4 Parameter Backup & Restore

(1) Select the edit mode;

(2) In the main menu, press [Parameters] key to enter the parameter interface;

(3) Press the [Management] key to enter management parameters interface;

(4) Move the cursor to 007 or 008 and select appropriate menu;

(5) Press the [EOB] key, the system confirms and performs backup and restore operation;

(6) The backup operation will generate the SYSCONF.BAK file in the root directory of disk D. Please save this file for backup in the future.

(7) For restore operation, also save the SYSCONF.BAK file in the root directory of disk D. The system will recognize this file automatically in the process of restoring.

3.8.5 Entering BISO Interface

(1) If the system can't be started due to irreversible error, you can enter the BIOS to upgrade and maintain the program;

(2) To enter BIOS: after the controller is turned on and before the application is turned on, press and hold the [Cancel] button for three seconds, and the system enters a blue background screen; if the BIOS has password, it will pop up a prompt: "Input Pass:", enter the password "26722719" and enter BIOS

(3) Enter the BIOS to perform the maintenance: Format C, D disk, copy USB disk files and upgrade, initialize parameters;

3.9 System Parameters

According to occasions and functions, the parameters contain comprehensive parameters, IO configuration parameters, management parameters and coordinate setting parameters.

Comprehensive parameters are complete, and contain basic operation and usage settings of the controller, including spindle, handwheel, home, etc.;

IO configuration parameters are mainly used for machine installation and test, adapting to the interface characteristics of machine tool and motor drive;

Coordinate setting parameters are tool setting configuration in [Coordinate] interface;

(1) It is required to confirm user identity to modify the parameter table. The controller has two levels of user authority, which are super user and operator; super user can modify all parameters and user passwords; while operator only can operate the parameters that require modification, and modify the operator password; in P3.1 of management parameters, the system enters the corresponding mode automatically according to the entered password.

(2) According to the application, the parameters will take effect immediately or after restarted; the parameters that require restart are marked with <Restart>.

(3) Certain parameters are set in binary system (parameter descriptor has bit symbol); the conversion between binary system and decimal system follows:

Bit0: Set to 1 to correspond to decimal 1;

Bit1: Set to 1 to correspond to decimal 2;

Bit2: Set to 1 to correspond to decimal 4;

Bit3: Set to 1 to correspond to decimal 8;

Bit4: Set to 1 to correspond to decimal 16;

Bit5: Set to 1 to correspond to decimal 32;

Bit6: Set to 1 to correspond to decimal 64;

Bit7: Set to 1 to correspond to decimal 128;

For more bits, multiply the decimal system corresponding to binary system of previous position by 2. If only the corresponding bit is 1, accumulate the numbers of corresponding decimal system according to the comparison table to get the setting value.

For example: set Bit0, Bit1 and Bit5 to 1, and the parameter will be 1+2+32=35.

3.9.1 Parameters Index List

Parameters type	S/N	Description	Effective mode	Default value	Page
General parameter (P1.)	001	Manual speed (mm/min)	Instant	1000	
General parameter (P1.)	002	Maximum feeding speed (mm/min)	Instant	20000	
General parameter (P1.)	003	Hndwheel speed (mm/min)	Instant	9000	
General parameter (P1.)	004	Hndwheel acceleration (mm/sec)	Instant	1000	
General parameter (P1.)	005	Home mode	Instant	Program homing	
General parameter (P1.)	006	Spindle alarm detection level	Instant	0	
General parameter (P1.)	007	Inverter alarm detection level	Instant	0	
General	008	External start 2 test voltage	Instant	0	

Parameters type	S/N	Description	Effective mode	Default value	Page
parameter (P1.)		level			
General parameter (P1.)	009	External pause 2 test voltage level	Instant	0	
General parameter (P1.)	010	External emergency stop 2 test voltage level	Instant	0	
General parameter (P1.)	011	Automatic home mode configuration (bit)	Instant	ZA-XY	
General parameter (P1.)	012	Interpolation acceleration mode	Instant	1	
General parameter (P1.)	013	Home test enable after alarm	Instant	0	
General parameter (P1.)	014	System home test enable	Instant	1	
General parameter (P1.)	015	Default processing plane	Instant	G17(X_Y)	
General parameter (P1.)	016	IO filter level (1~8)	Instant	0	
General parameter (P1.)	017	M code action delay time (ms)	Instant	100	
General parameter (P1.)	018	Warning lamp output port number	Instant	OUT18	
General parameter (P1.)	019	Running lamp output port number	Instant	OUT19	
General parameter (P1.)	020	External start 2 test port No.	Instant	=======	
General parameter (P1.)	021	External pause 2 test port No.	Instant	=======	
General parameter (P1.)	022	External emergency stop 2 test port No.	Instant	=======	
General parameter (P1.)	023	Reset off IO configuration 00~15	Instant	65535	
General parameter (P1.)	024	Reset off IO configuration 16~23	Instant	243	

Parameters type	S/N	Description	Effective mode	Default value	Page
General parameter (P1.)	025	Lubricant output port No.	Instant	OUT5	
General parameter (P1.)	026	Cooling output port number	Instant	OUT4	
General parameter (P1.)	027	Spindle CW output port number	Instant	OUT0	
General parameter (P1.)	028	Spindle CCW output port number	Instant	OUT1	
General parameter (P1.)	029	Spindle alarm detection port number	Instant	=======	
General parameter (P1.)	030	Inverter alarm detection port number	Instant	=======	
General parameter (P1.)	031	Output port number of variable frequency level 0	Instant	OUT23	
General parameter (P1.)	032	Output port number of variable frequency level 1	Instant	OUT22	
General parameter (P1.)	033	Output port number of variable frequency level 2	Instant	OUT21	
General parameter (P1.)	034	Output port number of variable frequency level 3	Instant	OUT20	
General parameter (P1.)	035	Line number increment	Instant	0	
General parameter (P1.)	036	System baud rate	Instant	115200	
General parameter (P1.)	037	Controller ID No.	Instant	1	
General parameter (P1.)	038	G73(M) loop tool retraction amount (mm)	Instant	2	
General parameter (P1.)	039	G83(M) loop tool retraction amount (mm)	Instant	2	
General parameter (P1.)	040	Arc interpolation feed amount (mm)	Instant	0.02	
General	041	Arc interpolation acceleration	Instant	100	

Parameters type	S/N	Description	Effective mode	Default value	Page
parameter (P1.)					
General parameter (P1.)	042	Interpolation speed mode	Instant	0	
General parameter (P1.)	043	Code pretreatment mode	Instant	0	
General parameter (P1.)	044	Variable frequency analog control mode	Instant	0	
Axis parameter (P2.)	001	X axis instruction frequency multiplication ratio	Instant	1	
Axis parameter (P2.)		Y axis instruction frequency multiplication ratio	Instant	1	
Axis parameter (P2.)		Z axis instruction frequency multiplication ratio	Instant	1	
Axis parameter (P2.)		A axis instruction frequency multiplication ratio	Instant	1	
Axis parameter (P2.)	002	X axis instruction frequency division coefficient	Instant	1	
Axis parameter (P2.)		Y axis instruction frequency division coefficient	Instant	1	
Axis parameter (P2.)		Z axis instruction frequency division coefficient	Instant	1	
Axis parameter (P2.)		A axis instruction frequency division coefficient	Instant	1	
Axis parameter (P2.)	003	X axis rapid traverse rate(mm/min)	Instant	6000	
Axis parameter (P2.)		Y axis rapid traverse rate(mm/min)	Instant	6000	
Axis parameter (P2.)		Z axis rapid traverse rate(mm/min)	Instant	6000	
Axis parameter (P2.)		A axis rapid traverse rate(mm/min)	Instant	6000	
Axis parameter	004	X axis start rate (mm/min)	Instant	600	

Parameters type	S/N	Description	Effective mode	Default value	Page
(P2.)					
Axis parameter (P2.)		Y axis start rate (mm/min)	Instant	600	
Axis parameter (P2.)		Z axis start rate (mm/min)	Instant	600	
Axis parameter (P2.)		A axis start rate (mm/min)	Instant	600	
Axis parameter (P2.)	005	X axis acceleration(Kpps)	Instant	1500	
Axis parameter (P2.)		Y axis acceleration(Kpps)	Instant	1500	
Axis parameter (P2.)		Z axis acceleration(Kpps)	Instant	1500	
Axis parameter (P2.)		A axis acceleration(Kpps)	Instant	1500	
Axis parameter (P2.)	006	X axis positive soft limit (mm)	Instant	+9999.999	
Axis parameter (P2.)		Y axis positive soft limit (mm)	Instant	+9999.999	
Axis parameter (P2.)		Z axis positive soft limit (mm)	Instant	+9999.999	
Axis parameter (P2.)		A axis positive soft limit (mm)	Instant	+9999.999	
Axis parameter (P2.)	007	X axis negative soft limit (mm)	Instant	-9999.999	
Axis parameter (P2.)		Y axis negative soft limit (mm)	Instant	-9999.999	
Axis parameter (P2.)		Z axis negative soft limit (mm)	Instant	-9999.999	
Axis parameter (P2.)		A axis negative soft limit (mm)	Instant	-9999.999	
Axis parameter (P2.)	008	Feeding speed (mm/min)	Instant	3000	

Parameters type	S/N	Description	Effective mode	Default value	Page
Axis parameter (P2.)		Initial feeding speed(mm/min)	Instant	800	
Axis parameter (P2.)		Feeding acceleration (mm/sec)	Instant	1500	
Axis parameter (P2.)	009	X axis reverse clearance compensation(pulse)	Instant	0	
Axis parameter (P2.)		Y axis reverse clearance compensation(pulse)	Instant	0	
Axis parameter (P2.)		Z axis reverse clearance compensation(pulse)	Instant	0	
Axis parameter (P2.)		A axis reverse clearance compensation(pulse)	Instant	0	
Axis parameter (P2.)	010	X hardware positive limit enable	Instant	1	
Axis parameter (P2.)		Y hardware positive limit enable	Instant	1	
Axis parameter (P2.)		Z hardware positive limit enable	Instant	1	
Axis parameter (P2.)		A hardware positive limit enable	Instant	1	
Axis parameter (P2.)	011	X hardware negative limit enable	Instant	1	
Axis parameter (P2.)		Y hardware negative limit enable	Instant	1	
Axis parameter (P2.)		Z hardware negative limit enable	Instant	1	
Axis parameter (P2.)		A hardware negative limit enable	Instant	1	
Axis parameter (P2.)	012	X hard limit effective voltage level	Restart	0	
Axis parameter (P2.)		Y hard limit effective voltage level	Restart	0	
Axis parameter		Z hard limit effective	Restart	0	

Parameters type	S/N	Description	Effective mode	Default value	Page
(P2.)		voltage level			
Axis parameter (P2.)		A hard limit effective voltage level	Restart	0	
Axis parameter (P2.)	013	X axis external home enable	Instant	1	
Axis parameter (P2.)		Y axis external home enable	Instant	1	
Axis parameter (P2.)		Z axis external home enable	Instant	1	
Axis parameter (P2.)		A axis external home enable	Instant	1	
Axis parameter (P2.)	014	Effective voltage level for X external home	Instant	0	
Axis parameter (P2.)		Effective voltage level for Y external home	Instant	0	
Axis parameter (P2.)		Effective voltage level for Z external home	Instant	0	
Axis parameter (P2.)		Effective voltage level for A external home	Instant	0	
Axis parameter (P2.)	015	X axis home direction	Instant	1	
Axis parameter (P2.)		Y axis home direction	Instant	1	
Axis parameter (P2.)		Z axis home direction	Instant	1	
Axis parameter (P2.)		A axis home direction	Instant	1	
Axis parameter (P2.)	016	X axis home speed (mm/min)	Instant	1000	
Axis parameter (P2.)		Y axis home speed (mm/min)	Instant	1000	
Axis parameter (P2.)		Z axis home speed (mm/min)	Instant	1000	

Parameters type	S/N	Description	Effective mode	Default value	Page
Axis parameter (P2.)		A axis home speed (mm/min)	Instant	1000	
Axis parameter (P2.)	017	X axis home deceleration speed	Instant	100	
Axis parameter (P2.)		Y axis home deceleration speed	Instant	100	
Axis parameter (P2.)		Z axis home deceleration speed	Instant	100	
Axis parameter (P2.)		A axis home deceleration speed	Instant	100	
Axis parameter (P2.)	018	X axis home scanning speed	Instant	60	
Axis parameter (P2.)		Y axis home scanning speed	Instant	60	
Axis parameter (P2.)		Z axis home scanning speed	Instant	60	
Axis parameter (P2.)		A axis home scanning speed	Instant	60	
Axis parameter (P2.)	019	X axis home offset(pulse)	Instant	0	
Axis parameter (P2.)		Y axis home offset(pulse)	Instant	0	
Axis parameter (P2.)		Z axis home offset(pulse)	Instant	0	
Axis parameter (P2.)		A axis home offset(pulse)	Instant	0	
Axis parameter (P2.)	020	Effective voltage level for servo X axis alarm	Instant	0	
Axis parameter (P2.)		Effective voltage level for servo Y axis alarm	Instant	0	
Axis parameter (P2.)		Effective voltage level for servo Z axis alarm	Instant	0	
Axis parameter		Effective voltage level for	Instant	0	

Parameters type	S/N	Description	Effective mode	Default value	Page
(P2.)		servo A axis alarm			
Axis parameter (P2.)	021	Effective voltage level for servo X axis reset	Instant	1	
Axis parameter (P2.)		Effective voltage level for servo Y axis reset	Instant	1	
Axis parameter (P2.)		Effective voltage level for servo Z axis reset	Instant	1	
Axis parameter (P2.)		Effective voltage level for servo A axis reset	Instant	1	
Axis parameter (P2.)	022	Servo X axis Z phase home enable	Instant	0	
Axis parameter (P2.)		Servo Y axis Z phase home enable	Instant	0	
Axis parameter (P2.)		Servo Z axis Z phase home enable	Instant	0	
Axis parameter (P2.)		Servo A axis Z phase home enable	Instant	0	
Axis parameter (P2.)	023	Effective voltage level for servo X axis Z phase	Instant	0	
Axis parameter (P2.)		Effective voltage level for servo Y axis Z phase	Instant	0	
Axis parameter (P2.)		Effective voltage level for servo Z axis Z phase	Instant	0	
Axis parameter (P2.)		Effective voltage level for servo A axis Z phase	Instant	0	
Axis parameter (P2.)	024	X servo home direction	Instant	0	
Axis parameter (P2.)		Y servo home direction	Instant	0	
Axis parameter (P2.)		Z servo home direction	Instant	0	
Axis parameter (P2.)		A servo home direction	Instant	0	

Parameters type	S/N	Description	Effective mode	Default value	Page
Axis parameter (P2.)	025	X pulse command format	Restart	1	
Axis parameter (P2.)		Y pulse command format	Restart	1	
Axis parameter (P2.)		Z pulse command format	Restart	1	
Axis parameter (P2.)		A pulse command format	Restart	1	
Axis parameter (P2.)	026	X pulse logic direction	Restart	1	
Axis parameter (P2.)		Y pulse logic direction	Restart	1	
Axis parameter (P2.)		Z pulse logic direction	Restart	1	
Axis parameter (P2.)		A pulse logic direction	Restart	1	
Axis parameter (P2.)	027	X axis specified interface axis No.	Restart	1	
Axis parameter (P2.)		Y axis specified interface axis No.	Restart	2	
Axis parameter (P2.)		Z axis specified interface axis No.	Restart	3	
Axis parameter (P2.)		A axis specified interface axis No.	Restart	4	
Axis parameter (P2.)		Main axis specified interface axis No.	Restart	0	
Axis parameter (P2.)	028	X axis encoder wire number (p)	Instant	2500	
Axis parameter (P2.)		Y axis encoder wire number (p)	Instant	2500	
Axis parameter (P2.)		Z axis encoder wire number (p)	Instant	2500	
Axis parameter		A axis encoder wire number	Instant	2500	

Parameters type	S/N	Description	Effective mode	Default value	Page
(P2.)		(p)			
Axis parameter (P2.)	029	X axis encoder pulse logic direction	Restart	0	
Axis parameter (P2.)		Y axis encoder pulse logic direction	Restart	0	
Axis parameter (P2.)		Z axis encoder pulse logic direction	Restart	0	
Axis parameter (P2.)		A axis encoder pulse logic direction	Restart	0	
Axis parameter (P2.)	030	X axis pulse logic voltage level	Restart	0	
Axis parameter (P2.)		Y axis pulse logic voltage level	Restart	0	
Axis parameter (P2.)		Z axis pulse logic voltage level	Restart	0	
Axis parameter (P2.)		A axis pulse logic voltage level	Restart	0	
Axis parameter (P2.)	031	X axis characteristics (rotation: 0, linear: 1)	Instant	1	
Axis parameter (P2.)		Y axis characteristics (rotation: 0, linear: 1)	Instant	1	
Axis parameter (P2.)		Z axis characteristics (rotation: 0, linear: 1)	Instant	1	
Axis parameter (P2.)		A axis characteristics (rotation: 0, linear: 1)	Instant	1	
Axis parameter (P2.)	032	X axis ROUND settings (Restart)	Instant	0	
Axis parameter (P2.)		Y axis ROUND settings (Restart)	Instant	0	
Axis parameter (P2.)		Z axis ROUND settings (Restart)	Instant	0	
Axis parameter (P2.)		A axis ROUND settings (Restart)	Instant	0	

Parameters type	S/N	Description	Effective mode	Default value	Page
Axis parameter (P2.)	033	X axis 360 reset	Instant	0	
Axis parameter (P2.)		Y axis 360 reset	Instant	0	
Axis parameter (P2.)		Z axis 360 reset	Instant	0	
Axis parameter (P2.)		A axis 360 reset	Instant	0	
Axis parameter (P2.)	034	X axis rotation display mode	Instant	0	
Axis parameter (P2.)		Y axis rotation display mode	Instant	0	
Axis parameter (P2.)		Z axis rotation display mode	Instant	0	
Axis parameter (P2.)		A axis rotation display mode	Instant	0	
Axis parameter (P2.)	035	X axis rotation path optimization	Instant	0	
Axis parameter (P2.)		Y axis rotation path optimization	Instant	0	
Axis parameter (P2.)		Z axis rotation path optimization	Instant	0	
Axis parameter (P2.)		A axis rotation path optimization	Instant	0	
Axis parameter (P2.)	036	X axis maximum acceleration(Kpps)	Instant	2000	
Axis parameter (P2.)		Y axis maximum acceleration(Kpps)	Instant	2000	
Axis parameter (P2.)		Z axis maximum acceleration(Kpps)	Instant	2000	
Axis parameter (P2.)		4 axis maximum acceleration(Kpps)	Instant	2000	
Axis parameter	037	Rotation axis speed	Instant	0	

Parameters type	S/N	Description	Effective mode	Default value	Page
(P2.)		optimization features			
Axis parameter (P2.)	038	A-axis maximum rotation speed	Instant	50	
Axis parameter (P2.)	039	Handwheel encoder direction	Instant	1	
Management parameter (P3.)	001	Type password to select management mode	Instant	Superuser	
Management parameter (P3.)	002	Modify super user password	Instant	*******	
Management parameter (P3.)	003	Modify operation user password	Instant	*******	
Management parameter (P3.)	004	Initialize comprehensive parameters to default	Restart	=======	
Management parameter (P3.)	005	Initialize IO configuration to default	Restart	=======	
Management parameter (P3.)	006	Reset all parameters	Restart	=======	
Management parameter (P3.)	007	Back up parameters	Instant	=======	
Management parameter (P3.)	008	Restore parameters	Restart	=======	
Management parameter (P3.)	009	Generate password file	Instant	=======	
Management parameter (P3.)	010	Menu clicking mode	Instant	=======	
Management parameter (P3.)	011	Clear accumulated processing pieces	Instant	=======	
Management parameter (P3.)	012	Clear current processing pieces	Instant	=======	
Management parameter (P3.)	013	Import CSV system configuration	Restart	=======	
Management	014	Default boot interface	Restart	Rel	

Parameters type	S/N	Description	Effective mode	Default value	Page
parameter (P3.)					
Management parameter (P3.)	015	System language packs	Restart	Chinese	
Management parameter (P3.)	016	Macro keyword effective enable	Instant	ON	
Management parameter (P3.)	017	Boot interface mode	Instant	=======	
Management parameter (P3.)	018	System debugging info enable	Instant	OFF	
Management parameter (P3.)	019	Axis control composite key enable	Instant	ON	

3.9.2 General parameter (P1.)

001	Manual speed

Range : 1 ~ 9999

Unit : mm/min

Authority : Operation admin

Default : 1000

Effective time : Instant

Note : Manual speed setting

In this mode, the initial velocity and acceleration of the specified axis is determined by the initial velocity and acceleration of the specified axis;

002	Maximum feeding speed

Range : 1 ~ 20000

Unit : mm/min

Authority : Operation admin

Default : 20000

Effective time : Instant

Note : Set the maximum feeding speed F;

The maximum feeding speed restricts the F instruction during processing, i.e. no matter what F is set to, the actual speed can't exceed this parameter value. Setting this parameter will prevent the damage caused by accidental speed programming error when transferring processing files.

003	Handwheel speed
004	Handwheel acceleration

Range : 500 ~ 9999, 100 ~ 3000

Unit : mm/min, mm/sec

Authority : Operation admin

Default : 9000, 1000

Effective time : Instant

Note : Set manual speed and handwheel speed;

Motion speed and acceleration of specified axis in this mode;

005	Home mode

Range : 0~1

Unit : None

Authority : Operation admin

Default : 0 (program)

Effective time : Instant

Note : Program home 0, mechanical home 1

Program home: coordinates return to home position

Mechanical home:

Mechanical home includes three modes: use detection signal only, use detection signal and enable servo Z phase detection, enable servo Z phase detection only

Use detection signal only: external detection switch should be used to locate the home position; during homing, press [Start] or negative button of a single home axis to make the home axis move to the home direction; stop at a slow speed when signal

is detected, and then move back slowly until the home point can't be detected, then slowly move back again until reverse signal is detected, and homing ends when the signal is valid again.

Use detection signal and enable servo Z phase detection: during homing, press [Start] or negative button of a single home axis to make the home axis move to the home direction, and stop at a slow speed when signal is detected; when servo Z phase detection enable switch in IO configuration parameters is turned on, the detection device automatically enables Z-phase position as the home position.

Enable servo Z phase detection only: position by servo Z-phase home.

Perform home position offset in mechanical home mode; at this moment, set parameter No. 019 home point offset in [axis configuration].

006	Spindle alarm detection level
007	Inverter alarm detection level

Range : 0 ~ 1, 0 ~ 1

Unit : Logic Voltage Level

Authority : Operation admin

Default : 0

Effective time : Instant

Note : Check the above alarm when the system is running. Once detection is effective, the system immediately alarms.

This port is affected by the setting of IO configuration.

008	External start 2 test voltage level
009	External pause 2 test voltage level
010	External emergency stop 2 test voltage level

Range : 0 ~ 1, 0~1, 0 ~ 1

Unit : Logic Voltage Level

Authority : Operation admin

Default	:	0
Effective time	:	Instant
Note	:	Expansion external start, pause, stop button of the system. This port is affected by the setting of IO configuration.

011	Automatic home mode configuration

Range	:	
Unit	:	Instruction line
Authority	:	Operation admin
Default	:	500
Effective time	:	Instant
Note	:	Set the auto-homing mode; this parameter specifies the order of axis homing; the system default is ZA-XY; the homing is divided into three sections

012	Interpolation acceleration mode

Range	:	
Unit	:	
Authority	:	Operation admin
Default	:	1
Effective time	:	Instant
Note	:	Set the performance of S-curve acceleration/deceleration

013	Home test enable after alarm
014	System home test enable

Range	:	0 ~ 1, 0 ~ 1
Unit	:	
Authority	:	Operation admin
Default	:	0, 1
Effective time	:	Instant
Note	:	Used to set whether prompt user to home in corresponding occasions, ensuring that the user has performed the home operation; If it is set to 0, the system will run directly without performing home operation.

015	Default process plane

Range : G17(X_Y), G18(Y_Z), G19(X_Z)

Unit :

Authority : Operation admin

Default : G17 (X_Y)

Effective time : Instant

Note : Set the default processing plane to G17(X_Y), G18(Y_Z) or G19(X_Z); modify the default plane, so that it isn't need to specify the modal plane value while programming, and write plane related instructions directly instead;

016	IO filter level (Restart)

Range : 0 ~ 8

Unit : None

Authority : Super admin

Default : 0

Effective time : After restarted

Note : Set the filter constant;

If the environment has too much interference, e.g. rain and thunder, please enter a filter value. Higher value indicates longer test time and high reliability; range: 50~4000ms

017	M code waiting time

Range : 0 ~ 9999

Unit : ms

Authority : Operation admin

Default : 100

Effective time : Instant

Note : Set the waiting time (unit: ms) after executing M code

018	Warning lamp output port No.
019	Running lamp output port No.
020	External start 2 test port No.

021	External pause 2 test port No.
022	External emergency stop 2 test port No.

Range : 0 ~ 23

Unit :

Authority : Super admin

Default : Port Comparison Table in the Manual

Effective time : Instant

Note : The output port number assigned to corresponding system function; to shield this function, enter "8888", and the parameter will display "=========".

023	IO Conf in RESET 00~15
024	IO Conf in RESET 16~23

Range : 0 ~ 65535

Unit :

Authority : Super admin

Default : 65535, 243

Effective time : Instant

Note : Used to configure the output port to low level in resetting

Use binary system to configure pin position; please configure in parameters interface

For example: 65404, is 1111 1111 0111 1100 in binary system; which indicates that OUT0, OUT1 and OUT7 keep original output status when the system is reset.

025	Lubricant output port No.
026	Cooling output port number
027	Spindle CW output port number
028	Spindle CCW output port number
029	Spindle alarm detection port number
030	Variable frequency alarm detection port number
031	Variable frequency segment position 0 output port number
032	Variable frequency segment position 1 output port number
033	Variable frequency segment position 2 output port number

034	Variable frequency segment position 3 output port number

Range : 0 ~ 23

Unit :

Authority : Super admin

Default : I/O port number

Effective time : Instant

Note : Set the input or output port number assigned to configure system functions; To shield this function, you can type 8888, and hide the parameter to "=========".

035	Line number increment

Range : 0 ~ 100

Unit : None

Authority : Operation admin

Default : 0

Effective time : Instant

Note : While editing G code manually, add a line number Nxxxxx automatically in a new line;

0 indicates that this function is disabled;

036	System baud rate

Range : 9600~115200

Unit : None

Authority : Operation admin

Default : 115200

Effective time : Restart

Note : The communication rate setting when DNC or other PC software and this controller are in RS232 communication mode

037	Controller ID No.

Range : 1~255

Unit	:	None
Authority	:	Operation admin
Default	:	1
Effective time	:	Restart
Note	:	The ID number setting of the controller when DNC or other PC software and this controller are in MODBUS communication mode

038	G73(M) circulating tool retraction amount
039	G83(M) circulating tool retraction amount

Range	:	0.1 ~ 100
Unit	:	mm
Authority	:	Operation admin
Default	:	2.000
Effective time	:	Instant
Note	:	Set the tool retracting amount after Q is fed in G73 and G83 instructions; this parameter (default: 2mm) is set according to actual chip removal effect.

040	Arc interpolation feed rate

Range	:	0 ~ 1
Unit	:	mm
Authority	:	Operation admin
Default	:	0.2
Effective time	:	Instant
Note	:	Set the arc interpolation equivalent If this value is too small, the arc has higher approximation accuracy, but the computation will be too high, making the pause during processing obvious and affecting the processing effect.

041	Arc interpolation acceleration

Range	:	10 ~ 500
Unit	:	mm/sec
Authority	:	Operation admin
Default	:	100
Effective time	:	Instant
Note	:	Set the arc acceleration equivalent If this value is too small, the acceleration will be slow; please set to a higher value according to the actual arc size.

042	Interpolation speed mode

Range	:	0 ~ 1
Unit	:	None
Authority	:	Operation admin
Default	:	0
Effective time	:	Instant
Note	:	In pretreatment mode, set to 0 to use corner speed balancing algorithm, or set to 1 to use axis acceleration constraints balancing algorithm.

043	Code pretreatment processing mode

Range	:	0 ~ 1
Unit	:	None
Authority	:	Operation admin
Default	:	0 (real-time processing)
Effective time	:	Instant
Note	:	Real-time processing is suitable for machine test. In pretreatment mode, the system enters processing state buffs for two seconds and pre-reads. The pretreatment mode only can check the direction and size of feeding segment to adjust the speed automatically and process at optimized speed.

044	Variable frequency analog control mode

Range	:	0 ~ 1
Unit	:	None
Authority	:	Operation admin
Default	:	0
Effective time	:	Instant
Note	:	Control mode corresponding to spindle S code (frequency conversion mode) 0 : Analog output 1: Section speed regulation (4-digit code), as below: OUT23------S0 OUT22------S1 OUT21------S2 OUT20------S3 In analog output mode, the analog voltage is: V=S/MaxRPM S is the rotation set by the user, and MaxRPM is the maximum rotation of main axis set by the parameters; In switching quantity mode, constitute block 0-15 according to four-digit code to output; S code value is restricted to 0-15;

3.9.3 Axis parameter configuration (P2.)

001	X axis instruction frequency multiplication ratio
	Y axis instruction frequency multiplication ratio
	Z axis instruction frequency multiplication ratio
	A axis instruction frequency multiplication ratio
002	X axis instruction frequency division coefficient
	Y axis instruction frequency division coefficient
	Z axis instruction frequency division coefficient
	A axis instruction frequency division coefficient

Range	:	1 ~ 65535, 1 ~ 65535
Unit	:	None
Authority	:	Operation admin
Default	:	1
Effective	:	Instant

time

Note : When screws of different pitches and motors of different step angles or servo motors of different pulses are matched, or connected through gears, it allows keeping the program and actual motion distance consistent through electronic gear ratio setting of the system.

CMR/CMD =P/ (L×1000)

CMR: gear ratio numerator

CMD: gear ratio denominator

P: Pulses corresponding to one rotation of the motor

L: Machine tool movement corresponding to one rotation of the motor (mm)

CMD/CMR is the pulse equivalent actually, i.e. the motion distance corresponding to every pulse (unit: 0.001mm).

Ex 1: the motor rotates one cycle every 5000 pulses, and the machine tool moves 5mm when the motor rotates one cycle, then

CMR/CMD=5000/ (5*1000)=1/1

Then, CMR=1, CMD=1, the pulse equivalent is 0.001mm

Ex 2: the motor rotates one cycle every 5000 pulses, and the machine tool moves 10mm when the motor rotates one cycle, then

CMR/CMD=5000/ (10*1000)=1/2

Then, CMR=1, CMD=2, the pulse equivalent is 0.002mm

CMR/CMD =P/(L×1000)

The machine tool moves 0.001mm when the controller sends one pulse

003	X axis rapid traverse rate(mm/min)
	Y axis rapid traverse rate(mm/min)
	Z axis rapid traverse rate(mm/min)
	A axis rapid traverse rate(mm/min)
004	X axis start rate (mm/min)
	Y axis start rate (mm/min)
	Z axis start rate (mm/min)
	A axis start rate (mm/min)

005	X axis acceleration(Kpps)
	Y axis acceleration(Kpps)
	Z axis acceleration(Kpps)
	A axis acceleration(Kpps)

Range : 1 ~ 20000, 1 ~ 20000, 1 ~ 8000

Unit : mm/min, mm/min, mm/sec

Authority : Operation admin

Default : 3000, 600, 1500

Effective time : Instant

Note : This parameter is the trapezoid acceleration/deceleration setting

Used for G00 instruction

About start speed, 1-2 rotation motor speed is recommended for step motor; as above, the machine tool moves 5mm when the motor rotates one cycle, and the speed is 5-10mm/sec (300-600mm/min). For servo motor, the start and stop shouldn't have vibration. If this speed is too high, it will cause vibration during motion, and the step motor will be out of step.

Parameters affected by the acceleration and start speed include manual speed, home speed and other non-interpolation motions;

006	X axis positive soft limit (mm)
	Y axis positive soft limit (mm)
	Z axis positive soft limit (mm)
	A axis positive soft limit (mm)
007	X axis negative soft limit (mm)
	Y axis negative soft limit (mm)
	Z axis negative soft limit (mm)
	A axis negative soft limit (mm)

Range : -9999 ~ 9999

Unit : mm

Authority : Operation admin

Default : Maximum positive/negative value

Effective time : Instant

Note : Generally, the machine tool has hard limit signal. In this case, software limit isn't required. Please set the positive limit to +9999.999, and negative limit to -9999.999.

If hard limit switch isn't installed, please use soft limit, which uses machine tool coordinate system as the base point. Positive limit and negative limit are subject to actual distance (unit: mm).

Since soft limit decelerates and stops at the limit point, it may exceed the set distance, which depends on acceleration time and speed. Please keep certain margin when setting this parameter.

008	Feeding speed(mm/min)
	Initial feeding speed(mm/min)
	Feeding acceleration(mm/sec)

Range : 1 ~ 9999, 1 ~ 9999, 1 ~ 8000

Unit : mm/min,mm/min,mm/sec,mm/min

Authority : Operation admin

Default : 3000, 800, 1500

Effective time : Instant

Note : The feeding instructions G01, G02 and G03 move at the speed of F instruction. If the F instruction isn't specified in the program, the above instructions move at the speed set by this parameter. If the F instruction is specified, this parameter will be invalid.

009	X axis reverse clearance compensation(pulse)
	Y axis reverse clearance compensation(pulse)
	Z axis reverse clearance compensation(pulse)
	A axis reverse clearance compensation(pulse)

Range	:	1 ~ 20000
Unit	:	Pulse
Authority	:	Operation admin
Default	:	0
Effective time	:	Instant
Note	:	A small clearance is left when power transfer process is conducted between moving machine part and its drive device (such as ball screw backlash), because the machine tool will be worn seriously if the mechanical structure has no clearance. The mechanical clearance will result in error in displacement travel. For example, when the machine tool changes from the CCW to CW movement, the position of the machine tool is one clearance slower than the actual encoder (encoder corresponds to the theoretical position when the machine tool has no clearance); when the machine tool changes from the CW to CCW movement, the position of the machine tool is one clearance faster than the encoder. Therefore, clearance compensation is required to eliminate the effects of mechanical clearance on machine tool. In addition, the parameter uses the minimum control unit pulse to compensate. The specific corresponding number should be converted according to the gear ratio. For example: X-axis clearance is 0.02mm, X-axis electronic gear ratio is 1:1, and X-axis backlash compensation value is 20, calculated as follows: (0.02 / 10) * 10000 = 20

010	X hardware positive limit enable
	Y hardware positive limit enable
	Z hardware positive limit enable
	A hardware positive limit enable
011	X hardware negative limit enable
	Y hardware negative limit enable

	Z hardware negative limit enable
	A hardware negative limit enable

Range : 0~1

Unit : None

Authority : Super admin

Default : 0

Effective time : After restarted

Note : Hard limit has two modes, i.e. hardware response and software scanning;

Hardware response mode:

This mode is integrated by the motion chip, and is triggered by the effective voltage level of the circuit test limit pin. Therefore, it is highly real-time, but it also has a defect. If the external interference is serious, the normal pulse will be affected and the system doesn't alarm because it can't read the error state in time, which will cause loss; therefore, this function requires that the wiring switch uses normally closed connection, i.e. high effective level; this function considers the complexity of field environment and the default value is off.

Scanning mode:

Scanning mode is integrated by the system and can't be shielded. The scanning mode input signal by accessing specified function number, and uses software anti-interference detection technology to check whether limit alarm occurs or has no interference. This requires certain time to check, and thus the real time isn't as well as interrupted limit. However, in most cases (at 10mm/min processing speed), it can meet the requirement on processing safety check.

The hardware response function of hard limit is prior to scanning response function, i.e. if the hardware response is enabled, it will quicken the response speed directly. It should be noted that the hardware response function only can stop pulse in instant mode. Therefore, the instant stop mode may cause

mechanical vibration if the speed is too high. While software scanning mode uses maximum acceleration mode and decelerates according to the maximum acceleration set to every axis by the user, and therefore overshot will occur.

The hard limit enable is not used to control the hardware response in the control system; instead it is used to control whether the hardware limit detection port is available; the default setting is available; when it is set to unavailable, the hardware limit detection port can be used as common input port.

012	X hard limit effective voltage level (Restart)
	Y hard limit effective voltage level (Restart)
	Z hard limit effective voltage level (Restart)
	A hard limit effective voltage level (Restart)

Range : 0 ~ 1

Unit : Logic Voltage Level

Authority : Super admin

Default : 0, 1

Effective time : Instant

Note : Input port detects valid signal of hard limit

013	X axis external home enable
	Y axis external home enable
	Z axis external home enable
	A axis external home enable

Range : 0 ~ 1

Unit : None

Authority : Super admin

Default : 1

Effective time : Instant

Note : When mechanical home mode is selected, this parameter determines whether external deceleration switch should be

searched. If this parameter is set to 0, and servo Z phase enable is also set to 0, the home mode sets current point as the home directly in mechanical mode.

0: No

1: Yes

014	Effective voltage level for X external home
	Effective voltage level for Y external home
	Effective voltage level for Z external home
	Effective voltage level for A external home

Range : 0~1

Unit : LOGIC VOLTAGE LEVEL

Authority : Super admin

Default : 0

Effective time : Instant

Note : Set the effective voltage level of external home sensor switch during home operation.

0: Low level

1: High level

015	X axis home direction
	Y axis home direction
	Z axis home direction
	A axis home direction

Range : 0 ~ 1

Unit : None

Authority : Operation admin

Default : 1, 1, 0, 0

Effective time : Instant

Note : Set the mechanical home direction of every processing axis

0: Positive

1: Negative

016	X axis home speed
	Y axis home speed
	Z axis home speed
	A axis home speed

Range : 0 ~ 9999

Unit : mm/min

Authority : Operation admin

Default : 1000

Effective time : Instant

Note : Set the home speed of every axis separately

017	X axis home deceleration speed
	Y axis home deceleration speed
	Z axis home deceleration speed
	A axis home deceleration speed
018	X axis home scanning speed
	Y axis home scanning speed
	Z axis home scanning speed
	A axis home scanning speed

Range : 1~20000

Unit : mm/min

Authority : Operation admin

Default : 100, 60

Effective time : Instant

Note : Used to set different speed parameters for mechanical home; the specific effect is as follows:

Home speed - (when external switch home is detected) Deceleration speed - Scanning speed.

019	X axis home offset
	Y axis home offset
	Z axis home offset
	A axis home offset

Range : -9999 ~ 9999

Unit : Pulse

Authority : Operation admin

Default : 0

Effective time : Instant

Note : Set the home offset (unit: mm) after axis home operation

The specific process is to perform mechanical homing first, offset corresponding pulse, and then set the point to mechanical home.

Note: This parameter is invalid during program home operation.

020	Effective voltage level for servo X axis alarm
	Effective voltage level for servo Y axis alarm
	Effective voltage level for servo Z axis alarm
	Effective voltage level for servo A axis alarm
021	Effective voltage level for servo X axis reset
	Effective voltage level for servo Y axis reset
	Effective voltage level for servo Z axis reset
	Effective voltage level for servo A axis reset

Range : 0 ~ 1

Unit : Logic voltage Level

Authority : Super admin

Default : 0, 1

Effective time : Instant

Note : Alarm active level is the active level of servo alarm input controller signal; servo reset active level is the level to reset servo signal; specific parameter settings depend on servo interface level.

022	Servo X axis Z phase home enable
	Servo Y axis Z phase home enable

	Servo Z axis Z phase home enable
	Servo A axis Z phase home enable
023	Effective voltage level for servo X axis Z phase
	Effective voltage level for servo Z axis Z phase
	Effective voltage level for servo Y axis Z phase
	Effective voltage level for servo A axis Z phase

Range : 0~1

Unit : Logic Voltage Level

Authority : Super admin

Default : 0

Effective time : Instant

Note : When this parameter is enabled, encoder Z phase positioning of corresponding axis will be enabled automatically in mechanical home mode, i.e. the “servo home” positioning; in this mode, the accuracy of repeated home positioning depends on servo positioning accuracy, and therefore it is recommended to enable this function for servo motor. Step motor doesn’t have encoder and can’t enable this option, or else the signals can’t be scanned during mechanical home operation and will move constantly.

024	X servo home direction
	Y servo home direction
	Z servo home direction
	A servo home direction

Range : 0 ~ 1

Unit : None

Authority : Super admin

Default : 0

Effective time : Instant

Note : This parameter determines the Z phase search direction when servo Z phase enable parameter is enabled in P2.009~P2.016.

0: Positive

1: Negative

025	X pulse command format (Restart)
	Y pulse command format (Restart)
	Z pulse command format (Restart)
	A pulse command format (Restart)

Range : 0 ~ 1

Unit : None

Authority : Super admin

Default : 1

Effective time : Restart

Note : Pulse command format setting is to configure the mode of output pulse. The compatible command format of the motor drive should be known in advance.

0: Pulse + pulse

1: Pulse + direction

026	X pulse logic direction (Restart)
	Y pulse logic direction (Restart)
	Z pulse logic direction (Restart)
	A pulse logic direction (Restart)

Range : 0 ~ 1

Unit : None

Authority : Super admin

Default : 1

Effective time : Restart

Note : Set pulse direction; if the controller direction is reverse to actual drive direction, please modify this parameter to adjust the rotation direction of motor.

027	X axis specified interface axis No. (Restart)
	Y axis specified interface axis No. (Restart)
	Z axis specified interface axis No. (Restart)
	A axis specified interface axis No. (Restart)

Range	:	0 ~ 4
Unit	:	Pulse port sequence No.
Authority	:	Super admin
Default	:	
Effective time	:	Restart
Note	:	In default mode, the actual number of every axis corresponds to the silk screen number on the shell. If certain function axis is abnormal, you can replace the axis through this function. For example, set P2.045 to 4, P2.048 to 1, then, any operation to X axis will be the operation to A axis encoder port on the shell. 0: no such axis 1~4: corresponding to 1#-4# axis

	Spindle specified interface axis No. (Restart)

Range	:	0~4
Unit	:	Pulse port sequence No.
Authority	:	Super admin
Default	:	
Effective time	:	Restart
Note	:	It is set to 0 by default, indicating that the main axis is in variable frequency control mode, i.e. analog or gear position control mode. To use servo main axis, a coding port is required (servo spindle must be in position control mode); you can modify this parameters to specify the function. 0: variable frequency main axis adjusted by analog 1~4: corresponding to 1#-4# axis Note: If a pulse port is specified as the function port of spindle, the pulse port should be removed from previously corresponding function axis number, or else the system will assign to spindle after restarted and the original function axis will be invalid.

028	X axis encoder wire number
	Y axis encoder wire number
	Z axis encoder wire number
	A axis encoder wire number

Range : 64 ~ 99999

Unit : Wire number

Authority : Super admin

Default : 2500

Effective time : Instant

Note : Set the encoder wires connected to every pulse port (AB phase pulse). Since four times frequency division is performed for internal transfer, the value of this parameter should be the pulses collected by the encoder for one cycle divided by 4.

029	X axis encoder pulse logic direction (Restart)
	Y axis encoder pulse logic direction (Restart)
	Z axis encoder pulse logic direction (Restart)
	A axis encoder pulse logic direction (Restart)

Range : 0~1

Unit : None

Authority : Super admin

Default : 0

Effective time : Restart

Note : If the logic direction obtained by the encoder is reverse to the actual motion direction of the axis, please set this parameter.

0: Positive direction

1: Negative direction

030	X axis pulse logic voltage level(Restart)
	Y axis pulse logic voltage level(Restart)
	Z axis pulse logic voltage level(Restart)
	A axis pulse logic voltage level(Restart)

Range	:	0~1
Unit	:	Logic Voltage Level
Authority	:	Super admin
Default	:	0
Effective time	:	Restart
Note	:	Set the normal voltage level when the pulse is working. If the setting is different from the normal voltage level required by motor drive, a direction will have accumulative error during every positive and negative motion (independent of pulses). Therefore, if the positioning axis of the machine has accumulative error in a direction, please check whether this parameter matches.

031	X axis characteristics (rotation: 0, linear: 1)
	Y axis characteristics (rotation: 0, linear: 1)
	Z axis characteristics (rotation: 0, linear: 1)
	A axis characteristics (rotation: 0, linear: 1)

Range	:	0 ~ 1
Unit	:	None
Authority	:	Super admin
Default	:	1
Effective time	:	Instant
Note	:	Set axis characteristics. 0: Rotary axis 1: Linear axis The settings of this parameter and the rotation display of specified axis will affect the ROUND setting of specified axis. When set to rotation characteristics, the ROUND value of specified axis is 359999

032	X axis ROUND settings (Restart)
	Y axis ROUND settings (Restart)

	Z axis ROUND settings (Restart)
	A axis ROUND settings (Restart)

Range : 0 ~ 10000000

Unit : Pulse

Authority : Super admin

Default : 0

Effective time : Instant

Note : Round function is available on hardware version 1.5 or later only;

This function is used to prevent the logic counting of axis exceeding the maximum counting range (2147483648) and causing overflow error;

Generally, overflow occurs only when the axis is set to rotary. The system will calculate the corresponding pulse limit according to the gear ratio of current axis and assign to the ROUND parameter of corresponding axis, if current axis is set to rotary and uses 360° display mode after the system getting the specified axis parameters. The user can check the change of this parameter when the rotary axis display function is enabled. The user can modify the changed parameters, and the finally displayed number will be effective.

033	X axis 360 reset
	Y axis 360 reset
	Z axis 360 reset
	A axis 360 reset

Range : 0~1

Unit : None

Authority : Super admin

Default : 0

Effective time : Instant

Note : None

034	X axis rotation display mode
	Y axis rotation display mode
	Z axis rotation display mode
	A axis rotation display mode

Range : 0~1

Unit : None

Authority : Super admin

Default : 0

Effective time : Instant

Note : Set the coordinate display mode of the axis. This parameter is valid when the characteristic value of specified axis is set to 0

0: 0 ~ 360 °display

1: -9999.999 ~ 9999.999 °display

The settings of this parameter and the characteristic value of specified axis will affect the ROUND setting of specified axis.

035	X axis rotation path optimization
	Y axis rotation path optimization
	Z axis rotation path optimization
	A axis rotation path optimization

Range : 0 ~ 1

Unit : None

Authority : Super admin

Default : 1

Effective time : Instant

Note : This parameter is valid when the characteristic value and rotation display of specified axis are set to 0; set whether looking for shortest path automatically; if it is rotary axis and is positioning but doesn't process, enable this function to shorten the motion time.

0: Do not optimize the path

1: Enable the shortest path

Note: If processing is required during the motion, the shortest path may be not your desired processing track.

036	X axis maximum acceleration
	Y axis maximum acceleration
	Z axis maximum acceleration
	A axis maximum acceleration

Range : 100 ~ 8000

Unit : Kpps (Kilo Pulse Per Second)

Authority : Super admin

Default : 2000

Effective time : Instant

Note : Set the maximum acceleration of every axis. This setting will affect the track speed optimization of pretreatment to every axis. If a high value is set, the axis response time will be shortened and characteristics of the motor will be improved according to the machine tool.

This parameter also affects the home function and limit stop function.

Hard limit function: Use hard limit in software scanning mode, in which the hard limit decelerates and stops according to the maximum acceleration of this axis. Therefore, if this value is too high, the machine tool will stop in emergency, and if this value is too low, it will cause too much overshoot.

Home function: the home acceleration of every axis uses this value.

037	Rotation axis speed optimization feature

Range : 0 ~ 1

Unit : None

Authority : Super admin

Default : 0

Effective time : Instant

Note : None

038	Hand wheel encoder direction

Range : 0 ~ 1

Unit : None

Authority : Super admin

Default : 0

Effective time : Instant

Note : None

	Hand wheel encoder direction

Range : 0 ~ 1

Unit : None

Authority : Super admin

Default : 0

Effective time : Instant

Note : When the logic direction obtained by handheld box encoder is reverse to the actual motion direction of the axis, set this parameter to perform in-phase setting (this function in not available temporarily).

0: positive direction

1: negative direction

3.9.4 Management parameter (P3.)

001	Entering management mode
002	Modify super user password
003	Modify operation user password

Range : None

Unit : None

Authority : None

Default : None

Effective : Instant

time

Note : In this menu, type the password and press 'Insert'. If the password is valid, the system will enter this user mode;
After entering, this menu will turn into "Exit XXX admin mode", indicating entering successfully;
In the new menu, press the Insert key to exit the admin mode. To modify the parameter table at this moment, you need to enter the admin mode again;
The super user can modify all passwords, while the operation user only can modify the own password.
Password 0 indicates that the password isn't checked in this mode; it isn't required to enter the admin mode to modify the parameters.

004	Initialize comprehensive parameters to default
005	Initialize IO configuration to default
006	Reset all parameters

Range : None
Unit : None
Authority : Super user
Default : None
Effective time : Instant
Note : Initial parameters table only in super user mode

007	Back up parameters
008	Restore parameters

Range : None
Unit : None
Authority : Super user
Default : None
Effective time : Instant
Note : The parameters are backed up and restored only in super user

mode.

The parameters are backed up to the sysconf.bak file in the root directory of the controller. If this folder already has a file with same name, the latest backup will overwrite this file.

The sysconf.bak file in the root directory is also used for restoring. During restoring, it will check whether the parameter versions are same according to the backed up parameter version; if not, the system won't restore the parameter table.

After restoring, the system will restart automatically.

009	Generate password file

Range : None

Unit : None

Authority : None

Default : None

Effective time : Instant

Note : If you have forgotten the password, you can generate a PassMeg.DAT file with this function, provide this file to controller manufacturer and ask the manufacturer to reset the password.

010	Menu clicking mode

Range : 0~1

Unit : None

Authority : None

Default : 0 (click)

Effective time : Instant

Note : The function is to be developed

011	Clear accumulated processing pieces
012	Clear current processing pieces

Range : None

Unit	:	None
Authority	:	None
Default	:	None
Effective time	:	Instant
Note	:	Clear the accumulated value of current processing pieces

013	Import CSV system configuration

Range	:	None
Unit	:	None
Authority	:	None
Default	:	None
Effective time	:	Restart
Note	:	Import the CSV system configuration of the manufacturer into the system

014	Default boot interface

Range	:	Select
Unit	:	None
Authority	:	Operation admin
Default	:	ABS
Effective time	:	Instant
Note	:	Select default boot interface from absolute position, relative position and comprehensive position.

015	System language packs

Range	:	0~1
Unit	:	None
Authority	:	Operation admin
Default	:	0 (Chinese)
Effective time	:	Instant

Note : Select system language

Chinese

English

016	Macro keyword effective enable

Range : 0~1

Unit : None

Authority : Operation admin

Default : 0 (Chinese)

Effective time : Instant

Note : Macro keyword effective parameter is used to set whether the macro expression symbol set on the face is valid; 1: valid, 0: invalid.

017	Boot interface mode

Range : 0~6

Unit : None

Authority : Operation admin

Default : 1S

Effective time : Instant

Note : Used to configure logo display mode; if it is set to 0, press any key to enter the system; for non-zero value, it enters into the system automatically after corresponding time delay.

019	System debugging info enable

Range : 0~1

Unit : None

Authority : Super user

Default : OFF/0

Effective time : Instant

Note : Used to configure whether RS232 of current system outputs

debugging information during running.

This parameter is dedicated for programmers, and users are not recommended to enable this parameters.

If debugging information is enabled, the system performance will deteriorate, so that it is disabled in the normal process. If networking is enabled, this feature must be turned off, or else the networking will crash.

020	Axis control composite key enable

Range : 0~1

Unit : None

Authority : Super user

Default : ON/1

Effective time : Instant

Note : Used to configure whether enable the key for axis motion on the controller panel.

This parameters is used to shield the composite function of the axis motion control button on the NC panel when use additional panel; if additional panel isn't used, it must be enabled or else axis movement cannot be controlled through the buttons.

021	Additional panel enable

Range : 0~1

Unit : None

Authority : Super user

Default : OFF/0

Effective time : Instant

Note : Used to configure whether NC uses additional panel, which must match ADT series, or compatible with the same protocol interface.

To use additional panel, the system debugging information must be disabled (P3.19).

Chapter 4 Programming

4.1 Motion direction naming of control axes

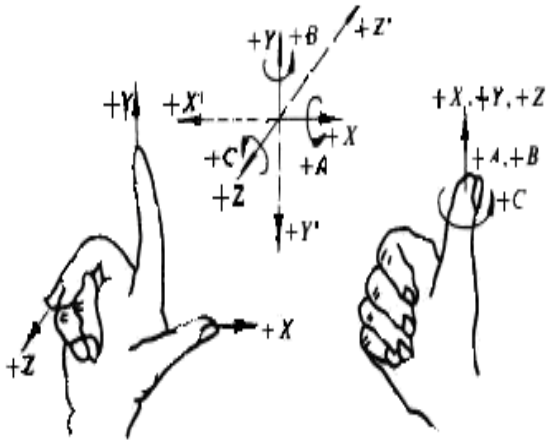


Fig. 2.1 Name of the machine tool coordinate axis

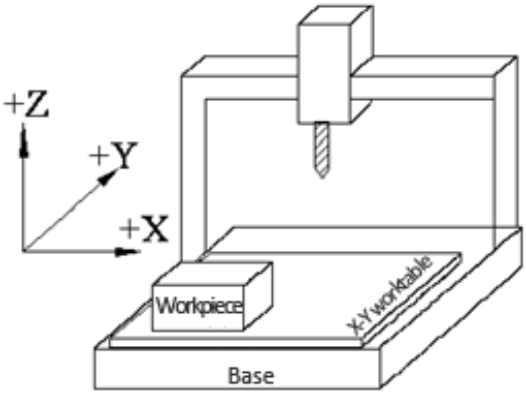


Fig. 2.2 Rotation axis direction determination of the machine tool

This system can control the rapid traverse, feeding and interpolation of four axes. The axis direction is defined in Cartesian coordinate system, as shown below (facing to the machine tool):

Z axis:

The up and down movement of the tool relative to the workpiece is Z axis motion, with the upward movement the positive motion and the downward movement the negative motion.

X axis:

The left and right movement of the tool relative to the workpiece is X axis motion, with the rightward movement the positive motion and the leftward movement the negative motion.

Y axis:

The forward and backward movement of the tool relative to the workpiece is Y axis motion, with the forward movement the positive motion and the backward movement the negative motion.

Spindle:

Look down to the workpiece, the clockwise rotation is spindle positive rotation and the counterclockwise rotation is negative rotation.

A, B, C axis:

The positive directions of rotation axes correspond to the positive directions of X, Y, Z axis, which are determined according to the forward direction of right hand screw.

- **Notice:**

The X, Y, Z, A, B, C axis motion described in this manual is the tool's motion relative to the workpiece, i.e. it is assumed that the workpiece coordinate system has been set.

4.2 Coordinate systems of machine tool and workpiece (G53, G54~G599)

Machine tool coordinate system G53:

- **Function and purpose**

G53 machine tool coordinate system is fixed on the machine tool, that is, the mechanical coordinate system of the machine tool. The machine tool coordinate system is established through the operation returning to the reference point after NC power on each time. Through G53 instruction and coordinate instructions thereafter, the tool moves to the specified location on the basic mechanical coordinate system.

- **Instruction format:**

Selection of basic mechanical coordinate system (G90/G91) G53 X_ Y _ Z_;

- **Note:**

1. When the system powers on, set basic mechanical coordinate system automatically with the return position of automatic or manual reference point (origin) as the benchmark
2. G53 instruction is non-modal instruction and is only valid in single section
3. When the G53 instruction is incremental instruction mode (G91), move the incremental value in the selected coordinate system, which relates to the reference point of current coordinate system rather than the position of each axis in current coordinate system
4. All the axes of G53 instruction are moving at fast feed rate, which is set in the system
5. The coordinate values specified in G53 instruction indicate the distance from the return position of reference point (origin) to zero point of the mechanical coordinate system
6. If G53 and G28 instructions (return to the reference point) are used in the same block, the following instructions are valid
7. The mechanical coordinate system will not change according to G92 instruction
8. Even if G53 instruction is executed, the tool radius compensation amount of designated axis will not be canceled

- **Example:**

Example 1:

G90 G54 G17

G90 G53 X10 Y10 Z10

If the origin of the workpiece coordinates of X-axis, Y-axis and Z-axis is (20, 20, 20), the current position of the system after executing abovementioned instructions is (-10, -10, -10)

Example 2:

G90 G54 G17

G00 X10 Y10 Z10

G91 G53 X10 Y10 Z10

If the origin of the workpiece coordinates of X-axis, Y-axis and Z-axis is (20, 20, 20), the current position of the system after executing abovementioned instructions is (-10, -10, -10)

Workpiece coordinate system G54~G59 (G591~G599):

- **Function and purpose:**

1. Workpiece coordinate system is used in program processing. It sets a reference point on the workpiece to the coordinate origin and is used to simplify the programming of workpieces.

2. You can move the tool to the position of the workpiece coordinate system through this instruction. The workpiece coordinate system is used by the programmer for programming. The system allows presetting six workpiece coordinate systems and nine extension coordinate systems (G591-G599). Set the offset of the origin of each workpiece coordinate system relative to the origin of the machine tool coordinate system, and then use G5X (5X indicates the number of specific workpiece coordinate system; the same below); G5X are modal instructions, which represent 1#~6# preset workpiece coordinate systems respectively.

3. Reset the workpiece coordinate system in currently selected workpiece coordinate system through this instruction to make the current position of the tool become the coordinate value of the instruction. (The current position of the tool includes compensation of tool radius, tool length and tool position)

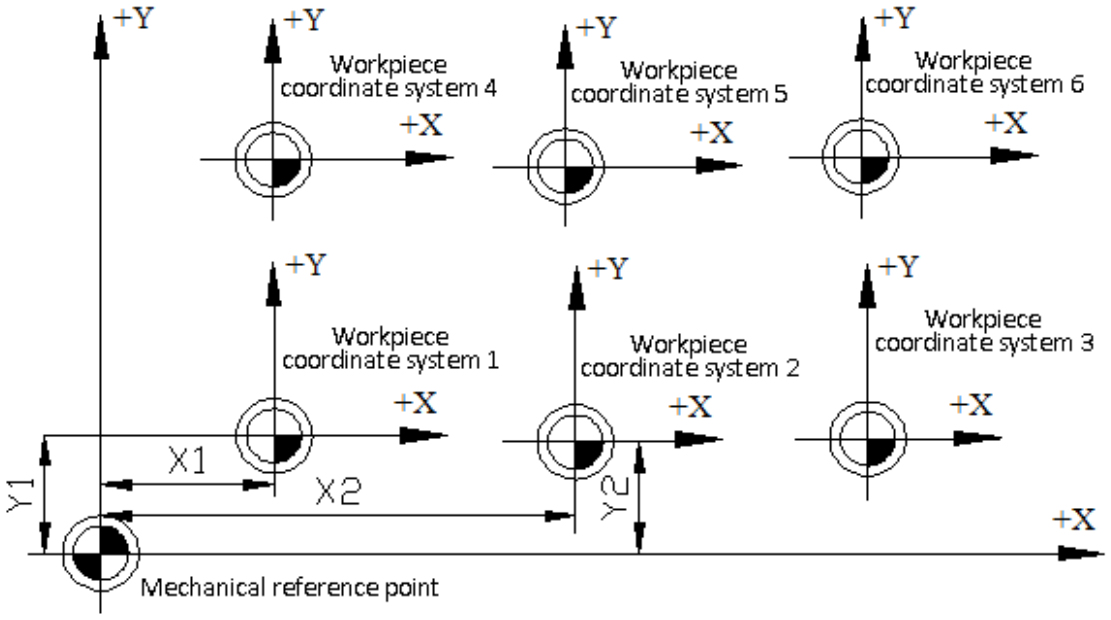


Fig. 4-2 Mechanical Coordinate System and Workpiece Coordinate System

➢ **Instruction format:**

(G90)G54 (X_ Y_ Z_)

➢ **Note:**

1. When the power is turned on, G54 coordinate system is selected by default

2. G54~G59 and G591~G599 are modal instructions

3. The reference point (origin) of workpiece coordinate system is the coordinate position of the origin in the mechanical coordinate system

4. G92 in workpiece coordinate system will move the coordinate system

5. Use G92 instruction in G54 (workpiece coordinate system 1) mode to set a new workpiece coordinate system, and other workpiece coordinate systems 2~6 (G55~G59) will also shift the workpiece coordinate system and set to new workpiece coordinate systems 2-6

➢ **Example:**

G90 G54 G17

If the origin of the workpiece coordinates of X-axis, Y-axis and Z-axis is (20, 20, 20), the current position of the system after executing abovementioned instructions is (-20, -20, -20)

Absolute coordinates and relative coordinates programming (G90, G91)

➢ **Function and purpose:**

Tool motion instructions include absolute value instruction and increment value instruction. In absolute value instruction mode, the coordinate value of the motion end in current coordinate system is specified; in increment value instruction, the distance of every coordinate axis relative to the start point motion is specified. However, the arc radius instruction specified by R and arc center instruction specified by I, J, K are always incremental instructions.

➢ **Instruction format:**

G90 X_ Y_ Z_; absolute value instruction

G91 X_ Y_ Z_; increment value instruction

➢ **Note:**

1. In absolute value G90 mode, the program running position is independent of current position and can be moved to the position of the workpiece coordinate system specified by the program.

In incremental value G91 mode, the program running position is independent of current position; when the program is running, the current location is the starting point (origin), and the value specified in the program is the relative value for movement.

The origin instructions of the workpiece coordinate system are the same coordinates in either one of absolute and incremental value mode.

2. For the next block, the program runs with the finally designated G90/G91 as the modal.

➢ **Example:**

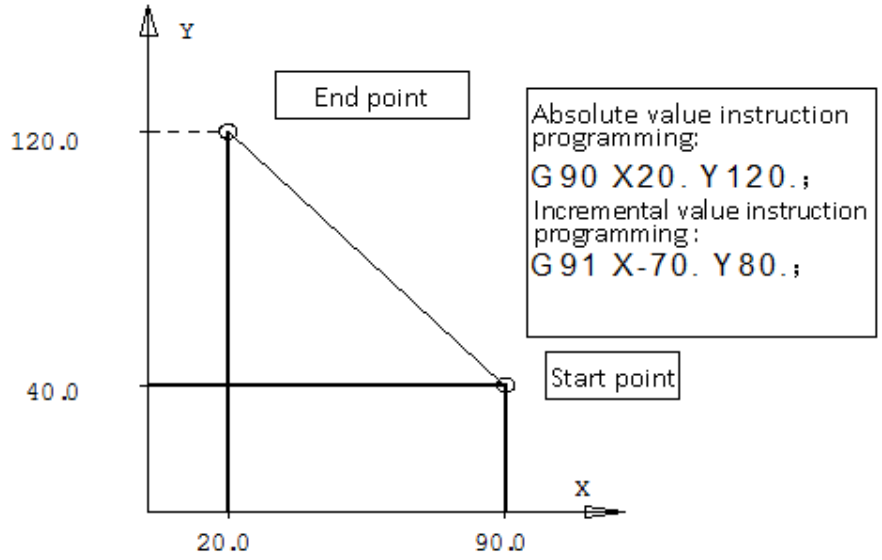


Fig. 4-3 Example Diagram

4.3 Modal function and non-modal function

➢ **Description**

The so-called modal function refers that once a code has been specified in the current block, it will remain in effect until another code of the same group appears in the block; the instruction doesn't need to be assigned in the next block.

The so-called non-modal function refers that a code only takes effect in its block. The instruction must be reassigned if it is used in next block.

➢ **Example:**

G00 and G01 are modal G code instructions

```
G01X_;
   Y_;      } G01 is valid in this range
   Z_;
G00X_;
```

4.4 Feeding function

The feeding of CNC machine tool is classified into quick positioning and cutting feeding. The quick positioning feeding appears in the motion between quick feeding and positioning during instruction G00, G28, G53, manual rapid traverse and fixed cycle. The speed of quick positioning feeding is specified by machine tool parameters. During quick positioning feeding, the motions among feeding axes are

disrelated, and move at the rapid traverse speed set by the parameters respectively. Generally, tool track is a broken line or straight line.

Cutting feeding appears in the processing feeding in G01, G02/03 and fixed cycle, and cutting feeding speed is specified by address F (unit: mm/min). In processing program, F is a modal value, i.e. original programmed F value is always valid before a new F value is specified. When KYX60A system is just electrified, F value is specified by system parameter. The axes of feeding are in interpolation relation, and the composition of their motions is cutting feeding motion.

The maximum value of F is controlled by system parameter; if the programmed F value is larger than this value, the actual feeding cutting speed is also this value. The cutting feeding speed also can be controlled by the feeding rate switch on the operation panel, and the actual cutting feeding speed is the product of specified F value and feeding rate. The range of rate is 10%-150%.

4.5 KYX60A program structure

4.5.1 Program structure

KYX60A processing program consists of the following parts:

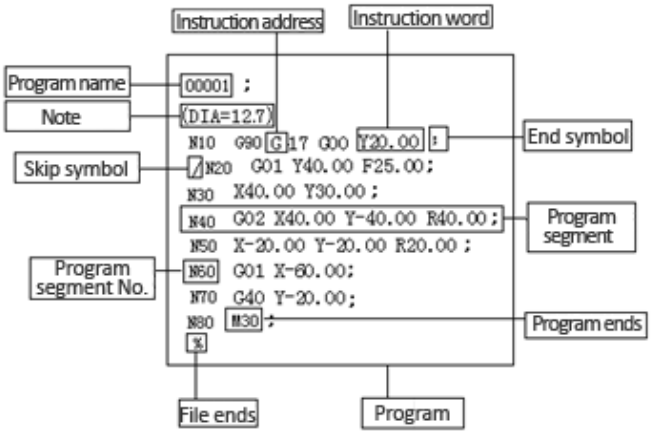


Fig. 4.5.1 KYX60A Program Structure Diagram

Program name:

Used to mark different programs, and consists of O and four digits.

If the start of the program doesn't have program name, the program segment No. of the program start will be considered as the program name by default;

If the program segment No. contains five digits, the latter four digits will be used as the program name;

If the latter four digits are 0, add 1 automatically to use as the program name;

N0 can't be used as program name;

When saving the program, if both program name and program segment No. don't exist, it is necessary to make a program name through MDI panel.

Note:

The content in the parentheses, in which the user can specify notes, guide, etc.:

The note doesn't have limit on length; if the program has a long note, the axis motion will pause for a while; therefore, if a long note is required, please put it at the place that motion pauses or without motion;

If there is only one ")" without "(", ")" will be ignored;

The note may have multiple lines and are separated with space;

During processing, the note can't be executed.

Instruction address:

One English letter in the text of the processing program ("Address" hereinafter)

Instruction word:

Adding a number after the instruction address will constitute an instruction word.

Program segment No.:

Consist of letter N and number (≤5 digits), and can be randomly arranged.

The sequence of executing program segments only related to the storage position rather than program segment No.;

If program segment N20 appears before program segment N10, N20 shall be executed first.

Program segment:

A program segment consists of one or several instruction word and ends with ";";

N_	G_	X_ Z_	F_	S_	T_	M_ ;
Program segment No.	Preparation	Size definition	Feeding speed	Spindle rotation	Tool change	Auxiliary function

Skip symbol:

If the first character of a program segment is "/", this program segment is conditional, i.e. skip switch. In upper position, this program segment isn't executed; when the skip switch is in lower position, this program segment is executed.

Program end:

Generally, the following codes are used when program ends:

Code	Action
M30	End main program
M99	End subroutine

✧ **Note:**

After M30 is executed, KYX60A stops executing and returns to program start;

After M99 is executed, KYX60A returns to the program that calls this subroutine and continues executing.

File end:

If the program end doesn't have %, KYX60A is reset.

Instruction word is the basic unit of program segment. Every address has unique meaning, and the following values also have different formats and ranges, as in the Table below:

Table 4.5.1 Instruction Address and Range of Command Value

Function	Address	Range	Meaning
Program name	O	1~9999	Program No.
Program segment No.	N	1~9999	Sequence No.
Preparation function	G	00~99	Specify motion mode (linear, arc…)
Size definition	X, Y, Z	±99999.999mm	Coordinate position value
	R	±99999.999mm	Arc radius, corner radius
	I, J, K	±9999.9999mm	Arc center coordinate position value
Feeding rate	F	1~100,000mm/min	Feeding rate
Spindle rotation	S	1~4000rpm	Spindle rotation
Select tool	T	0~99	Tool No.
Auxiliary function	M	0~99	Auxiliary function M code No.
Tool offset No.	H, D	1~200	Specify tool offset No.
Pause time	P, X	0~65sec	Pause time (ms)
Specify	P	1~9999	To call subroutine

subroutine No.			
Repeat times	P, L	1~999	To call subroutine
Parameters	P, Q, R	P is 0~99999.999 Q is ±99999.999mm R is ±99999.999	Fixed cycle parameters

4.5.2 Main program and subroutine

The processing programs include main programs and subroutines. Generally, NC executes the instructions of main program; however, NC will turn to execute subroutine when executes a subroutine calling instruction, and will return to the main program when executes the return instruction in subroutine.

When the processing program needs to run same track for several times, edit this track into the subroutine and save in the program memory of the machine tool, and this subroutine can be called when this track should be executed in the program.

When the main program calls a subroutine, this subroutine can call another subroutine, which is called double nesting. Generally, the machine tool allows up to quadruple subroutine nesting. In calling subroutine instruction, the subroutine can be repeated for 999 times.

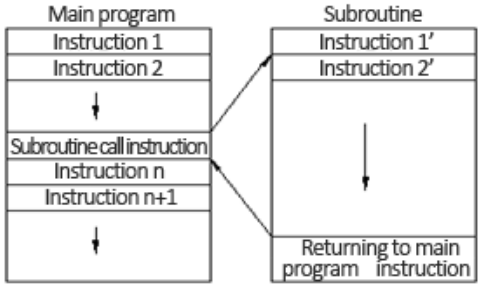


Fig. 4.5.2 Main Program and Subroutine

Subroutine format:

```
OXXXX     ; Subroutine name
……        ;
……        ; Subroutine name
……        ;
M99       ; Subroutine ends, and returns to previous program
```

Example: X100.0 Y100.0 M99;

✧ **Note:**

Program start should have a subroutine name specified by address O

M99 doesn't need to appear in a program segment separately.

Subroutine call format:

```
M98P        XXX        XXXX
```

✧ **Note:**

In the number following address P, the latter four digits are used to specify the program No. of called subroutine, and the former three digits are used to specify the repeat times of calling.

Example:

M98 P41005; call subroutine 1005, repeat four times

G90 G00 X-75. Y50. Z53. M98 P40035; this program segment specifies the X, Y, Z axis to fast locate the instruction position, and then call subroutine 0035 for four times.

✧ **Note:**

If the calling time isn't specified, the subroutine will be called only once;

M98 doesn't need to appear in a program segment separately;

Different from other M codes, M98 and M99 won't send signal to the machine tool when executing;

NC gives an alarm if can't find the program No. specified by address P;

Subroutine call instruction M98 can't be executed in MDI mode; to execute a subroutine separately, please edit the following program in the editing mode, and execute in automatic running mode.

```
O×××;
M98 P××××;
M30;
```

4.6 Standard G codes list

Table 4.6 G codes list

G code	Group	Function
*G00	01	Positioning (rapid traverse)
G01		Linear interpolation (cutting feeding)
G02		Arc interpolation CW (clockwise)
G03		Arc interpolation CCW(counterclockwise)
G04	00	Pause, accurate stop
*G17	02	XY plane selection
G18		ZX plane selection

G code	Group	Function
G19		YZ plane selection
G20	06	Imperial data entry
*G21		Metric data entry
G27	00	Return to and check reference point
G28		Return to reference point
G29		Return from reference point
*G40	07	Tool radius compensation cancel
G41		Left tool radius compensation
G42		Right tool radius compensation
G43	08	Positive tool length offset
G44		Negative tool length offset
*G49		Tool length offset cancel
G52	00	Local coordinate system setting
G53		Select machine tool coordinate system
*G54	05	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
G591		Extended workpiece coordinate system 7
G592		Extended workpiece coordinate system 8
G593		Extended workpiece coordinate system 9
G594		Extended workpiece coordinate system 10
G595		Extended workpiece coordinate system 11
G596		Extended workpiece coordinate system 12
G597		Extended workpiece coordinate system 13
G598		Extended workpiece coordinate system 14
G599		Extended workpiece coordinate system 15
G65	00	Macro program command
G73	09	Deep hole drilling fixed cycle
G74		Reverse threading fixed cycle

G code	Group	Function
G76		Boring fixed cycle
*G80		Cancel fixed cycle
G81		Drilling fixed cycle
G82		Drilling fixed cycle
G83		Deep hole drilling fixed cycle
G84		Taping fixed cycle
G85		Boring fixed cycle
G86		Boring fixed cycle
G87		Reverse boring fixed cycle
G88		Boring fixed cycle
G89		Boring fixed cycle
*G90	03	Absolute value programming
G91		Increment value programming
G92	01	Programmable workpiece coordinate system setting
*G98	10	Return to initial plane in fixed cycle
G99		Return to point R plane in fixed cycle

- **Note:**

The items marked with * are the default modal values of G codes of the system;

4.7 Positioning interpolation function (G00, G01, G02, G03)

4.7.1 Rapid positioning G00

➢ **Function and purpose**

This instruction is to locate the end point specified by the coordinates through linear or non-linear path with current point as the starting point.

➢ **Instruction format:**

G00 X_ Y_ Z_α_; (α is additional axis)

X Y Z α is coordinate value; absolute or increment programming mode is determined according to G90 or G91 state specified by the program.

➢ **Note:**

1. The motion axis specified by G00 moves to the specified location at rapid traverse rate, the movement between designated axes is unrelated, that is, tool movement trajectory is a straight line or polygonal line.

2. Under G00 instruction, the motion axis moves at the speed set in the system; the speed isn't controlled by

current F value. When motion axis reaches the end point, KYX60A considers that the block has ended and turns to the next block.

3. After G00 is specified, G00 mode remains before changing other G functions of G00 mode, that is, before issuing G01, G02, and G03of group 01. Therefore, if the instruction is still G00, just specify the coordinate value of motion axis.

4. G function of group 09 (G71~G89) is canceled (G80) mode in G00 instruction.

➢ **Example:**

The position of start point is X-50, Y-75; instruction G00 X150. Y25; the tool will have the track shown in Fig. 4-5.

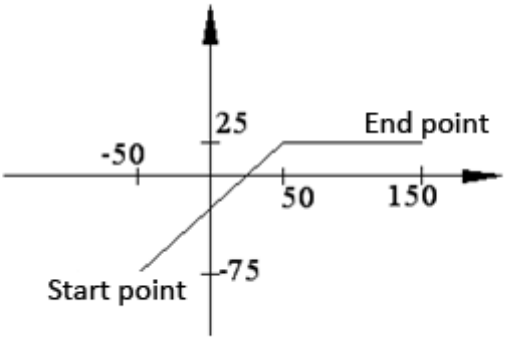


Fig. 4-5 G00 Programming Diagram

4.7.2 Linear interpolation G01

- **Function and purpose:**

This instruction is to move the tool from current point to the end point specified by the coordinates in linear (interpolation) at the speed specified in address F by combining the coordinates and the feed speed. The feed rate specified by address F always plays as the line speed in the feeding direction relative to the tool center.

- **Instruction format:**

G01 X_ Y_ Z_ α_ F_; (α is additional axis) X Y Z α is coordinate value; absolute or increment programming mode is determined according to G90 or G91 state specified by the program. F indicates the speed of linear motion (unit: mm/min)

- **Note:**

1. The motion axis specified by G01 moves from current point to the end point specified by the coordinates in linear (interpolation). It is synchronous motion between specified motion axes.

2. After G01 is specified, G01 mode remains before changing other G functions of G01 mode, that is, before issuing G00, G02, and G03 of group 01. Therefore, if the instruction is still G01 and the feeding speed doesn't change, just specify the coordinate value of motion axis.

3. G function of group 09 (G71~G89) is canceled (G80) mode in G00 instruction.

- **Example:**

The feeding speed of rotation axis is expressed in °/min. (F300=300 °/min)

Suppose that the current point of the tool is X-50. Y-75., the following program segment

N1 G01 X150. Y25. F100;

N2 X50. Y75;

will make the tool have the track shown in the figure below.

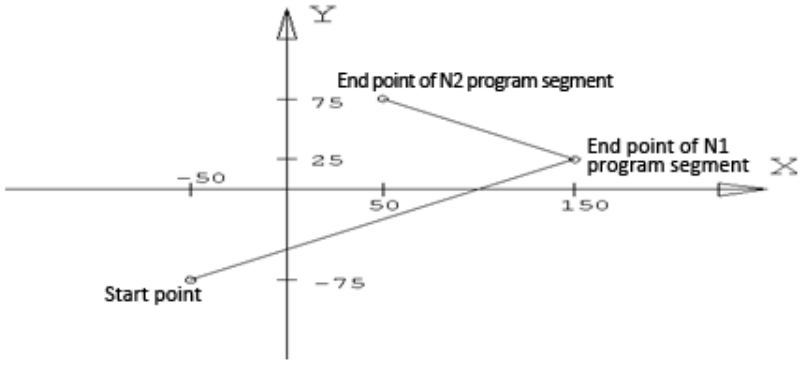


Fig. 4-6 G01 Programming Diagram

4.7.3 Arc interpolation (G02/G03)

- **Function and purpose:**

Used to move the tool in arc track. Two programming methods are available: specify the vector from start point of the arc and radius.

- **Instruction format:**

XY plane (G17):

G02/G03 X_ Y_ I_ J_ F_;

G02/G03 X_ Y_ R_ F_;

XZ plane (G18):

G02/G03 X_ Z_ I_ J_ F_;

G02/G03 X_ Z_ R_ F_;

YZ plane (G19):

G02/G03 Y_ Z_ I_ J_ F_;

G02/G03 Y_ Z_ R_ F_;

The parameters are described in the following table:

Table 4.6.1 Arc Interpolation Command Format Description

S/N	Data content		Instruction	Meaning
1	Plane selection		G17	Specify the arc interpolation on X-Y plane
			G18	Specify the arc interpolation on Z-X plane
			G19	Specify the arc interpolation on Y-Z plane
2	Arc direction		G02	Arc interpolation in clockwise direction CW
			G03	Arc interpolation in counterclockwise direction CCW
3	End point	G90 mode	Two axes instruction in X, Y, Z	The coordinate value of the end point position in current workpiece coordinate system
		G91 mode	Two axes instruction in X, Y, Z	Distance from start point to end point (directional)
4	Distance from start point to circle center			Distance from start point to circle center (directional)
	Arc radius			Arc radius
5	Feeding rate			The speed of arc motion

➢ **Note:**

G02 (G03) is modal instruction.

The arc crossing multiple quadrants can be specified in one program segment.

Arc direction
X-Y plane: look to negative direction from Z axis
X-Z plane: look to negative direction from Y axis
Y-Z plane: look to negative direction from X axis

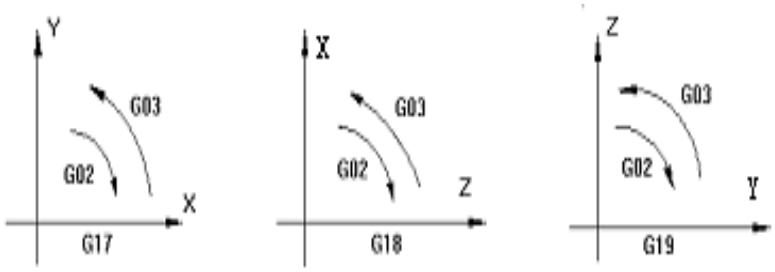


Fig. 4-7 G02/G03 Arc Interpolation

The end point of the arc is determined by address X, Y and Z. In G90 mode, i.e. absolute value mode, address X, Y and Z specify the coordinate value of arc end in current coordinate system; in G91 mode, i.e. increment value mode, address X, Y and Z specify the distance from the point of current tool to the end point in the direction of every axis.

In X direction, the address I gives the vector value from current point of the tool to the center; in Y and Z direction, the vector value from current point of the tool to the center is specified by address J and K respectively.

To program a segment of arc, in addition to specifying end point and circle center position, it is also possible by specifying radius and end point position. If the radius is specified with address R, the value of R can be either positive or negative; a positive R value can be used to determine an arc smaller than 180 °, and a negative value can be used to determine an arc larger than 180 °. Programming a full circle is only possible by specifying circle center.

➢ **Example:**

G02 J50 F500;
G91 G02 X50 Y50 J50 F500;
The processing tracks are shown in the figures below (full circle and 3/4 arc)

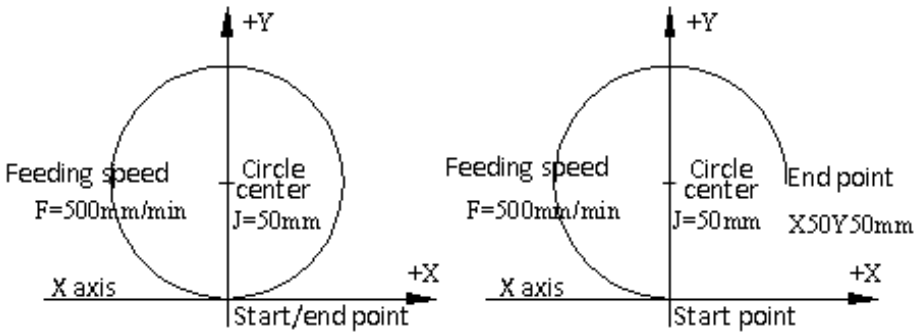


Fig. 6.4 Instruction Diagram of Processing Full Circle

To program a segment of arc, in addition to specifying end point and circle center position, it is also possible by specifying radius and end point position. If the radius is specified with address R, the value of R can be either positive or negative; a positive R value can be used to determine an arc smaller than 180°, and a negative value can be used to determine an arc larger than 180°. Programming a full circle is only possible by specifying circle center.

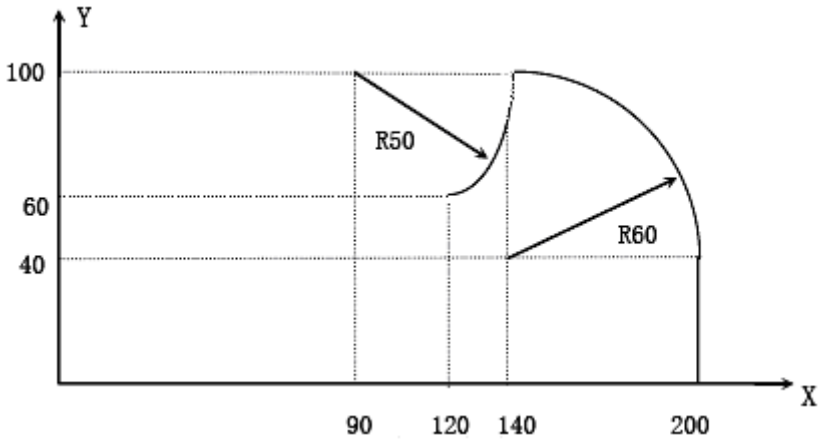


Fig. 6.5 Absolute/Increment Programming Diagram

Above tracks are programmed in absolute value and increment value mode as follows:

- Absolute value mode

```
G00 X200.0 Y40.0 Z0    ;
G90 G03 X140.0 Y100.0 I-60.0 F300.0    ;
G02 X120.0 Y60.0 I-50.0    ;
or:

G00 X200.0 Y40.0 Z0    ;
G90 G03 X140.0 Y100.0 R60.0 F300.0    ;
```

```
G02 X120.0 Y60.0 R50.0    ;
```

➢ Increment mode

```
G91 G03 X-60.0 Y60.0 I-60.0 F300.0    ;
G02 X-20.0 Y-40.0 I-50.0    ;
or:
G91 G03 X-60.0 Y60.0 R60.0 F300.0    ;
G02 X-20.0 Y-40.0 R50.0    ;
```

The feeding speed of arc interpolation is specified with F, which is the speed of tool in arc tangent direction.

4.8 Pause instruction (G04)

➢ **Function and purpose:**

Pause for a period of time between two program segments.

➢ **Instruction format:**

G04 P_ or G04 X_

Address P specifies the pause time, and the minimum unit of its instruction is 0.001 second if there is no radix point.

Address X specifies the pause time, and the minimum unit of its instruction is 1 second if there is no radix point.

➢ **Example:**

- G04 P 1000 : pause for 1000ms, equal to 1sec
- G04 X 1 : pause for 1sec

4.9 Plane selection (G17, G18, G19)

➢ **Function and purpose:**

Specify the plane in which the tool moves. Plane selection is used for arc interpolation plane, tool radius compensation plane and fixed cycle positioning plane.

➢ **Instruction format:**

G17……select XY plane

G18……select ZX plane

G19……select YZ plane

X, Y, Z indicate the coordinate axes or parallel axes

➢ **Note:**

In the program segment without instruction G17, G18 or G19, the plane doesn't have any change.

- **Example:**

```
G18 X_ Z_   ; ZX plane
X_ Y_   ; plane doesn't change (ZX plane)
```

Motion instruction is disrelated to plane selection.

```
G17 Z_   ;
Z axis doesn't exist on XY plane, and Z axis motion is disrelated to XY plane.
```

About the instructions related to plane selection, please refer to the content related to arc interpolation and tool compensation instructions.

4.10 Coordinate system instructions (G54~G59, G591~G599, G52)

4.10.1 Preset workpiece coordinate system (G54~G59, G591~G599)

- **Function and purpose:**

1. Workpiece coordinate system is used in program processing. It sets a reference point on the workpiece to the coordinate origin and is used to simplify the programming of workpieces.

2. You can move the tool to the position of the workpiece coordinate system through this instruction. The workpiece coordinate system is used by the programmer for programming. The system allows presetting six workpiece coordinate systems and nine extension coordinate systems (G591-G599). Set the offset of the origin of each workpiece coordinate system relative to the origin of the machine tool coordinate system, and then use G5X (5X indicates the number of specific workpiece coordinate system; the same below); G5X are modal instructions, which represent 1#~6# preset workpiece coordinate systems respectively.

3. Reset the workpiece coordinate system in currently selected workpiece coordinate system through this instruction to make the current position of the tool become the coordinate value of the instruction. (The current position of the tool includes compensation of tool radius, tool length and tool position)

- **Instruction format:**

(G90)G54 (X_ Y_ Z_)

- **Note:**

1. When the power is turned on, G54 coordinate system is selected by default

2. G54~G59 and G591~G599 are modal instructions

3. The reference point (origin) of workpiece coordinate system is the coordinate position of the origin in the mechanical coordinate system

4. G92 in workpiece coordinate system will move the coordinate system

5. Use G92 instruction in G54 (workpiece coordinate system 1) mode to set a new workpiece coordinate system, and other workpiece coordinate systems 2~6 (G55~G59) will also shift the workpiece coordinate

system and set to new workpiece coordinate systems 2-6

6. The system allows presetting 15 workpiece coordinate systems (six workpiece coordinate systems: G54~G59, and nine extension coordinate systems: G591~G599); set the offset of the origin of each workpiece coordinate system relative to the origin of machine tool coordinate system, and then use G54~G59 and G591~G599 instructions to select; G54~G59 and G591~G599 are modal instructions, corresponding to 1#~15# preset workpiece coordinate systems respectively.

- **Example:**

  Offset of preset 1# workpiece coordinate system: X-150.000 Y-210.000 Z-90.000.

  Offset of preset 4# workpiece coordinate system: X-430.000 Y-330.000 Z-120.000.

Table 4-10.1 Example of Preset Workpiece Coordinate System

Program segment content	Coordinates of end point in machine tool coordinate system	Note
N1 G90 G54 G00 X50. Y50.;	X-100, Y-160	Select 1# coordinate system, quick positioning
N2 Z-70.;	Z-160	
N3 G01 Z-72.5 F100;	Z-160.5	Linear interpolation, F value is 100
N4 X37.4;	X-112.6	(Linear interpolation)
N5 G00 Z0;	Z-90	Quick positioning
N6 X0 Y0 A0;	X-150, Y-210	
N7 G53 X0 Y0 Z0;	X0, Y0, Z0	Select to use machine tool coordinate system
N8 G57 X50. Y50. ;	X-380, Y-280	Select 4# coordinate system
N9 Z-70.;	Z-190	
N10 G01 Z-72.5;	Z-192.5	Linear interpolation, F value is 100 (modal value)
N11 X37.4;	X392.6	
N12 G00 Z0;	Z-120	
N13 G00 X0 Y0 ;	X-430, Y-330	

Seen from above samples, the function of G54~G59 instruction is to move the coordinate origin used by NC to the point that the coordinates in machine tool coordinate system are preset value; please refer to the operation section in this manual for the method of presetting.

After returning to the home of machine tool, coordinate systems 1~6 of the workpiece are created. G54 is the initial mode after electrified. The absolute position of the position screen is the coordinates in current coordinate system.

In CNC programming of machine tool, unless otherwise specified, X, Y, Z of interpolation instruction and other instructions related to coordinates are the coordinate position in current coordinate system (the coordinate system used when the instruction is executed). In most cases, the current coordinate system is one of G54~G59, and machine tool coordinate system are seldom used directly.

4.10.2 Programmable workpiece coordinate system (G92)

- **Function and purpose:**

  When G92 is specified, the machine does not move, but the coordinate system of absolute coordinate values and the current position will be preset according to the instruction.

- **Instruction format:**

  (G90)G92 X_Y_Z_;

- **Note:**

G92 instruction is a non-modal instruction, but the workpiece coordinate system created with this instruction is modal.

Actually, this instruction also specifies an offset, which is specified indirectly. It is the coordinate value of new workpiece coordinate system origin in original workpiece coordinate system; seen from G92 function, this offset is the difference between the coordinate value of the tool in original workpiece coordinate system and IP_ instruction value (as shown in Fig. 8.1)

If G92 instruction is used for several times, the offset specified by G92 instruction will superpose. For every preset workpiece coordinate system (G54-G59), the superposed offset is valid.

New coordinate system of the part is set in above instruction, e.g. the coordinate value of tool tip is IP_. Once the coordinates are confirmed, the position of the absolute value instruction is the coordinates in this coordinate system.

- **Example:**

  The coordinates of the tool in original coordinate system are (200, 100), after executing (G92 X100 Y50):
  The origin of new coordinate system offsets to the position A in the lower right figure;
  The offset of coordinate system is (100, 50), (the difference between the coordinates of the tool in original coordinate system and IP_ instruction value).
  The coordinates of the tool in new coordinate system are (100, 50).

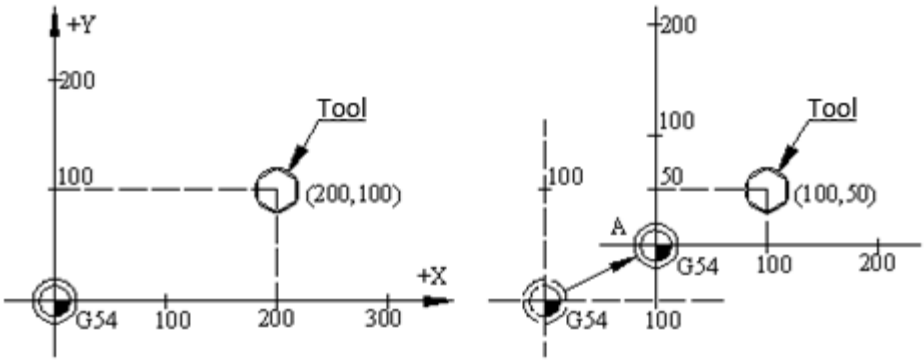


Fig. 8.1 G92 Instruction Function Diagram

4.10.3 Local coordinate system (G52)

- **Instructio Function:**

G52 can create a local coordinate system, which is a sub-coordinate system equivalent to G54~G59.

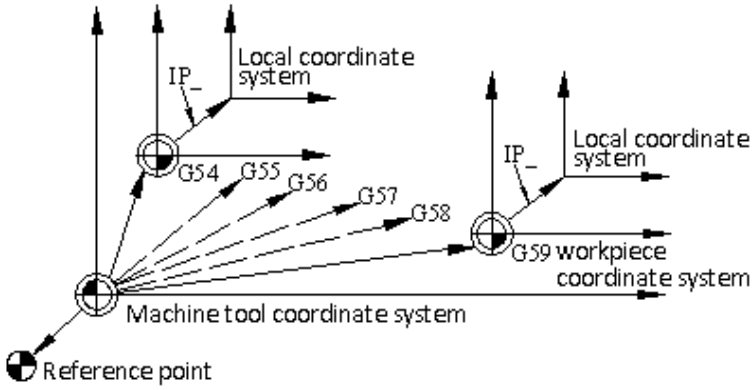


Fig. 8.2 Local Coordinate System Diagram

Instruction format:

G52 X_Y_Z_;	
X_Y_Z_;	Offset relative to current G54~G59 coordinate systems,

Note:

In this instruction, IP_ specifies the offset equivalent to current G54~G59 coordinate systems, i.e. IP_ specifies the position coordinates of local coordinate system origin in current G54~G59 coordinate system.

G52 instruction is always valid after specified until next G52 instruction is specified.

G52 instruction can set the processing coordinate system without changing the workpiece coordinate system.

G52 IP0 (G52 X0 Y0 Z 0 α0) can be used to cancel local coordinate system.

The setting of local coordinate system doesn't change the machine tool coordinate and workpiece coordinate system.

G52 instruction can replace G92 instruction to specify the offset between the origin of processing program and workpiece origin.

Example:

Local coordinate system in absolute value mode

①G28 X0 Y0;	
②G00 G90 X1000 Y1000;	
③G92 X0 Y0;	Define workpiece coordinate system
④G00 X500 Y500;	
⑤G52 X1000 Y1000;	Define local coordinate system
⑥G00 X0 Y0;	
⑦G01 X500 F100;	
⑧Y500;	
⑨G52 X0 Y0;	Cancel local coordinate system
⑩G00 X0 Y0;	

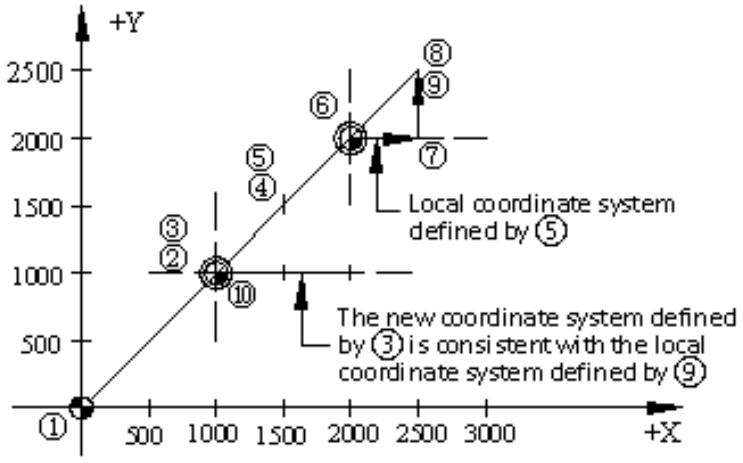


Fig. 8.3 Local Coordinate System Usage Diagram in Absolute Value Mode

4.11 Operation related to reference point

The machine tool coordinate system is created through returning to reference point after NC is electrified every time. The reference point is a fixed point on the machine tool, and its position is determined by the installation position of stopper switch of every axis and the home position of the servo motor of every axis. When this machine tool returns to the reference point, the coordinates of the

reference point in the machine tool coordinate system is X0, Y0, Z0.

4.11.1 Auto return to reference point (G28)

➢ **Instructio Function:**

This instruction makes the axis return to reference point of the machine tool through the center point specified by IP at the feeding speed of quick positioning.

Instruction format:

G28 X_ Y_ Z_α_; (α is additional axis) X Y Z α indicate the coordinates of center point.

➢ **Note:**

The center point may be specified either in absolute value mode or increment value mode, which depends on current mode.

Generally, this instruction is used to move the workpiece out of the processing area when the entire processing program ends, so as to unload processed parts and load the parts to be processed.

When execute G28 instruction before returning to reference point manually, the motion of every started from center point is same as returning to reference point manually, and the motion direction started from the center point is positive.

The coordinates in G28 instruction is saved as center point by NC; on another hand, if an axis isn't contained in G28 instruction, the coordinates of the center pointed saved by NC will use the value G28 instruction specified previously.

➢ **Example:**

```
N0010   X20.0   Y54.0;
N0020   G28   X-40.0   Y-25.0;   coordinates of center point (-40.0,-25.0)
N0030   G28   Z31.0;          coordinates of center point (-40.0,-25.0,31.0
```

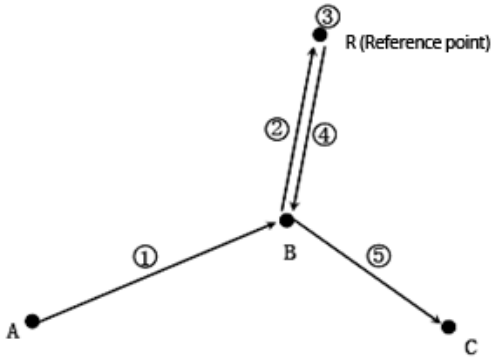


Fig. 8.2 Diagram of Automatically Returning to Reference Point

✧ **Caution:**

The coordinates of this center point are mainly used by G28 instruction.

In tool offset mode, tool offset is also valid for G27; For safety reasons, tool offset should be disabled before executing G28 instruction (radius offset and length offset).

4.11.2 Auto return from reference point (G29)

- **Instructio Function:**

This instruction makes the axis move from reference point to instruction position through center point at the feeding speed of quick positioning; the position of center point is confirmed by previous G28 instruction.

Instruction format:

G29 X_ Y_ Z_α_; (α is additional axis)
X Y Z α indicate the coordinates of end point of the tool motion.

Note:

Generally, after this instruction is used for G28, the instructed axis is on reference point or second reference point.

In increment value mode, the instruction value is the distance from center point to end point (instruction position).

In program, the specific movement amount from center point to reference point doesn't need to be calculated.

Example:

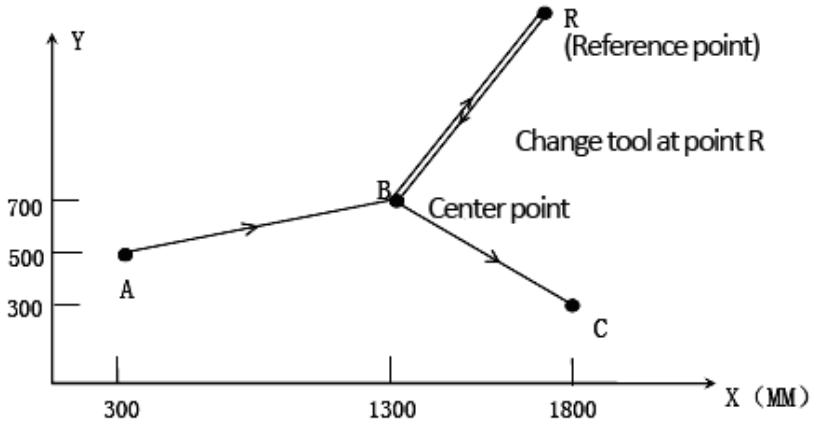


Fig. 8.3 G28, G29 Usage Diagram

```
G28 X1300.0 Y700.0   ;        (A→B program)
……
G29 X1800.0 Y300.0   ;        (B→C program)
```

✧ **Notice:**

When change part coordinate system after moving to reference point through center point with G28 instruction, the center point also moves to new coordinate system; when instruct G29 later, positioning at instructed position through center point in new coordinate system.

4.11.3 Reference point return checking (G27)

➢ **Instructio Function:**

This instruction makes the axis move to the position of IP instruction at the feeding speed of quick positioning, and then checks whether this point is reference point; if yes, sends the finishing signal that this axis returns to reference point (reference point arriving indicator of this axis is lighted); if not, gives an alarm and interrupts the running program.

➢ Instruction format:

G27 X_ Y_ Z_ P_;
X Y Z indicate that reference point returns to control axis.
P reference point returns number (the first reference point by default)

➢ Note:

The axes of simultaneous reference point return check are same to simultaneously controlled axes.

If the reference point isn't reached after instruction is executed, the program alarms.

4.12 Tool compensation function (G43, G44, G49)

➢ **Instructio Function:**

Assume the difference between tool length and actual tool length when correct the programming.

➢ **Instruction format:**

G43 Z_ H_; positive offset
G44 Z_ H_; negative offset
G49 Z_; (or H00) tool length compensation cancel

Move the end point position of Z axis instruction for an offset according to above instruction, and preset the difference between tool length and the tool length of actual processing assumed during programming in offset memory, and therefore the operator only needs to change the tool compensation to process parts with tools of different lengths without changing the program.

➢ **Note:**

In either absolute value or increment value mode, for G43, add the offset specified by H code (set in offset memory) to Z axis motion instruction end point coordinates in the program; for G44, subtract the offset specified by H code, and use the calculated coordinates as the end point coordinates.

When Z axis motion is omitted, if the offset is positive, G43 instruction will move an offset in positive direction and G44 will move an offset in negative direction. If the offset is negative, it moves to reverse direction.

G43 and G44 are modal G codes, which are always valid before the G codes of same group appear.

Specifying offset:

H code specifies the offset No., the corresponding offset will add or subtract Z axis motion instruction when the program is running, and thus creates new motion instruction of Z axis. Offset No. can be specified between H00 and H18, while the offset corresponding to H00 can't be set to static 0.

Enter tool compensation menu, and preset the offset to corresponding offset No. in the offset memory.

	Mm input	inch input
Offset	0-±999.999	0-±99.9999

Cancel tool length compensation:

Cancel tool length compensation with G49 or H00.

- **Example:**

Tool compensation processing (hole #1, #2, #3)

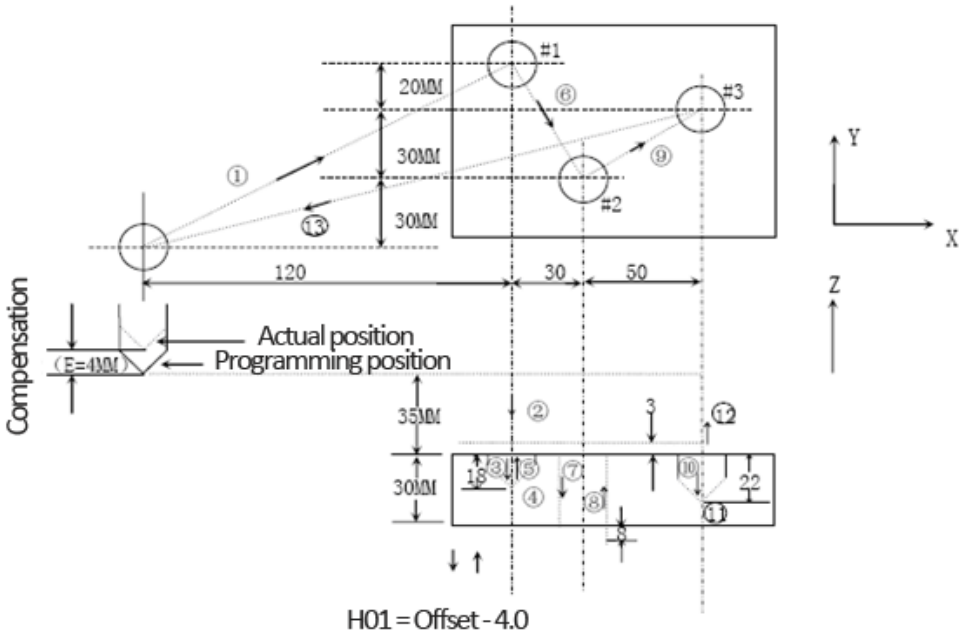


Fig. 9.1 Tool Compensation Processing Hole Example

```
N1 G91 G00 X120.0 Y80.0; ……  (1)
N2 G43 Z-32.0 H01; ……  (2)
N3 G01 Z-21.0; ……  (3)
N4 G04 P2000; ……  (4)
N5 G00 Z21.0; ……  (5)
```

```
N6 X30.0 Y-50.0; ……  (6)
N7 G01 Z-41.0;  ……  (7)
N8 G00 Z41.0;  ……  (8)
N9 X50.0 Y30.0; …… ……  (9)
N10 G01 Z-25.0;  ……  (10)
N11 G04 P2000;  ……  (11)
N12 G00 Z57.0 H00;  ……  (12)
N13 X-200.0 Y-60.0; ……  (13)
N14 M30;
```

✧ **Notice:**

When the offset No. is changed, it only changes to new offset, rather than adding the new offset to the old offset.

```
H01……offset 20.0
H02……offset 30.0
G90 G43 Z100 0 H01……Z moves to 120.0
G90 G43 Z100 0 H02……Z moves to 130.0
```

4.13 Hole processing function (G73~G89)

With hole processing fixed cycle, the functions that require several segments in other method can be finished in one segment. Table 10.1 lists all hole processing fixed cycles.

Table 10.1: Hole Processing Fixed Cycle

G code	Processing motion (Z axis negative)	Hole bottom action	Return motion (Z axis positive)	Application
G73	Sub, cutting feeding	-	Quick positioning feeding	High speed deep hole drilling
G80	-	-	-	Cancel fixed cycle
G81	Cutting feeding	-	Quick positioning feeding	Common drilling cycle
G82	Cutting feeding	Pause	Quick positioning feeding	Drilling to rough boring
G83	Sub, cutting feeding	-	Quick positioning feeding	Deep hole drilling cycle
G84	Cutting feeding	Pause - Spindle reverse rotation	Cutting feeding	Right thread taping
G85	Cutting feeding	-	Cutting feeding	Boring cycle
G86	Cutting feeding	Spindle stop	Quick positioning feeding	Boring cycle
G88	Cutting feeding	Pause- Spindle stop	Manual	Boring cycle
G89	Cutting feeding	Pause	Cutting feeding	Boring cycle

➢ **Instruction format:**

After G73/G74/G76/G81~G89, give hole processing parameters,
The format follows: (See Table 10.2 for details)
G××X_ Y_ Z_ R_ Q_ P_ F_ K_ ;
G×× : hole processing method
X_ Y_ Z_ : position parameters of hole being processed
R_ Q_ P_ F_ : hole processing parameters
K_ : repeat times

➢ **Details:**

Generally, one hole processing fixed cycle completes the following six steps (see Fig. 10.1):

G73/G74/G76/G81~G89,
1. X, Y axis quick positioning. 2. Z axis quickly positions to point R. 3. Hole processing. 4. Hole bottom action. 5. Z axis returns to point R. 6. Z axis quickly returns to the start point.

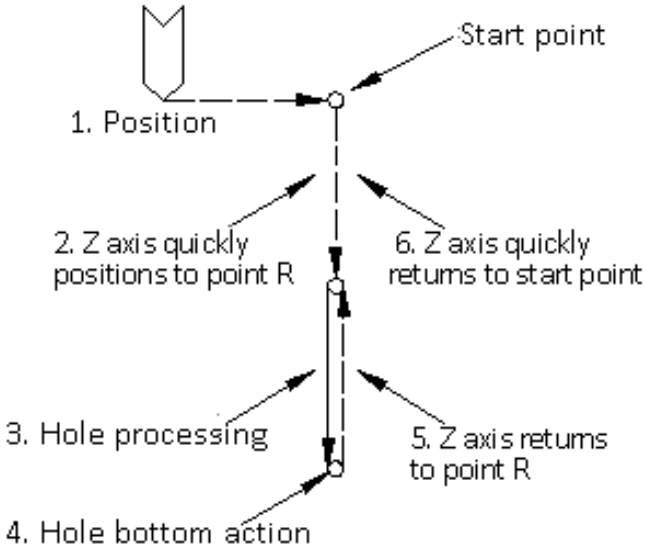


Fig. 10.1 Six Steps of Hole Processing Fixed Cycle

The instructions that have influence on the execution of hole processing fixed cycle instruction include G90/G91 and G98/G99. Fig. 10.2 shows the effect of G90/G91 on hole processing fixed cycle instruction.

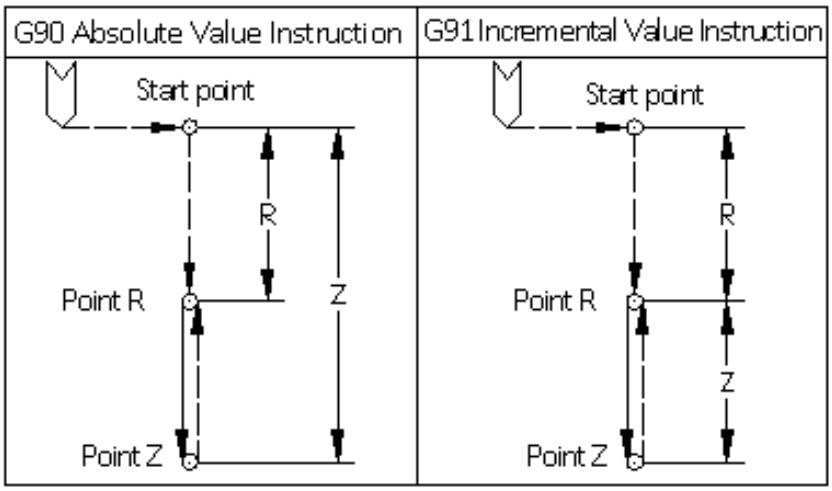


Fig. 10.2 Effect of G90/G91 on Hole Processing

G98/G99 determines fixed cycle returns to point R or the start point after hole processing; in G98 mode, Z axis returns to the start point after hole processing; in G99 mode, it returns to point R.

Generally, if the holes being processed are on a flat plane, we can use G99 instruction, because it will position next hole after returning to point R in G99 mode; in general programming, point R is close to workpiece surface, it will shorten part processing time; but if the workpiece surface has convex platform or tendon, the tool and workpiece may collide if G99 is used; at this moment, G98 should be used to return Z axis to the start point and then position next hole to ensure the safety. See the figure below.

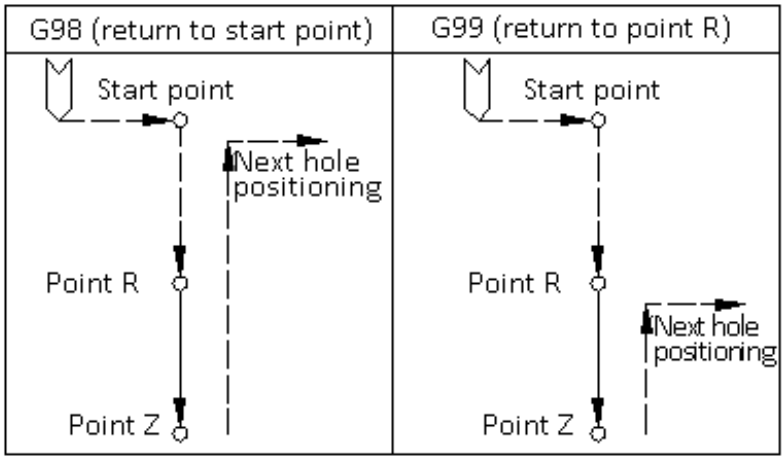


Fig. 10.3 Effect of G98/G99 on Hole Processing

Table 10.2 Meaning of Every Address in Hole Processing Fixed Cycle

Address	Meaning
Position parameter X, Y of holes being processed	Specify the position of the hole being processing in increment or absolute mode; the track and speed of the tool moving to processed hole are same to G00
Position parameter Z of holes being	In absolute value mode, specify the position of hole bottom in Z axis direction; in increment value mode, specify the distance from point R to hole bottom

Address	Meaning
processed	
Hole processing parameter R	In absolute value mode, specify the position of point R in Z axis direction; in increment value mode, specify the distance from the start point to point R
Hole processing parameter Q	Used to specify the tool feeding of deep hole drilling cycle G73 and G83, and the offset of fine boring cycle G76 and reverse boring cycle G87 (always increment value instruction no matter G90 or G91 mode)
Hole processing parameter P	Used to specify the pause time (unit: sec) in the fixed cycle that hole bottom action has pause
Hole processing parameter F	Used to specify the cutting feeding speed in fixed cycle; in the fixed cycle, the motion from start point to point R and from point R to start point executes in the speed of quick feeding, the motion from point R to point Z executes in the cutting feeding speed specified by F, while the motion from point Z to point R executes either in the speed specified by F or quick feeding speed.
Repeat times K	Specify the repeat times of fixed cycle in current positioning point; if K isn't specified, NC considers that K=1; if K is specified as 0, the fixed cycle won't be executed at current point.

The hole processing specified by Gxx is modular, and the fixed cycle can be canceled with G80 or 01 G instruction.

Hole processing parameter is also modular, and will be retained before changed or fixed cycle is canceled, even hole processing mode is changed.

A hole processing parameter can be specified or changed when specifying a fixed cycle or at any moment in the fixed cycle.

Repeat times K isn't a modular value, and is only specified when required.

Feeding speed F is a modular mode, and it will be retained even the fixed cycle is canceled.

If NC system is reset while executing the fixed cycle, hole processing mode, hole processing parameter and repeat times K are canceled.

The following example describes above content better.

SN	Program content	Remark
1	S_ M03	Specify the rotation, and specify the spindle to rotate positively
2	G81X_ Y_ Z_ R_ F_ K_	Locate specified X, Y point quickly, process with the hole processing parameter specified by Z, R, F and in the hole processing mode specified by G81, and repeat for K times; when the fixed cycle is started, Z, R, F are necessary hole processing parameters.
3	Y_	X axis doesn't move, Y axis quickly positions the instruction point and processes the hole; the hole processing parameter and hole processing mode retain the modular value in 2. The K value in 2 is invalid.
4	G82X_ P_ K_	Hole processing mode is changed; hole processing

		parameter Z, R, F retain the modular value, specify the value of hole processing parameter P and specify the repeat times K.
5	G80X_ Y_	The fixed cycle is canceled, and all hole processing parameters except F are canceled.
6	G85X_ Y_ Z_ R_ P_	Since the fixed cycle has been canceled when executing 5, all necessary hole processing parameters except F must be re-specified, even if these parameters aren't changed.
7	X_ Z_	X axis positions the instruction point and processes the hole, and hole processing parameter Z is changed in this segment.
8	G89X_ Y_	Position the XY instruction point and process the hole, and the hole processing mode is changed to G98. R, P are specified by 7, and Z is specified by 7.
9	G01X_ Y_	The fixed cycle mode is canceled, and all hole processing parameters except F are canceled.

In the following diagrams, we use the modes below to indicate the feeding of every segment:

Indicate motion in quick feeding speed ·····→
Indicate motion in cutting feeding speed ——→
Indicate manual feeding - - - - - - - - - - →

4.13.1 High-speed deep-hole drilling cycle (G73)

- **Instruction format:**

Format: G73 X_ Y_ Z_ R_ Q_ F_

- **Details:**

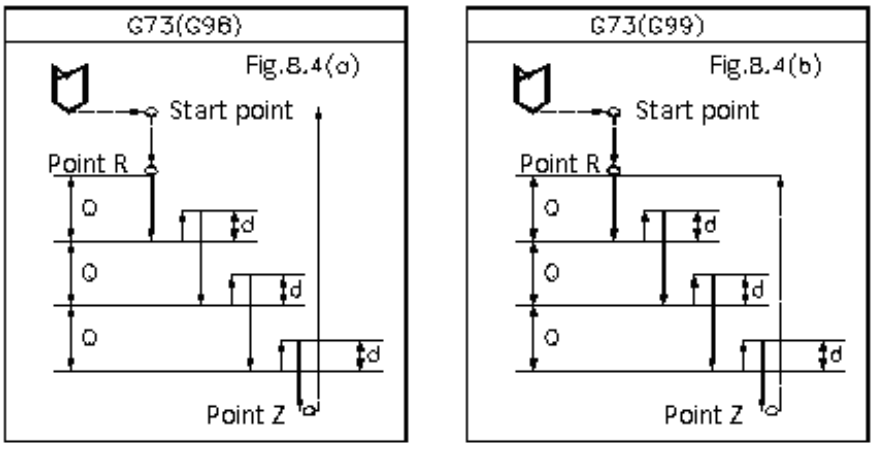


Fig. 10.4 High Speed Deep Hole Drilling Cycle Diagram

The feeding from point R to point Z is finished in several segments; after cutting every segment, Z axis lifts for certain distance, and then executes cutting feeding for next segment.

The distance that Z axis lifts every time is d, which is specified by parameter 531#, and the depth of every feeding is determined by hole processing parameter Q.

This fixed cycle is mainly used for processing holes with small diameter/depth ratio (e.g. Φ5, depth 70), and Z axis lift has the effect of chip breaking after cutting and feeding every segment.

4.13.2 Reverse-threading cycle (G74)

- **Instruction format:**

Format G74 X_ Y_ Z_ R_ F_(D_)
X_Y_ : thread position
Z_ : thread depth
R_ : start point of tool feeding/retreating
F_(D_) : convert feeding speed according to screw distance, or specify the distance with D_ directly

- **Details:**

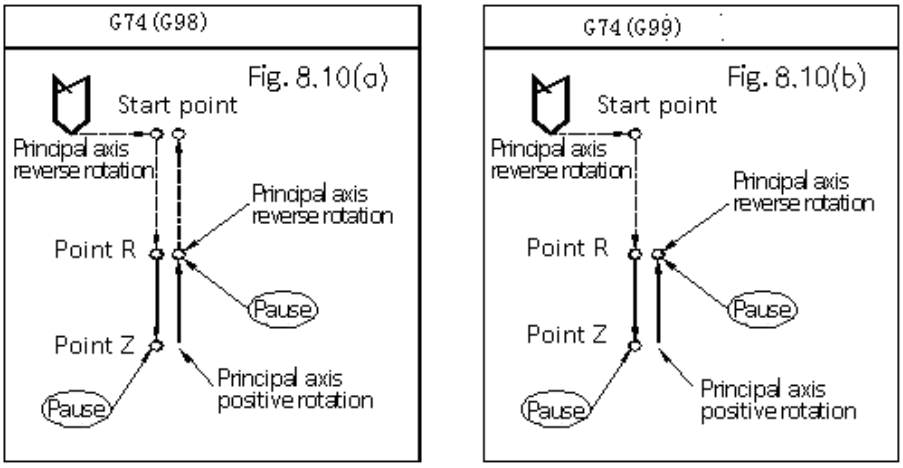


Fig. 10.5 Reverse Taping Cycle Diagram

- **Notice:**

In G74 and G84 cycle, the feeding rate switch and feeding retaining switch are ignored, i.e. feeding rate is retained at 100%, and can't be stopped before a fixed cycle completes; before cycle starts, the spindle should be specified to rotate in taping direction.

4.13.3 Cancel fixed cycle (G80)

After G80 instruction is executed, the fixed cycle (G73, G74, G81~g89) instruction is canceled, point R and point Z parameter and all hole processing parameter except F are canceled. In addition, the G codes of group 01 also have the same effect.

10.5 Drilling cycle (G81)

➢ **Instruction format:**

Format G81 X_ Y_ Z_ R_ F_

➢ **Details:**

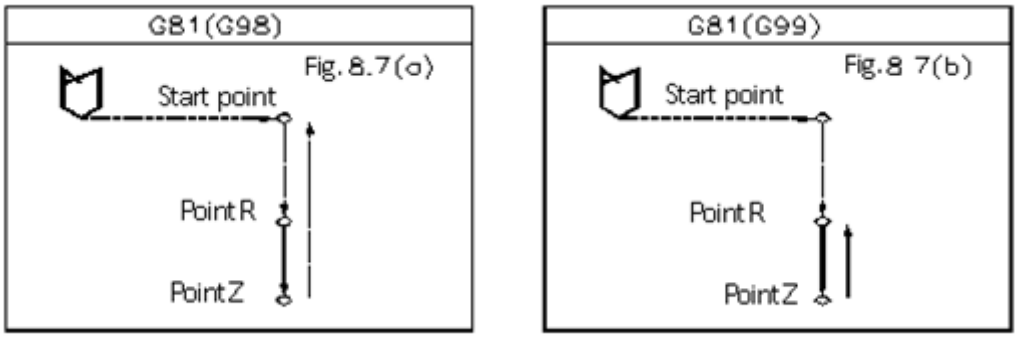


Fig. 10.6 Drilling Cycle Diagram

✧ **Note:**

G81 is the simplest fixed cycle, and its execution process follows:

X, Y positioning, Z axis moves to point R quickly, and feeds to point Z at F speed, Quickly returns to the start point (G98) or point R (G99), No hole bottom action

4.13.4 Drilling cycle, rough boring cycle (G82)

➢ **Instruction format:**

G82 X_ Y_ Z_ R_ P_ F_

➢ **Details:**

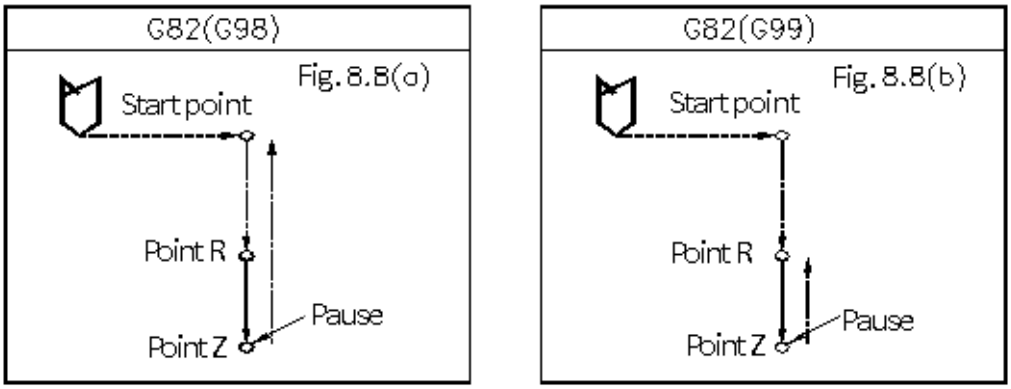


Fig. 10.7 Drilling Cycle, Rough Boring Cycle Diagram

✧ **Note:**

G82 fixed cycle has a pause action in the hole bottom, and others are same to G81. The pause of hole bottom can improve the precision of hole depth.

4.13.5 Deep-hole drilling cycle (G83)

➢ **Instruction format:**

```
G83 X_ Y_ Z_ R_ Q_ F_
```

➢ **Details:**

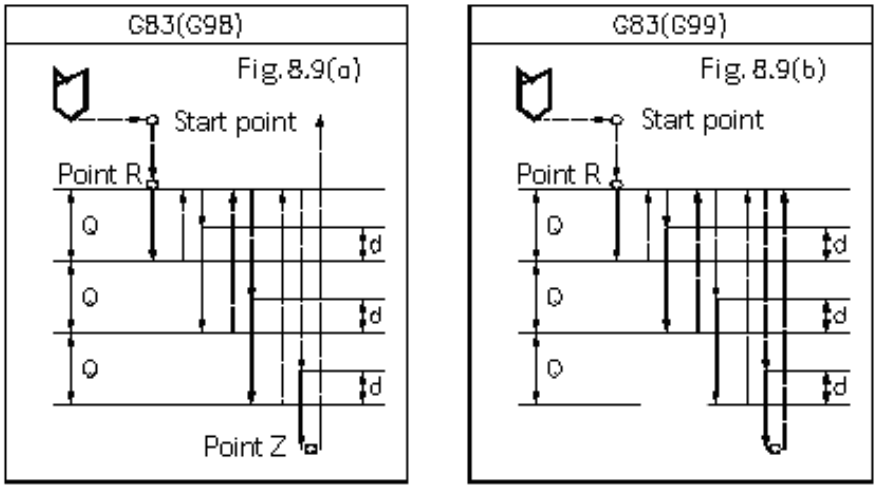


Fig. 10.8 Deep Hole Drilling Cycle (G83) Diagram

✧ **Note:**

Similar to G73 instruction, the feeding from point R to point Z under G83 instruction is also finished in two segments; different from G73 instruction, Z axis returns to point R after feeding of every segment, and then moves to position d above the start point at the quick feeding speed and starts the feeding of next segment. The distance of every feeding is specified by hole processing parameter Q, which is always positive; the value of d is specified by 532# machine tool parameters.

4.13.6 Tapping cycle (G84)

➢ **Instruction format:**

```
G84 X_   Y_    Z_    R_   F_(D_)
X_Y_       : thread position
Z_         : thread depth
R_         : start point of tool feeding/retreating
F_(D_)     : convert the feeding speed according to screw distance, or specify
the screw distance with D_ directly
```

➢ **Details:**

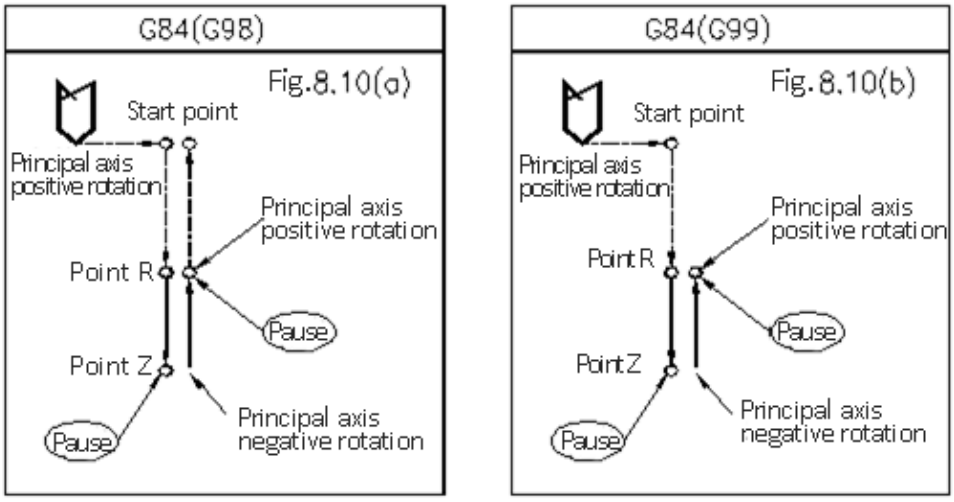


Fig. 10.9 Taping Cycle Diagram

- **Notice:**

In G74 and G84 cycle, the feeding rate switch and feeding retaining switch are ignored, i.e. feeding rate is retained at 100%, and can't be stopped before a fixed cycle completes; before cycle starts, the spindle should be specified to rotate in taping direction.

4.13.7 Boring cycle (G85)

➢ **Instruction format:**

G85 X_ Y_ Z_ R_ F_

➢ **Details:**

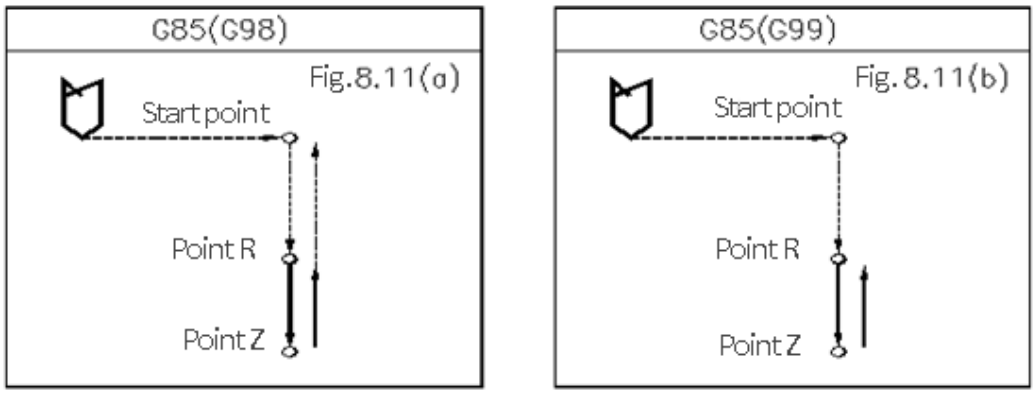


Fig. 10.10 Boring Cycle (G85) Diagram

This fixed cycle is very simple and the execution process follows:

X, Y positioning, Z axis quickly moves to point R, feeds to point Z at the speed specified by F, Returns to point R at the speed specified by F, In G98 mode, return to point R and return to the start point quickly.

4.13.8 Boring cycle (G86)

➢ **Instruction format:**

G86 X_ Y_ Z_ R_ F_

➢ **Details:**

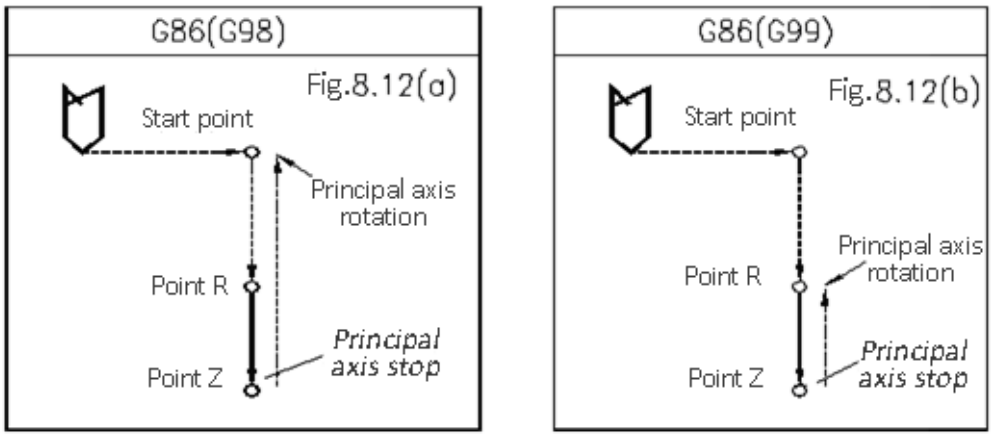


Fig. 10.11 Boring Cycle (G86) Diagram

✧ **Note:**

The execution of this fixed cycle is similar to G81; the difference is that the tool feeds to hole bottom in G86 to make the spindle stop, and quickly returns to point R or the start point to make the spindle to rotate in original direction and at original rotation.

4.13.9 Boring cycle (G88)

The fixed cycle G88 has manual return function and is used for boring.

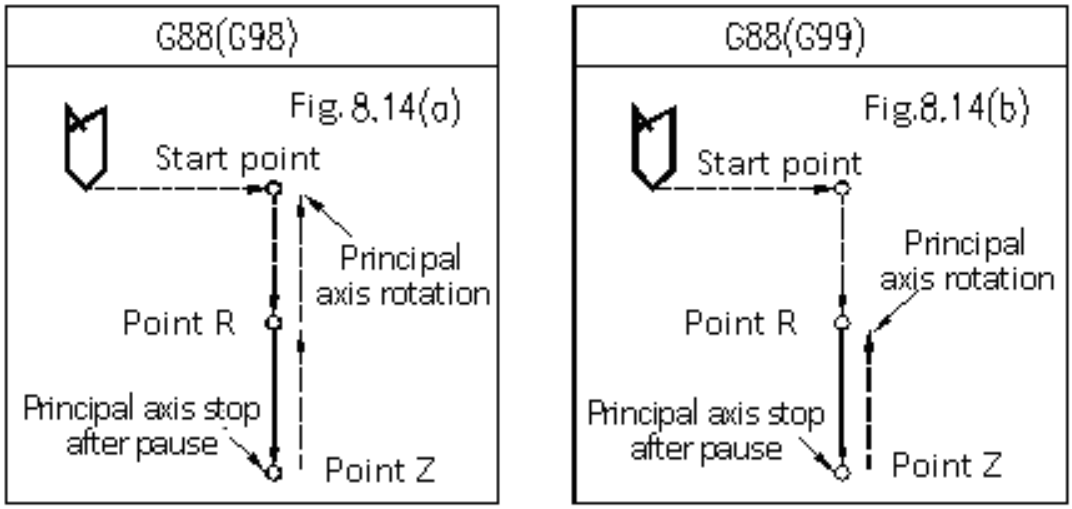


Fig. 10.12 Boring Cycle (G88) Diagram

4.13.10 Boring cycle (G89)

This fixed cycle increases pause in the hole bottom basing on G85, as shown below.

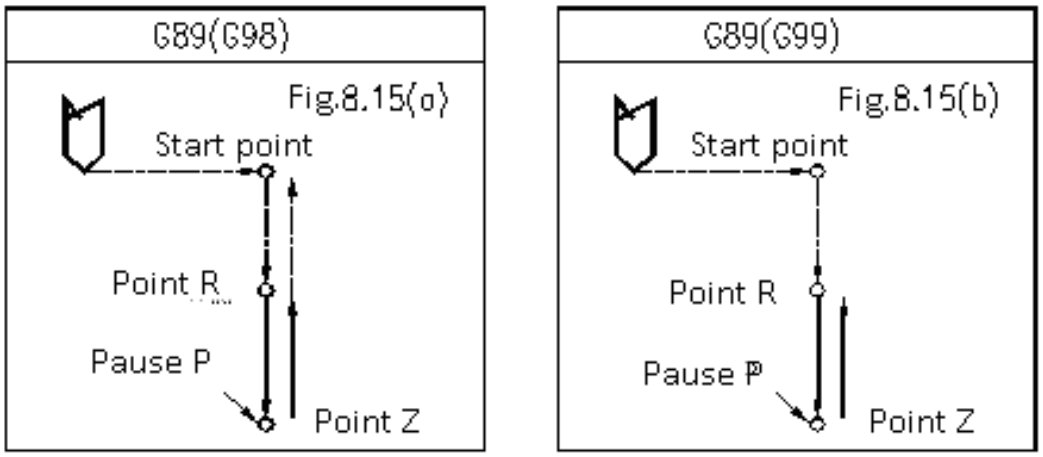


Fig. 10.13 Boring Cycle (G89) Diagram

4.14 Notes for using hole processing fixed cycle

(1) During programming, it is necessary to use S and M code to rotate the spindle before the fixed cycle instruction.

```
M03          ; Spindle positive rotation
.
G□□……; Correct
..
M05          ; Spindle stop
G□□……; False (instruction M03 or M04 is required before this segment)
```

(2) In fixed cycle mode, the segment containing X, Y, Z, R will execute the fixed cycle; if a segment contains neither address above, this segment won't execute the fixed cycle, except address X in G04. In addition, address P in G04 won't change the P value in hole processing parameters.

```
G00 X_;
G81 X_ Y_ Z_ R_ F_ K_ ;
;                    (Do not execute hole processing)
F_;                  (Do not execute hole processing , F value is updated)
M_;                  (Do not execute hole processing, only executes auxiliary function)
G04 P_;              (Do not execute hole processing, use G04 P_ to change hole processing data P)
```

(3) Hole processing parameterS Q, P must be specified in the segment in which the fixed cycle is executed, or else the specified Q, P value are invalid.

(4) During executing the fixed cycle (e.g. G76, G84, etc.) that contains spindle control, the spindle hasn't reached the specified rotation when the tool starts cutting feeding. In this case, it is required to insert G04 pause instruction during the hole processing operation.

(5) G code of group 01 also has the effect to cancel fixed cycle, and thus do not write fixed cycle instruction and G code of group 01 in the same segment.

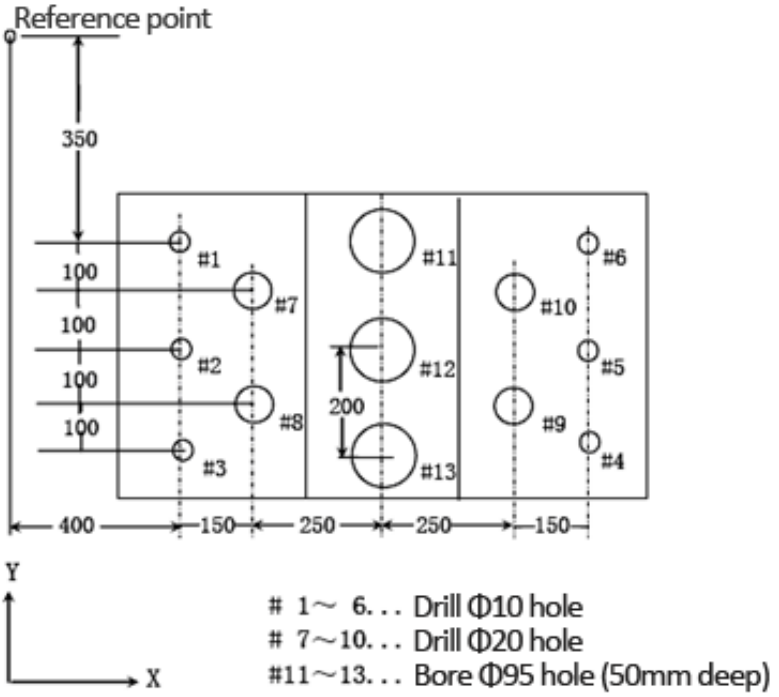
(6) If the segment that executes the fixed cycle specifies an M code, the M code will be executed while the fixed cycle is positioning, and the signal that M instruction is executed is sent when Z axis returns to point R or the start point. If the fixed cycle is repeated with K parameter instruction, the M code in the same segment will be executed when the fixed cycle is first executed.

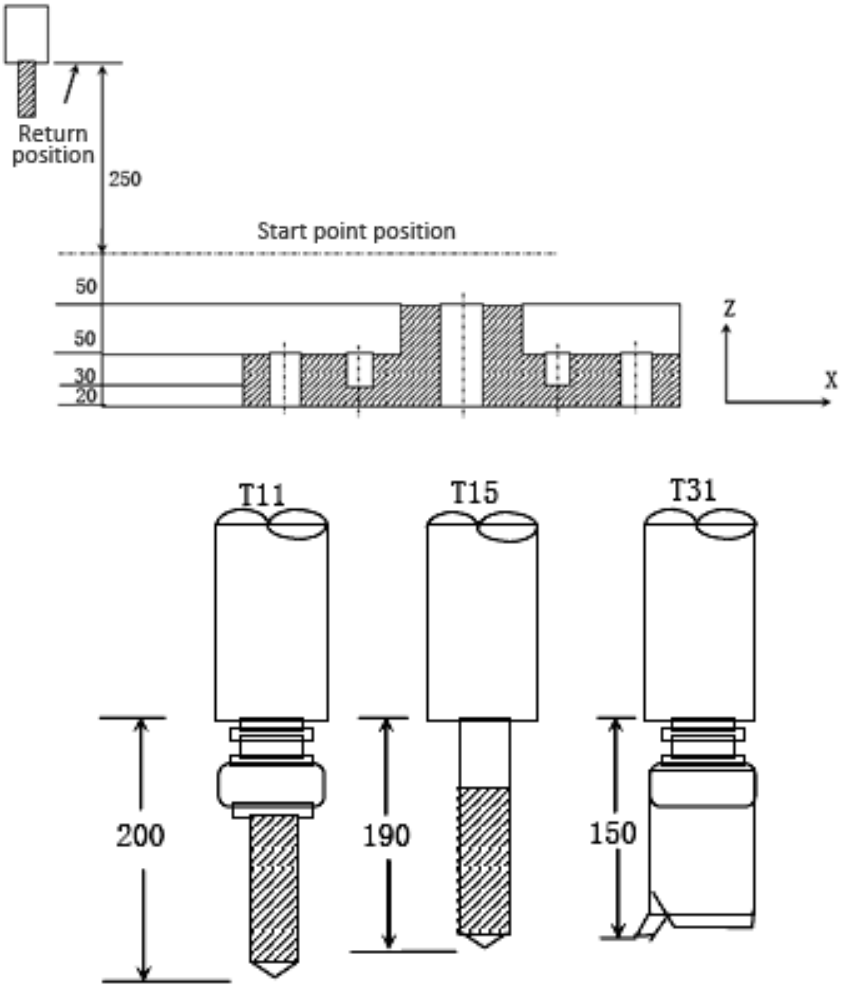
(7) In fixed cycle mode, tool offset instructions G45~G48 will be ignored (won't be executed).

(8) When single segment switch is in up position, the fixed cycle will stop after executes X, Y axis positioning, quickly feeds to point R and returns from the hole bottom (to point R or the start point). That is to say, it is required to press the cycle start button for three times to complete a hole processing. During the three stops, the first two stops are in feeding state, and the last one is in stopped state.

(9) While executing G74 and G84 cycles, if you press the retain button when Z axis moves to point R to point Z or reverse, the feeding retaining indicator will be lighted immediately, but the machine tool action won't stop immediately, until Z axis returns to point R. In addition, feeding rate switch is invalid in G74 and G84 cycles, and it is fixed at 100%.

4.15 Examples of using tool length compensation and fixed cycle





The value of offset No. 11 is 200.0,

The value of offset No. 15 is 190.0,

The value of offset No. 31 is 150.0,

The offsets are set separately. The program follows:

```
N001 G92 X0 Y0 Z0    ;          The coordinate system is set at reference point
N002 G90 G00 Z250.0 T11 M6;    Change tool
N003 G43 Z0 H11    ;              Execute plane tool length compensation at the start point
N004 S30 M3    ;                   Spindle starts
N005 G99 G81 X400.0 Y-350.0
Z-153.0 R-97.0 F120.0    ;          Process #1 hole after positioning
N006 Y-550.0    ;             Process #2 hole after positioning, return to point R plane
N007 G98 Y-750.0    ;             Process #3 hole after positioning, return to start point plane
N008 G99 X1200.0 ;                Process #4 hole after positioning, return to point R plane
N009 Y-550.0    ;                 Process #5 hole after positioning, return to point R plane
N010 G98 Y-350.0    ;             Process #6 hole after positioning, return to start point plane
N011 G00 X0 Y0 M5    ;    Return to reference point, spindle stops
N012 G49 Z250.0 T15 M6    ;       Cancel tool length compensation, change tool
N013 G43 Z0 H15    ;              Start point plane, tool length compensation
```

```
N014 S20 M3    ;                    Spindle starts
N015 G99 G82 X550.0 Y-450.0    ;
Z-130.0 R-97.0 P30 F70;          Process #7 hole after positioning, return to point R plane
N016 G98 Y-650.0    ;             Process #8 hole after positioning, return to start point plane
N017 G99 X1050.0    ;              Process #9 hole after positioning, return to point R plane
N018 G98 Y-450.0    ;            Process #10 hole after positioning, return to start point plane
N019 G00 X0 Y0 M5    ;          Return to reference point, spindle stops
N020 G49 Z250.0 T31 M6    ;        Cancel tool length compensation, change tool
N021 G43 Z0 H31    ;             Start point plane, tool length compensation
N022 S10 M3    ;                    Spindle starts
N023 G85 G99 X800.0 Y-350.0    ;
Z-153.0 R47.0 F50    ;             Process #11 hole after positioning, return to point R plane
N024 G91 Y-200.0    ;           Process #12, #13 holes after positioning, return to point R plane
Y-200.0 ;
N025 G00 G90 X0 Y0 M5    ;    Return to reference point, spindle stops
N026 G49 Z0    ;                  Cancel tool length compensation
N027 M30    ;%                    program stops
```

4.16 Auxiliary function (M, S, T)

This machine tool uses S code to program spindle rotation, uses T code to program tool selection, and other auxiliary functions are achieved with M code.

4.16.1 M codes list:

Table 11.1 M Codes List

M code	Function
M01	Program stop
M03	Spindle positive rotation
M04	Spindle negative rotation
M05	Spindle stop
M06	Too exchange instruction
M08	Cooling on
M09	Cooling off
M32	Lubrication on
M33	Lubrication off
M30	Program ends and returns to program header
M98	Call subroutine
M99	Subroutine ends and returns / Repeat
M56	Output 02# interrupt port is high voltage level
M57	Output 02# interrupt port is low voltage level
M58	Output 03# interrupt port is high voltage level
M59	Output 03# interrupt port is low voltage level
M10	Output 06# interrupt port is high voltage level
M11	Output 06# interrupt port is high voltage level

M code	Function
M20	Output 07# interrupt port is high voltage level
M21	Output 07# interrupt port is low voltage level
M12	Output 08# interrupt port is high voltage level
M13	Output 08# interrupt port is low voltage level
M14	Output 09# interrupt port is high voltage level
M15	Output 09# interrupt port is low voltage level
M16	Output 10# interrupt port is high voltage level
M17	Output 10# interrupt port is low voltage level
M18	Output 11# interrupt port is high voltage level
M19	Output 11# interrupt port is low voltage level
M40	Output 12# interrupt port is high voltage level
M41	Output 12# interrupt port is low voltage level
M42	Output 13# interrupt port is high voltage level
M43	Output 13# interrupt port is low voltage level
M44	Output 14# interrupt port is high voltage level
M45	Output 14# interrupt port is low voltage level
M46	Output 15# interrupt port is high voltage level
M47	Output 15# interrupt port is low voltage level
M48	Output 16# interrupt port is high voltage level
M49	Output 16# interrupt port is low voltage level
M50	Output 17# interrupt port is high voltage level
M51	Output 17# interrupt port is low voltage level
M66	Output 20# interrupt port is high voltage level
M67	Output 20# interrupt port is low voltage level
M64	Output 21# interrupt port is high voltage level
M65	Output 21# interrupt port is low voltage level
M62	Output 22# interrupt port is high voltage level
M63	Output 22# interrupt port is low voltage level
M60	Output 23# interrupt port is high voltage level
M61	Output 23# interrupt port is low voltage level
M88 Pn Lm	Check whether input IO (IN n) voltage level is m (high/low), continue to wait if not true
M89 Pn Lm Qt	Output: OUT n, voltage level: m, delay t ms output, or execute immediately if there is no t

In the machine tool, M code has two effects: one is to control the execution of the program, and the other is IO operation, which is used to control the execution of spindle, cooling system and other auxiliary devices.

M code for program control

M00……program stops. When NC executes M00, the program execution is interrupted; after reset, press the Start button to continue executing the program.

M30……program ends, and returns to program header
M98……call subroutine
M99……subroutine ends, and returns to the main program

Other M codes

M03……spindle positive rotation. Use this instruction to rotate the spindle counterclockwise (CCW) with currently specified spindle rotation.
M04……spindle reverse rotation. Use this instruction to rotate the spindle clockwise (CW) with currently specified spindle rotation.
M05……spindle stops.
M06……tool change starts. M06 T02 instruction is to change tool #2.
M08……cooling on
M09……cooling off
M32……lubrication on
M33……lubrication off
M88……specify input IO port to check the voltage level; continue to execute if the levels are same, or else wait. If no voltage level signal is specified, it is low voltage level signal by default. For example: M88 P0 L1, wait until IN0 is high voltage level, or else wait all along.
M89……specify output IO port to check the voltage level; if no voltage level signal is specified, it is low voltage level signal by default; if Q value is specified, the operation needs to delay for Q ms and then output IO signal. For example: M89 P5 L0, specify OUT5 to output low voltage level.

- **Notice:**

If the motion instruction and M are in the same segment, M instruction will be executed first.

If the program has several M codes in current line, only one is valid, i.e. the last defined M code.

4.16.2 M code for program control

M codes for program control include M00, M30, M98 and M99, and the functions are explained as follows:

M00……program stops. When NC executes M00, the program execution is interrupted; after reset, press the Start button to continue executing the program.

M30……program ends, and returns to program header

M98……call subroutine

M99……subroutine ends, and returns to the main program

4.16.3 Other M codes

M03……spindle positive rotation. Use this instruction to rotate the spindle counterclockwise (CCW) with currently specified spindle rotation.

M04……spindle reverse rotation. Use this instruction to rotate the spindle clockwise (CW) with currently specified spindle rotation.

M05……spindle stops.

M06……tool change starts. M06 T02 instruction is to change tool #2.

M08……cooling on

M09……cooling off

M32……lubrication on

M33……lubrication off

M88……specify input IO port to check the voltage level; continue to execute if the levels are same, or else wait. If no voltage level signal is specified, it is low voltage level signal by default.

M89……specify output IO port to check the voltage level; if no voltage level signal is specified, it is low voltage level signal by default; if Q value is specified, the operation needs to delay for Q ms and then output IO signal.

4.16.4 S code

The rotation instruction of the spindle is specified by the S code, which is modular, i.e. always valid after the rotation is specified, until another S code changes the modular value.

The maximum value of S instruction is limited by the maximum spindle rotation set by parameter

S instruction includes two modes. When the frequency mode is set to 1, the S instruction value is 0~15, indicating level 0~15; when the frequency mode is set to 0, multiply the ratio of S value and the maximum speed by 10V to get the analog voltage output; after specifying S instruction, execute M03 or M04 to rotate the spindle.

4.16.5 T code

Machine tool magazine uses random tool selection mode, i.e. two digits T code TXX specifies the tool No., regardless which tool set it is in; the range of address T is any integral between 1 and 99.

Warning: Tool meter must be set properly, or else it will damage the machine tool and cause unpredictable results.

4.17 Key processing instructions

Key processing instructions mainly include drill point key processing instruction, bilateral internally threaded key processing instruction, bilateral externally threaded key processing instruction, S-shaped slot key processing instruction, flat-bottom double-layer key processing instruction, mortise key processing instructions, snap type key processing instruction, and code engraving processing instruction. The list of processed key instructions follows:

Table 4.17.1 List of Key Processing Instructions

Instruction code	Function
G199	Key code selection instruction
G200	Common drill point key processing instruction
G201	Complex drill point key processing instruction
G202	Sawtooth key processing instruction (three-axis processing)
G203	Sawtooth key processing instruction (cylinder processing)
G210	Bilateral externally threaded key processing instruction (external-internal-external-internal-external)
G211	Bilateral internally threaded key processing instruction (external-internal-external-internal-external)
G212	S-shaped slot key processing instruction
G213	Bilateral externally threaded key processing instruction (original

Instruction code	Function
	G66)
G214	Bilateral internally threaded key processing instruction (original G67)
G220	Flat-bottom double-layer key processing instruction
G221	Mortise key processing instruction
G250	Code engraving processing instruction
Not available	Snap type key processing instruction
Not available	Convergence double-layer key processing instruction
Not available	Round bottom double-layer key processing instruction
Not available	Angle code key processing instruction
Not available	Bilateral antisymmetric key processing instruction
Not available	Bilateral outer layer key processing instruction (tooth peak and tooth valley have the same width, the angle between the tooth peaks is not fixed)
Not available	Bilateral inner layer key processing instruction (width of fluted disc is fixed, the angle between the tooth peaks is not fixed)

[Note]: Above aren't all the key processing instructions. Only commonly used key processing instructions are added to the system. For any questions about key processing, please contact the company.

4.17.1 G199 read code string instruction in code file

- **Format:**

  G199

- **Parameters:**

  1. Code strings read from code file

- **Note:**

  This instruction determines the number of code strings selected from code file for key processing. This instruction should be used in conjunction with key processing instructions.

- **Routine:**

Routine parameters:

1. The code string number read by G199 from code file is set to 4

2. The four parameters of G200 are 2, 6, 1 and 10 respectively. At the same time, "001. Number of key codes" in [Craft] -> [Craft] is set to 6, and "005. Z-axis, A-axis safe height" is set to 5

3. Set the six X-axis coordinate parameters (hole 00~05) in [Craft] -> [Hole position] to 2, 4, 6, 8, 10 and 12

Sample application:

```
O0001
G90 G54 G17
G00 Z5
G00 X0
G199
G200
```

G00 Z5
G00 X0
M30
%

4.17.2 G200 common drill point key processing instructions

- **Format:**

  G200

- **Parameters:**

  1. Select the number of code string for processing
  2. Number of G200 key tooth codes
  3. X-axis direction (1: CW; -1: CCW)
  4. Bottom borehole residence time (ms)

- **Note:**

1. If the test code isn't null, the test code will be selected for processing in prior; if the test code string is null, read the code in code file, and the system alarms if no code file is loaded.

2. During G200 processing, read the code in code file for key processing, if "select code string for processing" hasn't been set, will codes will be read in sequence; if it has been set, the selected code will be read according to set condition.

3. The coordinate position of G200 common drill point key processing instruction should be set in [Craft] -> [Hole position].

- **Graphics and routine:**

Diagram of common drill point key:

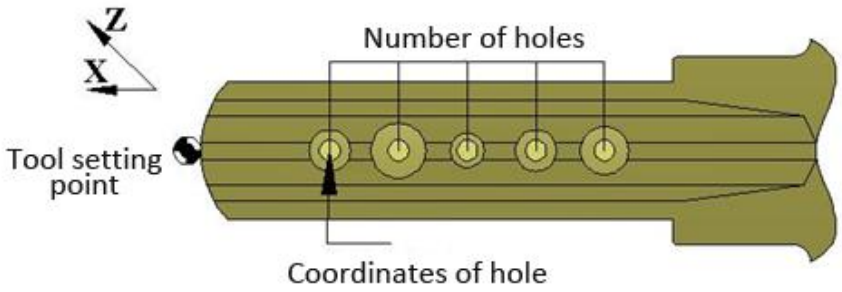


Fig. 4.17.1 Diagram of common drill point key

Routine parameters:

1. The code string number read by G199 from code file is set to 4

2. The four parameters of G200 are 2, 6, 1 and 10 respectively. At the same time, "001. Number of key codes" in [Craft] -> [Craft]] is set to 6, and "005. Z-axis, A-axis safe height" is set to 5

3. Set the six X-axis coordinate parameters (hole 00~05) in [Craft] -> [Hole position] to 2, 4, 6, 8, 10 and 12

Sample application:

O0001
G90 G54 G17
G00 Z5
G00 X0
G199

G200
G00 Z5
G00 X0
M30
%

4.17.3 G201 complex drill point key processing instructions

- **Format:**

  G201

- **Parameters:**

  1. Select the number of code string for processing
  2. Number of G201 key tooth codes
  3. X-axis direction (1: CW; -1: CCW)
  4. Y-axis direction (1: CW; -1: CCW)
  5. Bottom borehole residence time (ms)

- **Note:**

1. If the test code isn't null, the test code will be selected for processing in prior; if the test code string is null, read the code in code file, and the system alarms if no code file is loaded.

2. During G201 processing, read the code in code file for key processing, if "select code string for processing" hasn't been set, will codes will be read in sequence; if it has been set, the selected code will be read according to set condition.

3. The coordinate position of G201 complex drill point key processing instruction should be set in [Craft] -> [Hole position].

- **Graphics and routine:**

Diagram of key with complex drill points:

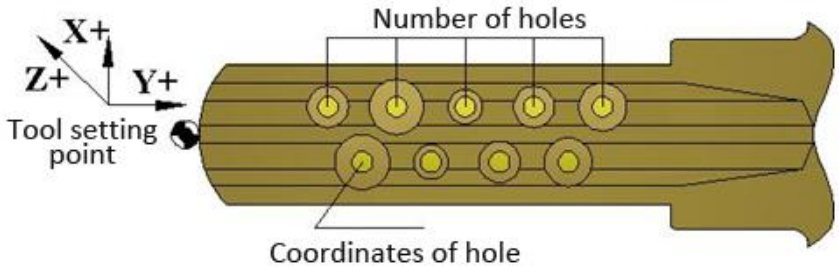


Fig. 4.17.1 Diagram of key with complex drill points

Routine parameters:

1. The code string number read by G199 from code file is set to 4

2. The five parameters of G201 are 2, 12, 1 and 10 respectively. At the same time, "001. Number of key codes" in [Craft]—>[Craft] is set to 12, and "005. Z-axis, A-axis safe height" is set to 5

3. Set the six X-axis and Y-axis coordinate parameters (hole 00~05) in [Craft] -> [Hole position] to (-2, 2), (-2, 4), (-2, 6), (-2, 8), (-2, 10), (-2, 12), (2, 3), (2, 5), (2, 7), (2, 9), (2, 11)

4. For KY460A system, set "010. Single spindle: 0; double spindles: 1" in [Craft] -> [Hole position] to 1 for double spindles processing

Sample application:

O0001
G90 G54 G17
G00 Z5

G00 X0 Y0
G199
G201
G00 Z5
G00 X0 Y0
M30
%

4.17.4 G202 sawtooth key processing instructions (three-axis processing)

- **Format:**

  G202

- **Parameters:**

  1. Select the number of code string for processing
  2. Number of G202 key tooth codes
  3. Tooth start distance
  4. Tooth pitch
  5. X-axis offset distance
  6. X-axis direction (1: CW; -1: CCW)
  7. Y-axis direction (1: CW; -1: CCW)

- **Note:**

1. If the test code isn't null, the test code will be selected for processing in prior; if the test code string is null, read the code in code file, and the system alarms if no code file is loaded.

2. During G202 processing, read the code in code file for key processing, if "select code string for processing" hasn't been set, will codes will be read in sequence; if it has been set, the selected code will be read according to set condition.

- **Graphics and routine:**

Sawtooth Key Diagram:

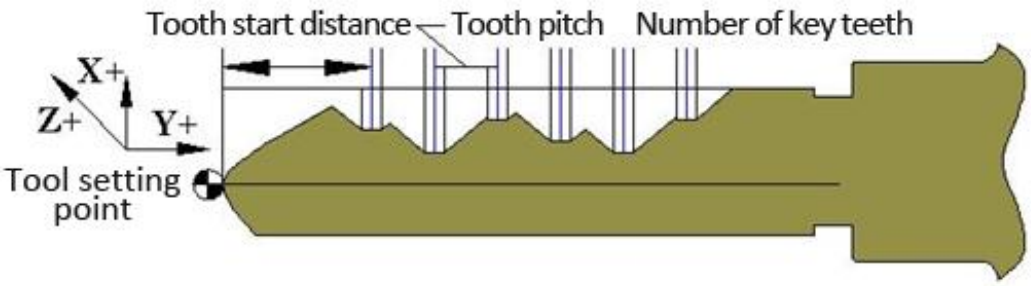


Fig. 4.17.4 Sawtooth Key (Three-axis Processing) Diagram

Routine parameters:

1. The code string number read by G199 from code file is set to 4

2. The seven parameters of G202 are 2, 6, 5, 3.5, 10, 1 and 1 respectively. At the same time, "001. Number of key codes" in [Craft] -> [Craft] is set to 6, and "005. Z-axis, A-axis safe height" is set to 5

Sample application:

O0001
G90 G54 G17
G00 Z5
G00 X0 Y0

G199
G202
G00 Z5
G00 X0 Y0
M30
%

4.17.5 G203 sawtooth key processing instructions (cylinder processing)

- **Format:**

  G203

- **Parameters:**

  1. Select the number of code string for processing
  2. Number of G202 key tooth codes
  3. Tooth start distance
  4. Tooth pitch
  5. Cylinder forward detection signal port
  6. Cylinder backward detection signal port
  7. Output port that controls cylinder
  8. Cylinder action delay time

- **Note:**

1. If the test code isn't null, the test code will be selected for processing in prior; if the test code string is null, read the code in code file, and the system alarms if no code file is loaded.

2. During G202 processing, read the code in code file for key processing, if "select code string for processing" hasn't been set, will codes will be read in sequence; if it has been set, the selected code will be read according to set condition.

3. The instruction uses cylinder processing method, which has a same processing result as three-axis system in G202.

- **Graphics and routine:**

Sawtooth Key Diagram:

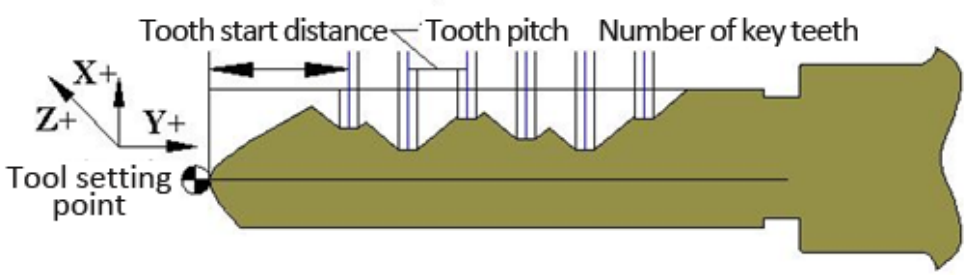


Fig. 4.17.4 Sawtooth Key (Cylinder Processing) Diagram

Routine parameters:

1. The code string number read by G199 from code file is set to 4

2. The seven parameters of G202 are 2, 6, 5, 3.5, 10, 1 and 1 respectively. At the same time, "001. Number of key codes" in [Craft] -> [Craft] is set to 6, and "005. Z-axis, A-axis safe height" is set to 5

Sample application:

O0001
G90 G54 G17

G00 Z5
G00 X0
G199
G203
G00 Z5
G00 X0
M30
%

4.17.6 G210 bilateral externally threaded key processing instruction (external-internal-external-internal-external)

- **Format:**

  G210

- **Parameters:**

  1. Select the number of code string for processing
  2. Key width
  3. Left tooth start distance
  4. Right tooth start distance
  5. Tooth pitch
  6. Width of fluted disc
  7. Processing depth
  8. Tool diameter
  9. Number of left codes
  10. Number of right codes
  11. Affiliate decoration (1: Yes, 0: No)
  12. Whether process port triangle
  13. X-axis direction (1: CW; -1: CCW)
  14. Y-axis direction (1: CW; -1: CCW)

- **Note:**

1. If the test code isn't null, the test code will be selected for processing in prior; if the test code string is null, read the code in code file, and the system alarms if no code file is loaded.

2. During G202 processing, read the code in code file for key processing, if "select code string for processing" hasn't been set, will codes will be read in sequence; if it has been set, the selected code will be read according to set condition.

- **Graphics and routine:**

  Diagram of Bilateral Externally Threaded Key:

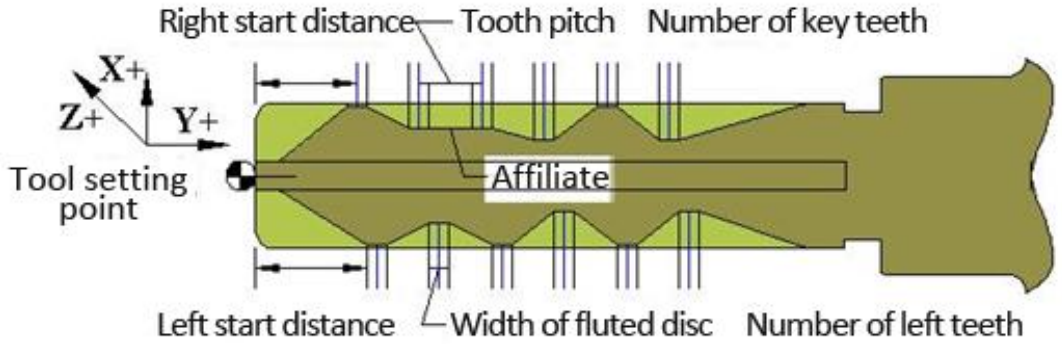


Fig. 4.17.4 Diagram of Bilateral Externally Threaded Key

Routine parameters:

Not available

Sample application:

O0001
G90 G54 G17
G00 Z5
G00 X0 Y0
G199
G210
G00 Z5
G00 X0 Y0
M30
%

4.17.7 G211 bilateral internally threaded key processing instruction (external-internal-external-internal-external)

- **Format:**

  G211

- **Parameter:**

  1. Select the number of code string for processing
  2. Key width
  3. Left tooth start distance
  4. Right tooth start distance
  5. Tooth pitch
  6. Width of fluted disc
  7. Processing depth
  8. Tool diameter
  9. Number of left codes
  10. Number of right codes
  11. Affiliate decoration (1: Yes, 0: No)
  12. Whether process port triangle
  13. X-axis direction (1: CW; -1: CCW)
  14. Y-axis direction (1: CW; -1: CCW)

- **Note:**

1. If the test code isn't null, the test code will be selected for processing in prior; if the test code string is null, read the code in code file, and the system alarms if no code file is loaded.

2. During G202 processing, read the code in code file for key processing, if "select code string for processing" hasn't been set, will codes will be read in sequence; if it has been set, the selected code will be read according to set condition.

- **Graphics and routine:**

  Diagram of Bilateral Internally Threaded Key:

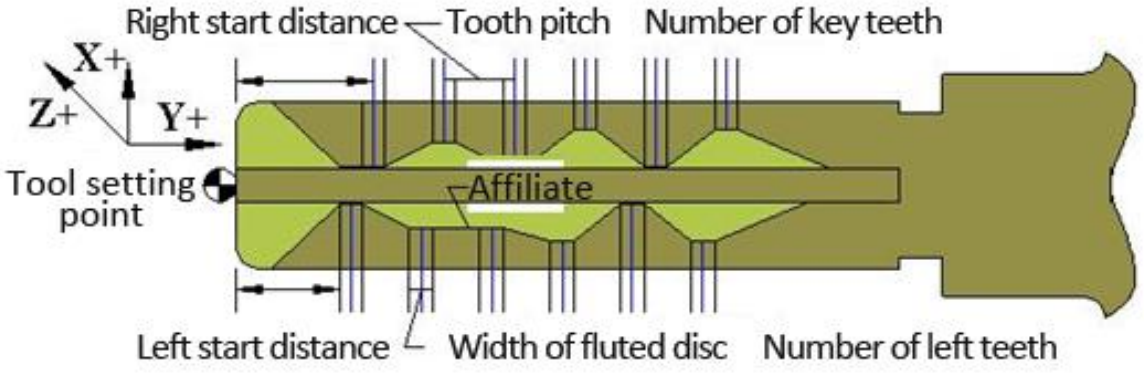


Fig. 4.17.4 Diagram of Bilateral Internally Threaded Key

Routine parameters:

Not available

Sample application:

```
O0001
G90 G54 G17
G00 Z5
G00 X0 Y0
G199
G211
G00 Z5
G00 X0 Y0
M30
%
```

4.17.8 G212 S-shaped key processing instruction

- **Format:**

  G211

- **Parameters:**

  1. Select the number of code string for processing
  2. Key width
  3. Width from left side of the key to center line
  4. Width from right side of the key to center line
  5. X coordinates of front end of port triangle
  6. Y coordinates of front end of port triangle
  7. Tooth start distance
  8. Tool diameter

9. Tooth pitch
10. Width of fluted disc
11. Slot width
12. Processing depth
13. Number of key codes
14. Feeding times
15. Margin of last tool
16. Affiliate decoration (1: Yes, 0: No)
17. X-axis direction (1: CW; -1: CCW)
18. Y-axis direction (1: CW; -1: CCW)

- **Note:**

1. If the test code isn't null, the test code will be selected for processing in prior; if the test code string is null, read the code in code file, and the system alarms if no code file is loaded.

2. During G202 processing, read the code in code file for key processing, if "select code string for processing" hasn't been set, will codes will be read in sequence; if it has been set, the selected code will be read according to set condition.

- **Graphics and routine:**

S-shaped slot key diagram:

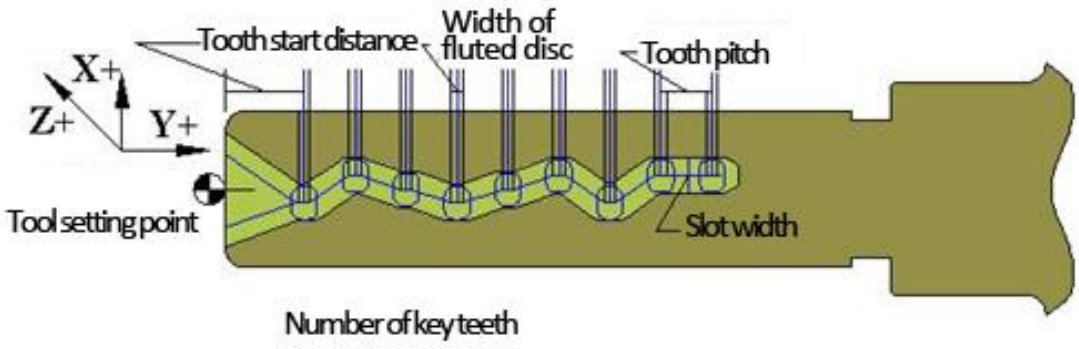


Fig. 4.17.4 Diagram of Bilateral Internally Threaded Key

Routine parameters:

Not available

Sample application:

```
O0001
G90 G54 G17
G00 Z5
G00 X0 Y0
G199
G212
G00 Z5
G00 X0 Y0
M30
%
```

4.17.9 G213 bilateral externally threaded key processing instruction

- **Format:**

  G213

- **Parameters:**

  1. Select the number of code string for processing
  2. Key width
  3. Left tooth start distance
  4. Right tooth start distance
  5. Tooth pitch
  6. Width of fluted disc
  7. Processing depth
  8. Tool diameter
  9. Number of left codes
  10. Number of right codes
  11. Affiliate decoration (1: Yes, 0: No)
  12. Distance from origin to key vertex
  13. Left start angle
  14. Right start angle
  16. Width of key slot
  16. Angle of left tail end of the key
  17. Angle of right tail end of the key
  18. Distance between two ends of the key
  19. Whether mill key bit (1: Yes, 0: No)
  20. X-axis direction variable (1: CW; -1: CCW)
  21. Y-axis direction variable (1: CW; -1: CCW)

- **Note:**

1. If the test code isn't null, the test code will be selected for processing in prior; if the test code string is null, read the code in code file, and the system alarms if no code file is loaded.

2. During G202 processing, read the code in code file for key processing, if "select code string for processing" hasn't been set, will codes will be read in sequence; if it has been set, the selected code will be read according to set condition.

- **Graphics and routine:**

Diagram of Bilateral Externally Threaded Key:

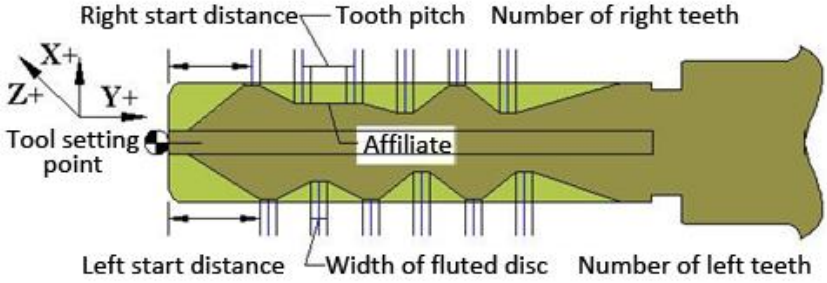


Fig. 4.17.4 Diagram of Bilateral Externally Threaded Key

Routine parameters:

Not available

Sample application:

O0001
G90 G54 G17
G00 Z5
G00 X0 Y0
G199
G213
G00 Z5
G00 X0 Y0
M30
%

4.17.10 G214 bilateral internally threaded key processing instruction

- **Format:**

  G210

- **Parameters:**

  1. Select the number of code string for processing
  2. Key width
  3. Left tooth start distance
  4. Right tooth start distance
  5. Tooth pitch
  6. Width of fluted disc
  7. Processing depth
  8. Tool diameter
  9. Number of left codes
  10. Number of right codes
  11. Solid key: 1, Hollow key: 0
  12. Angle of key opening
  13. Length of key tooth
  14. Size of minimum key tooth
  15. Width of hollow slot of the key
  16. Whether mill key bit mark (1: Yes, 0: No)
  17. Y-axis direction variable (1: CW; -1: CCW)
  18. Y-axis direction variable (1: CW; -1: CCW)

- **Note:**

1. If the test code isn't null, the test code will be selected for processing in prior; if the test code string is null, read the code in code file, and the system alarms if no code file is loaded.

2. During G202 processing, read the code in code file for key processing, if "select code string for processing" hasn't been set, will codes will be read in sequence; if it has been set, the selected code will be read according to set condition.

- **Graphics and routine:**

  Diagram of Bilateral Internally Threaded Key:

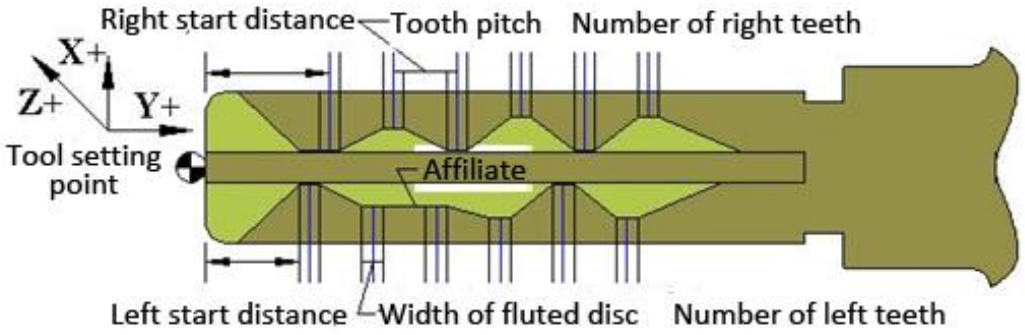


Fig. 4.17.4 Diagram of Bilateral Internally Threaded Key

Routine parameters:

Not available

Sample application:

O0001
G90 G54 G17
G00 Z5
G00 X0 Y0
G199
G214
G00 Z5
G00 X0 Y0
M30
%

4.17.11 G220 flat bottom double-layer key processing instruction

- **Format:**

  G214

- **Parameters:**

  1. Select the number of code string for processing
  2. Upper key offset
  3. Lower key offset
  4. Upper tooth start distance
  5. Lower tooth start distance
  6. Upper tooth processing depth
  7. Lower tooth processing depth
  8. Tooth pitch
  9. Width of fluted disc
  10. Tool diameter
  11. Number of upper tooth codes
  12. Number of lower tooth codes
  13. X-axis direction (1: CW; -1: CCW)
  14. Y-axis direction (1: CW; -1: CCW)

- **Note:**

  1. If the test code isn't null, the test code will be selected for processing in prior; if the test code

string is null, read the code in code file, and the system alarms if no code file is loaded.

2. During G202 processing, read the code in code file for key processing, if "select code string for processing" hasn't been set, will codes will be read in sequence; if it has been set, the selected code will be read according to set condition.

- **Graphics and routine:**

Diagram of key with complex drill points:

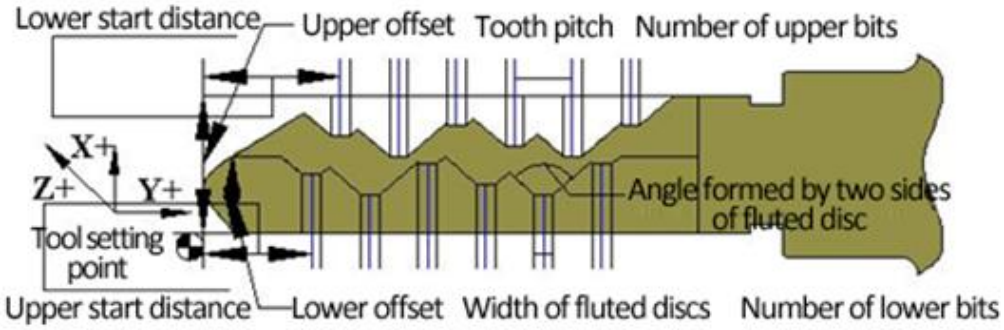


Fig. 4.17.4 Diagram of Bilateral Externally Threaded Key

Routine parameters:

Not available

Sample application:

O0001
G90 G54 G17
G00 Z5
G00 X0 Y0
G199
G220
G00 Z5
G00 X0 Y0
M30
%

4.17.12 G221 mortise key processing instruction

- **Format:**

G221

- **Parameters:**

1. Select the number of code string for processing
2. Key width
3. Left layer offset
4. Right layer offset
5. Left tooth start distance
6. Right tooth start distance
7. Left tooth processing depth
8. Right tooth processing depth
9. Tooth pitch
10. Width of fluted disc
11. Tool diameter
12. Left code digits

13. Right code digits

14. Select CW/CCW direction for left tooth processing (0: CW; 1: CCW)

15. Select CW/CCW direction for right tooth processing (0: CW; 1: CCW)

16. Time required changing to reverse tooth (note: right in the parameter indicates that the tooth facing the reference direction)

17. X-axis direction (1: CW; -1: CCW)

18. Y-axis direction (1: CW; -1: CCW)

- **Note:**

1. If the test code isn't null, the test code will be selected for processing in prior; if the test code string is null, read the code in code file, and the system alarms if no code file is loaded.

2. During G202 processing, read the code in code file for key processing, if "select code string for processing" hasn't been set, will codes will be read in sequence; if it has been set, the selected code will be read according to set condition.

- **Graphics and routine:**

Diagram of key with complex drill points:

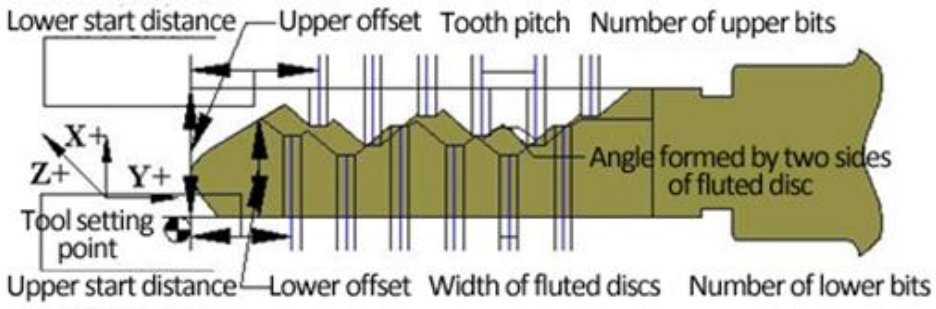


Fig. 4.17.4 Diagram of Bilateral Externally Threaded Key

Routine parameters:

Not available

Sample application:

```
O0001
G90 G54 G17
G00 Z5
G00 X0 Y0
G199
G221
G00 Z5
G00 X0 Y0
M30
%
```

4.17.13 G250 code engraving instruction

- **Format:**

G250

- **Parameters:**

1. Start position of X-axis processing
2. Start position of Y-axis processing
3. Start position of Z-axis processing
4. Start position of A-axis processing

5. Font X-axis offset
6. Font Y-axis offset
7. Font Z-axis offset
8. Font A-axis offset
9. Font X-axis size scaling
10. Font Y-axis size scaling
11. Font Z-axis size scaling
12. Font A-axis size scaling
13. Select CW/CCW direction for left tooth processing (0: CW; 1: CCW)
14. Select CW/CCW direction for right tooth processing (0: CW; 1: CCW)

➢ **Note:**

1. If the test code isn't null, the test code will be selected for processing in prior; if the test code string is null, read the code in code file, and the system alarms if no code file is loaded.

2. During G202 processing, read the code in code file for key processing, if "select code string for processing" hasn't been set, will codes will be read in sequence; if it has been set, the selected code will be read according to set condition.

➢ **Graphics and routine:**

Diagram of engraving code:

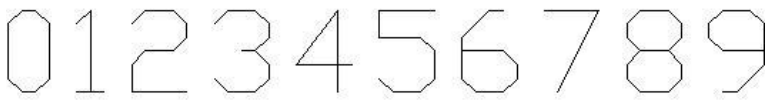


Fig. 4.17.4 Code Engraving Diagram

Routine parameters:

Not available

Sample application:

```
O0001
G90 G54 G17
G00 Z5
G00 X0 Y0
G199
G250
G00 Z5
G00 X0 Y0
M30
%
```

4.18 Category B macro function

4.18.1 Variable instruction

➢ **Function:**

All the address values in the program are not described with fixed value, and are replaced with variables; when the program is running, variables are referenced to improve the versatility of the program. This function is called as variable instruction.

Instruction format:

#Δ Δ Δ =○○○○○○○○○ or #Δ Δ Δ =[expression]

Details:

(1) Representation of variables:

(a) # m	M=0~9 constituted value	#100
(b) #[f]……	f has the following meanings	
	Value m	123
	Variable	#543
	Expression	#110+#119
	- (symbol) expression	-#120
	Function expression	SIN [#110]

Note:

Standard operating symbols are +, -, ×, /.

When the function expression is ignored, the function can't be executed.

The variable No. can't be negative, e.g. # -100 is illegal.

Below are false variable representations:

False		Correct
# 6 / 2	→	#[6 / 2]
#-[# 1]	→	#[- # 1]
#——5	→	#[- [- 5]]

(2) Types of variables

Type	Variable address	Function description
Global variable	#100~#199 #500~#999	Both main program and subroutine can be called #100~#199 are non-retentive variables, and will be reset automatically when the system is repowered #500~#999 are retentive variables, and the values still exist when the system power system is cut off.
Local variable	#1~#32	Can be called in the same program
System variable	Not available	

⑶ Variable reference

(a)	Except O, N and / (slash)
(b)	Specify with variables directly G01X#1Y#100
(c)	Take the complement of the variables directly G01X-#2
(d)	Variable defines variable #3= - #105 ; take the complement of #105 directly and evaluate to #3 #4=1000 ; evaluate 1000 to #4 directly
(e)	Define the evaluation with expression #1=#3+#2-100; the value #1 equals to the result of #3+#2-100 X[#1+#3+1000]; the value of X is the result of expression [#1+#3+1000]

Note:

Function evaluation and expression evaluation must be written separately, and can't be in the same line.

False		Correct
X#1=#3+100	→	#1=#3+100 X#1

[ ] can be embedded up to five levels.

```
#543=-[[[[[#120]/2+15.]*3-#100]/ #520+#125+#128]* #130+#132
```

The variable values must be 0~±9999999 (seven significant figures); if exceeding the maximum value, the calculation error will be enlarged.

4.18.2 Macro program call

- **Using macro calling function**

Function:

Same as subroutine calling, the macro program can transfer variables to subroutine during calling, which is different from M98 subroutine calling.

The following G codes are instructions to call macro program:

Table 12.1 Macro Program Calling Instruction

G code	Function
G65	Macro program calling
G66	Macro program calling mode A (call motion instruction)
G661	Macro program calling mode B (call every segment)
G67	Cancel macro program calling mode

Details:

The macro programs specified after G66 (or G661) instruction is specified, before G67 (cancel) instruction, and after the segments with motion instruction are executed (or every segment is executed).

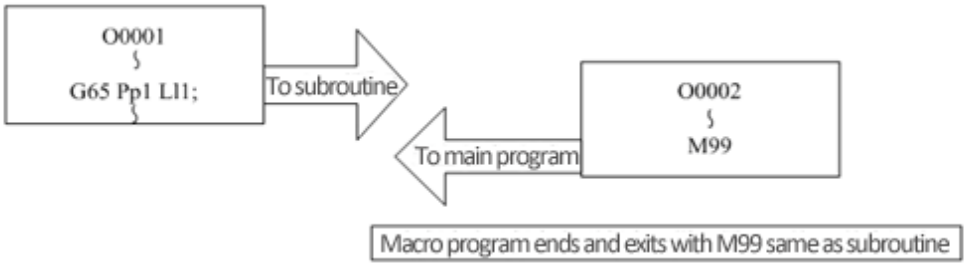
G66 (or G661) and G67 instructions must be used in pair in the same program.

- **Macro program calling command**

Function and purpose:

Macro program calling instructions include simply callings that are called by calling instruction only, calling modes (A&B) of single segment fixed calling.

(1) Simply calling



Format:

G65 P_ L_ < argument >;
P_ : subroutine No.
L_ : repeat times

The <argument> function in G65 is a method that the main program uses bit address to transfer parameters to subroutine; this method uses local variable to transfer; the argument is described below.

Argument format:

A_B_C_...X_Y_Z_

Details:

Except G, L, N, O, P, all bit addresses can be specified as arguments.

The bit addresses that do not need to transfer can be ignored.

In G65 instruction segment, all the bit addresses are considered as the arguments of G65.

For example:

G65P0002N100G01G90X100.Y200.F400R1000,

G01 instruction isn't executed, and all bit addresses are considered as the arguments of G65.

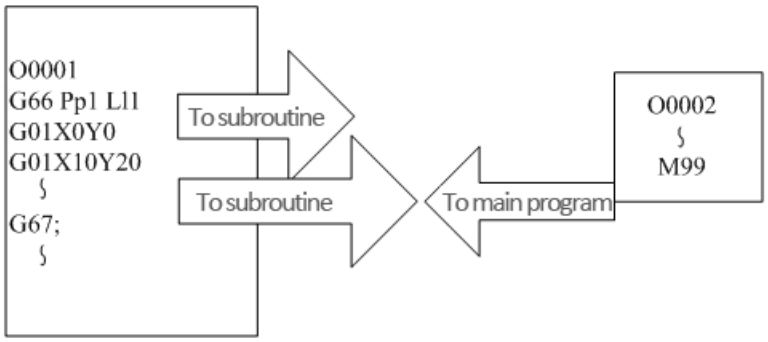
The comparison between the bit addresses specified by the arguments and local variable number follows:

Table 12.2 Comparison between Argument Specified Bit Addresses and Local Variables

Address	Variable No.	G65, G66, G661
A	#1	○
B	#2	○
C	#3	○
D	#7	○
E	#8	○
F	#9	○
G	×	×
H	#11	○
I	#4	○
J	#5	○
K	#6	○
L	×	×
M	#13	○
N	×	×
O	×	×
P	×	×
Q	#17	○
R	#18	○
S	#19	○
T	#20	○
U	#21	○
V	#22	○
W	#23	○
X	#24	○
Y	#25	○
Z	#26	○

○: can be used; ×: can't be used

(2)Mode calling A (motion instruction calling)



Between G66 and G67, after the segment with motion instruction is executed, all the specified macro subroutines are called and executed, and the execution times are specified by L.

Format:

G66 P_ L_ <argument>;	
P_	: subroutine No.
L_	: repeat times

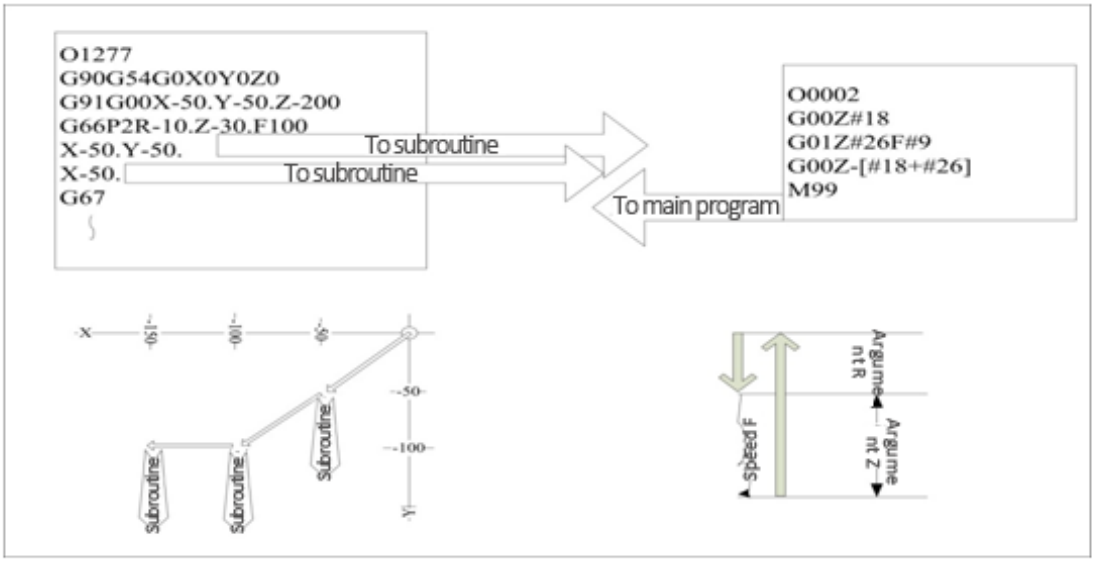
Details:

After G66 instruction is specified and before G67 (cancel) instruction is specified, all the segments with motion instruction will call G66 specified macro subroutine automatically after executed.

G66 and G67 instructions are in the same program, and must be specified in pair. If G66 instruction isn't executed first and G67 instruction is executed directly, the system will alarm.

In G66 instruction segment, all the bit addresses are considered as the arguments of G65.

For example: drilling cycle



Note:

G66 instruction executes the subroutine for the first time, and later motion instructions will call the subroutine automatically.

After G67 instruction takes effect, the subroutine won't be executed.

(3)Mode calling B (every segment calls)

Between G661 and G67, every instruction segment will call the specified macro subroutine unconditionally.

Format:

G661 P_ L_ <argument>;	
P_	: subroutine No.
L_	: repeat times

Details:

In G661 mode, all the read codes except O, N and G codes of every segment will be used as arguments.

In G661 instruction segment, all the bit addresses are considered as the arguments of G661.

For example:

G661P0002N100G01G90X100.Y200.F400R1000,

G01 instruction isn't executed, and all bit addresses are considered as the arguments of G661.

4.18.3 Variable

Function and purpose:

Variable is a useful function of macro. Four types of variables are available, which are local variable, global non-retentive variable, global retentive variable and system variable. These variables make the writing of macro very convenient and universal.

Using multiple variables:

- Macro calls variable, and the variable can be specified by multiple or expression. As below:

#1=10 #10=20 #20=30 #5=#[#[#1]]; #10=5 #10=20 #20=30 #5=1000 #[#[#1]]=#5	According to #1=10,#[#[#1]]=#[#10] According to #10=20,#[#10]=#20 Therefore, #5=#20 or #5=30 According to #1=10,#[#[#1]]=#[#10] According to #10=20,#[#10]=#20 Therefore, #20=#5 or #20=1000

- Example of specifying multiple variables:

#10=5 #5=100 #6=##10	##10 and #[#10] have the same meaning

- Replace the number with expression:

#10=5	
#[#10+1]=1000	#6=1000
#[#10-1]=-1000	#4=-1000
#[10*3]=100	#15=100
#[#10/2]=-100	#2=-100

Undefined variables:

The variables haven't been defined after the system is started are blank by default. The local variables that the arguments haven't been specified are also used as blank variables. The #0 of the system is also blank variable. In the calculation, blank variables can be used as 0; generally, #0 can't be used as expression L-value for calculation. However, if the programmers edit falsely, the program won't

report error and this measure doesn't have any effect.

➢ Calculation formula

#1=#0; ……#1=<blank>	Please note that the <blank> in the calculation formula indicates 0.
#2=#0+1; ……#2=1	<blank>+<blank>=0;
#3=1+#0; ……#3=1	<blank>+<fixed number>=<fixed number>
#4=#0*10;……#4=0	<fixed number>+<blank>=<fixed number>
#5=#0+#0;……#5=0	

➢ Variable reference

#1=<blank>
G0X#1Y1000; ……equals to G0X0Y1000
G0X#1+10Y1000;……equals to G0X10Y1000

➢ Conditional

In conditional determination, blank variable is equivalent to 0 in logic conditional operator.

4.18.4 Types of variables

⑴ Public variables

Any bit address can use public variables, which contain 600 groups; among those, #100~#199 are non-retentive public variables after power failure, #500~#999 are retentive public variables.

⑵ Local variables (#1-#32)

When calling subroutine, local variables can be defined with <argument> and only can be used in programs; the local variable of every macro program is independent, and thus can be repeated. (up to four levels)

G65 Pp1 Ll1 <argument>;
p1 : subroutine No.
l1 : repeat times

<Arguments> are Aa1 Bb1 Cc1… Zz1, etc.; the bit address specified by <argument> and the local variables in the subroutine are shown below:

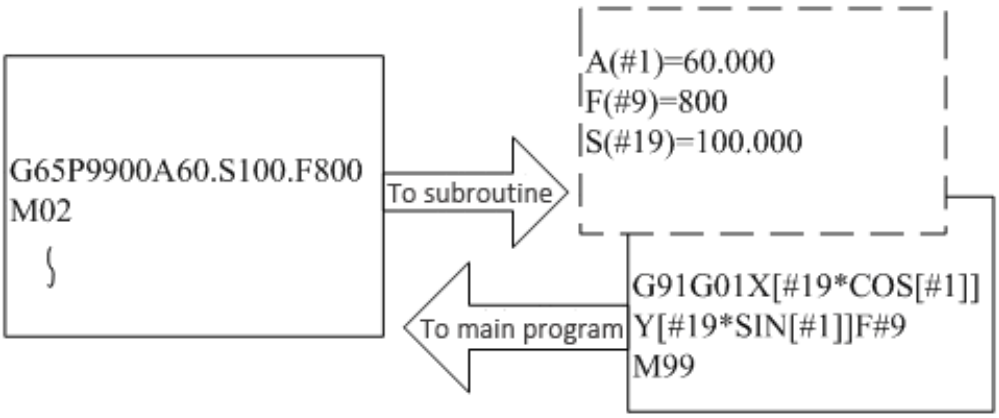
Bit address	Variable No.	Subroutine	Bit address	Variable No.	Subroutine
A	#1	○	N	×	×
B	#2	○	O	×	×
C	#3	○	P	×	×
D	#7	○	Q	#17	○
E	#8	○	R	#18	○

F	#9	○	S	#19	○
G	×	×	T	#20	○
H	#11	○	U	#21	○
I	#4	○	V	#22	○
J	#5	○	W	#23	○
K	#6	○	X	#24	○
L	×	×	Y	#25	○
M	#13	○	Z	#26	○

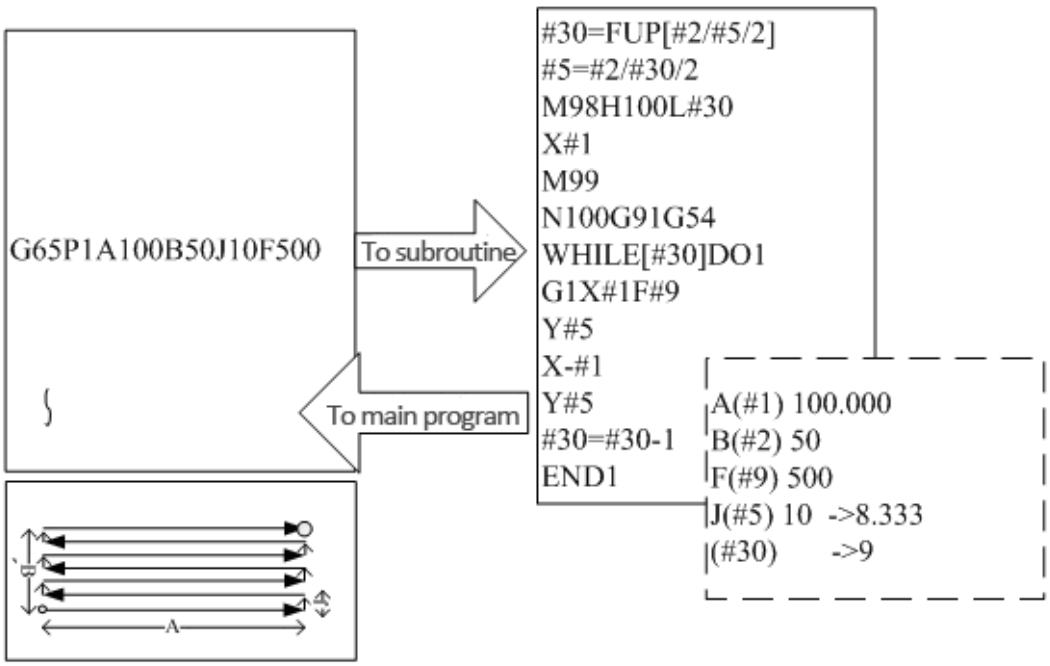
The argument bit addresses marked with "×" can't be used.

The argument bit addresses marked with "○" can be used.

① While calling macro program, the local variables in subroutine can be defined by specifying the <argument>



② Local variables can be used in respective subroutine freely.

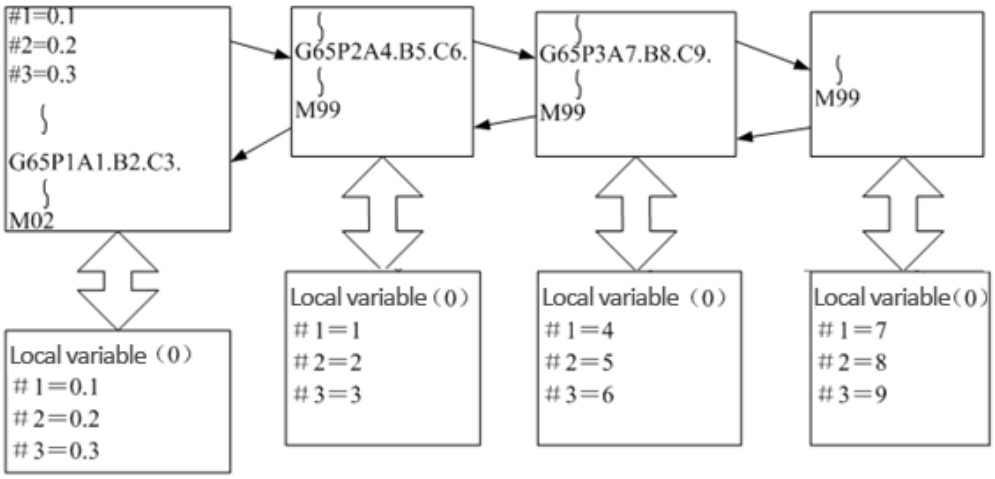


In face milling examples, argument J indicates that the spacing is 10mm during face milling; however, to ensure equal spacing processing, the spacing is changed to 8.333mm.

Secondly, local variable #30 is the calculation result of reciprocating processing times data.

② Local variables can be used for macro calling of every level independently up to four levels.

③ The main program (macro level 0) provides specific local variables; however, local variables can't use argument at level 0.



4.18.5 Calculus instruction

The variables allow various calculus expressions.

Instruction format:

#i=[expression]

The expressions may be combinations of constants, variables, functions or subexpressions.

In the table below, #j, #k can be replaced with constants.

Calculation method	#i=#j	Definition/replacement
Addition and subtraction	#i=#j+#k	Addition
	#i=#j-#k	Subtraction
	#i=#j OR #k or #i=#j|#k	32-bit OR calculation (logical AND)
	#i=#j XOR #k or #i=#j^#k	32-bit XOR calculation
Multiplication and division	#i=#j*#k	Multiplication
	#i=#j/#k	Division
	#i=#j MOD #k	Remainder
	#i=#j AND #k or #i=#j & #k	32-bit AND calculation (logical product)
Function calculation	#i=SIN[#k]	Sine
	#i=COS[#k]	Cosine
	#i=TAN[#k]	Tangent tanθ equals to sinθ/cosθ
	#i=ASIN[#k]	Arcsine
	#i=ATAN[#k]	Arctangent
	#i=ACOS[#k]	Arc cosine
	#i=SQRT[#k]	Square root
	#i=ABS[#k]	Absolute value
	#i=ROUND[#k]	Rounding
	#i=FIX[#k]	Abandon the decimal point
	#i=FUP[#k]	Carry the decimal point
	#i=LN[#k]	Natural logarithm
	#i=EXP[#k]	e(=2.718…) is exponent of the base

Note:

The values without decimal point are considered same as the values with decimal point (1=1.000)

The expression after the function must be bracketed with [ ].

Expression calculation priority:

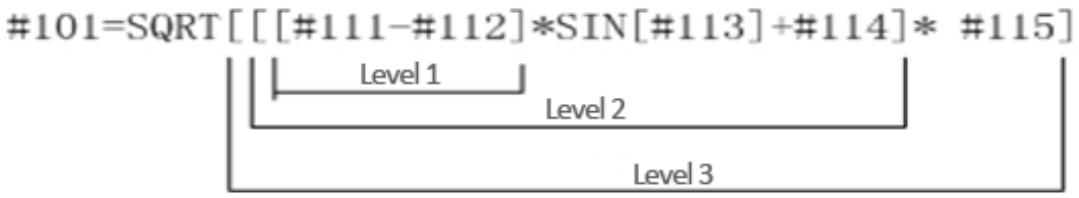
Smaller number indicates higher priority	Calculation symbol
1	#
2	[ ]
3	Function (SIN,COS,EXP…)
4	*,/,MOD
5	+,-
6	GE,GT,LE,LT
7	EQ,NE
8	AND,XOR,OR
9	=

Note:

The calculation expression of the same level follows the sequence from left to right.

The calculation expression has more priorities; if the expression is too long, please enforce the priority with [ ].

[ ] can be embedded in the calculation for up to five levels. As below:



Example of calculation commands:

(1) Specifying main program and argument	#i=#j	Definition/replacement
(2) Definition/replacement (=)	#1=1000 #2=1000 #3=#101 #4=#102 #5=#41	#1 1000.000 #2 1000.000 #3 100.000 #4 200.000 #5 -10.000
(3) Addition and subtraction (+ -)	#11=#1+1000 #12=#2-50 #13-#101+#1 #14=#41-3 #15=#41+#102	#11 2000.000 #12 950.000 #13 1100.000 #14 -13.000 #15 190.000
(4) Logical AND (OR)	#3=100 #4=#3 XOR 14	#3=01100100 14=00001110 #4=01101110=110
(5) XOR	#3 = 100 #4 = #3 XOR 14	#3=01100100 14=00001110 #4=01101010=106
(6) Multiplication and division (* /)	#21=100*100 #22=100.*100 #23=100*100. #24=100.*100 #25=100/100 #26=100./100. #27=100/100. #28=100./100. #29=#41*#101 #30=#41/#102	#21 10000.000 #22 10000.000 #23 10000.000 #24 10000.000 #25 1.000 #26 1.000 #27 1.000 #28 1.000 #29 -1000.000 #30 -0.050
(7) Remainder (MOD)	#31=#19 MOD #20	#19 48.000 #20 9.000 #31 3.000
(8) Logical product (AND)	#9 = 100 #10= #9 AND 15	#9 =01100100 15 =00001111 #10=00000100=4
(9) Sine (SIN)	#501=SIN[60] #502=1000*SIN[60]	#501 0.860 #502 866.025
(10) Cosine (COS)	#541=COS[45] #542=1000*COS[45.]	#541 0.707 #542 707.107

(1) Specifying main program and argument	#i=#j	Definition/replacement
(11) Tangent (TAN)	#551=TAN[60] #552=1000*TAN[60]	 #551 1.732 #552 1732.051
(12) Arcsine (ASIN)	#531=ASIN[100.500/201.] #532=ASIN[0.500] #533=ASIN[-0.500]	#531 30.000 #532 30.000 #533 -30.000
(13) Arc tangent (ATAN)	#561=ATAN[173205/100000] #562=ATAN[173205/100.] #563=ATAN[173.205/100000] #564=ATAN[173.205/100.] #565=ATAN[1.732]	#561 60.000 #562 60.000 #563 60.000 #564 60.000 #565 59.999
(14) Arc cosine (ACOS)	#521=ACOS[100./141.421] #522=ACOS[10/14.142] #523=ACOS[0.707]	#521 45.000 #522 44.999 #523 45.009
(15) Square root (SQRT)	#571=SQRT[1000] #572=SQRT[10.*10.+20.*20] #573=SQRT[#14*#14+#15*#15]	#571 31.623 #572 22.361 #573 190.444
(16) Absolute value (ABS)	#576=-1000 #577=ABS[#576] #3 = 70. #4=-50. #580=ABS[#4-#3]	#576 -1000.000 #577 1000.000 #580 120.000
(17)		
(18) Rounding (ROUND)	#21=ROUND[14/3] #22=ROUND[-14/3]	#21 5.000 #22 -5.000
(19) Abandon the decimal point (FIX)	#21=FIX[14/3] #22=FIX[-14/3]	#21 4.000 #22 -4.000
(20) Carry the decimal point (FUP)	#21=FUP[14/3] #22=FUP[-14/3.]	#21 5.000 #22 -5.000
(21) Natural logarithm (LN)	#101=LN[5] #102=LN[0.5] #103=LN[-5]	#101 1.609 #102 -0.693 error
(22) Exponent (EXP)	#104=EXP[2] #105=EXP[1] #106=EXP[-2]	#104 7.389 #105 2.718 #106 0.135

Calculation precision

Macro variable contains seven significant figures, and thus the precision may be reduced if single calculation value is too large or too small (9999999.000~0.0000001), and repeated calculation will cause cumulative error. Therefore, the macro variable should be in a reasonable range; in addition, while calculating trigonometric and exponential functions, too large value is also a reason of doubled error due to calculation error of the functions.

4.18.6 Control instruction

➢ **Conditional instruction**

Instruction format:

IF[conditional expression] GOTO n; (n is the order No. in the program)

The types of [conditional expression] are shown in the table below:

#i EQ #j	= when #i equals to #j
#i NE #j	≠ when #i doesn't equal to #j
#i GT #j	> when #i is larger than #j
#i LT #j	< when #i is smaller than #j
#i GE #j	≥ when #i is larger than or equals to #j
#i LE #j	≤ when #i is smaller than or equals to #j

Details:

When the condition is established, the program will go to execute line n; if it isn't established, it will execute the following in sequence.

When the [conditional expression] is ignored, the program will execute the GOTO sentence unconditionally.

The n of GOTO sentence must exist in the program, or else the program will alarm.

#i, #j, and n can be replaced with variables. For the segments that contain the order No. n specified by GOTO n, the order No. n must be in front of the segment, or else error may occur due to lack of keywords when the program jumps.

If the specified segment contains "/" in the front and is followed by Nn, the ignoring function of the segment will be invalid, and this segment will still go to execute.

When GOTO instruction is executed, the system will search downwards first; if not found, the system will return and search downwards from the program header; if still not found until the calling segment, the system will send alarm information.

EQ and NE only can be used for integers, and the values with decimal fraction should be compared with GT, GE, LT, and LE instructions.

➢ **Cycle conditional instruction**

Instruction format:

```
WHILE[expression]DO m;(m=1,2,3…127)
…
END m;
```

Details:

When the conditional expression is established, the programs between WHILE and END will be executed repeatedly; if not established, the program will go to next segment of END m directly.

WHILE [expression] DO m and END m should be used in pair. If the LEaan line of WHILE [expression] is ignored, the segments between DO m and END m will be repeated endlessly. The range of M is 1-127.

WHILE allows nesting up to 27 levels.

(1) Same identifier and can be used repeatedly

```
WHILE[...]DO1
...
END1
...
WHILE[...]DO1
...
END1
M30
```

Correct

(2) The identifier of WHILE~Dom can be specified freely

```
WHILE[...]DO1
...
END1
...
WHILE[...]DO2
...
END2
...
WHILE[...]DO3
...
END3
...
WHILE[...]DO4
...
END4
M30
```

Correct

(3) WHILE~DO m contains up to 27 level; the range of m is 1~127, and can be specified freely.

```
WHILE[...]DO1
WHILE[...]DO2
...
WHILE[...]DO27
...
END27
...
END2
...
END1
M30
```

Correct

Note: m can't be used repeatedly after specified in nesting.

(4) The level of WHILE~DO m can't exceed 28

```
WHILE[...]DO1
WHILE[...]DO2
...
WHILE[...]DO27
WHILE[...]DO28
...
END28
END27
...
END2
END1
M30
```

Correct

(5) WHILE~DO m must be specified before END m

```
END 1
...
WHILE~DO 1
```

False

(6) WHILE~DO m must be corresponded in one program

```
WHILE~DO1
...
WHILE~DO2
...
END1
```

False

(7) WHILE~DO m can't be crossed

```
WHILE~DO1
...
WHILE~DO2
...
END1
...
END2
```

False

(8) The calling of M98, G65, G66 and other subroutines can be executed between WHILE~DO m

```
WHILE~DO1
G65 P100
...
END1
```

Allow

(9) GOTO can't jump into WHILE cycle

```
IF~GOTO n
WHILE~DO1
Nn;
...
END1
```

False

(10) GOTO can jump out of WHILE cycle

```
WHILE~DO1
IF~GOTO n
;
...
END1
Nn
```

Correct

(11) Call subroutine in WHILE~DO cycle, and execute WHILE~DO cycle in the subroutine; at this moment, the nesting levels of WHILE is the sum of main program and subroutine, and can't exceed 27.

```
WHILE~DO1
G65 P100 To subroutine
...
END1

O100
WHILE~DO1
...
END1
```

(12) In macro program, M99 will cause program error if WHILE and END are not used in pair.

False

```
WHILE~DO1
G65 P100 To subroutine
...
END1

O100
WHILE~DO1
...
M99
END1
```

M99 causes that DO and END can't be matched

4.18.7 Notes of using macro

Macro program uses variables to calculate and combine the NC program described by the logic, making the program more versatile. However, since the logical calculation is flexible, it may lead to some hidden errors; to avoid logic errors, it is necessary to note the mode when writing macros.

(1) Variable initialization; all the variables used in the program should be initialized at the beginning of the program; the variables for transfer also require an intermediate variable, in order to avoid error due to parameters modified by the program during multiple processing.

(2) In main program, subroutine or macro, please use local variables as much as possible; all the local variables will be cleared during program calling, in order to keep a clean environment for programming. Even if the reference is false, it will be located easily.

(3) Same as subroutine, macro can't be used in tool radius compensation; therefore, please cancel the compensation before calling.

4.18.8 Macro variable user parameters system configuration

- Macro variables contain [User] menu, which is used to rename the macro variable addresses related to process parameters, in order to make the operation more intuitional; the specific method is to configure the system with csv file;
- CSV is an Excel format. Please create a configuration datasheet in the following format in Excel, save as CSV file, name the file as SYSTABLE.CSV, and save it in directory ADT. Select [Parameter > Management > Import CSV system configuration], and the system will automatically check whether the file exists; if yes, the system menus will be configured with this file.
- The configuration of CSV macro variable user parameters follows:

Example of user macro configuration: the range of sequence number is 17~100 and the range of corresponding macro address is 500~999; this macro address is nonvolatile. The user can customize up to 50

	A	B	C	D	E
1	User macro configuration	S/N	User-defined name	Corresponding macro address	
2	User-defined macro variable name 1	17	User-defined name 1	500	
3	User-defined macro variable name 2	18	User-defined name 2	501	
4	User-defined macro variable name 3	19	User-defined name 3	502	
5	User-defined macro variable name 4	20	User-defined name 4	503	
6	User-defined macro variable name 5	21	User-defined name 5	504	
7					
8					
9	User-defined alarm setting	S/N	External alarm S/N	User-defined alarm prompt	
10	User-defined alarm setting 1	200	1	External alarm 1	
11	User-defined alarm setting 2	201	2	External alarm 2	
12	User-defined alarm setting 3	202	3	External alarm 3	
13	User-defined alarm setting 4	203	4	External alarm 4	
14	User-defined alarm setting 5	204	5	External alarm 5	
15					

Example of user-defined alarm configuration: the range of the sequence number is 200~215, the range of corresponding external alarm sequence number is 1~16, sequence number corresponds to bit number 1~16 of external alarm register, and

Appendix 1: Program burning

- **Copying upgrade program with USB disk**

(1) In the main menu, press [Prog] to enter the program interface

(2) Press [File] to enter the file interface

(3) Insert the USB disk, open the disk in the root directory and enter directly

(4) Move cursor to the upgrade file ADTROM.BIN, select [Copy], enter ADT directory in disk D and paste;

(5) Select the second upgrade file NC_RES.NC; skip this step if the file doesn't exist. Also select Copy, enter disk D, and paste in directory ADT.

(6) If the ADT directory has KEY directory and LOGO.BMP, it is not necessary to operate these two files. If not, select and copy KEY directory and LOGO.BMP, enter D disk, and paste in the ADT directory.

(7) If the CODE directory in D disk has files associated with key processing, it is not necessary to operate this part. If not, select CODE directory in USB disk, enter D disk, and paste.

(8) Restart the system after upgrade, press and hold the [Cancel] button for three seconds when the system is starting, the system pops up the interface asking the user to enter the password: "Input Pass:", enter the password "26722719", and the system enter the BIOS interface, press [4] to select "4 --- Start-up mode"; if USB disk is selected, press the [Cancel] button, and then press [.] and [9] successively to restart the system and make the program in disk D take effect. If normal start is selected, press the [Cancel] button, and then press [1] and select "1 --- BIOS Settings", then press [1] and select "1 --- Program area", press [Y] according to the prompt, then press [Cancel], and then press [.] and [9] successively to restart the system and make the program in disk D take effect.

(9) Enter System Info in Diagnosis menu to view the system version and compilation date, and check whether the upgrade is successful.

- **Enter BIOS and upgrade system via USB disk**

This method is not commonly used

- **Connect to the control system via PC and upgrade the system**

(1) Connect the equipped data cable to "XS11 USB" interface of the control system, and insert the USB interface of data cable into the USB port of the computer

(2) In the main menu, press [Prog] to enter the program interface;

(3) Press [File] to enter the file management interface;

(4) Insert the USB disk, open the disk in the root directory and enter directly

(5) Move cursor to the upgrade file ADTROM.BIN, select [Copy], enter ADT directory in disk D and paste;

(6) Select the second upgrade file NC_RES.NC; skip this step if the file doesn't exist. Also select Copy, enter disk D, and paste in directory ADT.

(7) If the ADT directory has KEY directory and LOGO.BMP, it is not necessary to operate these two files. If not, select and copy KEY directory and LOGO.BMP, enter D disk, and paste in the ADT directory.

(8) If the CODE directory in D disk has files associated with key processing, it is not necessary to operate this part. If not, select CODE directory in USB disk, enter D disk, and paste.

(9) After upgrade, press [Cancel] to disconnect the communication between the control system and PC; restart the system, press and hold the [Cancel] button for three seconds when the system is starting, the system pops up the interface asking the user to enter the password: "Input Pass:", enter the password "26722719", and the system enter the BIOS screen, press [4] to select "4 --- Start-up mode"; if USB disk

is selected, press the [Cancel] button, and then press [.] and [9] successively to restart the system and make the program in disk D take effect. If normal start is selected, press the [Cancel] button, and then press [1] and select "1 --- BIOS Settings", then press [1] and select "1 --- Program area", press [Y] according to the prompt, then press [Cancel], and then press [.] and [9] successively to restart the system and make the program in disk D take effect.

(10) Enter System Info in Diagnosis menu to view the system version and compilation date, and check whether the upgrade is successful.

- **Caution:**

The update of KYX60A system is different from KYX00 system. KYX60A system should be updated according to corresponding item number, or else the registration number of the system will not match, and the interface display is proper, but X, Y, Z, A-axis of the system do not move on the machine tool. The item number and registration number of KYX60A are as follows:

Item No.	Model	Number of axes	Hardware platform	Registration No.
BZ001B098A	KY260A	2	CNC4620	0X64088c40
BZ001B099A	KY360A	3	CNC4640	0X64088c60
BZ001B100A	KY460A	4	CNC4640	0X64088c40

After the update, check the last line in [System Info] of [Diagnosis] if ID is registered successfully; if yes, the registration is successful; if not, the registration failed.

Appendix 2: Processing example

Not available

History (I)

Feedback	Qu Chong	Date	2013.5.3	Current version / total pages	V1.2, 185
Problem Description	Optimize the system, integrate the comprehensive parameters, axis configuration, management parameters and IO configuration parameters into comprehensive parameters, axis configuration parameters and management parameters, and delete parameters not commonly used. The comprehensive parameters have different sequences and the manual should be rewritten.				
Confirmed by	Qu Chong				
Revised version		Total pages after revision		Revised by	

History (II)

Feedback	Mo yunfeng	Date	2013.6.4	Current version / total pages	V1.2, 185
Problem Description	1. “USB and Serial Port” of (6)XS9 and XS11 on page 12 of the maintenance manual should be “Serial Port and USB” 2. It is recommended to revise (8) Controller 4VDC switching power supply for system power supply on page 13 of the maintenance manual into “(8) Controller 24VDC switching power supply for control system power supply only rather than other load power supply” 3. Delete the analog spindle description on page 28 of the maintenance manual.				
Confirmed by	Qu Chong				
Revised version		Total pages after revision		Revised by	Qu Chong

History (III)

Feedback		Date		Current version / total pages	
Problem Description					
Confirmed by					
Revised version		Total pages after revision		Revised by	

History (IV)

Feedback		Date		Current version / total pages	
Problem Description					
Confirmed by					
Revised version		Total pages after revision		Revised by	

History (V)

Feedback		Date		Current version / total pages	
Problem Description					
Confirmed by					
Revised version		Total pages after revision		Revised by	

History (VI)

Feedback		Date		Current version / total pages	
Problem Description					
Confirmed by					
Revised version		Total pages after revision		Revised by	