TC5510/TC5522/TC5530/TC5522R/TC5530R

Motion Controller (Stepping Motor Controller) Manual

Full figure

Left View

Back view

Right View

TC55 Breakout Board (TC55-MOD)

Note: The TC55 series are exactly the same panel size and shell film. TC55-MOD is an optional product.

1. Outline

The TC55 panel-type motion controller (CNC system) uses high-performance 32-bit CPU. The drive device adopts subdivision stepping motor or AC servo motor. It is equipped with LCD monitor and fully enclosed touch-type operating keyboard. The system has high reliability, high precision, low noise, easy to operate etc. This controller can control 1-3 motor motion and realize Point to Point, linear interpolation, circular interpolation and other operations. It has loop, jump and simple PLC functions etc. Simple and clear parameters bring you convenient and fast operation. Input/output setting functions are convenient for your to use and maintain， and applicable to all kinds of 1-3 axis motion device.

Product features:

The boot screen can be modified

Controller or upper computer dual-mode programming

Independent 24V power supply reverse connection protection

IO optically-coupled isolation

Output short circuit protection

Manual positive or reverse rotation can be controlled by the external switches synchronously

Simple PLC logic

The parameter area password can be set

Applicable product type:

l CNC drilling machine system, CNC lathe system, CNC milling machine system, CNC grinding machine system

l cutting machine control system, welding control system, dispenser control system, feeding control system

l displacement platform, one-dimensional control platform, two-dimensional control platform, three-dimensional control platform

l threading machine control system, screwdriving machine control system

l spraying production line control system, assembly production line control system, the meter counter control system

Technical characteristics:

- Automatically execute： Actual running, Empty running, Single-step execution, Terminate program, Start and Pause function
- Manual operation：Manual high speed, Manual low speed, Jog operation, Back to program zero, Back to mechanical zero
- Program management： Edit program, Program read-in, Delete program, Save program
- Parameter settings： Set various control parameters of processing and operation, get the best status of processing effect

System parameters:

High-performance, high-speed 32-bit CPU

High-grade black and white double color LCD monitor (resolution: 192 × 64)

Special motion control chip (signal output: 5V TTL)

Universal customizable input / output (16 photoelectric isolated 24V inputs, 8 relay outputs)

User processing program memory (can store 20 programs)

Minimum data unit 0.001 mm

Maximum data size ± 99999.999mm

Maximum pulse output frequency 150KHz

System main functions are automatic, manual, program editing, system parameters, self-check, settings, etc.

Installation:

panel type installation structure

dimension 172 x 94 x 30, installing size 164 x 86

Power supply requirements

≥ DC 24V/40W

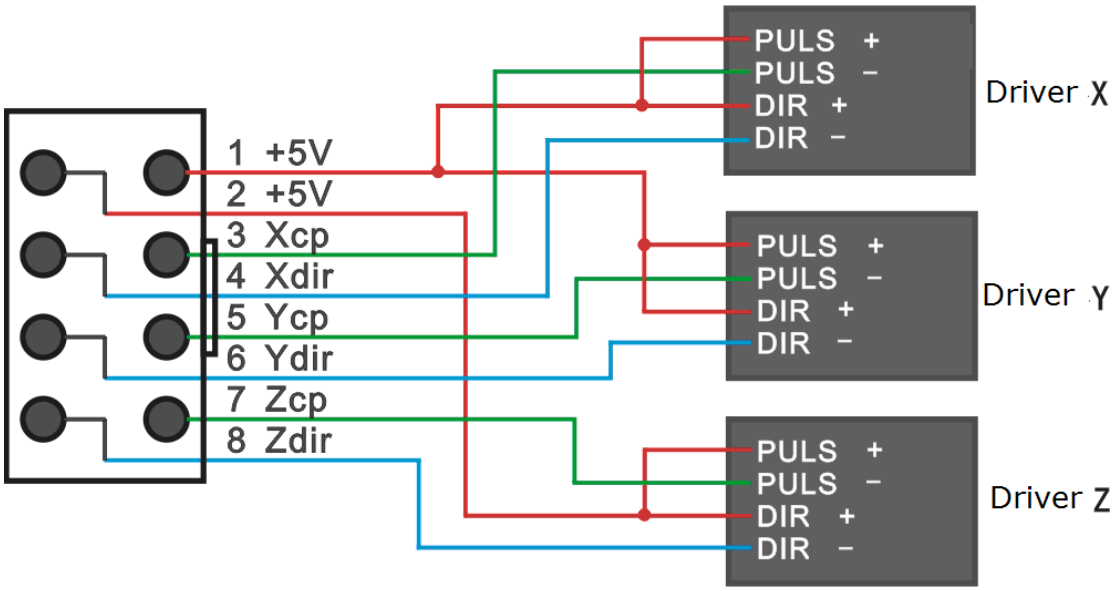
Environmental requirements

working temperature: 0 ~ 60 ℃

relative humidity: 5 ~ 90% no condensation

2. Connection

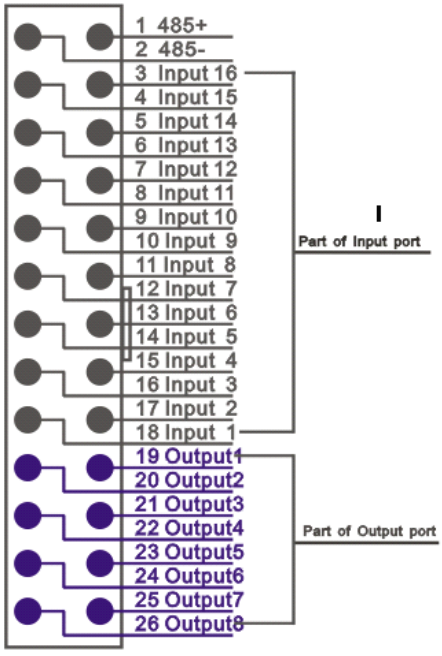
2.1 Pulse interfaces and drives wiring diagram



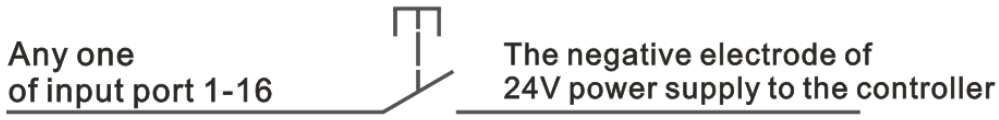
Note: the red line is No.1 pin

TC5510 only take the first 4 wires or terminals of 8P. Similarly TC5522 or TC5522R only take the first 6 wires or terminals of 8P.

2.2 Input/output and switches and relays wiring diagram.



2.2.1 Input wiring diagram

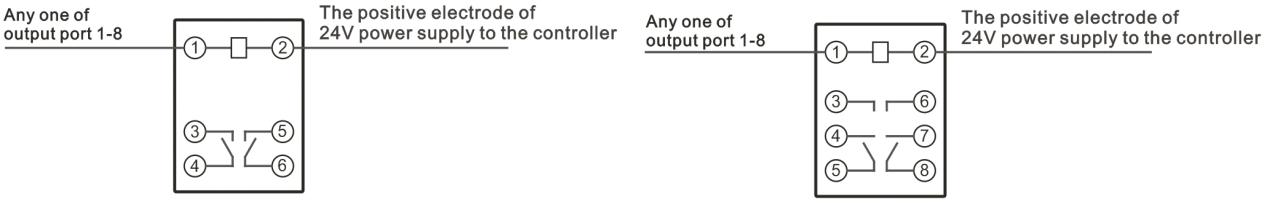


Any one of input port 1-16
The negative electrode of 24V power supply to the controller

The picture shows the two-wire standard mechanical switch and photoelectric switch connection method. One end of the switch is connected to the selected and defined input line. The other end of the switch must be connected to the negative electrode of 24V power supply to the controller.

Note: the photoelectric switch must be NPN type.

2.2.2 Output line wiring diagram



Any one of output port 1-8
The positive electrode of 24V power supply to the controller

As shown in figure, contacts 1 and 2 are the coil contacts of relay.

Output line is connected to one end of the relay coil. The other end of the coil is connected to the positive electrode of 24V power supply to the controller.

Note 1: If the connected relay or other electrical component is required for power supply positive or negative access, then wire according to the electrical diagram of the relay or the electrical component.

Note 2: the red line is No.1 pin

Any one of input port 1-16 can be connected to one end of the switch. The specific switch definition depends on the input parameter values of IO settings of

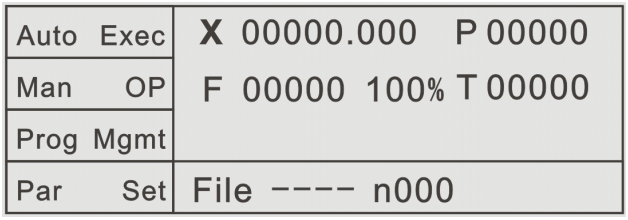
the controller parameters area. The other end of the switch must be connected to the negative electrode of 24V power supply to the controller.

Any one of output port 1-8 can be connected to one end of the relay coil. The specific output port definition depends on the output parameter values of IO settings of the controller parameters area. The other end of the relay coil must be connected to the positive electrode of 24V power supply to the controller.

Controller power supply interface definition is on the back of the controller shell. Please wire according to the mark on the back. The power supply is 24V DC. Power is not less than 40W.

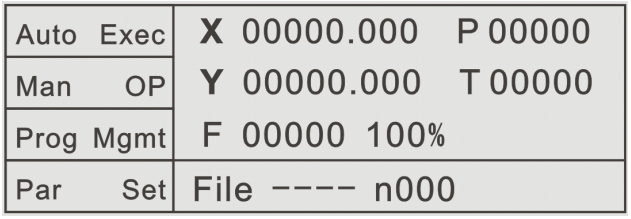
3. Operation

Boot and enter into the main operation interface after showing the boot screen



AutoExec Automatically execute
Man OP Manual operation
ProgMgmt Program management
File
Pars Set Parameter settings

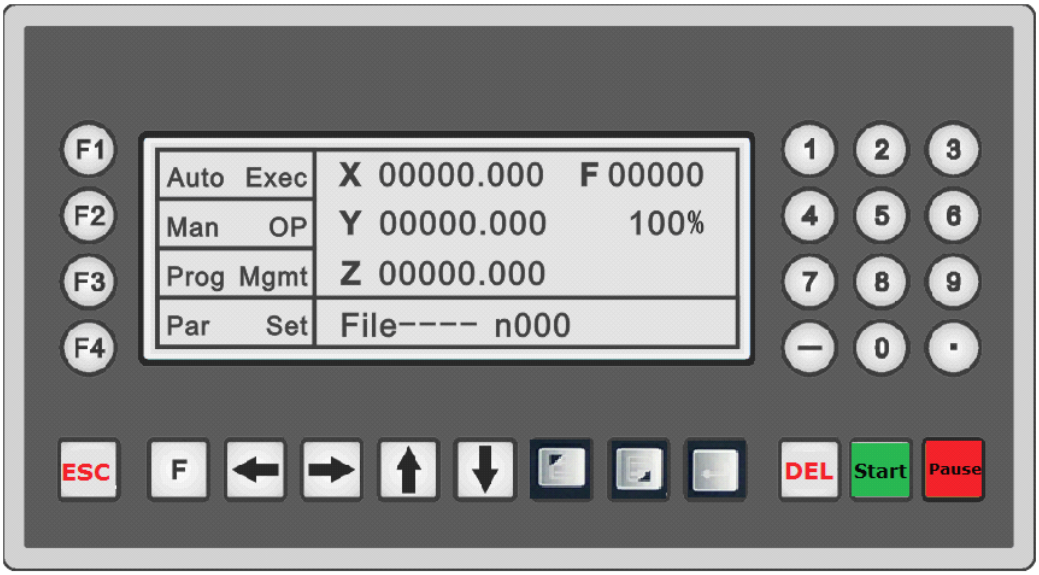
The picture shows the main interface of TC5510 after boot.



The picture shows the main interface of TC5522 or TC5522R after boot.

Auto Exec	X 00000.000 F 00000
Man OP	Y 00000.000 100%
Prog Mgmt	Z 00000.000
Par Set	File ---- n000

The picture shows the main interface of TC5530 or TC5530R after boot.



1. The screen area

AutoExec： Automatically execute. Click to enter the program execution interface, including the default Actl Run(Actual running), Mt Run (Empty running), SS Mo(Single-step mode running), TrmtProg(Terminate program).

Man OP: Manual operation. Click to enter the manual operation interface, including the default Man LSpd(Manual low speed), Man HSpd(Manual high speed), Jog Op(Jog operation), BkTProg0(Back to program zero), BkTMech0(Back to mechanical zero).

ProgMgmt: Program management. Click to enter the program editing interface, including Ed Prog(Edit program), Prog RI(Program read-in), Del Prog(Delete program), Sv Prog(Save program).

Pars Set: Parameters settings. Click to enter the parameter setting interface, including Sys Pars(System parameters), Sys SC(System self-check), IO Set(IO settings), Usr Mgmt(User management).

X: The current coordinate position

Y: The current coordinate position

Z: The current coordinate position

F: The current running speed

Unit: subject to the selected electronic gear calculation formula. When electronic gear default is 1:1, the unit default is the pulse number.

100%: the speed percentage (rate) of the current F value

P: the countdown of loops

T: the countdown of delay time

2. The Key area

F1: In the current state, the function key corresponding to the F1 key position, is valid in the current state.

F2: In the current state, the function key corresponding to the F2 key position, is valid in the current state.

F3: In the current state, the function key corresponding to the F3 key position, is valid in the current state.

F4: In the current state, the function key corresponding to the F4 key position, is valid in the current state.

0-9： Valid in the Ed Prog(Edit program) state or Pars Set(Parameters settings) state.（0、1、2、3、4、5、6、7、8、9）

1-3：In the Man Op(Manual operation) state, long press 1 key for three seconds, you can modify the X-axis coordinate to "X-axis mechanical reference point value", which is set in the Sys Pars(System parameters) area of Pars Set (Parameters settings) area. In the Man Op(Manual operation) state, long press 2 key for three seconds, you can modify the Y-axis coordinate to "Y-axis mechanical reference point value", which is set in the Sys Pars(System parameters) area of Pars Set (Parameters settings) area. In the Man Op(Manual operation) state, long press 3 key for three seconds, you can modify the Z-axis coordinate to "Z-axis mechanical reference point value", which is set in the Sys Pars(System parameters) area of Pars Set (Parameters settings) area. The three values defaults are 0.

“-”：In the Ed Prog(Edit program) state, it is used to distinguish between movement direction. In the Man Op(Manual operation) state, it is used to operate the Z-axis reverse rotation.

“.”：In the Ed Prog(Edit program) state, it is used to distinguish between numerical accuracy. In the Man Op(Manual operation) state, it is used to operate the Z-axis forward rotation.

Exit: Invalid in the program running process and valid in the rest of the state. It is used to return to previous operation interface.

“←”：In the Man Op(Manual operation) state, X-axis is reverse rotation. It is valid in

the ProgMgmt(Program management) state or Pars Set(Parameters settings) state and used to move the cursor.

"→": In the Man Op(Manual operation) state, X-axis is forward rotation. It is valid in the ProgMgmt(Program management) state or Pars Set(Parameters settings) state and used to move the cursor.

"↑": In the Man Op(Manual operation) state, Y-axis is forward rotation. It is valid in the ProgMgmt(Program management) state or Pars Set(Parameters settings) state and used to move the cursor.

"↓": In the Man Op(Manual operation) state, Y-axis is reverse rotation. It is valid in the ProgMgmt(Program management) state or Pars Set(Parameters settings) state and used to move the cursor.

PgUp: PageUp. In the main interface, AutoExec(Automatically execute) or Man Op(Manual operation) state，it is the rate+ key of the speed F value and used to increase rate of the current speed F value, maximum to 150%. In the ProgMgmt(Program management) state or Pars Set(Parameters settings) state, it is used to page up the current page.

PgDn: PageDown. In the main interface, AutoExec(Automatically execute) or Man Op(Manual operation) state，it is the rate- key of the speed F value and used to reduce rate of the current speed F value, minimum to 10%. In the ProgMgmt(Program management) state or Pars Set(Parameters settings) state, it is used to page down the current page.

Cfm: Confirm. Valid in the Pars Set(Parameters settings) state. When modifying

parameters or entering password for user login, press the Cfm(Confirm) key to determine to modify parameters or user login successfully.

Spc: Space. It is valid in the Ed Prog(Edit program) state or Pars Set(Parameters settings) state and used to clear(delete) the value of the current cursor position.

St: Start. It is valid in the AutoExec(Automatically execute) state and used to start the execution of the current program file.

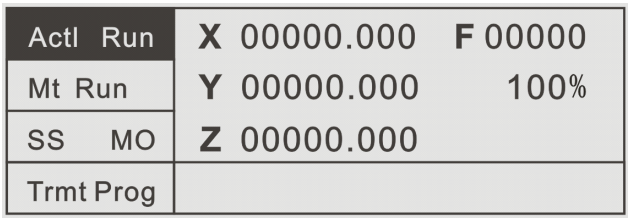
Paus: Pause. It is valid in the AutoExec(Automatically execute) state and used to pause the execution of the current program file.

3.1 AutoExec(Automatically execute)

(TC5530R is used as an example)

3.1.1 Actl Run(Actual running)

In the main interface, press the left F1 key to enter the AutoExec(Automatically execute) interface. At this time, the Actl Run(Actual running) key color is negative display, and the system default is in the Actl Run(Actual running) state in the program continuous running mode.



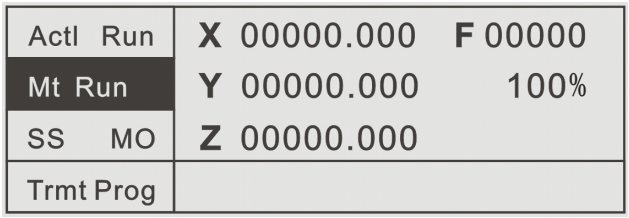
Actl Run (Actual running)
Mt Run(Empty running)
SS Mo (Single-step mode)
TrmtProg (Terminate program)

In this state, by pressing the St(Start) key on the controller panel or the external input line St(Start) key defined in the IO set(IO settings) area of Pars Set (Parameters

settings) area, the controller begins running with the default last read-in program file. Press the Paus(Pause) key to pause the running program. To continue to execute the paused program file, press the St(Start) key again. To exit the program and remain in the AutoExec(Automatically execute) interface, press the TrmtProg(Terminate program) key. If want to direct exit to the main interface, the program must be in the Paus(Pause) or TrmtProg(Terminate program) state, then press the Exit key to return to the main interface, or the Exit key is invalid.

3.1.2 Mt Run(Empty running)

In the AutoExec(Automatically execute) interface, press the left F2 key to enter the Mt Run(Empty running) state. At this time, the controller Mt Run(Empty running) key color is negative display, and the system is in the Empty running state in the program continuous running mode.

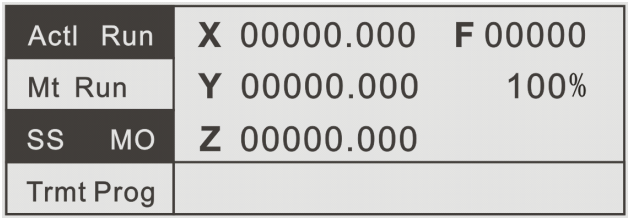


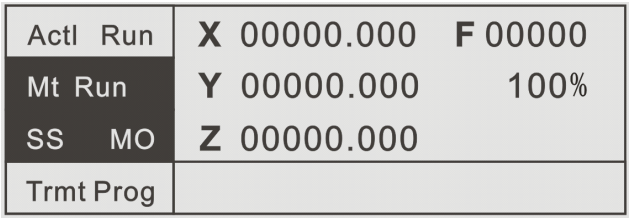
Cont Mo (Continuous mode)

In this state, press the St(Start) or Paus(Pause) key on the controller panel or the external input line St(Start) or Paus(Pause) defined in the IO set(IO settings) area of Pars Set (Parameters settings) area, to operate Empty running on program. At this time, the controller program runs normally without any external action. It is often used for simulation test after program editing and before actual running.

3.1.3 Single-step execution

In the Actl Run(Actual running) or Mt Run(Empty running) state, if SS Mo(Single-step mode) is not chosen, the system default is in the program continuous running mode and the SS Mo(Single-step mode) key color is not negative display. If press the left F3 key corresponding to SS Mo(Single-step mode), the SS Mo(Single-step mode) key color is negative display and in the Actl Run(Actual running) or Mt Run(Empty running) state, program runs with Single-step mode. Each step of the program running is controlled by the St(Start) key on the controller panel or the external input line St(Start) defined in the IO set(IO settings) area of Pars Set (Parameters settings) area. Press the key, the current file executes a line of program, press again, then continues to execute next line of program.



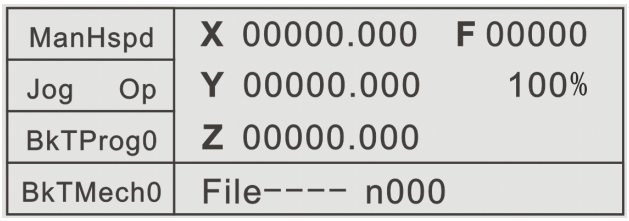


3.1.4 TrmtProg(Terminate program)

In the program pause or running state, by pressing this key, the program stops and jumps to the first line of the program in the AutoExec(Automatically execute) interface.

3.2 Man Op(Manual operation)

In the main interface, press the left F2 key to enter the Man Op （Manual operation） interface.



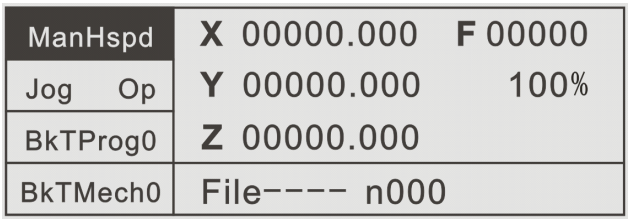
Man HSpd(Manual high speed)
Jog Op(Jog operation)
BkTProg0(Back to program zero)
BkTMech0(Back to mechanical zero)

At this time, the default is Man LSpd(Manual low speed). The speed parameter is set in the Spd Pars(Speed parameters) area of the Pars Set(Parameters settings) area, and is the fixed mode(the rate key is valid).

Through the keys（‘←’‘→’‘↑’‘↓’‘-’‘.’）on the controller panel or the direction keys of external IO settings, the forward or reverse rotation of the motor can be controlled.

3.2.1 Man HSpd(Manual high speed)

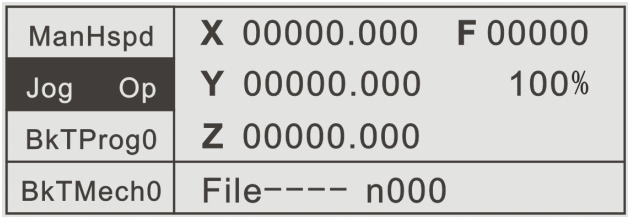
In the Man Op(Manual operation) interface, press the left F1 key to enter the Man HSpd(Manual high speed) interface. At this time, the Man HSpd(Manual high speed) key color is negative display, and the system is in the Manual high speed state. The speed parameter is set in the Spd Pars(Speed parameters) area of the Pars Set(Parameters settings) area, and is the fixed mode(the rate key is valid).



Through the keys（‘←’‘→’‘↑’‘↓’‘-’‘.’）on the controller panel or the keys of external IO settings, the forward or reverse rotation of the motor can be controlled.

3.2.2 Jog Op(Jog operation)

In the Man Op(Manual operation) interface, press the left F2 key to enter the Jog Op(Jog operation) interface. At this time, the Jog Op(Jog operation) key color is negative display, and the system is in the Jog operation state. The Jog increment parameter is set in the CtrlPars(Control parameters) area of the Pars Set(Parameters settings) area, and is the fixed mode.



Through the keys（‘←’‘→’‘↑’‘↓’‘-’‘.’）on the controller panel or the keys of external IO settings, the forward or reverse rotation of the motor can be controlled with the already set displacement increment.

3.2.3 BkTProg0(Back to program zero)

In the Man Op(Manual operation) interface, press the left F3 key to enter the BkTProg0(Back to program zero) interface. At this time, the BkTProg0(Back to program zero) key color is negative display, and the system is in the BkTProg0(Back to program zero) state. The speed parameter is set in the Spd Pars(Speed parameters) area of the Pars Set(Parameters settings) area, and is the resultant speed of the system high speed. That is, returning back to the program coordinates zero in linear interpolation (start at the same time, stop at the same time) mode. At this time, only Pause is valid. After returning back to the coordinates zero, the key color negative display is back to the normal status.

ManHspd	X 00000.000	F 00000
Jog Op	Y 00000.000	100%
BkTProg0	Z 00000.000	
BkTMech0	File---- n000	

ManHspd	X 00000.000	F 00000
Jog Op	Y 00000.000	100%
BkTProg0	Z 00000.000	
BkTMech0	File---- n000	

3.2.4 BkTMech0(Back to mechanical zero(mechanical reference point))

In the Man Op(Manual operation) interface, press the left F4 key to enter the BkTMech0(Back to mechanical zero) interface. At this time, the BkTMech0(Back to mechanical zero) key color is negative display, and the system is in the BkTMech0(Back to mechanical zero) state. The Bk0 HSpd(Back to zero high speed) or Bk0 LSpd(Back to zero low speed) speed parameter of Back to mechanical zero**(**reference point**)** is set in the Spd Pars(Speed parameters) area of the Pars Set(Parameters settings) area, and is the fixed mode(the rate key is valid).

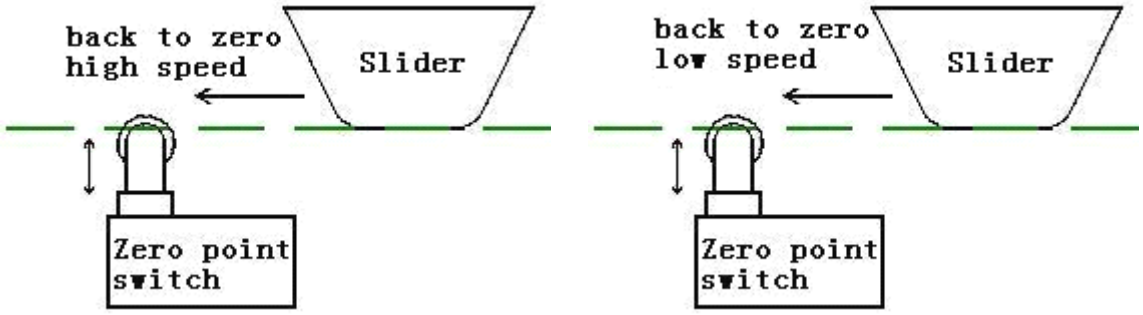
ManHspd	X 00000.000	F 00000
Jog Op	Y 00000.000	100%
BkTProg0	Z 00000.000	
BkTMech0	File---- n000	

Through the keys（‘←’‘→’‘↑’‘↓’‘-’‘.’）on the controller panel or the direction keys of external IO settings, the single direction Back to mechanical zero(mechanical reference point) operation of the motor can be controlled. Before completing the BkTMech0(Back to mechanical zero) operation, only the Paus(Pause) key is valid.

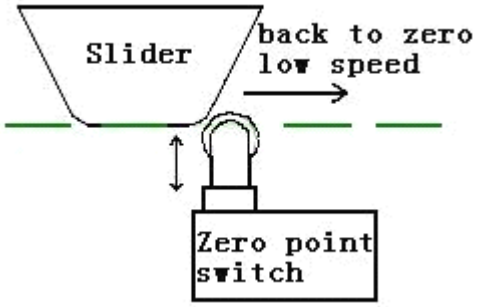
Back to mechanical zero is a set of combined actions. The premise is first to set a set of zero point switches(both mechanical and photoelectric switches are allowed), assuming that the switches are in the N.O.(Normally open) state.

1） .Firstly move to the pressed key direction (or the direction specified by the

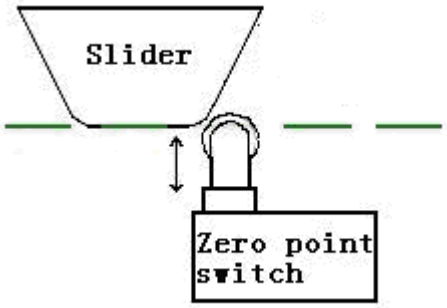
program instructions) with Bk0 HSpd(Back to zero high speed), until touch the zero point switch of the axis and the switch status changes from Open to Closed.



2）.After touching the zero point switch, move with the speed quickly reduced to Bk0 LSpd(Back to zero low speed), until pass the switch and the switch status changes from Closed to Open.



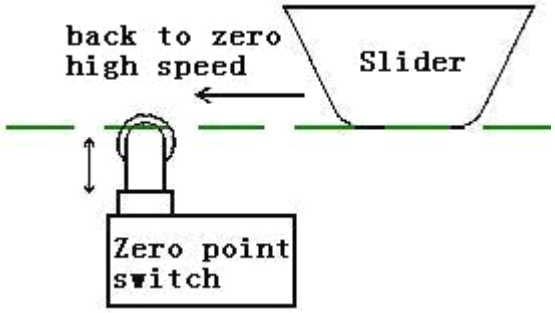
3）. After the switch status changes to Open, the axis will automatically continue to reverse move with Bk0 LSpd（Back to zero low speed）, until press the switch again and the switch changes from Open to Closed secondly. At this time, stop instantly, the action of Back to mechanical zero is completed and the axis coordinate on the controller interface changes to the axis reference point value. The axis reference point value is set in the CtrlPars (Control parameters) area of the Pars Set(Parameters settings) area.



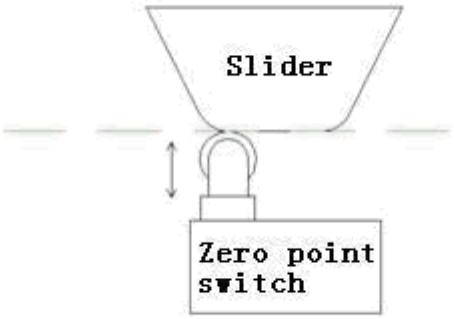
Note: The machine tool limit switch is used as an example in the above figures. Also the photoelectric switch can be used as the zero point switch, similarly.

Back to zero mode II:

1). Firstly move to the pressed key direction(or the direction specified by the program instructions) with Bk0 HSpd(Back to zero high speed), until touch the zero point switch of the axis and the switch status changes from Open to Closed.

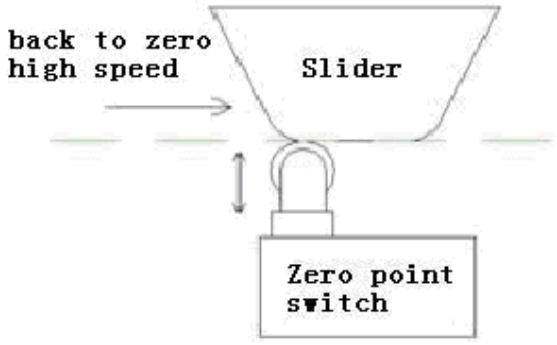


2). After the zero point switch is touched, the speed is quickly reduced to zero.



3). Automatically reverse move with Bk0 LSpd（Back to zero low speed）, until the

switch changes from Closed to Open. At this time, stop instantly, the action of Back to mechanical zero is completed and the axis coordinate on the controller interface changes to the axis reference point value. The axis reference point value is set in the CtrlPars (Control parameters) area of the Pars Set(Parameters settings) area.



Note: Back to zero mode II has some risk. If adopting this mode, propose that Bk0 HSpd(Back to zero high speed) is set smaller to avoid the slider exceed the zero point switch when back to zero.

3.3 ProgMgmt(Program Management)

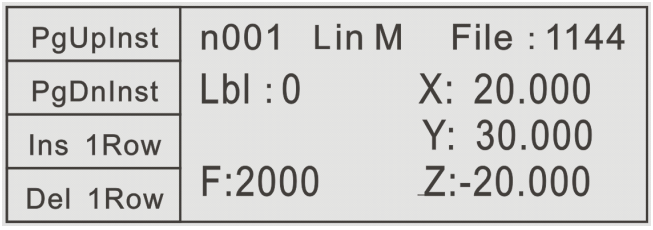
In the main interface, press the left F3 key to enter the ProgMgmt(Program Management) interface. In this interface, we can operate program to edit, read-in, delete and save.

Ed Prog	
Prog RI	
Del Prog	
SaveProg	

Ed Prog Edit program
Prog RI Program read-in
Del Prog Delete program
Sv Prog Save program

3.3.1 Ed Prog(Edit program)

In the ProgMgmt(Program Management) interface, press the left F1 key to enter the Ed Prog(Edit program) interface. In this interface, we can modify or rebuild the read-in file or the last open file before the shutdown.



PgUpInst Page up instruction
PgDnInst Page down instruction
Ins 1Row Insert a row
Del 1Row Delete a row
Lin M Linear motion
Lbl Label
Abs M Absolute motion
PTP M Point to Point motion
End End

3.3.1.1 PgUpInst(Page up instruction): In the current line number, through this key, we can page up all Instruction Names controlled and applied by the controller.

Example:

In the interface,

n001 Lin M File:1144

press PgUpInst once, then becomes

n001 Abs M File: 1144

That is, in the n001 line，modify the old Lin M(Linear motion) instruction to the Abs M （Absolute motion） instruction, and new instruction parameter shows simultaneously.

3.3.1.2 PgDnInst(Page down instruction): In the current line number, through this key, we can page down all Instruction Names controlled and applied by the controller.

Example:

In the interface,

n001 Lin M File:1144

press PgDnInst once, then becomes

n001 PTP M File: 1144

That is, in the n001 line，modify the old Lin M(Linear motion) instruction to the PTP M （Point to Point motion） instruction, and new instruction parameter shows simultaneously.

3.3.1.3 Ins 1Row(Insert a row): In the current line number, through this key, we can move wholly the all action instructions in the old line to the next line automatically. In the current line, new a line is inserted and the default is End instruction.

Example:

In the interface,

n001 Lin M File:1144

press Ins 1Row once, then the line becomes

n001 End File: 1144

press PgDn,

n002 Lin M File:1144

(The current all instructions and parameters move down the next line automatically.)

3.3.1.4 Del 1Row(Delete a row): In the current line number, through this key, we can delete the action instruction in the current line. The all instructions and parameters of the current file move up one line automatically.

Example:

In the interface,

n001 Lin M File:1144

n002 Outp File:1144

press Del 1Row once, then the line becomes

n001 Outp File: 1144

(The instruction in the old n001 line is deleted.)

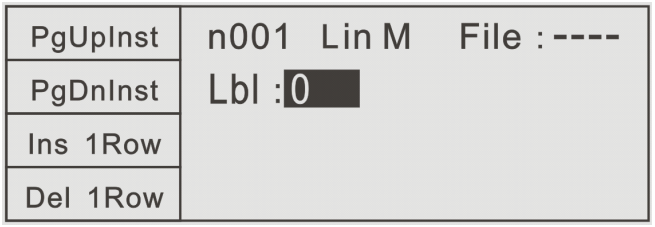
Modify program:

In the Ed Prog(Edit program)state, according to the open file, find the line number need to be modified through the PgUp or PgDn key on the controller panel, and find the required instruction through the F1 or F2 on the controller panel left side corresponding to PgUpInst(Page up instruction) or PgDnInst(Page down instruction). Then through the '↑''↓'key on the controller panel, move the cursor and modify the parameter in the cursor position. After modifying the current line, if continue to modify, once again find the line number need to be modified through the PgUp or PgDn key,and modify the program by the above method. if not modify, press Exit to return to the Sv Prog(Save program) in the ProgMgmt(Program Management) interface, and save the file according to the prompt of Sv Prog(Save program).

Newly build program:

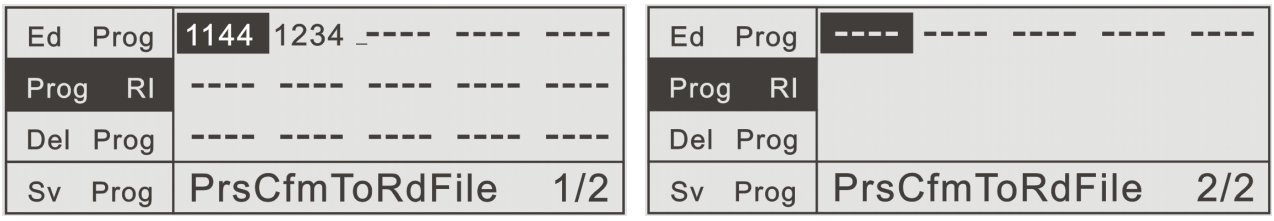
In the Ed Prog(Edit program)state, after the F key on the controller panel is pressed for three seconds, the interface will clear the previous open file and re-create a new file. At this time, the program current line is n001, the instruction is End, and

the file name is undefined.



3.3.2 Prog RI(Program read-in)

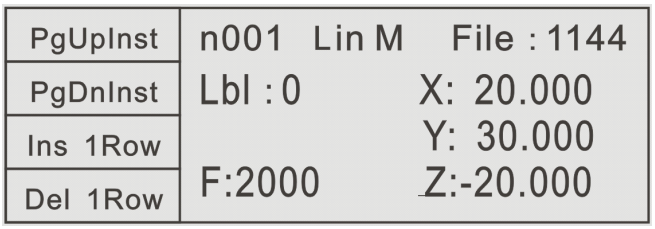
In the ProgMgmt(Program Management) interface, press the left F2 key to enter the Prog RI(Program read-in) interface. In this interface, we can read-in the saved file.



Del Prog Delete program

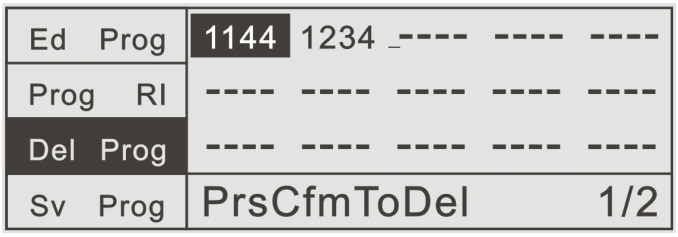
PrsCfmToRIFile Press the Confirm key to read-in the file

Move the cursor to choose the saved file through‘←’‘→’‘↑’‘↓’, and press the Cfm(Confirm) key to read-in the chosen file according to the prompt. After read-in, automatically enter the Ed Prog(Edit Program) interface, and operate the file according to demand.



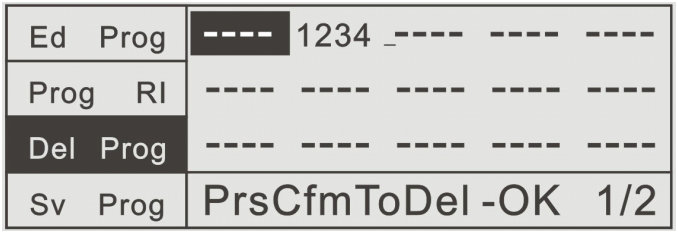
3.3.3 **Del Prog(Delete program)**

In the ProgMgmt(Program Management) interface, press the left F3 key to enter the Del Prog(Delete program) interface. In this interface, we can delete the saved file.



PrsCfmToDel Press the Confirm key to delete the file

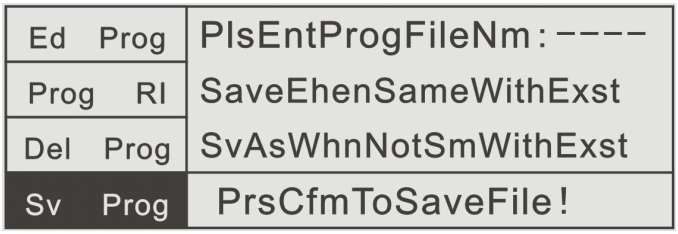
Move the cursor to choose the saved file through‘←’‘→’‘↑’‘↓’, and press the Cfm(Confirm) key to delete the chosen file according to the prompt.



PrsCfmToDel-OK Press the Confirm key to delete the file – Success

3.3.4 Sv Prog(Save program)

In the ProgMgmt(Program Management) interface, press the left F4 key to enter the Sv Prog(Save program) interface. In this interface, we can save or save as the edited file.



PlsEntProgFileNm Please enter a program file name:
SaveWhenSameWithExst Save the file when the same name with the existing file
SvAsWhnNotSmWithExst Save as the file when not the same name with the existing file
PrsCfmToSaveFile Press the Confirm key to save the file

Input the file name through the number keys 0-9, and press the Cfm(Confirm) key to save or save as the chosen file according to the prompt.

Ed Prog	PlsEntProgFileNm: 1144
Prog RI	SaveWhenSameWithExst
Del Prog	SvAsWhnNotSmWithExst
Sv Prog	SvAs Prog OK!

Ed Prog	PlsEntProgFileNm: 1234
Prog RI	SaveWhenSameWithExst
Del Prog	SvAsWhnNotSmWithExst
Sv Prog	Save Prog OK!

SvAs Prog OK Save as the program successfully
Save Prog OK Save the program successfully

3.4 Pars Set(Parameters settings)

In the main interface, press the left F4 key to enter the Pars Set(Parameters settings) interface. In this interface, we can edit or modify the controller parameters.

Sys Pars	
Sys SC	
IO Set	
UsrMgmt	

Sys Pars System parameters
Sys SC System self-check
IO Set IO settings
Usr Mgmt User management

Sys Pars(System parameters): the Control parameters and Speed parameters can be edited or modified.

Sys SC(System self-check): the Actual input, Setting input, Actual output, Program output state can be checked.

IO Set(IO settings): the external input or output ports can be defined or modified, and the factory values can be resumed.

Usr Mgmt(User management): We must enter the Usr Mgmt(User management) interface to edit or modify all the parameters. Input the password to login.

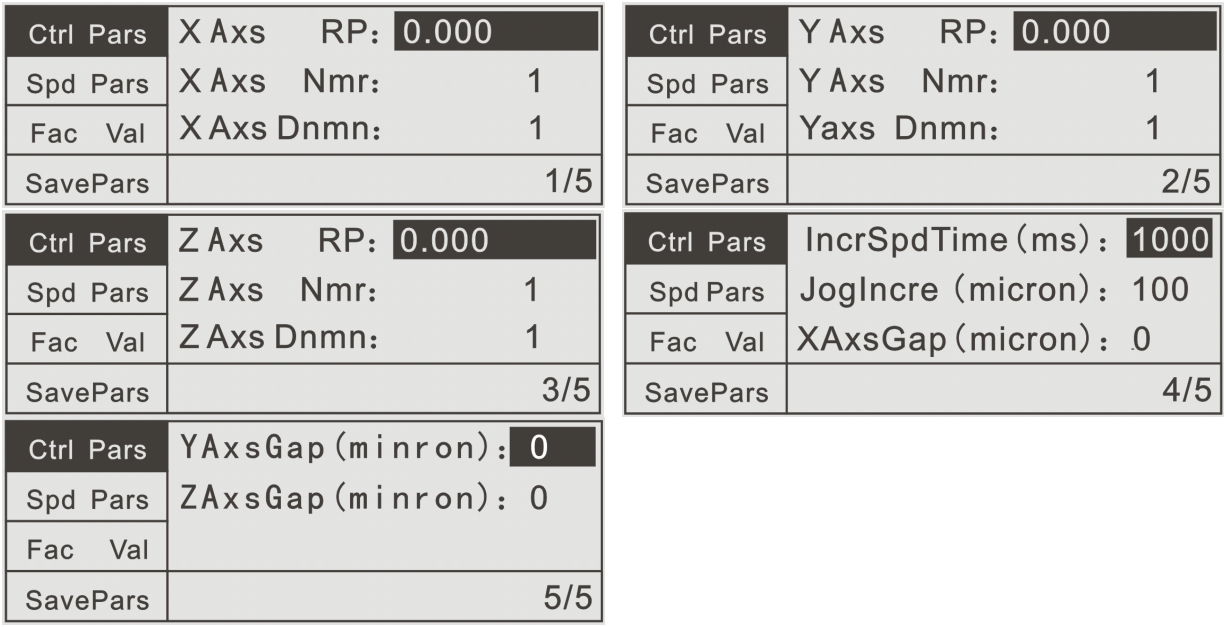
3.4.1 Sys Pars(System parameters)

In the Pars Set (Parameters settings) interface, press the left F1 key to enter the Sys Pars(System parameters) interface. In this interface, we can edit or modify the

axis reference points, Electronic gear, Increase speed time, Jog increment, the axis gaps, a lot of speed parameters of the axises.

3.4.1.1 CtrlPars(Control parameters)

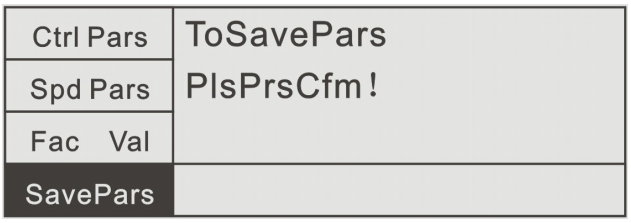
When enter the Sys Pars(System parameters) interface, the default is CtrlPars(Control parameters). Or in the Sys Pars(System parameters) interface, press the left F1 key to enter the CtrlPars(Control parameters) interface.

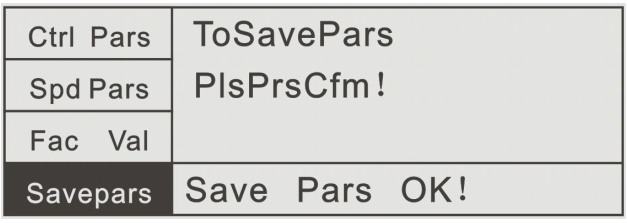


CtrlPars Control parameters
Spd Pars Speed parameters
FacVal Factory value
SavePars Save parameters
XAxs RP X-axis reference point
XAxsNmr X-axis numerator
XAxsDnmn X-axis denominator
IncrSpdTime(ms) Increase speed time (ms)
JogIncre(micron) Jog increment (micron)
XAxsGap(micron) X-axis gap (micron)

Through the‘↑’‘↓’‘PgUp’‘PgDn’keys on the controller panel, move the cursor to find the parameter need to be modified， and edit or modify the parameter. Through pressing the left F4 key corresponding to SavePars(Save parameters) and then pressing the Cfm(Confirm) key on the controller panel according to the prompt, save

the modified data.

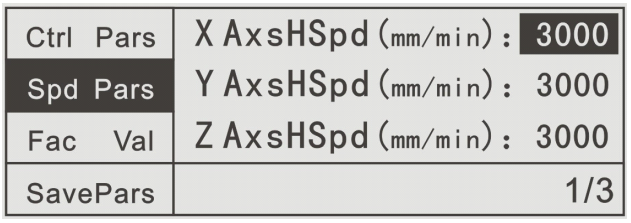


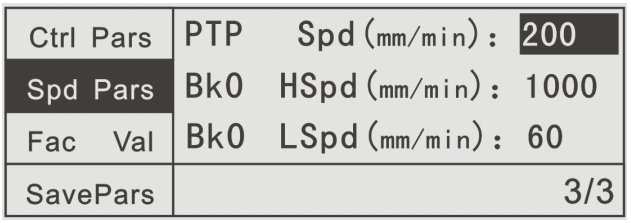


ToSavePars Press save to save the parameters
PlsPrsCfm Press the Confirm key to save the parameters
Save Pars OK Save the parameters successfully

3.4.1.2 Spd Pars(Speed parameters)

In the Sys Pars(System parameters) interface, press the left F2 key to enter the Spd Pars(Speed parameters) interface. In this interface, we can edit or modify the axis maximum speeds, start speed, manual high speed, manual low speed, point to point speed, back to zero high speed, back to zero low speed.

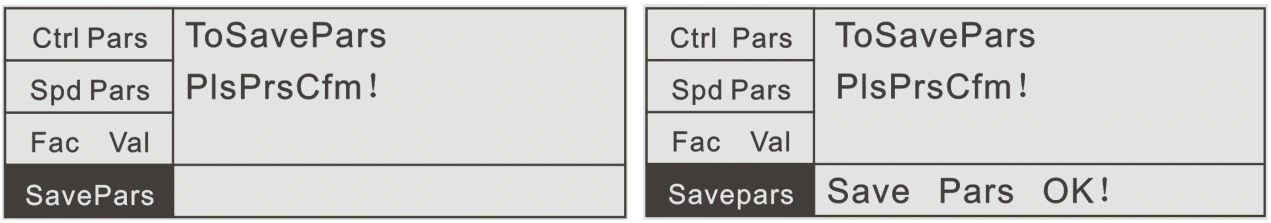




X-axis high speed XAxsHSpd
Start high speed St HSpd
Manual high speed Man HSpd
Manual low speed Man LSpd
Point to Point speed PTP Spd
Back to zero high speed Bk0 HSpd
Back to zero low speed Bk0 LSpd

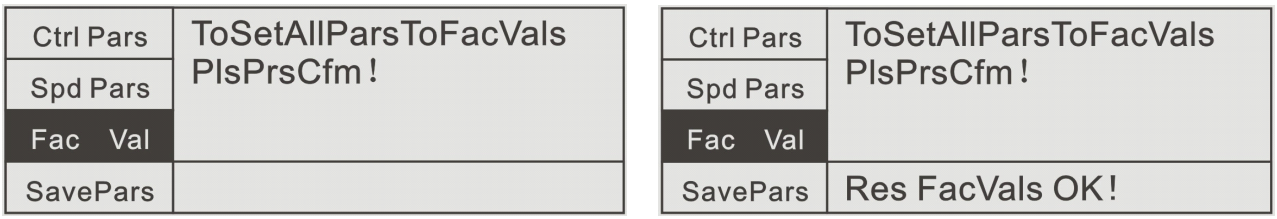
Through the‘↑’‘↓’‘PgUp’‘PgDn’keys on the controller panel, move the cursor to find the parameter need to be modified， and edit or modify the parameter. Through

pressing the left F4 key corresponding to SavePars(Save parameters) and then pressing the Cfm(Confirm) key on the controller panel according to the prompt, save the modified data.



3.4.1.3 FacVal(Factory values)

In the Sys Pars(System parameters) interface, press the left F3 key to enter the FacVal(Factory values) interface. In this interface, we can resume all the parameters to the Factory values, and according to the interface prompt, press the Cfm(Confirm) key on the controller panel to confirm to resume the output values.



ToSetAllParsToFacVal To set all parameters to factory values
PlsPrsCfm Press the Confirm key to set all parameters to factory default values
ResFacValsOK Resume factory values successfully

3.4.2 Sys SC(System self-check)

Actl Inp: Actual input. Through the on/off test of the external corresponding input ports switches, detect whether or not the input ports are normal and one-to-one.

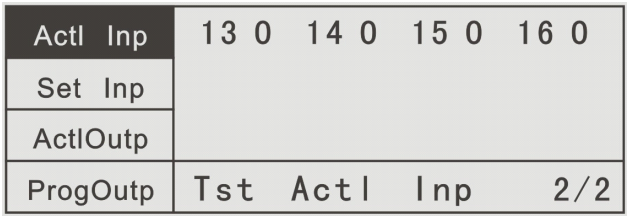
Set Inp: Setting input. Through the on/off test of the external input ports switches,

detect whether or not the set input ports functions are normal and accurate corresponding.

ActlOutp: Actual output. Through the controller manual operation to test the output ports relays on/off, detect whether or not the output ports are normal and one-to-one.

ProgOutp: Program output. Through the controller manual operation to test the set output ports relays on/off, detect whether or not the output ports are normal and accurate corresponding.

3.4.2.1 In the Pars Set(Parameters settings) interface, press the left F2 key to enter the Sys SC (System self-check) interface. The controller interface default is Actl Inp(Actual input).



Through controlling the external input ports on/off, detect whether or not the input ports are valid and one-to-one.

Example: switch on the external input 1, the controller interface input port 1 should change from O(Off) to I(On).

Actl Inp Actual input
Set Inp Setting input
ActlOutp Actual output
ProgOutp Program output
O off
I on
Tst Actl Inp Test actual input

Note: when there is no change, may be as follows

24V power supply is not working properly

The input signal line connection is not normal

3.4.2.2 Set Inp(Setting input)

In the Sys SC (System self-check) interface, press the left F2 key to enter the Set Inp(Setting input) interface.

Actl Inp	X+Lim:0	X-Lim:0
Set Inp	Y+Lim:0	Y-Lim:0
ActlOutp	Z+Lim:0	Z-Lim:0
ProgOutp	Tst Set Inp	1/3

Actl Inp	Alm :0	Stop :0
Set Inp	Xzero:0	Yzero:0
ActlOutp	Zzero:0	St :0
ProgOutp	Tst Set Inp	2/3

Actl Inp	P :0	IcSp:0
Set Inp	RdSp:0	
ActlOutp		
ProgOutp	Tst Set Inp	3/3

X+Lim X positive limitation
X-Lim X negative limitation
Tst Set Inp Test setting input
Alm Alarm
Stop Emergency stop
XZero X zero
St Start
Paus Pause
IcSp Increase speed
RdSp Reduce speed

Through controlling the external input ports on/off, detect whether or not the set input ports are valid and accurate corresponding.

Example: In the IO Set(IO settings) area of the Pars Set(Parameters settings) area,

ExtStart(External start) is set N.O.(Normally open) and input port is set 3. Through switching on the external input port 3, the St(Start) status should change from O(Off) to I(On) on the second page of the controller interface.

Actl Inp	Alm :0	Stop :0
Set Inp	Xzero:0	Yzero:0
ActlOutp	Zzero:0	St :0
ProgOutp	Tst Set Inp	2/3

Actl Inp	Alm :0	Stop :0
Set Inp	Xzero:0	Yzero:0
ActlOutp	Zzero:0	St :I
ProgOutp	Tst Set Inp	2/3

Note: when there is no change, may be as follows

The input point setting is wrong

The input signal is not normal

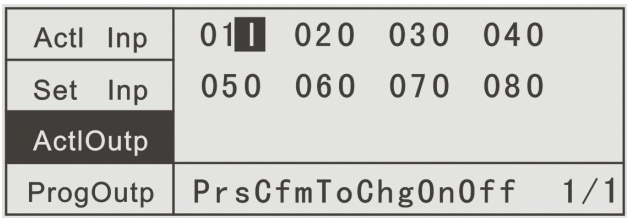
3.4.2.3 ActlOutp(Actual output)

In the Sys SC (System self-check) interface, press the left F3 key to enter the ActlOutp(Actual output) interface.

Actl Inp	01**0**	020	030	040
Set Inp	050	060	070	080
ActlOutp				
ProgOutp	PrsCfmToChgOnOff			1/1

PrsCfmToChgOnOff Press the Confirm key to change On or Off

Through pressing the ←, →, ↑, ↓ cursor keys to change the selected actual output point, move the cursor. Press the Cfm(Confirm) key on the controller panel, the corresponding state changes from O(Off) to I(On), or from I(On) to O(Off). At the same time the corresponding output will change from open to closed, or from closed to open.



Note: when there is no change, may be as follows

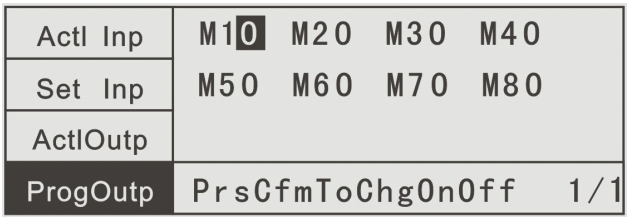
24V power supply is not working properly

The output signal line connection is not normal

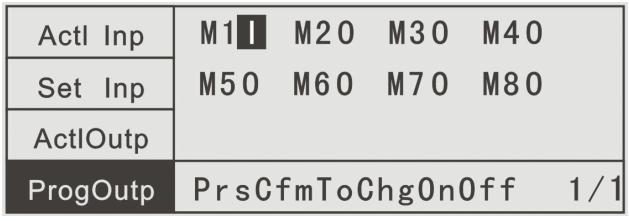
The corresponding relay is not working properly

3.4.2.4 ProgOutp(Program output)

In the Sys SC (System self-check) interface, press the left F4 key to enter the ProgOutp(Program output) interface.



Through pressing the ←, →, ↑, ↓ cursor keys to change the selected program output point, move the cursor. Press the Cfm(Confirm) key, the corresponding state changes from O(Off) to I(On), or from I(On) to O(Off). At the same time the corresponding output will change from open to closed, or from closed to open.



Note: when there is no change, may be as follows

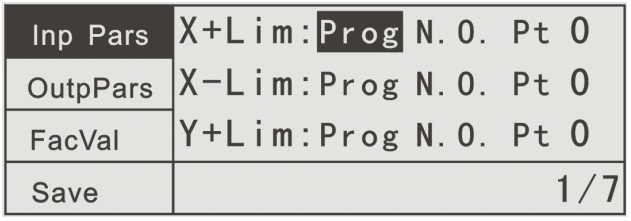
The program output point setting is wrong

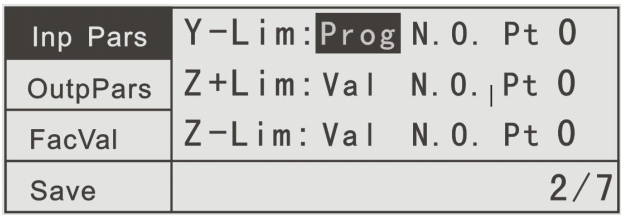
The output signal is not normal

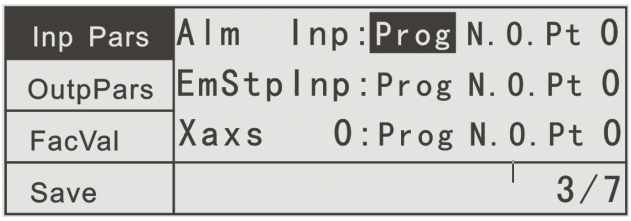
3.4.3 IO Set(IO settings)

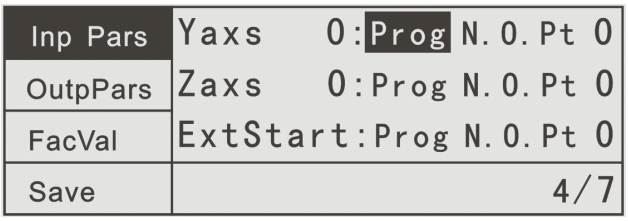
In the Pars Set(Parameters settings) interface, press the left F3 key to enter the IO Set(IO settings) interface. The system default is in the InpPars (Input parameters) setting interface.

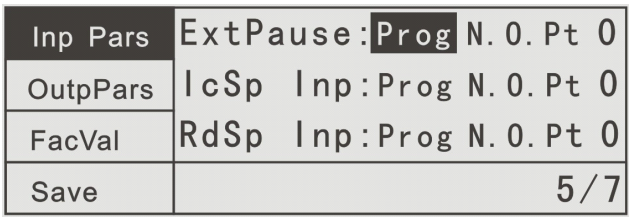
3.4.3.1 InpPars (Input parameters)

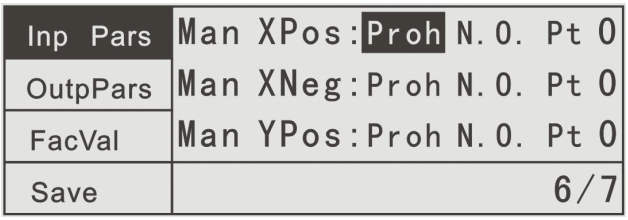


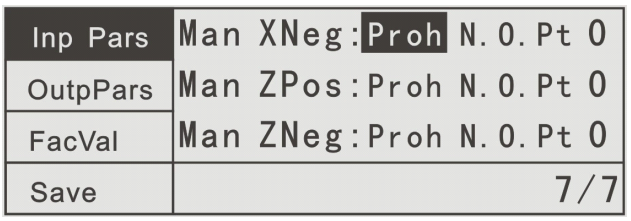












InpPars Input parameters
OutpPars Output parameters
Sv Save
X+Lim X-axis positive limitation
X-Lim X-axis negative limitation
Proh Prohibited

Val valid
N.O.Normally open
N.C. Normally closed
“Pt” Port
InpPt input port
Alm Inp Alarm Input
EmStpInp Emergency stop input
XAxs0X axis zero
ExtStart External start
ExtPause External Pause
IcSp Inp Increase speed input
RdSp Inp Reduce speed input
Man XPos Manual X positive direction
Man XNeg Manual X negative direction

Through setting the parameters, determine the specific definition of the external input ports. Through pressing the ←, →, ↑, ↓ cursor keys, move the cursor. Through pressing the PgUp, PgDn keys, turn the page. Through the Cfm(Confirm) key on the controller panel, switch Proh(Prohibited) and Vld (Valid), or N.O.(Normally open) and N.C.(Normally closed). Through the number keys 0-9 on the controller panel, modify the parameter of the Pt(Port, short for InpPt(Input port)).

Proh(Prohibited): This function is not used.

Vld (Valid): This function is valid.

N.O.(Normally open): The selected switch status corresponding to the input port is normally open switch.

N.C.(Normally closed): The selected switch status corresponding to the input port is normally closed switch.

Pt 0: Port 0. Corresponding to the input port, the following number ‘0’ indicates that the external input line is No. 0 line. Because No. 0 line does not exist, that is the input port is not set. When need to set, No. 1-16 line can be selected to fill the setting.

Example: The external Man XPos(Manual X positive direction rotation) is set Vld(Valid). The input line No.6 is selected to connect the switch of Man XPos(Manual X positive direction rotation). The switch is N.O.(Normally open) switch.

Inp Pars	Man XPos:Proh N.O. Pt 0
OutpPars	Man XNeg:Proh N.O. Pt 0
FacVal	Man YPos:Proh N.O. Pt 0
Save	6/7

Before modification

Inp Pars	Man XPos:Val N.O. Pt 6
OutpPars	Man XNeg:Proh N.O. Pt 0
FacVal	Man YPos:Proh N.O. Pt 0
Save	6/7

After modification

3.4.3.2 OutpPars(Output parameters)

In the IO Set(IO settings) interface, press the left F2 key to enter the OutpPars(Output parameters) interface.

Inp Pars	ProgOutPt1: 1
OutpPars	ProgOutPt2: 2
FacVal	ProgOutPt3: 3
Save	1/3

Inp Pars	ProgOutPt4: 4
OutpPars	ProgOutPt5: 5
FacVal	ProgOutPt6: 6
Save	2/3

Inp Pars	ProgOutPt7: 7
OutpPars	ProgOutPt8: 8
FacVal	
Save	3/3

ProgOutpPt Program output port

ProgOutpPt(Program output port) indicates the filled output port in the process of programming. The following number '1' '2' etc. indicates the corresponding external output port number, the controller default is 1 to 1. That is the applied output instruction in the process of programming. The entered number of the output port indicates the controlled number of the external output line. The purpose that it can be

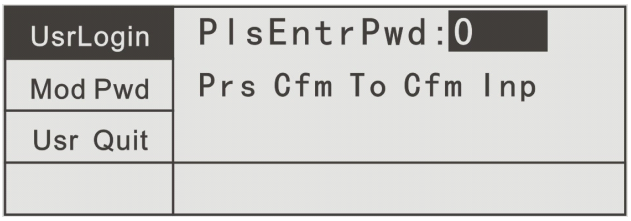
defined is when the external output line changes, the program can not be modified, just the parameter is modified simply.

Note: The FacVal(Factory value), Save functions and application methods are the same as the above (Sys Pars(System parameters)).

3.4.4 Usr Mgmt(User management)

In the Pars Set(Parameters settings) interface, press the left F4 key to enter the Usr Mgmt(User management) interface. The system default is in the UsrLogin (User login) state.

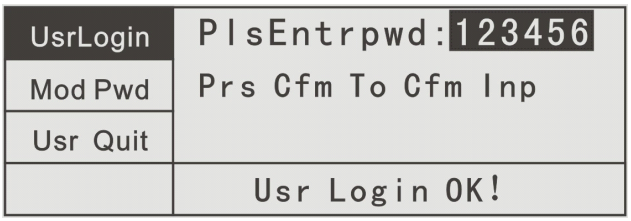
3.4.4.1 UsrLogin (User login)

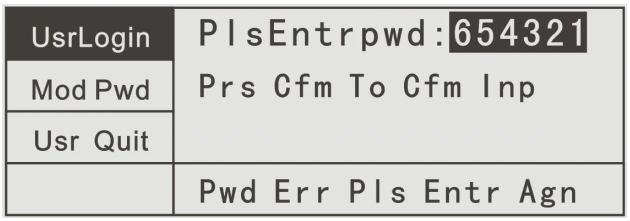


UsrLogin User login
Mod Pwd Modify password
Usr Quit User quit
PlsEntrPwd Please enter the password
PrsCfmToCfmInp Press the Confirm key to confirm the input

After entering the user password and pressing the Cfm(Confirm) key, if enter correctly, the prompt is Usr Login OK(User login successfully), if not, the prompt is Pwd Err Pls Entr Agn(Password mistake, please enter again). The factory value of the user password is 123456. After the user logs in successfully, the processing parameters can be modified and saved. Otherwise the processing parameters can not be modified and saved.

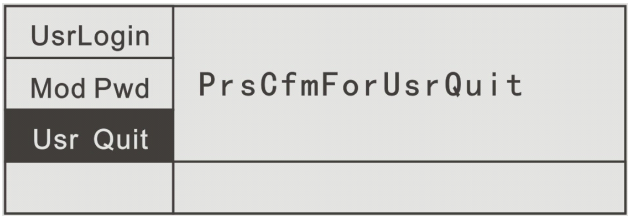
Example: If need to modify the parameters, firstly, the user must log in. The password is 123456.

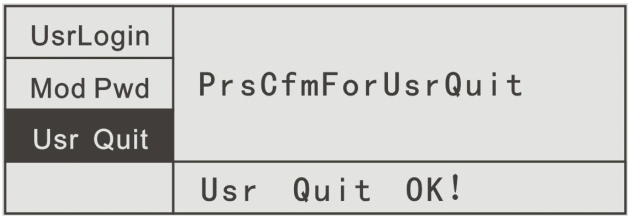




Usr Login OK　User login successfully
Pwd Err Pls Entr Agn Password mistake, please enter again

If the login is successful and pressing directly the Exit key on the controller panel, then the Pars Set(Parameters settings) area can be edited. After the editing is completed, you can re-enter the Usr Mgmt(User management), and select the Usr Quit(User quit). After pressing the Cfm(Confirm) key, the current all contents of Pars Set(Parameters settings) can not be changed. Also you may not choose to re-enter, after power down, the status is 'User not logged in' automatically. If need to modify, enter again the Usr Mgmt(User management) to login.





PrsCfmForUsrQuit Press the confirm key for user's quit
Usr Quit OK　User quit successfully

3.4.4.2 Mod Pwd(Modify password)

In the Usr Mgmt(User management) interface, press the left F2 key to enter the Mod Pwd(Modify password) interface.

Through this setting, the old password of Usr Mgmt(User management) can be modified and set to the password grasped by the user.

Note: If want to modify the parameters in the Pars Set(Parameters settings) area, you need to login Usr Mgmt(User management), and then you can modify and save. Or

you can not modify the Pars Set(Parameters settings) area.

4. Programming

The controller detailed instructions and cases

In a new file, the default instruction is End in the all n001-n480 lines.

Through pressing the left side F1(PgUpInst) or F2(PgDnInst) key, find the instruction need to be applied. In the instruction staying interface, the parameters show automatically. Move the cursor and modify the parameters through the keys on the controller panel. After finishing the current line and pressing the PgDn key on the controller panel, then continue to write the next line instruction. If not writing, directly exit to save the program, or return to the main interface to enter the AutoExec(Automatically execute) interface, and demonstrate the written program, if no problem, enter the ProgMgmt(Program Management) to save the program. (The premise is that the power supply is not interrupted, or after not saving and entering the AutoExec interface to demonstrate, the last modified program will be not saved if the power supply is interrupted.)

The controller instructions

Note: All data units are subject to the selected units of the electronic gear calculation formula. The parameter Lbl(Label) inside the instruction names will be detailed separately.

End: When this line is executed, the controller action will be finished, and no longer continue.

1）PTP M(Point to Point motion): Run with the system most high speed in relative

coordinates.

Example:

n003 PTP M File: 3399

Lbl: 0 X： 20.000

Y： -34.125

Z： 12.320

Meaning: After completing the n002 line execution, then execute this line n003. X/Y/Z will run with the most high speed set in the Sys Pars(System parameters). The speed is the resultant speed. The motion action is the linear interpolation (start at the same time, stop at the same time). Subject to the previous position, the X axis will move 20.000 to the positive direction, the Y axis will move 34.125 to the negative direction, the Z axis will move 12.320 to the positive direction. After completing the execution, automatically execute the n004 line action.

2）Lin M(Linear motion): Run with the given F value speed of this line, in relative coordinates. The Start speed≤the F value≤the axis high speed. When the input F value is less than or equal to the start speed, the running speed is the start speed value. When the input F value is greater than or equal to the axis high speed, the running speed is the axis high speed value. The start speed(St HSpd(Start high speed)) and the axis high speed are set in the Pars Set(Parameters settings)→Sys Pars(System parameters)→Spd Pars(Speed parameters).

Example:

n006 Lin M File: 3399

Lbl: 0 X: 10.000

Y: 0

F: 660 Z: -16.880

Meaning: After completing the n005 line execution, then execute this line n006. X/Y/Z will run with the F value 660 speed. The speed is the resultant speed. The motion action is the linear interpolation (start at the same time, stop at the same time). Subject to the previous position, the X axis will move 10.000 to the positive direction, the Y axis will not move, the Z axis will move 16.880 to the negative direction. After completing the execution, automatically execute the n007 line action.

3）Abs M(Absolute motion): Run with the given F value speed of this line, in absolute coordinates. The Start speed≤the F value≤the axis high speed. When the input F value is less than or equal to the start speed, the running speed is the start speed value. When the input F value is greater than or equal to the axis high speed, the running speed is the axis high speed value. The start speed(St HSpd(Start high speed)) and the axis high speed are set in the Pars Set(Parameters settings)→Sys Pars(System parameters)→Spd Pars(Speed parameters).

Example:

n009 Abs M File: 3399

Lbl: 0 X: 30.000

Y: ---------

F: 880 Z: -16.880

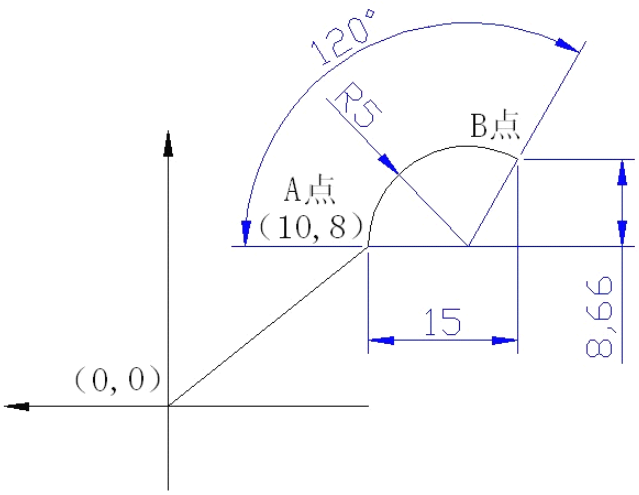
Meaning: After completing the n008 line execution, then execute this line n009.

X/Y/Z will run with the F value 880 speed. The speed is the resultant speed. The motion action is the linear interpolation (start at the same time, stop at the same time). Subject to the last powered or set coordinates, the X axis will move to the coordinate 30.000 position, the Y axis will not move, the Z axis will move to the coordinate -16.880 position. After completing the execution, automatically execute the n010 line action.

Note: Moving in absolute coordinates, if don't want to make the current an axis movement, that is, only make the other axis movement and maintain the position of the axis not to change, change the value to '--------' through the Cfm(Confirm) key on the controller panel.

4 ） CirIntrp(Circular Interpolation): The X, Y axises, in the interpolation way, clockwise fit the arc curve through the set radius R and the resultant normal velocity value. The parameters X, Y are the coordinates of the circular arc end relative to the starting point. Through the positive or negative value of R, the drawn curve can be determined to the inferior arc(less than 180°) or the major arc (greater than 180°).

Example 1: The controller is powered, quickly move to the X10, Y8 position of the point A, and then Z slowly moves down -6, and then clockwise draw the circular arc to the point B, the radius is 5, the circular arc is 1/3 of the full circle, and then Z moves up 6, then X,Y return back to program zero.



Programming is as follows:

n001 Lin M File：7788

Lbl：0 X： 10.000

Y： 8.000

F：1000 Z： 0.000

n002 Lin M File：7788

Lbl：0 X： 0.000

Y： 0.000

F：100 Z： -6.000

n003 CirIntrp File：7788

Lbl：0

X： 15.000 Y: 8.660

R： 5 F：100

n004 Abs M File：7788

Lbl：0 X： --------

Y： --------

F：100　　Z：　0.000

n005　Abs M　File：7788

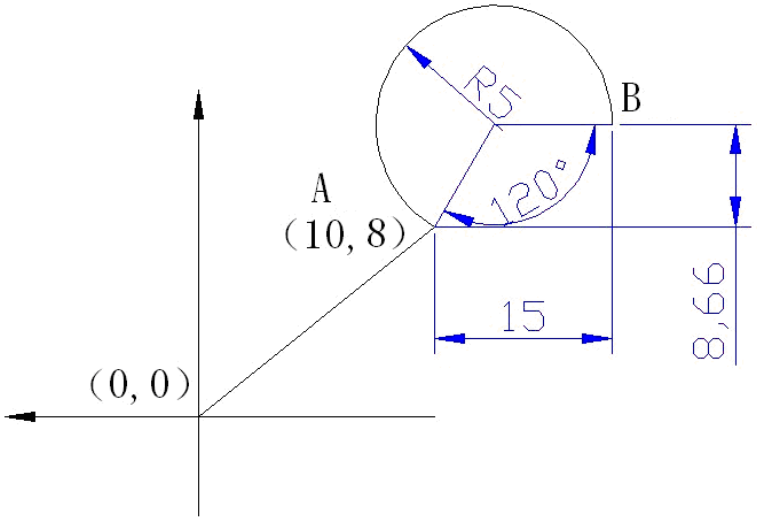
Lbl：0　　X：　0.000

Y：　0.000

F：1000　　Z：　--------

Example 2: The controller is powered, quickly move to the X10, Y8 position of the point A, and then Z slowly moves down -6, and then clockwise draw the circular arc to the point B, the radius is 5, the circular arc is 2/3 of the full circle, and then Z moves up 6, then X,Y return back to program zero.

Programming is as follows:



n001　Lin M　File：7788

Lbl：0　　X：　10.000

Y：　8.000

F：1000　　Z：　0.000

n002　Lin M　File：7788

Lbl：0　　X：　0.000

Y： 0.000

F：100 Z： -6.000

n003 CirIntrp File：7788

Lbl：0

X： 15.000 Y: 8.660

R： -5 F：100

n004 Abs M File：7788

Lbl：0 X： --------

Y： --------

F：100 Z： 0.000

n005 Abs M File：7788

Lbl：0 X： 0.000

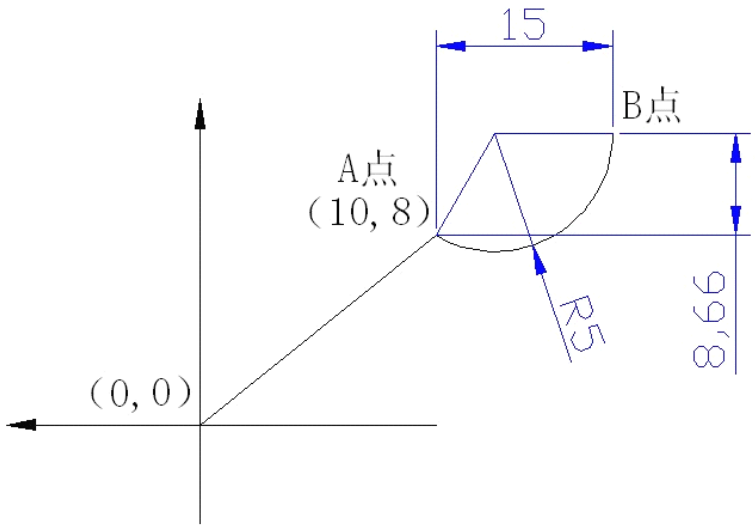
Y： 0.000

F：1000 Z： --------

(The only difference of the two program segment is that the radius is positive or negative.)

5）IvCrItrp(Inverse circular interpolation): The X, Y axises, in the interpolation way, counterclockwise fit the arc curve through the set radius R and the resultant normal velocity value. The parameters X, Y are the coordinates of the circular arc end relative to the starting point. Through the positive or negative value of R, the drawn curve can be determined to the inferior arc(less than 180°) or the major arc (greater than 180°).

Example 1: The controller is powered, quickly move to the X10, Y8 position of the point A, and then Z slowly moves down -6, and then counterclockwise draw the circular arc to the point B, the radius is 5, the circular arc is 1/3 of the full circle, and then Z moves up 6, then X,Y return back to program zero.



Programming is as follows:

n001 Lin M File：7788

Lbl：0 X： 10.000

Y： 8.000

F：1000 Z： 0.000

n002 Lin M File：7788

Lbl：0 X： 0.000

Y： 0.000

F：100 Z： -6.000

n003 IvCrItrp File：7788

Lbl：0

X： 15.000 Y: 8.660

R： 5 F：100

n004 Abs M File：7788

Lbl：0 X： --------

Y： --------

F：100 Z： 0.000

n005 Abs M File：7788

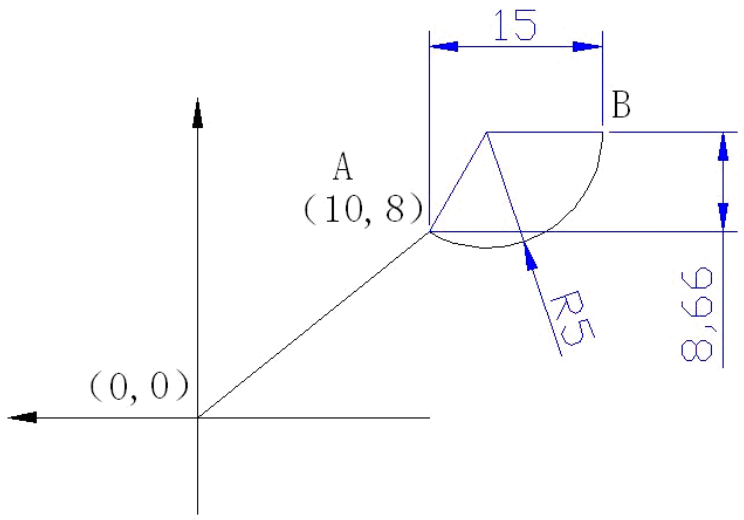
Lbl：0 X： 0.000

Y： 0.000

F：1000 Z： --------

Example 2: The controller is powered, quickly move to the X10, Y8 position of the point A, and then Z slowly moves down -6, and then counterclockwise draw the circular arc to the point B, the radius is 5, the circular arc is 2/3 of the full circle, and then Z moves up 6, then X,Y return back to program zero.

Programming is as follows:



n001 Lin M File：7788

Lbl：0 X： 10.000

Y： 8.000

F：1000 Z： 0.000

n002 Lin M File：7788

Lbl：0 X： 0.000

Y： 0.000

F：100 Z： -6.000

n003 IvCrItrp File：7788

Lbl：0

X： 15.000 Y: 8.660

R： -5 F：100

n004 Abs M File：7788

Lbl：0 X： --------

Y： --------

F：100 Z： 0.000

n005 Abs M File：7788

Lbl：0 X： 0.000

Y： 0.000

F：1000 Z： --------

(The only difference of the two program segment is that the radius is positive or negative.)

Note: The circular arc instructions are limited to TC5522R and TC5530R proprietary

instructions.

6） SetCoord(Set coordinates): Set the current position coordinates. When executing this line instruction, the controller will be subject to the current set new coordinates, the interface coordinates display will also be subject to the new coordinates.

Example: The power is turned on, with the 660 mm/min speed, the program moves the absolute coordinate three positions to 20.000 mm, and then set the current point to the program running coordinate 0,0,0 point, and then all run 25 mm, then set the current point to the program running coordinate 0,0,0 point.

n001 Abs M File：7788

Lbl：0 X： 20.000

Y： 20.000

F：660 Z： 20.000

（At this time, the controller interface coordinates display is X:20.000 Y:20.000 Z:20.000）

n002 SetCoord File：7788

Lbl：0 X： 0.000

Y： 0.000

Z： 0.000

（At this time, the controller interface coordinates display is X: 0.000 Y: 0.000 Z: 0.000）

n003 Abs M File：7788

Lbl：0 X： 25.000

Y： 25.000

F：660 Z： 25.000

（At this time, the controller interface coordinates display is X: 25.000 Y: 25.000 Z: 25.000）

n004 SetCoord File：7788

Lbl：0 X： 0.000

Y： 0.000

Z： 0.000

（At this time, the controller interface coordinates display is X: 0.000 Y: 0.000 Z: 0.000）

Note: After setting the current coordinates, all subsequent running instructions are subject to the last set coordinates position. The main associated instruction is the Abs M(Absolute motion) instruction in absolute coordinates, has nothing to do with the Lin M(Linear motion) and PTP M(Point to Point motion) instruction in relative coordinates.

7）Delay: Delay time. Link the previous line instruction, and then execute this Delay instruction, then link the next line instruction. Make the time extension for executing the next line instruction.

Example: X moves 10 mm to the positive direction, Y moves 15 mm to the negative direction, Z axis does not move, the resultant speed is 1000, and then delay 5.25 seconds, then the three axises return back to program zero.

n001 Lim M File：7788

Lbl：0 X： 10.000

Y：-15.000

F：1000 Z： 0.000

n002 Delay File：7788

Lbl：0

Dla Time： 5.25

n003 Abs M File：7788

Lbl：0 X： 0.000

Y： 0.000

F：1000 Z： 0.000

(Dla Time: Delay time)

8） Abs Jump(Absolute jump): When executing this line, jump to the place which Lbl(Label) is the same with the Dest Lbl(Destination label) to start to continue, equivalent to the infinite time (unlimited) loop instruction.

Example: The three axises are at the starting point coordinates 0,0,0. The first step X moves 10mm, Y moves 10mm. The second step Z moves -5 mm. The third step, delay 1 second. The fourth step Z return back to coordinate zero. The fifth step, start again from the first line to make the infinite loop of the first four steps.

n001 Lim M File：7788

Lbl：23 X： 10.000

Y： 10.000

F：800 Z： 0.000

（Because at last start from this line to make the infinite loop, in the Lbl(Label) position of the starting line at the beginning of the loop, take a numerical marking name, and in the program file, the Lbl(Label) of each line instruction can not repeat, except 0, that is not defined.）

n002 Lim M File：7788

Lbl：0 X： 0.000

Y： 0.000

Z： -5.000

n003 Delay File：7788

Lbl：0

Dla Time： 1

n004 Abs M File：7788

Lbl：0 X： ---------

Y： ---------

Z： 0.000

n005 Abs Jump File：7788

Lbl：0

Dest Lbl：23

（As long as executing this line instruction, then automatically jump to the line which program line Lbl(Label) is 23, to continue to start executing the program.）

(Dest Lbl:Destination label)

9）CondJump(Conditional jump): This line instruction and the external input port

must be applied at the same time. Detect the selected input port state(I(On) or O(Off)), if the set condition state of the instruction meets with the state of the external set input port, jump to the place which Lbl(Label) is the same as the current line Dest Lbl(Destination label), to start to continue, if does not meet with the external input port state, automatically execute the next line.

Example1:The first step，X moves 10mm，then detect the signal state of the external input port 11，if closed，then execute the third line，Y moves 20，Z moves -15，if open，then X axis continues to move 10mm，then detect again the signal state of the external input port 11，if closed，then execute the third line，if open，execute the first line instruction in this loop.

n001　Lin M　File：7788

Lbl：39　　X：10.000

Y：0.000

F：800　　Z：0.000

（The Label name is 39）

n002　CondJump　File：7788

Lbl：0

InpPt：11　　Cond：O

Dest Lbl：39

（Execute this line，detect the state of the external input port 11， if the input port 11 is externally connected, then does not meet with the set 'Cond: O' of the line, then continues directly to execute the next line, if met, that is the input port 11 is still

in the off state, then continues to loop the line instruction labeled 39.）

(InpPt: Input port, Cond: Condition, O: Off, Dest Lbl:Destination label)

n003 Lin M File：7788

Lbl：0 X： 0.000

Y： 20.000

F：800 Z： -15.000

Example 2: The program runs, waiting for the signal of the external input port 8, the input port closes once, while the three axises move 10mm at the same time, executes a total of 10 times.

n001 CondJump File：7788

Lbl：777

InpPt： 8 Cond：O

Dest Lbl：777

（Detect the state of the external input port 8, the switch is not pressed down, at this time it is off, and meets with the line instruction 'Cond: O', the program will jump to the Lbl: 777 line to continue to execute, because this line label is also 777, so long as the external switch is not pressed down, that is not closed, the program will always execute the line, rather than downwards, at this time no any external program action is equivalent to 'Pause', when the switch is pressed down, that is the switch is closed, does not meet with the line set 'Cond: O', then continues to execute the next line instruction.）

n002 Lin M File：7788

Lbl：0　　X：10.000

Y：10.000

F：800　　Z：10.000

n003　Loop　File：7788

Lbl：0

NumOfLps：9

Dest Lbl：777

（Execute this line, the program will automatically jump to the Lbl: 777 line instruction to continue to execute, the execution number of times is 9 times, accumulative total of 10 times, so when to execute a loop N times, the number of times of the loop instruction should be written as N-1 times.）

（NumOfLps: Number of loops）

10）Loop: Execute this line, the program will automatically jump to the line instruction which label meets with the set Dest Lbl to continue to execute, and execute N-1 times.

The case as above.

11）Output: Execute this line, according to the set parameters of the line, operate the external output port, the output port can be connected with the standard 24V relay, through controlling the relay coil on/off, control the contacts suck & shut, and then control the external devices connected with the relay N.O.(Normally open) or N.C.(normally closed), most of the cylinder, or the spindle motor, or the buzzer.

Example: The three axises move to position 10, and then open the output port 2, after

delaying 3 seconds, turn off the output port, the three axises move 20 again from the current point, open the output port 1 and 2, delay 1 second, turn off the output port 1, and then delay 2 seconds, turn off the output port 2, and then the three axises position return back to coordinate zero.

n001 Lin M File：7788

Lbl：0 X： 10.000

Y： 10.000

F：800 Z： 10.000

n002 Output File：7788

Lbl：0

OutpPt：2 Stat：I

n003 Delay File：7788

Lbl：0

Dla Time： 3

n004 Output File：7788

Lbl：0

OutpPt：2 Stat：O

n005 Lin M File：7788

Lbl：0 X： 20.000

Y： 20.000

F：800 Z： 20.000

n006 Output File：7788

Lbl：0

OutpPt：1　　Stat：I

n007　Output　　File：7788

Lbl：0

OutpPt：2　　Stat：I

n008　Delay　　File：7788

Lbl：0

Dla Time：　1

n009　Output　　File：7788

Lbl：0

OutpPt：1　　Stat：O

n010　Delay　　File：7788

Lbl：0

Dla Time：　2

n011　Output　　File：7788

Lbl：0

OutpPt：2　　Stat：O

n012　Abs M　　File：7788

Lbl：0　　X：　0.000

Y：　0.000

F：1000　　Z：　0.000

(OutpPt : Output port,　Stat: Status)

12）BkTMech0(Back to mechanical zero): Execute this instruction, through the set parameters of the instruction, execute the action of Back to the mechanical reference point, after completing the execution, the axis coordinate will clear the axis reference point value on the controller panel.(The reference point is set to 0 by default, and can be set any value according to demand.)

Example: The mechanical reference point Y, Z is located in the program processing origin of coordinates -20 position, X-axis is rotation axis, do not set the mechanical zero point (reference point), the controller is powered on, execute the program, the first line choose the Y-axis back to mechanical zero, then the Z-axis back to mechanical zero, the X-axis does not move, then the Y, Z axis back to coordinate zero, then the X-axis moves 20mm, then Y, Z move 30mm, loop these two actions 50 times, end.

n001　BkTMech0　File：7788

Lbl：0

Slctd Ax：Y

BkTo0Dir：Ne

n002　BkTMech0　File：7788

Lbl：0

Slctd Ax：Z

BkTo0Dir：Ne

n003　Abs M　File：7788

Lbl：0　X：----------

Y： 0.000

F：800 Z： 0.000

n004 Lin M File：7788

Lbl：555 X： 20.000

Y： 0.000

F：800 Z： 0.000

n005 Lim M File：7788

Lbl：0 X： 0.000

Y： 30.000

F：600 Z： 30.000

n006 Loop File：7788

Lbl：0

NumOfLps：49

Dest Lbl：555

(BkTMech0: Back to mechanical zero

Slctd Ax :The selected axis

BkTo0Dir:Back to zero direction

Ne:Negative)

13）Cl Subr(Call subroutine): Execute this line, put the prior written subroutine with the same name in here to execute, after execution, continues to execute the next line. This instruction and 'Subr Bgn(Subroutine begin)' 'Subr End(Subroutine end)' are used together.

14） Subr Bgn(Subroutine begin): Contains the subroutine name, and corresponding with Subr End(Subroutine end), both are indispensable.

15）Subr End(Subroutine end): Corresponding with Subr Bgn(Subroutine begin), both are indispensable.

Example 1: Drilling hole, the Z axis is the drill axis. A row of holes, the pitch is 20 mm, a total of 30 holes, the hole depth is 10 mm.(The plunging and lifting is set the subroutine.)

n001 Lin M File：7788

Lbl：123 X： 20.000

Y： 0.000

F：800 Z： 0.000

n002 Cl Subr File：7788

Lbl：0

Subr Nm：100

n003 Loop File：7788

Lbl：0

NumOfLps：29

Dest Lbl：123

n004 End File：7788

Lbl：0

n005 Subr Bgn File：7788

Lbl：0

Subr Nm：100

n006 Lin M File：7788

Lbl：123 X： 0.000

Y： 0.000

F：100 Z： -10.000

n007 Abs M File：7788

Lbl： 123 X： --------- （Change the value to '--------' through the Cfm(Confirm) key, indicate that the axis does not move.）

Y： 0.000

F：800 Z： 0.000

n008 Subr End File：7788

Lbl：0

(Subr Nm: Subroutine name)

Example 2: Drilling hole, use the cylinder, the pitch of holes is 20 mm, a total of 30 holes, the hole depth is 10 mm.(The plunging and lifting is set the subroutine.)

n001 Lin M File：7788

Lbl：123 X： 20.000

Y： 0.000

F：800 Z： 0.000

n002 Cl Subr File：7788

Lbl：0

Subr Nm：100

n003 Loop File：7788

Lbl：0

NumOfLps：29

Dest Lbl：123

n004 End File：7788

Lbl：0

n005 Subr Bgn File：7788

Lbl：0

Subr Nm：100

n006 Output File：7788

Lbl：0

OutpPt：1 Stat：I

n007 CondJump File：7788

Lbl：89

InpPt： 1 Cond：O

Dest Lbl：89

（The Input port 1 is the INPOS(In Position) signal. If touching the switch, the program will run downwards, or continue to wait.）

n008 Output File：7788

Lbl：0

OutpPt：1 Stat：O

n009 Output File：7788

Lbl：0

OutpPt：2 Stat：I

n010 CondJump File：7788

Lbl：98

InpPt： 1 Cond：O

Dest Lbl：98

n011 Output File：7788

Lbl：0

OutpPt：2 Stat：O

n012 Subr End File：7788

Lbl：0

5. Appendix

1. Electronic gear calculation and formula

- **Electronic gear setting**

The numerator, denominator, represent respectively, the X, Y axis electronic gear numerator, denominator. The range of the value is 1-99999.

The determined method of Electronic gear numerator, denominator:

$$\frac{\text{The required pulses number of motor single direction rotating a circle (n)}}{\text{The moving distance of motor single direction rotating a circle (in μm) (m)}}$$

It is simplified to the most simple fraction, the numerator and denominator are 1-99999 integer. When there is an infinite decimal (such as: π), the numerator and denominator are multiplied by the same number(Try to multiply many times with a calculator, and remember the multiplied total value, after determining, re-calculate to

eliminate the calculation error), so that the impact of the decimal omitted by the numerator or denominator is minimal.

Example 1: Screw Drive: The stepping motor drive is 5000 steps per revolution, or servo drive is 5000 pulses per revolution, the screw lead is 6 mm, the reduction ratio is 1:1, ie 1.0.

$$\frac{5000}{6\times1000\times1.0} \Longrightarrow \frac{5}{6}$$

Namely: the numerator is 5，the denominator is 6.

Example 2: Pinion and Rack: The stepping motor drive is 6000 steps per revolution, or servo drive is 6000 pulses per revolution, the teeth number is 20, the modulus is 2. Then the gear turns one revolution, the rack moves 20 × 2 × π.

$$\frac{6000}{20\times2\times3.14159265358979\times1000} \Longrightarrow \frac{1}{20.943951} \Longrightarrow \frac{107}{2241.00276} \Longrightarrow \frac{107}{2241}$$

Namely: the numerator is 107, the denominator is 2241, and the error is 3 microns difference within 2241 mm.

Example 3: Turns and Revolutions: the stepping motor drive is 4000 steps per turn(revolution), the servo drive motor is 4000 pulses per turn(revolution), the reduction ratio is 1:1, ie 1.0.

$$\frac{4000}{1\times1000\times1.0}=\frac{4}{1}$$

Namely: the numerator is 4，the denominator is 1.

Example 4: Angles and Degrees: the stepping motor drive is 6000 steps per turn(revolution), the servo drive motor is 6000 pulses per turn(revolution), the

reduction ratio is 1:1, ie 1.0.

$$\frac{6000}{1 \times 360 \times 1000 \times 1.0} = \frac{1}{60}$$

Namely: the numerator is 1，the denominator is 60.

- **Increasing and reducing speed curve settings**

The increasing and reducing speed curve is relative to the start speed, the X(Y,Z) axis high speed, and the increase speed time.

Explanation: According to the above three parameters, the system automatically computes to generate an S-shaped curve. The parameters settings of the actual increasing and reducing speed curve are relative to the used motor type and manufacturers, the motor maximum speed, the motor start frequency, the mechanical transmission gear ratio, the weight of the machinery, the inertia of the machinery, the backlash size, the mechanical transmission resistance, the coaxial degree of the motor shaft and the screw shaft, the power loss during transmission, the drive output power, the drive status settings and so on. Pay attention to that the settings should be reasonable, otherwise the following phenomena will appear:

Lost steps: the start speed is too high/ the increase speed time is too short/ the X(Y,Z) axis high speed is too high

Locked rotor: the start speed is too high/ the increase speed time is too short/ the X(Y,Z) axis high speed is too high

Vibration: the start speed is too high/ the increase speed time is too short

Slow: the start speed is too low/ the increase speed time is too long

When using a stepping motor, the increasing and reducing speed curve should not

stall and not lost steps. Through changing the start speed, the X(Y,Z) axis high speed, and the increase speed time, the motion process can achieve the ideal state.

When using a servo motor, the increasing and reducing speed curve should be efficient and no overshoot. Through changing the start speed, the X(Y,Z) axis high speed, and the increase speed time, the motion process can achieve the ideal state.

2 The controller common failures and treatment methods

- The position is not accurate: the reduction ratio of the calculation formula is input backwards, or the electronic gear ratio is filled out backwards, or the reduction of fraction is not accurate.
- White screen : the voltage is instable, or the power is not enough, or the LCD screen contrast knob is adjusted improperly.
- Only positive rotation or only reverse rotation: the motor drive input level doesn't match, or the connection line is virtual connection (the control line and the motor power line).
- Motor stall and vibrate: the subdivision is adjusted improperly, or the motor speed exceeds the rated speed or the motor load is too heavy, or the acceleration or deceleration time is too short.
- All the output ports failure: the output port lines are direct short circuit with +24V, or the relay coil is short circuit, or the relay rated current exceeds 100mA.
- The input port failure: : IO settings are defined improperly, or the input port contact isn't good.

6. Acronyms and Abbreviations

A	
Abs Jump	**Absolute jump**
Abs M	**Absolute motion**
Actl Inp	**Actual input**
Actl Run	**Actual running**
ActlOutp	**Actual output**
Alm	**Alarm**
Alm Inp	**Alarm Input**
AutoExec	**Automatically execute**
B	
Bk0 HSpd	**Back to zero high speed**
Bk0 LSpd	**Back to zero low speed**
BkTMech0	**Back to mechanical zero**
BkTo0Dir	**Back to zero direction**
BkTo0Mo	**Back to zero mode**
BkTProg0	**Back to program zero**
C	
Cl Subr	**Call subroutine**
Cond	**Cond**
CondJump	**Conditional jump**
CtrlPars	**Control parameters**
D	
Del 1Row	**Delete a row**
Del Prog	**Destination label**
Dest Lbl	**Delete program**
Dla Time	**Delay Time**
E	
EmStpInp	**Emergency stop input**
End	**End**
ExtStart	**External start**
ExtPause	**External Pause**

F	
FacVal	**Factory default value**
FailedToSavePwd	**Failed to save user's password**
FailToClrProg	**Failed to clear program**
FailToOwFileNm	**Failed to overwrite file name**
FailToOwProg	**Failed to overwrite program**
FailToResFacVal	**Failed to resume factory values**
FailToSvAsProg	**Failed to save as program**
FailToSvFileNm	**Failed to save file name**
FailToSvPars	**Failed to save parameters**
FailToSvProg	**Failed to save program**
File	**File**
Freq	**Frequency**
I	
I	**Power on**
IcSp	**Increase speed**
IcSp Inp	**Increase speed input**
IncrSpdTime(ms)	**Increase speed time (ms)**
InpPars	**Input parameters**
InpPt	**input port**
Ins 1Row	**Insert a row**
IO Set	**IO settings**
J	
JogIncre(micron)	**Jog increment (micron)**
Jog Op	**Jog operation**
L	
Lbl	**label**
Lin M	**Line ar motion**
Loop	**Loop**
M	
Man HSpd	**Manual high speed**

Man LSpd	**Manual low speed**
Man OP	**Manual operation**
Man XNeg	**Manual X negative direction**
Man XPos	**Manual X positive direction**
Mod Pwd	**Modify password**
Mt Run	**Empty running**
N	
Ne	**Negative**
N.C.	**Normally closed**
N.O.	**Normally open**
NumOfLps	**Number of loops**
O	
O	**Power off**
OutpPars	**Output parameters**
OutpPt	**Output port**
Output	**Output**
OverwrProgOK	**Overwrite Program Successfully**
P	
Par Sets	**Parameter settings**
Paus	**Pause**
PgDnInst	**Page down instruction**
PgUpInst	**Page up instruction**
PlsEntrPwd	**Please enter the password**
PlsEntProgFileNm	**Please enter a program file name**
PlsInpPars	**Please input parameters**
Po	**positive**
ProgFileFull	**Program file is full**
ProgMgmt	**Program Management**
ProgOutp	**Program output**
ProgOutpPt	**Program output port**
Prog RI	**Program read-in**
Proh	**Prohibited**
PrsCfmForUsrQuit	**Press the confirm key for user's quit**
PrsCfmToChgOnOff	**Press the Confirm key to change On or Off**
Prs Cfm To Cfm Inp	**Press the Confirm key to confirm input**
PrsCfmToDel	**Press the Confirm key to delete the file**

PrsCfmToSaveFile	**Press the Confirm key to save the file**
PrsCfmToSavePwd	按确认键保存密码
Pt	**port**
PTP M	**Point to Point motion**
PTP Spd	**Point to Point speed**
Pwd Err Pls Entr Agn	**Password mistake, please enter again**
R	
Res FacVals OK	**Resume factory default values successfully**
RdSp	**Reduce speed**
RdSp Inp	**Reduce speed input**
S	
Save	**Save**
SavePars	**Save parameters**
Save Pars OK	**Save the parameters successfully**
Save Prog OK	**Save the program successfully**
SaveUsr'sPwd Ok	**user's password saved successfully**
SaveWhenSameWithExst	**Save the file when the same name with the existing file**
SetCoord	**Set coordinates**
Set Inp	**Setting input**
Slctd Ax	**The selected axis**
Spd Pars	**Speed parameters**
SS Mo	**Single-step mode**
St	**Start**
Stau	**status**
St HSpd	**Start high speed**
Stop	**Emergency stop**
Subr Bgn	**Subroutine begin**
Subr Nm	**Subroutine name**
SvAs Prog OK	**Save as the program successfully**
SvAsWhnNotSmWithExst	**Save as the file when not the same name with the existing file**
Sv Prog	**Save program**
Sys Pars	**System parameters**
Sys SC	**kSystem self-chec**
SysUsrLogin	**System user login**

T	
Test Op OK	**Test operation successfully**
Test user login	**Test user log in**
ToSavePars,PlsPrsCfm	**Press the Confirm key to save the parameters**
ToSetAllParsToFacVals,PlsPrsCfm	**Press the Confirm key to set all parameters to factory default values**
TrmtProg	**Terminate program**
Tst Actl Inp	**Test actual input**
Tst Set Inp	**Test setting input**
U	
Usr LI	**User logged in**
UsrLogin	**User login**
Usr Login OK	**User login successfully**
Usr Mgmt	**User management**
Usr NLI	**User not logged in**
Usr Quit	**User quit**
Usr Quit OK	**User quit successfully**
V	
Val	**valid**
X	
XAxsDnmn	**X-axis denominator**
XAxsGap(micron)	**X-axis gap (micron)**
XAxsHSpd	**X-axis high speed**
XAxsNmr	**X-axis numerator**
XAxs RP	**X-axis reference point**
X+Lim	**X positive limitation（X-axis positive limitation）**
X-Lim	**X negative limitation（X-axis negative limitation）**
X zero	**X zero**
XAxs 0	**X axis zero**
X NL	**X negative limitation**
X PL	**X positive limitation**