User Manual

PLC of CNC System

V2.0

Shenzhen Guanhong Technology CO.,LTD
Add: Anxin Building, No 536 Shenhui Road, Liuyue community, Henggang Street ,Longgang District, Shenzhen City,Guangdong Province, China
Post code: 518115

-Catalogue-

Chapter 1 PROGRAMMING

1.1 PLC Specification

Specification	SZGHTECH CNC System
Programming Language	Ladder
Programming Software	GHPLCS
Programming Grade	2
Executive Cycle of Program	8ms
Basis code average treatment time	<1.5us
Max Step of Program	4700 steps
Programming Instruction	Basic Instruction + Function Instruction
Internal Assistant Relay (R)	R000~R511
PLC Alarm Detection (A)	A000~A031
Timer(T)	T000~T127
CUNTER (C)	C000~C127
Data Base (D)	D000~D255
Nonvolatile relay (K)	K000~K063
Counter preset value data register (DC)	DC000~DC127
Timer preset value data register (DT)	DT000~DT127
Subprogram (P)	P000~P099
Mark (L)	L000~L099
Input signal of NC side (F)	F000~F063
Signal outputs to the NC side (G)	G000~G063
Input(X)	X000~X063
Output(Y)	Y000~Y047

1.2 Sequential program

The sequential program is defined to logically control refer to the machine and relative devices.After converting the program into a certain format, CPU can be decoded and arithmetic processing,and stored in RAM. And CPU read the codes in high speed and executed by the arithmetic operation.

Sequential program is compiled in the beginning of ladder.

1.2.1 Distribution of I/Os (step1)

The interface can be distributed after control target is defined and the corresponding input/output signal points are counted. Refer to input/output interface signal list.

1.2.2 Edit of Ladder (step2)

Ladder can be edited online,& edited by SZGHPLCS.exe on computer.After finished well,it needs to debug.

1.2.3 Debug of Ladder (step3)

After finished, the ladder is debugged as follows:

A: Emulator

Use one emulator (TEST in PLC software) instead of machine to debug it. Machine signal state is represented with switch ON/OFF,and output signal state is done with indicator ON/OFF. Observe if ever indicator on the emulator is correct when test on software.

B: Diagnosis of CNC

After finished well & restore into system, we could observe if the diagnostic state of every signal is consistent with the function requirement when executing CNC. Check the ladder by checking condition of each I/Os in Diagnosis interface.

C: Actual Run

There may be unexpected result in the actual debugging machine and so we need to do preventive measures before debugging.

Chapter 2 Sequence Program

The operating principle is different with the common relay, because the PLC sequence controlling is carried out by the ladder diagram GHPLCS software compiling.

And therefore, it is better to thoroughly understand the sequence controlling principle when designing the PLC sequence programming.

2.1 Performance Process of Sequence Programming

In the general relay controlling circuits, each of them can be simultaneously operated. When the relay A is operated in the following figure, the replay D and E can be operated (when the contactor A and B are closed) at the same. Each replay in the PLC sequence control is operated in turn. The relay D is operated before relay A, and then the relay E operates (refer to the following figure). Namely, each relay is operated based upon the sequence of the ladder diagram (compiling sequence).

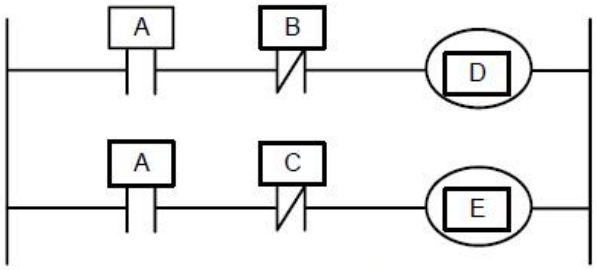


Fig. 2.1 (a) Circuit illustration

The differences between the relay circuit and PLC programming operation are shown below in the Fig. 2.1 (b) and Fig. 2.1 (c).

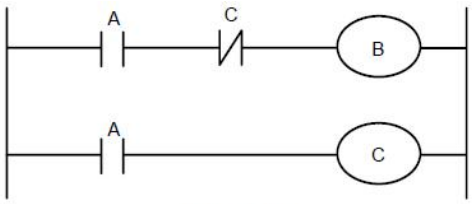


Fig. 2.1 (b)

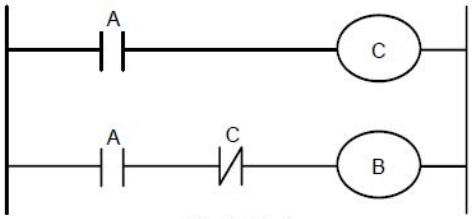


Fig. 2.1 (c)

(1) Relay circuit

Both Fig. 2.1 (b) and Fig. 2.1 (c) are shared a same operation. B and C are switched on after A is turned on. B is cut off after C is ON.

(2) PLC program

A same relay is shared a same circuit, refer to the Fig. 2.1 (b); B and C are switched on after A is turned on. B is cut off after one cycle of the PLC program is performed. In the Fig. 2.1 (c), C is ON instead of B, after C is turned on.

2.2 The Performance of the Cycle

PLC performs from the beginning to the end of the ladder diagram it performs again from the beginning of the ladder diagram after this diagram is performed. which is called cycle performance The performance time from the beginning to the end of the ladder diagram is abbreviated as a period of a cycle treatment The shorter of the treatment period is, the stronger of the response capacity of the signal is.

2.3 The Priority Sequence of the Performance

PLC program are composed of two parts: the 1st Level program and the 2nd level program. which are inconsistent with the performance period The P level program performs once each 8ms, which can be treated some fast corresponding and short pulse. The 2 program performs once each 8-nms N is the partition value of the 2nd level program PLC may divide the 2" level program into N parts when the 2 level program is executed it is performed one part for each 8ms

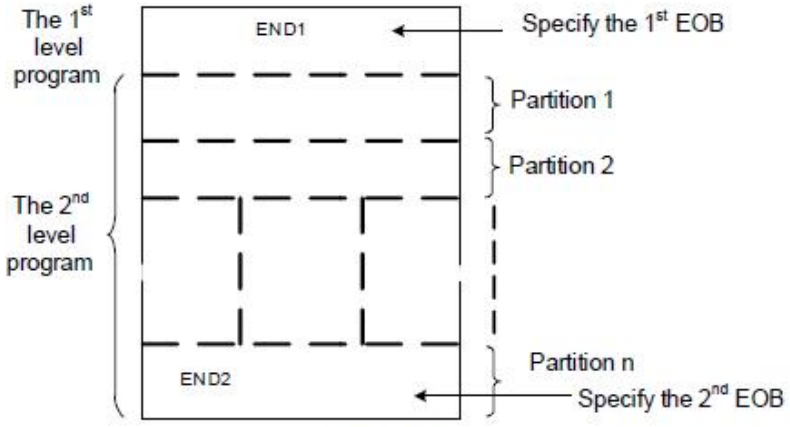


Fig. 2-3-1

PLC on CNC system is separately performed in the PLC-ARM7 The 1ms of each 8ms is the communication time for reading the PLC data from the CNC The 5ms is that the PLC gains the system control signal (F X). and uploads the control result data (G. Y parameter) external port 10 PLC is always performed me ladder diagram calculation Other than the interruption of hie response exchange data.

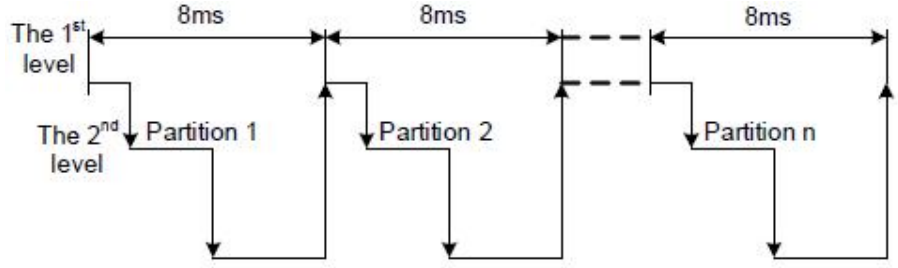


Fig. 2-3-2

When the last partition value of the 2nd level program of the n is performed, the program is then executed from the beginning of the program. In this case. when the partition value is n, the performance time of one cycle is 8*n ms. The 1st level program performs once each 8ms; the 2 level program performs once each 8.n ms. If its steps of the 1st level program is increased, and therefore the steps of the 2nd level

program within 8ms should be reduced correspondingly; the partition value may be increased, and the treatment time of the overall program will be longer. So, the compiling of the 1g level program should be shorter.

2.4 Sequence Programming Structure

The ladder diagram is compiled with sequence in the traditional PLC. It owns the following advantages in the ladder diagram language allowing the structured programming:

1. The program is easy to comprehend and compile.
2. It is more convenient to find the faults during the programming
3. It is easy to find some reasons when the operation malfunction occurs.

The methods of the main structure programming are shown below

1) Sub_program

The subprogram is regarded as a treatment unit based on the ladder diagram

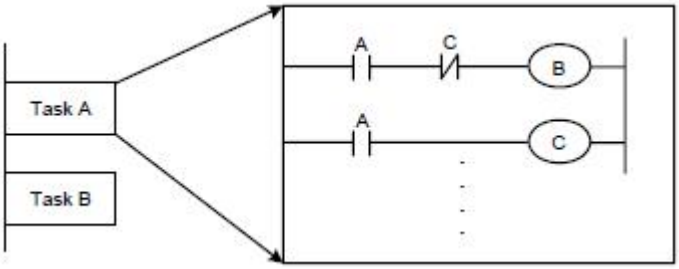


Fig. 2-4-1

2) Nesting

One subprogram can be performed the task by calling another one.

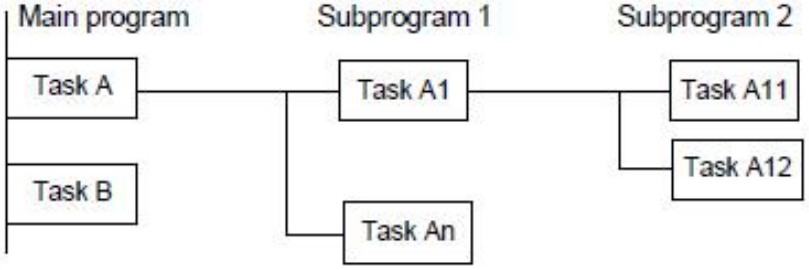


Fig 2-4-2

3) Conditional branch The main program is performed circularly, and checks whether its conditions are suitable. The corresponding subprograms are performed under these conditions, vice versa.

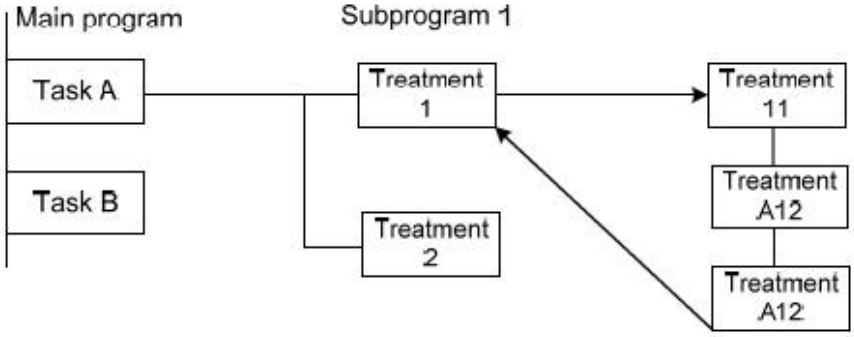


Fig. 2-4-3

2.5 The Treatment of Inputs/Output Signal

The treatment of the input signals:

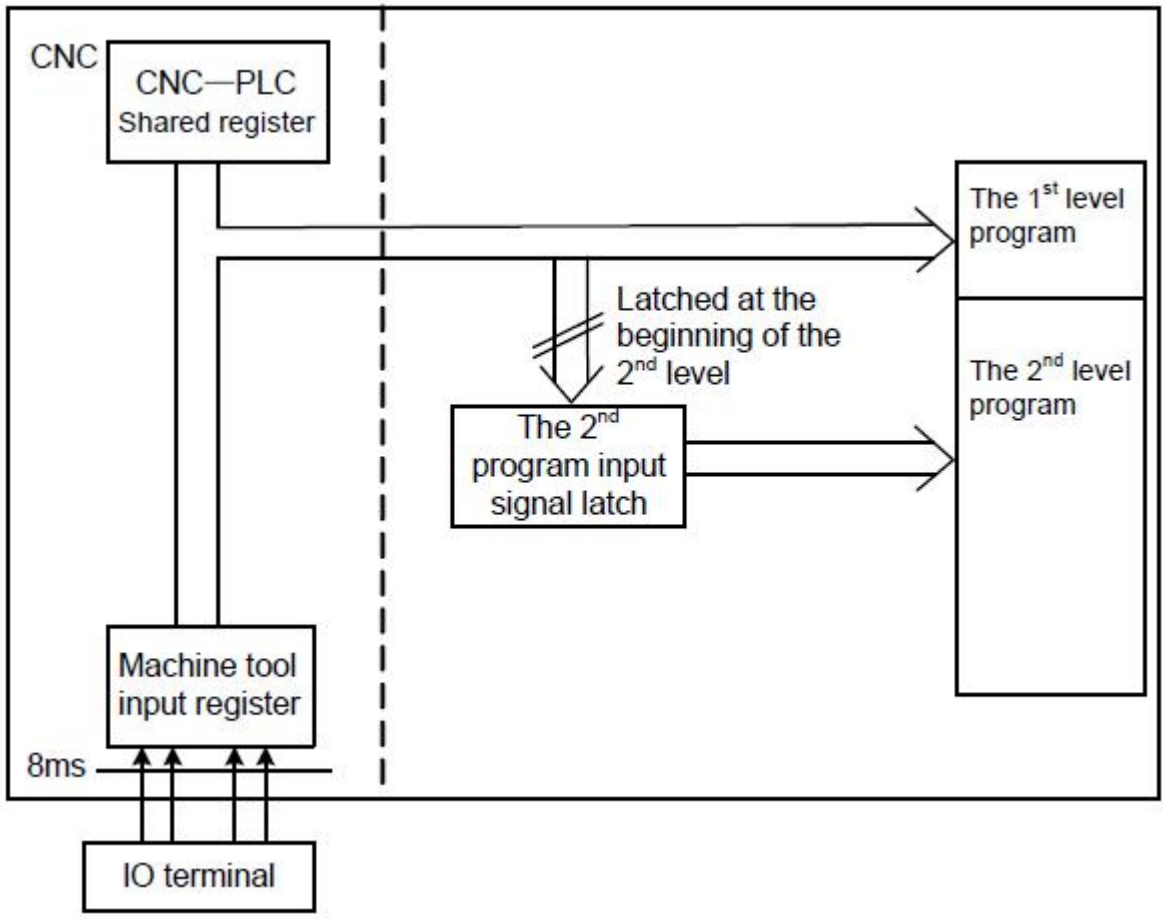


Fig. 2-5-1

The treatment of the Output signals:

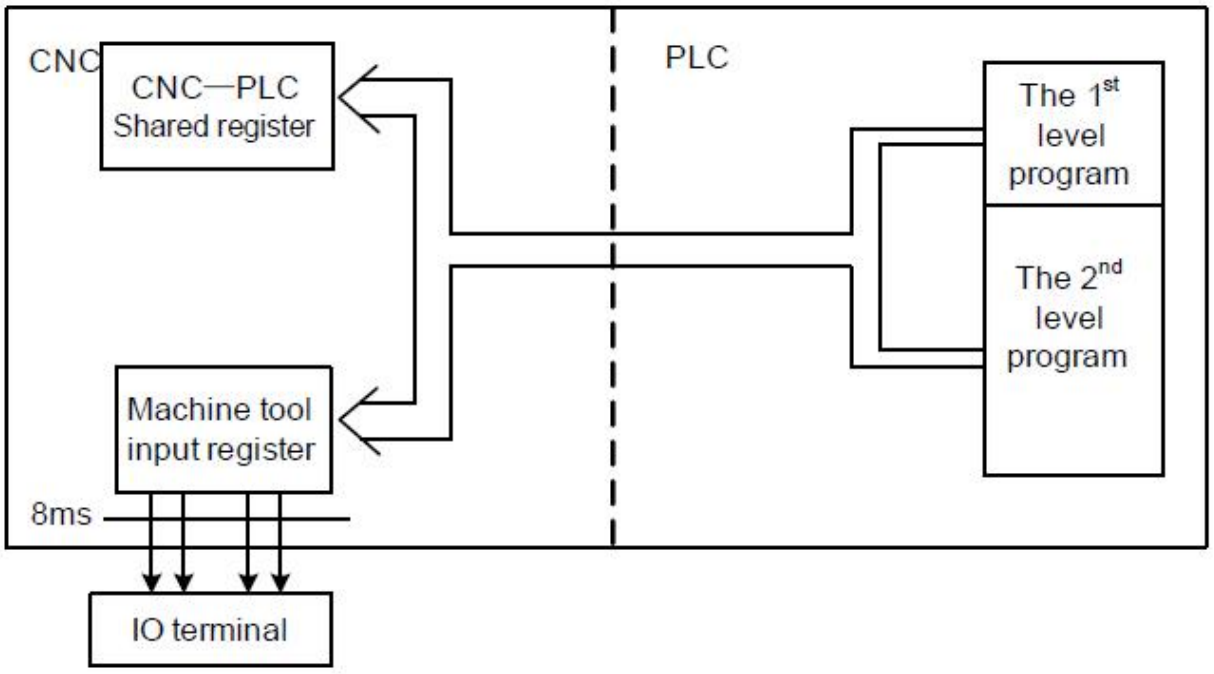


Fig. 2-5-2

2.5.1 Input Signal Treatment

(1) NC input register

The NC input signals from the NC are memorized into the NC input register, which are transferred to PLC each 8ms. The 1st level program performs the corresponding treatment using state of these signals.

(2)Machine tool input register

The machine tool Input register is scanned and memorized its input signal from the machine each 8ms. The 1st level program is also performed the corresponding treatment by using this signals directly.

(3)The 2nd level program input register

The 2nd level program input signal register is also called the 2no level program synchronic input signal register. Wherein, the stored input signal is treated by the 2nd level program. This signal state in the register is synchronic with the 2nd level one.

The signals both in the NC and machine tool input register can be locked to the 2nd level program input latch, as long as the 2nd level program performs. The signal state in this latch keeps invariable during the performance of the 2m level program

2.5.2 The Treatment of the Output Signal

(1) NC output register

The output signal transfers to the NC output register from the PLC each 8ms.

(2) Machine tool output register

The signal memorized in the machine tool Output register conveys to the machine tool each 8ms

Note: The signal states. such as the NC input register. NC output register. machine input register and machine output register, which can be displayed by the self-diagnosis function. The diagnosis number is the address number in the sequence programming.

2.5.3 The Distinguish of the Signal State between 1st level & 2nd level

As for the same input signal, their states may different between the 1st and 2nd level programming, that is the reason that different registers are used between two levels programming. Namely, the input signal used with the 2nd level program is the one of the 1st level who is locked. And therefore, the signal in the 2nd level program is later than the 1st level one. At the worst case, one 2nd level program performance cycle can be lagged.

It is better to remember this point when programming the ladder diagram.

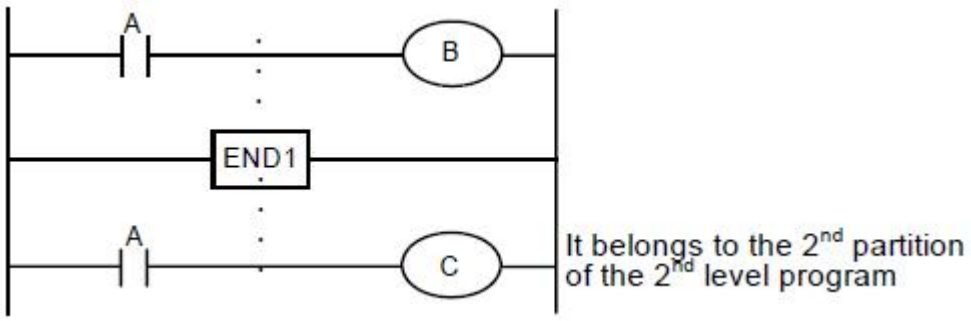


Fig. 2-5-3-1

A=1 performs the 1st level program when the 1st 8ms is performed, then B=1. And therefore, the 2nd level program is performed, the A=1 is latched to the 2nd level program, and then the first partition of the 2nd level program is completed.

A turns into 0 to perform the 1st level program when the 2nd 8ms is performed, then B=0. And therefore, the 2nd partition of the 2nd level program is performed; in this case, the state of the A is still latched as the one last time. So, C=1.

In this way, the state both B and C are different.

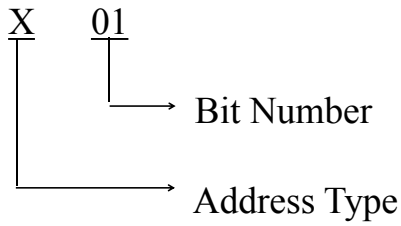
2.6 Interlocking

In the sequence control, the interlocking is very important from the safety issue.

It is necessary to use the interlocking in the sequence control programming. Simultaneously, the hard interlocking is used in the relay control circuit of the strong electric cabinet of the machine tool sides. This is the reason that the interlocking is disabled when the hardware of the performance sequence programming malfunctions, even if the interlocking is logically used in the sequence program (software). And therefore, the interlocking can be ensured the safety for the user, and prevent the machine tool from damaging in the strong electric cabinet of the machine sides.

Chapter 3 ADDRESS

Addresses are used for distinguishing signals. Different addresses separately correspond to input/output signal at machine side and CNC side, internal relay, counter, timer, holding relay and data list. An address number is consisted of address type, address number and bit number as follows:



Address type: X , Y , R ,F, G, K, A, D, C, & T
Bit Number: 0 ~ 999
Address number: Decimal number means one byte.

Bit number: Octonary number system, 0～7 are separately indicated the bytes (0～7 bits) in the front of the address number.

The address type of the GSK990MC PLC is shown below:

Table 3-1

Address	Address Explanation	Length
X	MT ----> PLC (64 bytes)	X000~X063
Y	PLC ----> MT (48 bytes)	Y000~Y047
F	CNC ----> PLC (64 bytes)	F000~F063
G	PLC ----> CNC (64 bytes)	G000~G063
R	Intermediate relay (512 bytes)	R000~R511
D	Data register	D000~D125
DC	The data register of the counter preset value	DC000~DC127
C	Counter (0~127)	C000~C127
A	PLC alarm detection	A000~A031
T	Timer (0~127)	T000~T127
DT	The data register of the timer preset vlaue	DT000~DT127
K	Nonvolatile relay (64 bytes)	K000~K063

INT8U data type is 8-bit character type without symbol, INT16U data type is 16-bit integral type without symbol.

3.1 Inputs (X)(Machine Tool --> PLC)

The X address of the GH CNC system PLC composes of two types:

1. The X address is assorted with the I/O input terminal: CN61 Input port & CN15 SP-Axis port(Reference as Appendix 3), total 36 pieces.

2. The X address is assorted with the input button on the MDI panel of the system.

3.2 Outputs (Y)(PLC ---> Machine Tool)

1. Output address(Y) are distributed to I/O interfaces,CN62 Output port , CN15 SP-Axis port(Reference as Appendix 3), total 36 pieces.

2. The Y address is assorted with indicators on the MDI panel of the system.

3.3 PLC ---> CNC Address (G)

The addresses from G0 to G63, its definition type: INT8U, totally 64 bytes.Defines of each address,please reference as Appendix4: G, F signals.

3.4 CNC ---> PLC Address (F)

The addresses from F0 to F63, its definition type: INT8U, totally 64 bytes.Defines of each address,please reference as Appendix4: G, F signals.

3.5 Internal Replay Address（R）

The address area is reset when the system is turned on. R510 and R511 are used by the system. Its definition type is: INT8U, totally 512 bytes.

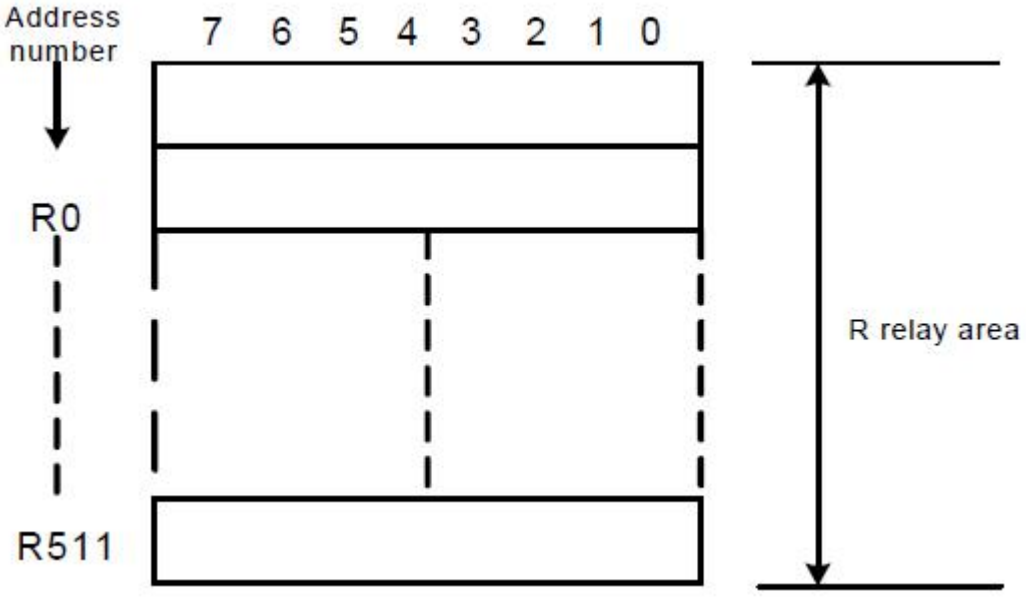


Fig. 3-5-1

System program administration area:

R510

The signal of R510.0 address is set to 1 when PLC starts and restarts,which is

used the signal set by the initial user. The R510.0 is reset to 0 after the ladder diaram is performed once.

R511 (System timer)

The following four signals can be used for system timer:

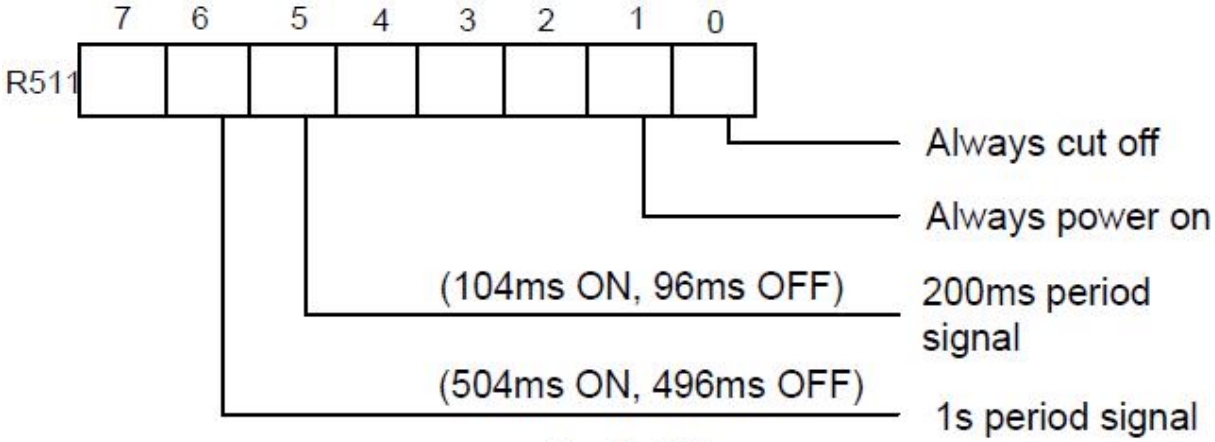


Fig. 3-5-2

3.6 Nonvolatile Relay Address (K)

This address area is used for nonvolatile replay and PLC parameter setting. This area is called nonvolatile relay area, namely, the content inside the register will not lose even if the system is turned off. K000~ ~K005 are used by the system, which is used to protect the PLC system parameter, it is very convenient for user to control PLC in the CNC system.

Its definition type: INT8U, totally 64 bytes.

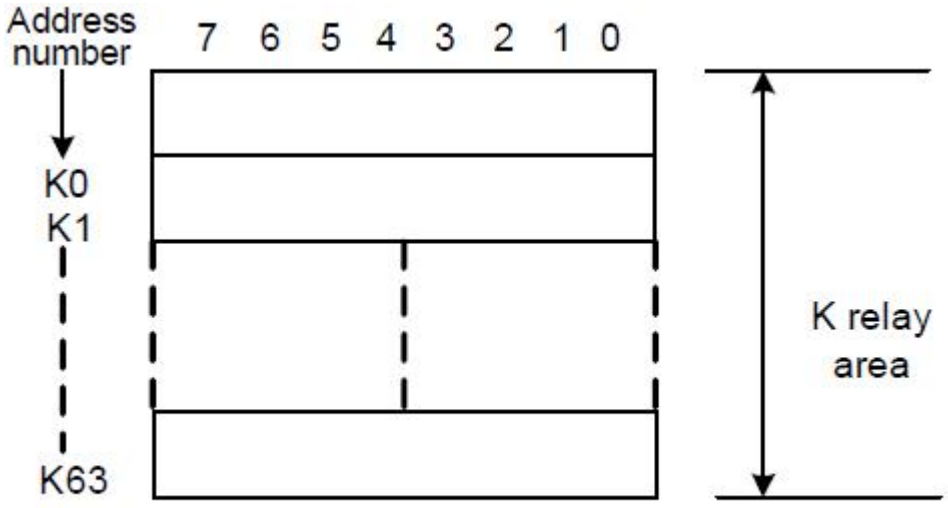


Fig. 3-6-1

Note: When PLC enters the debugging mode. All of the external alarms are canceled, and the machine interlocking signals are then canceled, the tool-change code can not be performed. The parameter can be modified only when comprehending the parameter, so that the damage in the machine tool or injury of the person may occur.

3.7 Information Display Request Address (A)

This address area is reset when the system is turned on.

Its definition type: INT8U, totally 32 bytes.

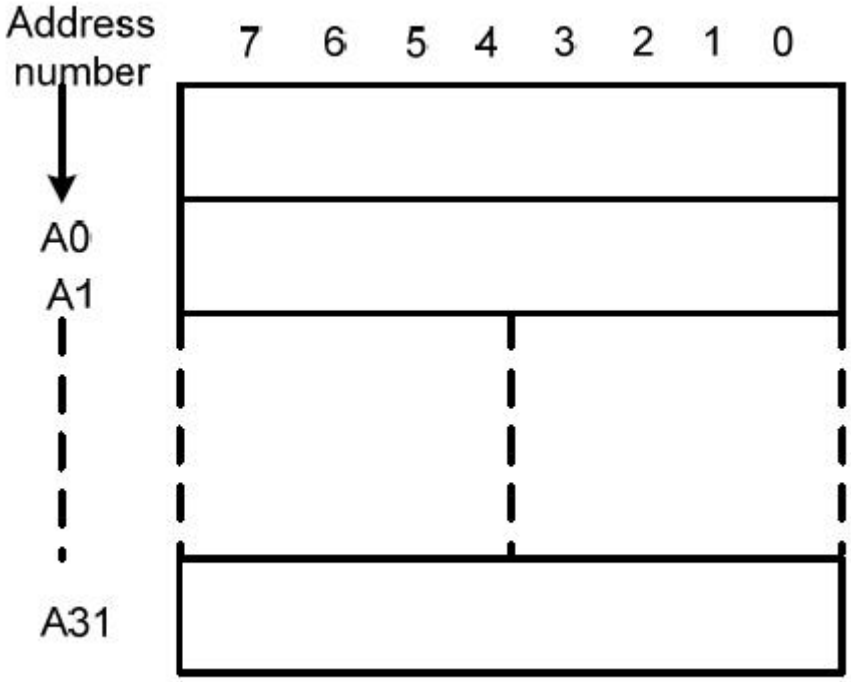


Fig. 3-7-1

3.8 Counter Timer (C)

This area is used for placing the current count value of the counter. The data is reset after the system is turned off.

Its definition type: 128 addresses.

3.9 Counter Preset Value Address (DC)

This address area is used for storing the counter preset, which is a nonvolatile storage area, that is, the memorized content may not loose even the system is power off.

Its definition type:128 addresses. The setting value of DC is only read instead of writing.

3.10 Timer Address (T)

This address area is used for storing the current numerical value of the timer. The data initial is presetting value after the system is power off. Current data is presetting value when it is set to 0.

Its definition type: 128 addresses

3.11 Presetting Value Address of the Timer (DT)

This address area is used for placing the timer preset value. This area is nonvolatile register area, namely, the content inside the register will not lose even if

the power of the system is turned off.

Its definition type: 128 addresses. The setting value of DT is only read instead of writing.

3.12 Data Table Address (D)

The content inside the memory will not lose even if the power of the system is turned off.

Its definition type: totally 256 addresses. Among them, D240~247 are used by the system instead of the user.

3.13 Label Address (L)

It is used to specify labels both skip object and the LBL code in the JMPB code.

Its range: 0～99

3.14 Subprogram Number (P)

It is used to specify the called object subprogram number in the CALL code and the subprogram number in the SP code .

Its range: 0～99

Chapter 4 PLC BASIC INSTRUCTIONS

The design of the sequence program begins from the compiling of the ladder diagram. The ladder diagram consists of relay contact and function code. The logic relationship in the ladder diagram composes of sequence program. There are two methods of the sequence program input: one is that the input method uses the mnemonic symbol language (The system is not temporarily supported the PLC instruction code of the RD, AND and OR); the other one that is used the relay symbol. The programming can be compiled using ladder diagram, and do not comprehend the PLC code based upon the latter.

Actually, the sequence program inside the system can be converted into corresponding PLC code even if it is input by the relay symbol.

The basis codes are commonly used codes when designing the sequence programming, which are performed one-digit calculation.

The basis instruction codes of the GH CNC System are shown below:

Instruction Name	Function
RD	Left shift one bit of the content of the register, the signal state specified by address set to ST0
RD.NOT	Left shift one bit of the content of the register, the signal state specified by address is set to ST0 after its state is set to NOT.
WRT	Output the logic calculation result to the specified address
WRT.NOT	Output the logic calculation result after NOT to the specified address.
AND	Logic AND
AND.NOT	Logic AND after the specified state is set to NOT.
OR	Logic OR
OR.NOT	Logic OR after the specified state is set to NOT.
OR.STK	Right shift one bit of the stacked memory after ST0 and ST1 logic OR
AND.STK	Right shift one bit of the stacked memory after ST0 and ST1 logic AND

4.1 RD, RD.NOT, WRT, WRT.NOT Code

◆ Mnemonic code and function

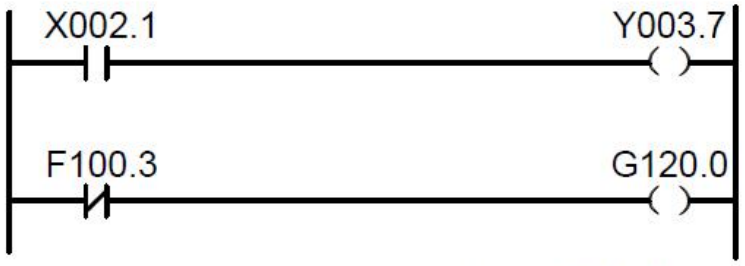
Mnemonic code	Function	Ladder Symbol
RD	Left shift one bit of the content of the register, the signal state specified by address set to ST0	
RD.NOT	Left shift one bit of the content of the register, the signal state specified by address is set to ST0 after its state is set to NOT.	
WRT	Output the logic calculation result to the specified address	
WRT.NOT	Output the logic calculation result after NOT to the specified address.	

Code explanation

WRT and WRT.NOT codes are the coil drive code of the output relay and internal relay, but the input relay can not be used.

The parallel WRT instruction can be used multiply, but it cannot outputs with multi-coil .

Refer to the following programming:




```
RD      X002.1
WRT     Y003.7
RD.NOT  F100.3
WRT     G120.0
```

Fig. 4-1-1

4.2 AND,AND.NOT Code

◆ Mnemonic code and function

Mnemonic code	Function	Ladder Symbol
AND	Logic AND	
AND.NOT	Logic AND after the specified state is set to NOT.	

◆ Code explanation

AND, ANI can connect one contact in serial. There can be many contacts in serial and the instructions can be used many times.

◆ Programming Example

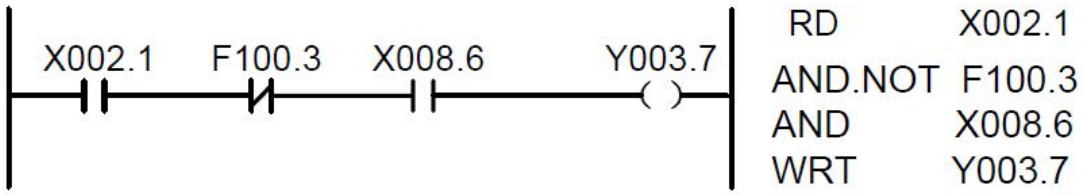


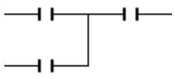
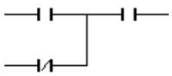
Fig. 4-2-1

Program Explanation:

If X002.1=1, F100.3=0 and X008.6 is 1 , the system output Y003.7 .

4.3 OR,OR.NOT Code

◆ Mnemonic code and function

Mnemonic code	Function	Ladder Symbol
OR	Logic OR	
OR.NOT	Logic OR after the specified state is set to NOT.	

◆ Instruction explanation

A: OR, OR.NOT can be connected to one contact in parallel. When more than two contacts are connected in series and the serial loop is connected with other loop in parallel, the system should use OR.NOT.

B: The system executes OR, OR.NOT from its current step with RD, RD.NOT in parallel.

◆ Programming Example

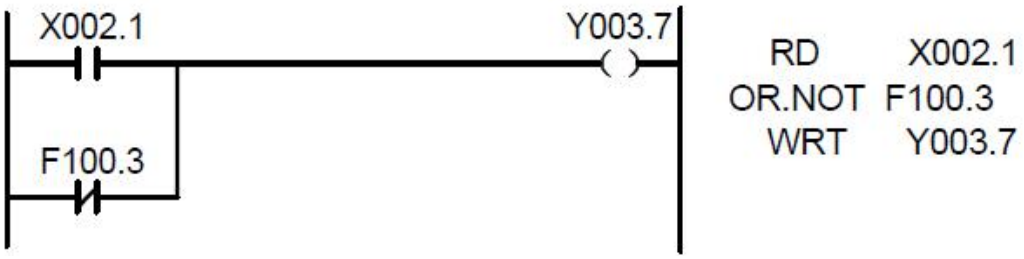
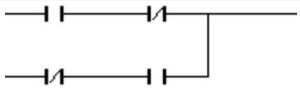


Fig. 4-3-1

4.4 OR.STK INSTRUCTION

◆ Mnemonic code and function

Mnemonic code	Function	Ladder Symbol
OR.STK	Right shift one bit of the stacked register after ST0 and ST1 logic OR	

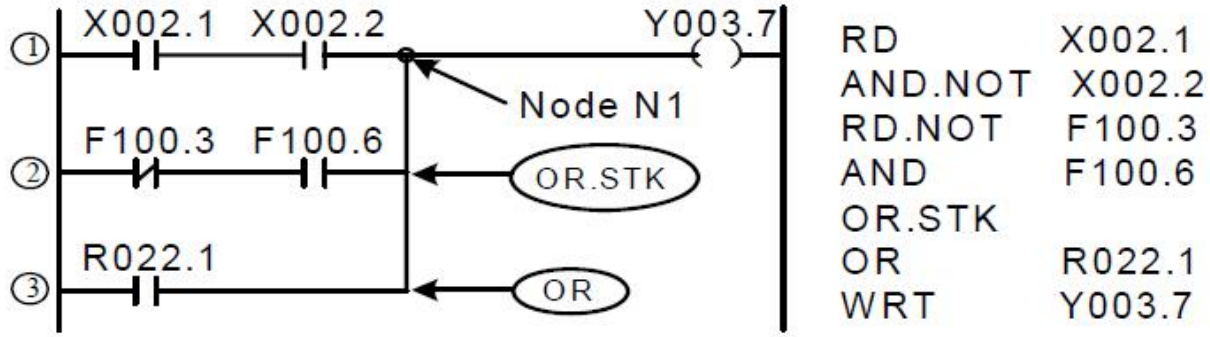
◆ Instruction explanation

OR.STK code is the separate code without any address.

A: Serial loop block is defined to its loop combined by more than contacts in series. When the serial loop is connected in parallel, starting point of branch uses RD and its end point uses OR.STK.

B: ORB is sole instruction without address.

◆ Programming Example



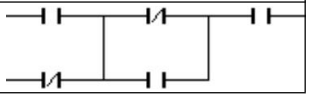
Program explanation：

There are three branches ①, ② and ③ from the left bus to the node N1. The branches ① and ② are series connection circuit block. When the series connection circuit block is performed between bus to node or among the nodes, other than the first branch, use the RD code when the following branch is ended. The branch ③ is not a series connection circuit block, which can be used by the OR code.

OR. STK and AND. STK are the code without operation component, which indicates the OR , AND relationships between circuit blocks.

4.5 AND.STK INSTRUCTION

◆ Mnemonic code and function

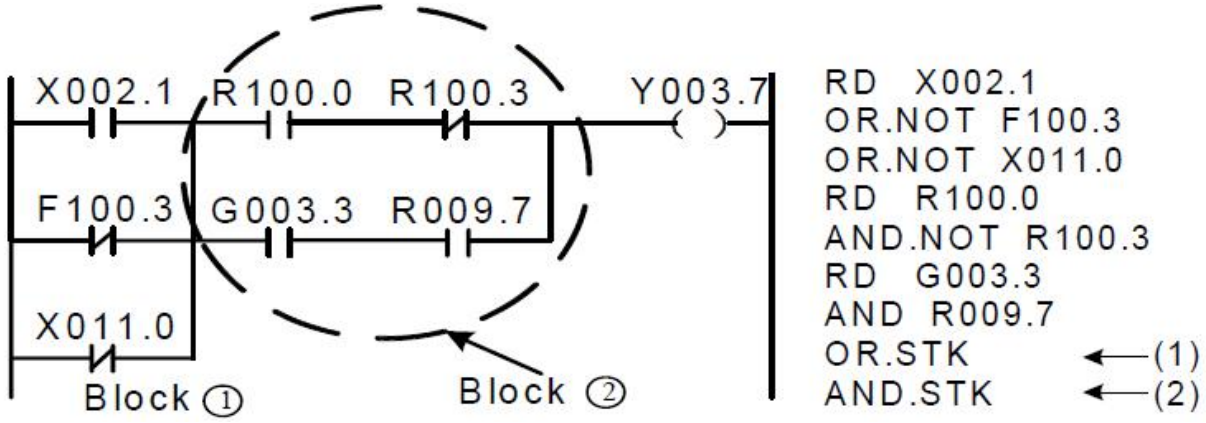
Mnemonic code	Function	Ladder Symbol
AND.STK	Right shift one bit of the stacked memory after ST0 and ST1 logic AND	

◆ Instruction explanation

◆ Use the AND. STK coded when the branch circuit (parallel circuit block) is connected with series connection with the front of the circuit. The start of the branch is used RD, RD.NOT code. Use the AND. STK code is connected with

series connection with the front of the circuit after the series connection circuit block is executed.

- AND. STK code is the separate code without any address.
- Programming Example



Program explanation：

As for the above mentioned ladder diagram and instruction table,(1)OR.STK indicates parallel connection of the series connection circuit block in the block②, (2)AND.STK expresses the series connection between circuit block ① and ②.

CHAPTER5 PLC Function Code

Basic codes such as controlling operations of machine tool are difficult to program, therefore,functional instructions are available to facilitate programming.

Table 5-1 (GH PLC functional instruction code)

No.	Code	Function	No.	Code	Function
1	END1	End of 1st level ladder program	19	ROT	Binary rotation control
2	END2	End of 2nd level ladder program	20	SFT	Register shift
3	CALL	Subprogram call	21	DIFU	Rising edge check
4	CALLU	Unconditional subprogram call	22	DIFD	Failing edge check
5	SP	Subprogram	23	COMP	Binary comparison
6	SPE	End of subprogram	24	COIN	Coincidence check
7	SET	Set	25	MOVN	Transfer of arbitrary number of bytes
8	RST	Reset	26	MOVB	Transfer of 1 byte
9	JMPB	Label jump	27	MOVW	Transfer of 2 bytes
10	LBL	Label	28	XMOV	Indexed data transfer
11	TMR	Timer	29	DSCH	Binary data search
12	TMRB	Fixed timer	30	ADD	Binary addition
13	TMRC	Timer	31	SUB	Binary subtraction
14	CTR	Binary meter	32	ANDF	Functional AND
15	DEC	Binary decoding	33	ORF	Functional OR
16	COD	Binary code conversion	34	NOT	Logical Negation
17	COM	Common line control	35	EOR	Exclusive OR
18	COME	End of common line control			

5.1 END1 (End of 1st Level Sequence Program)

Function: It must be specified once in a sequence program,either at the end of the 1st level sequence, or at the beginning of the 2nd level sequence when there is no 1st level sequence. It can write 500 steps.

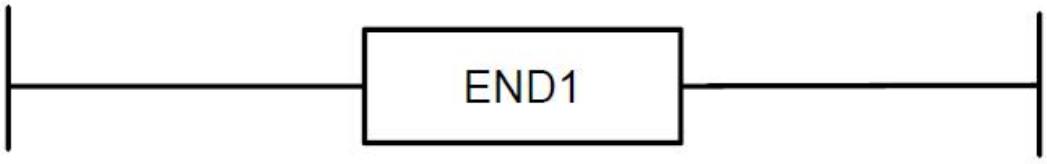


Fig. 5-1-1

5.2 END2 (End of 2nd Level Sequence Program)

Function: Specify at the end of 2nd level sequence.

Format:

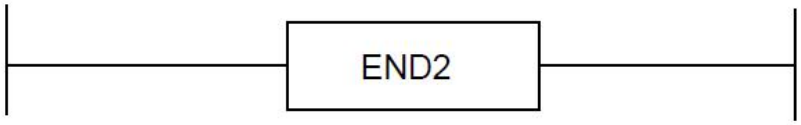


Fig. 5-2-1

5.3 CALL (Call Subprogram)

Function:Call a specified subprogram.

CALL has the following additional functions; The subprogram may be nested up to 18 levels by other subprograms, but if a dead cycle is made by the closed loop calling, an alarm will be issued by system. Therefore to execute the data volume under the control, the allowable subprogram calling times are 100, and the subprogram calling in the 1st level is disabled. Alarm will be issued for the instructions or network between SP and END2, SPE and SP which can't be executed by system.

Format:

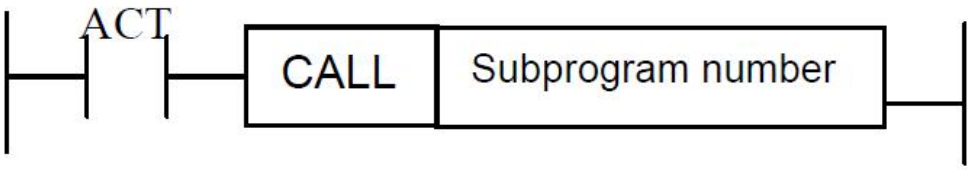


Fig. 5-3-1

Control Condition:

ACT = 0 , execute the next instruction behind CALL.

ACT = 1 , call subprogram which number is specified

Parameter: Subprogram number. specifies the subprogram number of a subprogram to be coded following this instruction. Range: 0-99.

5.4 CALLU (Unconditional Subprogram Call)

Function: Unconditionally call a specified subprogram .

It has the following characteristics and Limits: The subprogram may be nested up to 18 levels by other subprograms, but if a dead cycle is made by the closed loop calling,

an alarm will be issued by system Therefore to execute the data volume under the control. the allowable subprogram calling times are 100. and the subprogram calling in the 1sE level is disabled. Alarm will be issued for the instructions or network between SP and END2. after SPE and before SP which can't be executed by system.

Format:

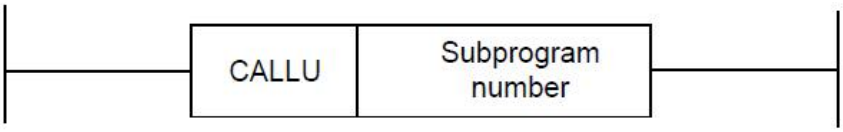


Fig. 5-4-1

Parameter: Subprogram number: specifies the called subprogram number. Range: 0-99.

5.5 SP(Subprogram)

Function: The SP functional instruction is used to create a subprogram. A subprogram number is specified as a subprogram name. SP is used with the SPE functional instruction to specify the subprogram range.

Note:

1. A subprogram must be written after END2.

2. Another subprogram cannot be nested into a subprogram.

Format:

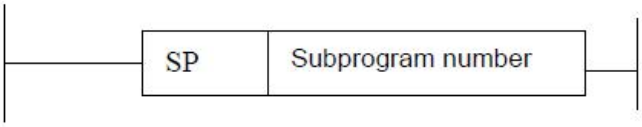


Fig. 5-5-1

Parameter:

Subprogram number: specifies the called subprogram number. Range: 0～99.

5.6 SPE (Subprogram End)

Function: * it is used to specify the range of subprogram when SPE is used with the S P.

* the control will return to the main program which called the subprogram when the instruction is executed.

* the subprogram is written after END2.

Format:

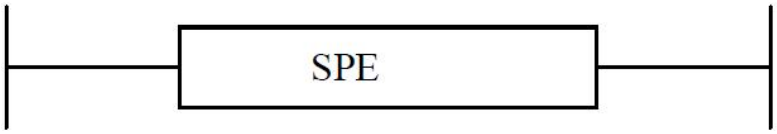


Fig. 5-6-1

Example:

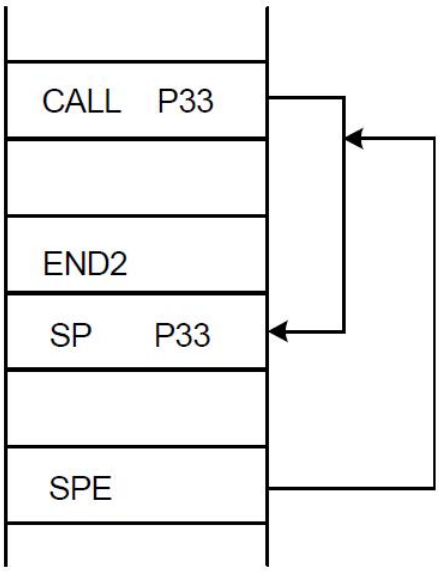


Fig. 5-6-2

5.7 SET (Set)

Function: Set to 1 for the specified address.

Format:

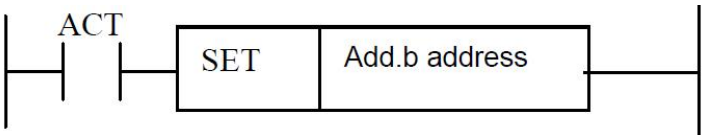


Fig. 5-7-1

Control condition:
ACT＝0 , keep add.b invariably.
ACT = 1, set add.b to 1.
Parameter: Add.b: Reset element address bit can be the output coil, Add= Y, G, R, K, A.

5.8 RST (Reset)

Function: Set to 0 for the specified address.
Format:

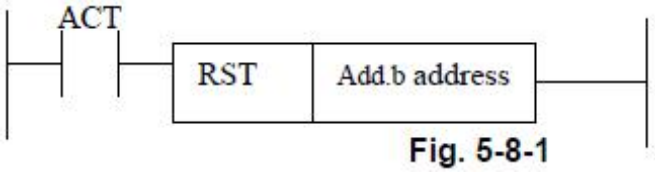


Fig. 5-8-1

Control condition:
ACT=0, keep add.b invariably.
ACT=1, Set add.b to 1.
Parameter: Add.b: reset element address bit can M the output coil. Add=Y, G, R, K, A.

5.9 JMPB (Label Jump)

Function: The JUMP function instruction transfer control to a Ladder immediately after the label set in a ladder program.

JMPB has the following additional functions:

More than one jump instruction can be coded for the same label

Jumped END1 and END2 are forbidden.

Jumped subprogram and subprogram are forbidden

Jump back is permitted, but the user should handle the infinite loop may be caused by it.

Jumped main program and subprogram are forbidden.

Format:

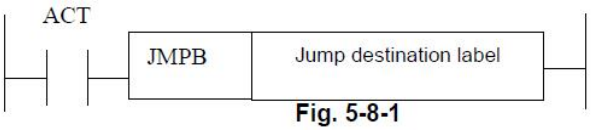


Fig. 5-8-1

Control conditions:

ACT=0 : The next instruction after the JMPB instruction is executed.

ACT=1 : Jump to the specified label and executes the next instruction behind the label.

Parameter:Lx:Specifies the label of the jump destination.A value from 0 10 99 can be specified.

5.10 LBL (Label)

Function: The LBL functional instruction specifies a label a ladder program. It specifies the jump destination for JMPB functional instruction.

Note: one xx label is only specified one time with LBL. Otherwise, the system alarms.

Format:

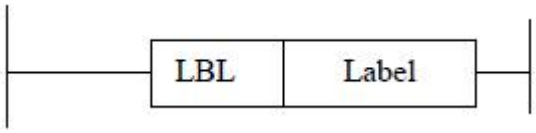


Fig. 5-10-1

Parameter: xx: specifies the label of the jump destination. Label number range: 0～99.

Example:

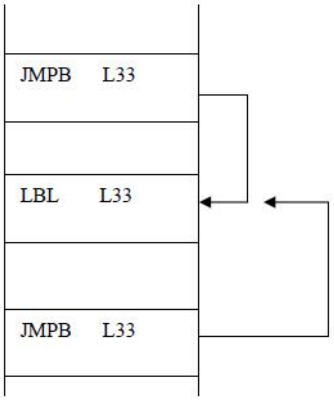


Fig. 5-10-2

5.11 TMR (Timer)

Function: This is an on-delay timer.

Format:

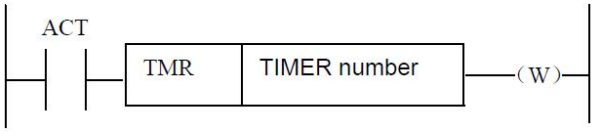


Fig5-11-1

Control condition:

ACT=0: turns off the timer relay.

ACT=1: initiates the timer. i.e. timing from 0.

Detailed functions:

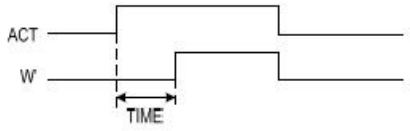


Fig. 5-11-2

Parameter:

TIMER: timer serial number is named with xxx which are numbers (0～127).

Output: W: output coil.

W=1 when the output reaches the preset value.

W=0 when the output does not reach the preset value.

Note:Timer is executed each 8ms,take ms as its setting unit, and 8ms is taken as execution base.

Those time less than 8ms are taken as 8ms. i.e. it is set for 54ms, 54=6*8+6, 2ms is needed to be added, so the actual execution time is 56ms.

The time of the timer is set under the 【TMR】of 【PLCPAR】in PRG window.

The system will automatically detect the range of the sequence number of the timer, alarm will be issued for those duplicate or beyond range sequence numbers.

5.12 TMRB (Fixed Timer)

Function: This is an on-delay timer.

Format:

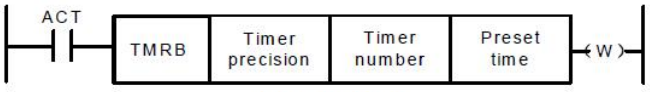


Fig. 5-12-1

Control condition:

ACT=0 : turns off the timer relay.

ACT=1 : initiates the timer.

Detailed functions:

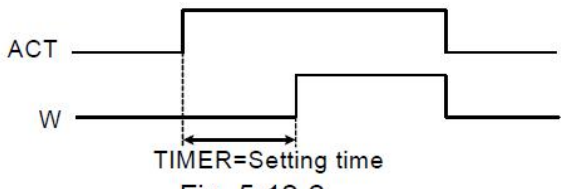


Fig. 5-12-2

Parameter:
TIMER: timer serial number is named with xxx which are numbers(0~127)

Output:

W: coil output. W=1 when the counter reaches the preset value.

Note 1: The system automatically check the counter's sequence number range. An alarm occurs when the serial number is repetitive or exceeds its range.

Note 2: After the ladder is upgraded. the current value of the counter is cleared.

To get reliable counter counting, reset the counter by the pulse signal before counting.

5.13 DEC (Binary Decode)

Function:

DEC can decode binary code data. Outputs 1 when the eight-digit BCD signal is equal to a specified number, and 0 when not. It is mainly used to decode M or T function.

Format:

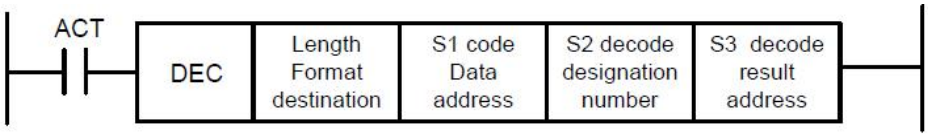


Fig. 5-15-1

Control condition:

ACT=0 : resets all the output data bit.

ACT=1 : decode data. Results of processing is set in the output data address.

Parameter:

Length : Set the size of code data to the 1st digit of the parameter.

0001 : code data is in binary format of 1 byte length.

0002 : code data is in binary format of 2 byte length. code data address.

S1 : Specifies an address at which code data is stored.

S2 : Number specification decode designation. Specifies the first of the 8 (1 byte) or 16 (2 bytes) continuous numbers to be decoded.

S3 : decode result address. Specifies an address where the decoded result shall be output. A one-byte or two-byte area is necessary in the memory for the output.

Example:

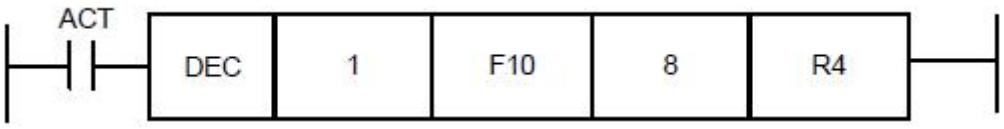


Fig. 5-15-2

When ACT=1 and F10=8, R4=0000,0001 ;

When ACT=1 and F10=9, R4=0000,0010 ;

... ...

When ACT=1 and F10=15 , R4 =1000,0000 .

5.14 COD (Binary Code Conversion)

Function:

COD instruction automatically creates a table with corresponding size used for user inputting

conversion table data when it inputs the data capacity. Each table has 10 lattices and if it is not

divided by 10, count the lattices by its quotient adding 1, but its capacity data does not change.

Format:

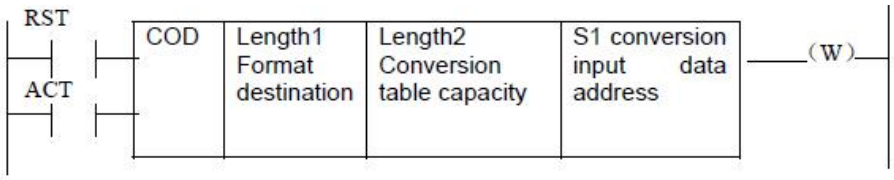


Fig.5-16-1

Table 5-16-1

S1	0	1	2	.....	9
S2	XXX	YYY	AAA	.....	
S1	10	11	12	.....	N-1
S2	.....	.....	.....	.....	UUU

Control conditions:

Reset (RST):

RST=0: do not reset.

RST=1: reset error output W.

Activate instruction (ACT):

ACT＝0: do not execute COD.

ACT＝1: execute COD. Take value of “Conversion input data address(S1)” as the table number of conversion table, take out a corresponding one version data which corresponds to the table number from the conversion table, output the output address used for the conversion data (S2).

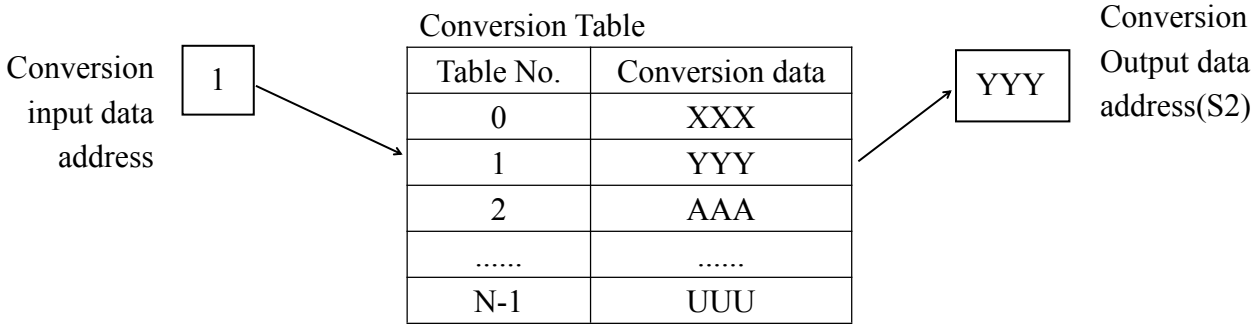


Fig. 5-16-2

Parameter:

Length1 : designates binary numerical size in the conversion table.

1 : Numerical data is binary 1-byte data.

2 : Numerical data is binary 2-byte data.

Length2 : Capacity of conversion table data. 100 data can be made. 100 bytes when designating 1 byte format, and 100 words when 2 byte format. All number is at most 512 bytes in COD conversion table.

S1 : Data in the conversion data table can be taken out by specifying the table number. The address specifying the table number is called conversion input data address, and 1-byte memory is required from the specified address.

S2 : S2： Conversion data output address. Memory of the byte length specified in the format designation is necessary from the specified address.

Output: If t here are an y abnormality when executing the C ODB instruction, W=1

Note:Size of the conversion data table is maximum 100. This conversion data table is programmed between the parameter conversion data output address of this instruction and the error output (W).

5.15 COM (Common Line Control)

Function:

This function can be used for specifying the number of coil till the common end code COME. If the common line end instruction is not specified, the system will alarm.

Format:

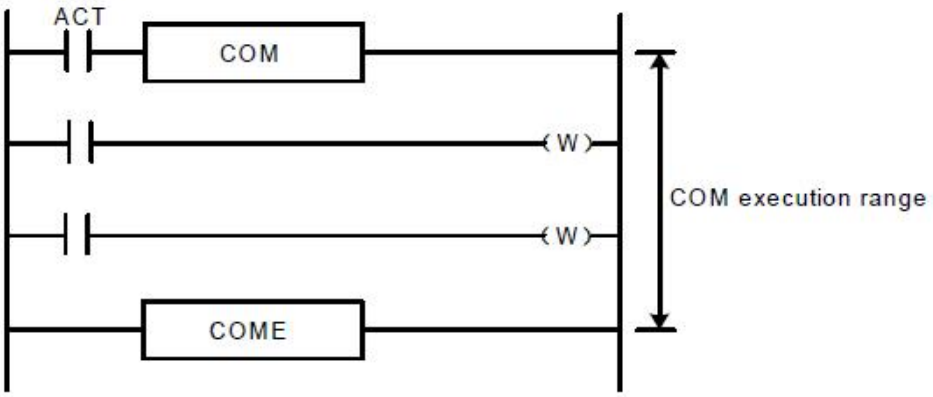


Fig. 5-17-1

Control condition:

ACT=0: The specified number of coils or the coils within the region specified are unconditionally turned off (WW=0).

ACT=1: it is thee same with COM which is not executed.

Note 1: In the range specified with a COM instruction, no additional COM instruction can be specified.

Note 2: The coil for WRT.NOT in the range specified with a COM instruction is singly set to 1 when COM ACT=0.

Note 3: Do not use JMPB,END, END2, CALL, CALLU, LBL,SSP,SPE, COM, COM between COM and COME, otherwise, an alarm occurs.

5.16 COME (Common Line Control End)

Function:

The instruction can be used to specify the control range of the common control line instruction (COM).This instruction cannot be used alone.It must be used together with COM instruction.

Format:

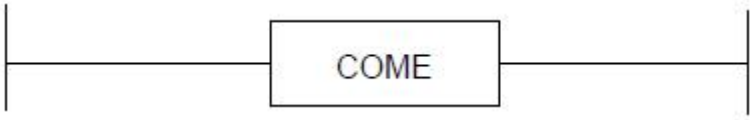


Fig.5-18-1

5.17 ROT (Binary Rotation Control)

Function:

Controls rotors, such as tool post, rotary table, etc., and it is used for the following functions.

1. Selection of the rotation direction via the shorter path.
2. Calculation of the number of steps between the current position and the goal position; calculation of the position on position before the goal to the number o f steps up to one position before the goal.
3. To calculate the position number just before the target position or the steps to the position just before the target position.

Format:

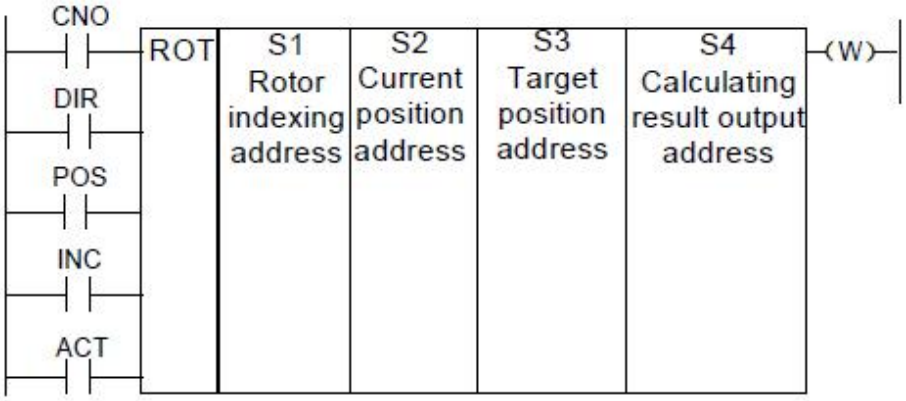


Fig. 5-19-1

Control conditions:

Specify the starting number of the rotor(CN0):

CNO=0: begins the number of the position of the rotor with 0.

CNO=1: begins the number of the position of the rotor with 1.

Select the rotation direction via the shorter path or not: (DIR):

DIR=0: no direction is selected. The direction of rotation is only forward.

DIR=1: selected. The direction of rotation is forward/backward.

Specify the operating conditions (POS):

POS=0: calculate the goal position.

POS=1: calculates the position one position before the goal position.

Specify the position or the number of steps(INC):

INC=0: calculates the number of the position. If the position one position before the goal position is to be calculated, specify INC=0 and POS=1.

INC=1: calculates the number of steps. If the difference between the current position and the goal position is to be calculated, specify INC=1 and POS=0.

Execution instruction (ACT):

ACT= 0: the ROT instruction is not executed. W does not change.

ACT＝1:executed. Normally, set ACT=0. If the operation results are required,set ACT=1.

Parameter:

S1: Specify the rotor indexing number.

S2: Specify the address storing the current position.

S3: Specify the address storing the goal position(or instruction value), for example the address storing the CNC output T code.

S4: Calculate the number of steps for the rotor to rotate, the number of steps up to the position one position before, or the position before the goal. When the calculating result is to be used, always check that if ACT=1.

Output:

W: The direction of rotation for control of rotation via the shorter path is output to W. When W=0, the direction is forward (FOR) when 1, reverse (REV). The definition of FOR and REV is shown in the following figure. If the number given to the rotor is ascending, the rotation is FOR; if descending, REV. The address of W can be determined arbitrarily. When, however, the result of W is to be used, always check that ACT=1.

Example: Rotor rotation direction:

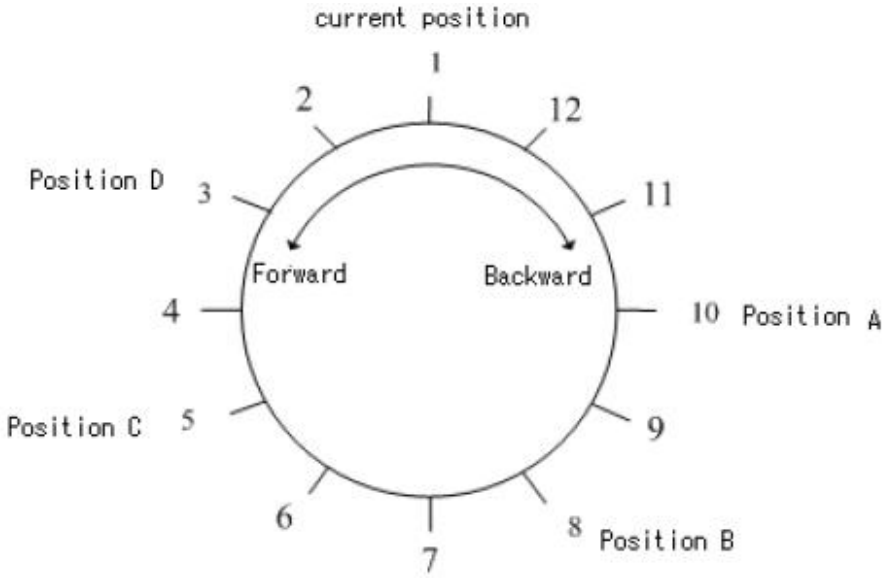


Fig. 5-19-2

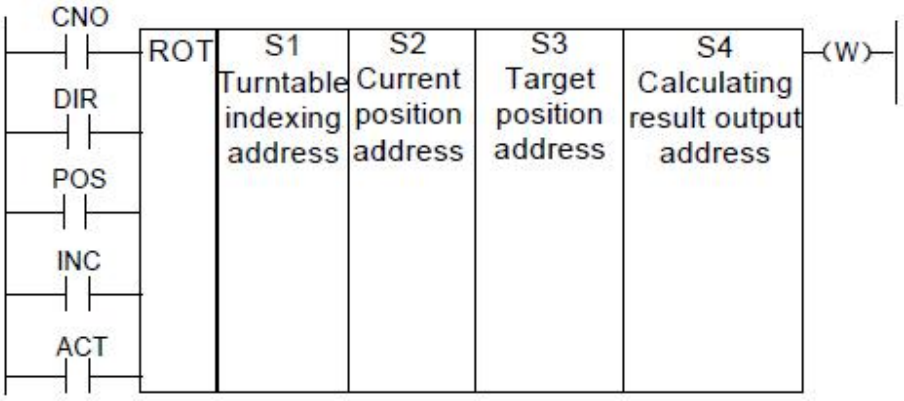


Fig. 5-19-3

Perform the short path rotation, and calculate the position number of previous one position of goal position.

Current position number S2=1, position number of rotation graduation S1 ＝ 12, CNO=1, DIR=1 POS=1, I NC=0:

When S3=10 goal position is A, and ACT=1, S4＝11, W＝1.

When S3=8 goal position is B, and ACT=1, S4＝9, W＝1.

When S3=5 goal position is C, and ACT=1, S4＝4, W＝0.

When S3= 3 goal position is D, and ACT=1, S4＝2,W＝0.

5.18 SFT (Shift Register)

Function:

This instruction can each time shift a byte data (8 bits) by a bits number set by a Parameter, For the circular shifting, each overflowing "1" will be added reversely, i.e. If the highest bit "1" is overflowed by the left shifting, so the lowest bit will be filled b y "1", vice versa.

Format:

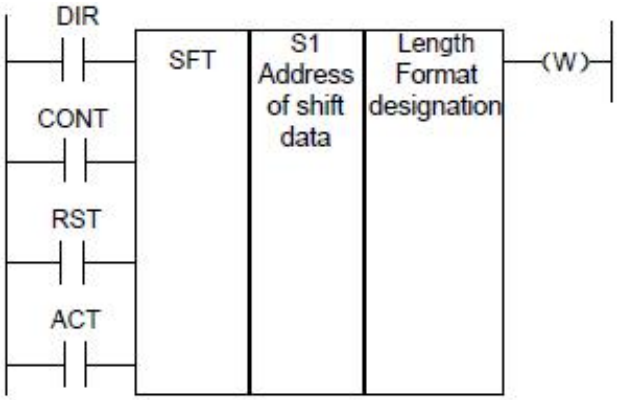


Fig. 5-20-1

Control conditions:

Shift direction specification (DIR)

DIR=0: Left shift

DIR=1: Right shift

Condition specification (CONT)

CONT=0: do not cycle shift

CONT=1: cycle shift

Reset (RST)
The shifted out data(W=1) is reset (W=0).
RST=0: W is not reset.
RST=1: W is reset (W=0).
Actuation signal (ACT)
ACT=0: do not execute SFT instruction.
ACT=1:shifting processing is done when ACT=1. For shifting one bit only, execute an
instruction when ACT=1, and then, set ACT to 0.
Parameter:
S1: set the big shift data address which consists of a storage area with one byte.
Length: a 4-digit number, its definition is shown below:

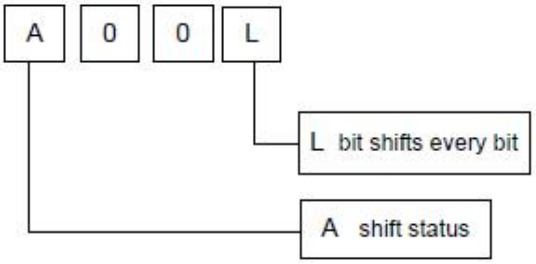


Fig. 5-20-2

L: range: 0~8.
A : bit parameter. A=0: When ACT=1 is shifting, the shift period is one bit.
A=1: ACT is taken as a pulse signal, it is 1 from 0, shift one bit.
Output:
W : W=0: "1" was not shifted out because of the shift operation.
W=1: "1" was shifted out because of the shift operation.

5.19 DIFU (Rising Edge Check)

Function:
The DIFU instruction sets the output signal to 1 for one scanning cycle on a rising edge of the
input signal.
Format:

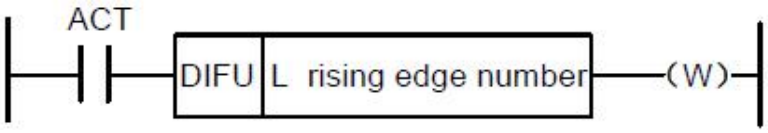


Fig. 5-21-1

Control condition:
Input signal: On a rising edge(0 -> 1)of the input signal, the output signal is set to 1.
Output signal: The output signal level remains at 1 for one scanning cycle of the ladder level
where this functional instruction is operating.
Parameter: Rising edge number
Parameter: L : rising edge number, range 0~255. Another DIFU instruction or DIFD

instruction , in the ladder uses the same number, an alarm occurs.
Operation:

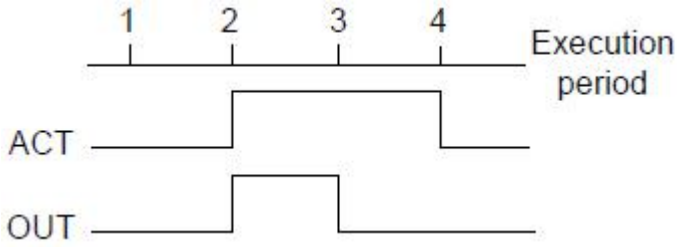


Fig. 5-21-2

The system will check the sequence number of the rising edge automatically, when the number exceeds the range or the number is duplicated, an alarm occurs.

5.20 DIFD (Falling Edge Check)

Function:

The DIFD instruction set the output signal to 1 for one scanning period on a falling edge of the input signal.

Format:

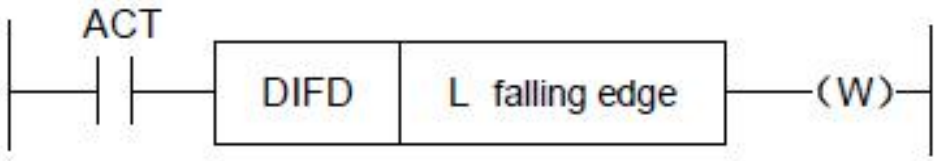


Fig. 5-22-1

Control conditions:

Input signal: on a falling edge（1 -> 0）of the input signal, the output signal is set to 1.

Output signal: the output signal level remains at 1 for one scanning period of the ladder level where this functional instruction is operating.

Parameter:

L : rising edge number, range 0~255. Another DIFU instruction or DIFD instruction in the ladder uses the same number, the system will alarm.

Operation

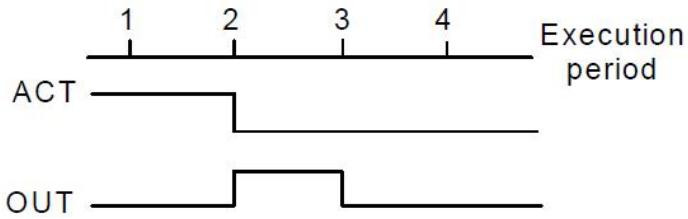


Fig. 5-22-2

The system checks the sequence number of the falling edge automatically, when the number exceeds the range or the number is duplicated, alarm occurs.

5.21 COMP (Binary Comparison)

Function:

Compares binary values. Specifies enough byte to store the input data and the comparison data in the memory.

Format:

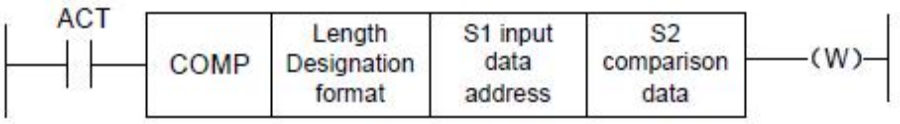


Fig. 5-23-1

Control conditions:

ACT＝0: The COMP instruction is not executed. W does not alter.

ACT＝1: The COMP instruction is executed.

Parameter:

Length: specification format(constant or address) and data length(1 or 2 bytes) for the input data.

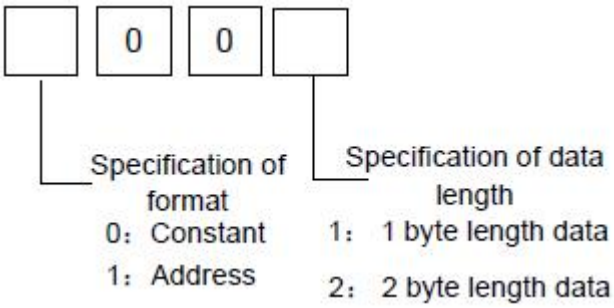


Fig. 5-23-2

S1, S2: content of comparison source 1 and comparison source 2. It can be constant and also be address number.

Address number: R, X, Y, F, G, K, A, D, T, C.

Output:

W=0: input data>comparison data

W=1: input data ≤comparison data

5.22 COIN (Coincidence Check)

Function:

Check whether the input value and comparison value coincide.

Format:

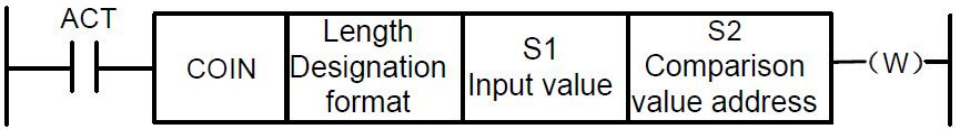


Fig. 5-24-1

Control conditions:

ACT＝0 ,the COIN instruction is not executed. W does not change.

ACT＝1 ,the COIN instruction is executed.

Parameter:

Length: specification format(constant or address) and data length(1 or 2 bytes)for input data.

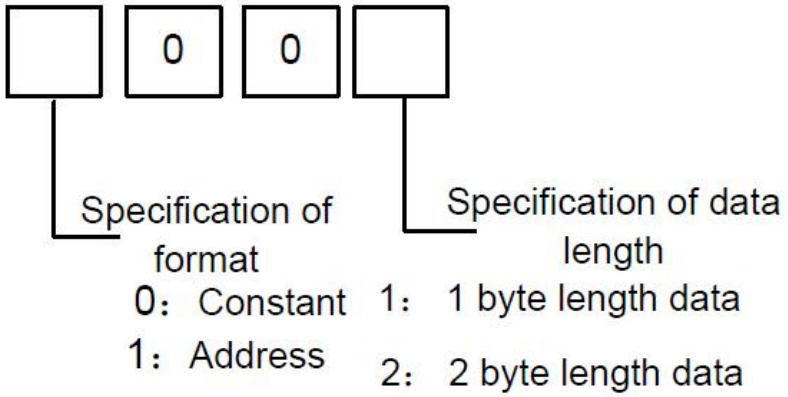


Fig. 5-24-2

S1 : The input data can be specified as either a constant or an address storing it.

S2 : Address storing of comparison data.

Output:

W : W=0: input value ≠comparison value

W=1: input value=comparison value

5.23 MOVN (Transfer of Data)

Function:

The MOVN instruction transfers data from source address and a specified binary data to a

specified destination address.

Format:

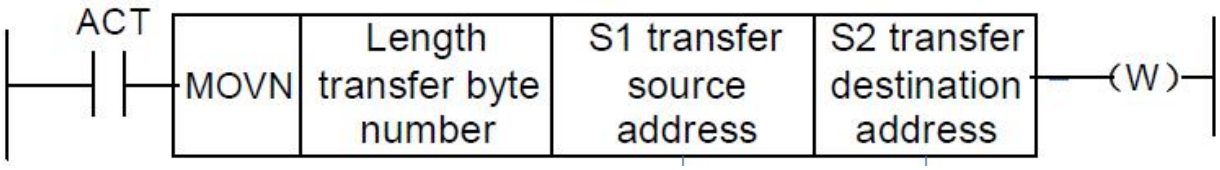


Fig. 5-25-1

Control condition:

ACT＝0: No data is transferred.

ACT＝1: The byte of specified number is transferred.

Parameter:

Length : Transferred byte n umber.

S1 : Stating byte of address or constant of source data.

Selecting transfer format according to S1:

1.S1 is constant: if S2 is single byte address, S1 in byte unit is copied to address corresponding to Length byte which takes S2 as the initial; if S2 is word unit, S2 in word unit is copied to the address corresponding to Length word which takes S2 as the initial.

2.S1 is address: S1 and S2 transmit the data in byte in spit if S1 and S2 address classifications are matched.

S2 : Starting byte of destination address.

Example:

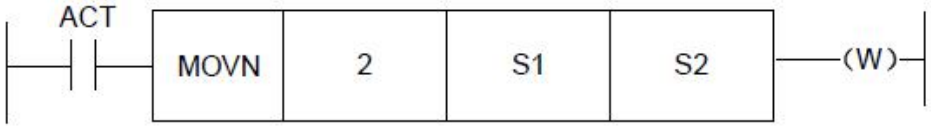


Fig. 5-25-2

1.When S1 is the constant ,5 , and S2 is R60, R60=00000101,R61=00000101

2.When S1 is the constant ,5 , and S2 is D60, D60=5, D6 1=5.

3.When S1 is the address , D50 , and S2 is D60, D60=50

W=1, the specified number byte is delivered.

W=0, no data be delivered.

If it detects that it exceeds the range of parameter type in transferring , an alarm occurs.

5.24 MOVB (Transfer of One Byte)

Function:

The MOVB instruction transfer one-byte data from a specified source address to a specified

destination address.

Format:

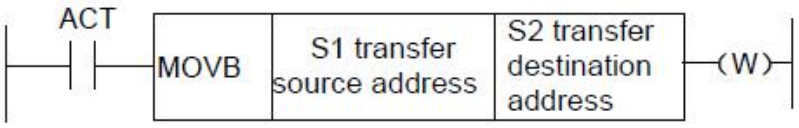


Fig. 5-26-1

Control conditions:

ACT＝0, No data is transferred.

ACT＝1, one-byte data is transferred.

Parameter:

S1: source address or constant.

when S2 is a single-byte address, S1 with byte value is copied to S2 address; when S2 is a word address, S1 with byte value is copied to S2 lower-byte address.

S2：destination address.

5.25 MOVW（Transfer of Two- Byte）

Function:

The MOVB instruction transfer two-byte data from a specified source address to a specified

destination address.

Format:

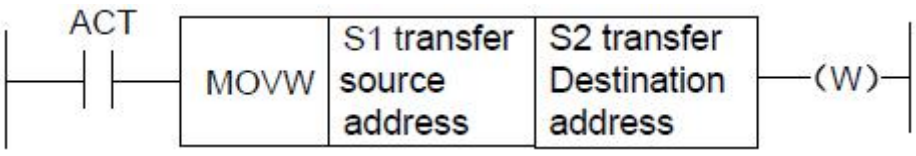


Fig. 5-27-1

Control conditions:

ACT＝0，no data is transferred.

ACT＝1，two-byte is transferred.

Parameter:

S1 ： source address or constant.

S2 ： destination address.

5.26 XMOV（Binary Index Modifier Data Transfer）

Function:

This function instruction instructs reading and rewriting of data in the data table. Number of data (table capacity) in the data table can be specified by specifying the address. In PLC run, perform the data table according to the user setting.

Format:

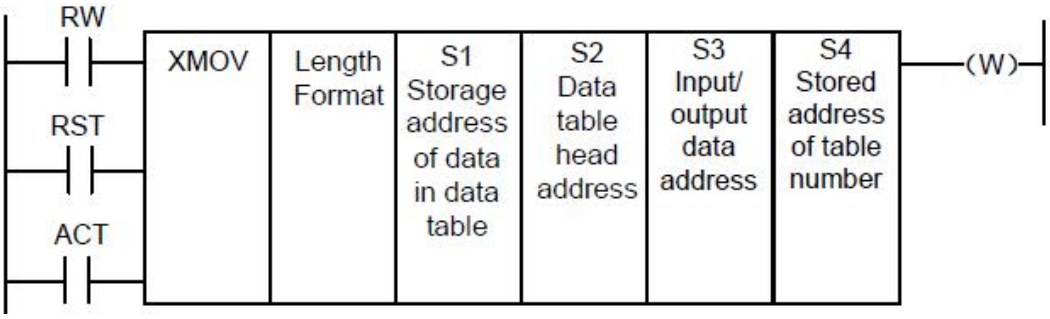


Fig. 5-28-1

Control conditions:

Read, write designation（RW）

RW=0：read data from data table.

RW=1：write data to data table.

Reset（RST）

RST=0： reset release.

RST=1： reset W=0.

Activation code(ACT)

ACT＝0 ： do not execute XMOV，and there is no change in W.

ACT＝1 ： execute XMOV code.

Parameter:

Length ： specifies the transferred data length.

1：1-byte long data.

2：2-byte long data

S1 ： data capacity’s stored address of the data table. The address is used to store the data

table’s data capacity. Its occupied byte quantity meets the length specified by Length, and the valid range of data is determined by the byte length specified by Length1 format.

1-byte length：1 to 512.

2-byte length：1 to 256. namely 256×2＝512 bytes，which is the PLC data table’s capacity.

S2 :set head address in the data table. The memory area of data table (byte length)

X number
of data table. The table head address must be the value set in D data table.

S3 : Input/output data address. In case of reading, set the address of memory which stores a reading result. In case of the writing, set the address of the memory which stores a writing result, its occupied byte number meets the setting of Length format. Limit the address to D register.

S4 ： Index storage address. It is used to read or write store an index value. Its occupied byte number meets the designation. When the set index address is m ore than the data stored in S1, error output W=1.

Actual transmission address=head address +index value ， index value is 0～(S1-1)，the actual transmission address cannot exceed the data table.

Output:

In the case where the index value exceeds the value set in S1, W=1， the reading or writing of the data table isn’t executed.

W=0，no error.

W=1，error found.

5.27 DSCH (Binary Data Search）

Function:

The DSCH instruction is used to search the binary data in data table. The number of data (table capacity) in the data table can be specified by specifying the address. Thus allowing change in table capacity even after writing the sequence program in the ROM.

Format:

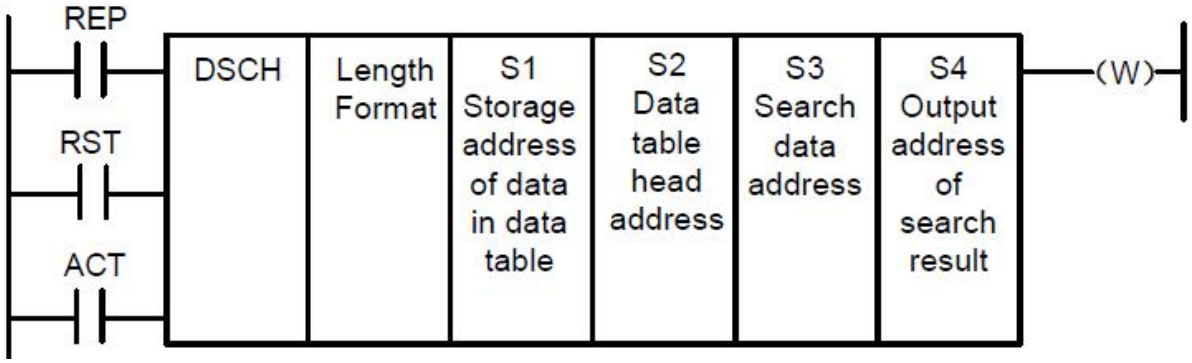


Fig. 5-29-1

Control conditions:

Check repetition（REP）

REP=0：Execute the DSCH instruction, search begins from the head address, and the repetition will be omitted, the search stops when the target data is found in the first time, and output its address. If the searched data is not found, W=1.

REP=1：Execute the DSCH instruction, if the searched data is not found or is two（or more than two）, W=1.

Reset（RST）RST=0：Release reset. RST=1：Reset. W=0.

Activation instruction (ACT):

ACT＝0 : Do not execute DSCH instruction, W does not change.

ACT＝1 : Execute DSCH instruction. If the search data is found, table number where the data is stored will be output. If the search data is not found, W be comes 1.

Parameter:

Length ： Specifies data length 1：1-byte long data.

2：2-byte long data.

S1 ：Storage address of number of data in data table. This address requires memory of number of byte according to the format designation. Number of data in the table is n+1(head number in the table is 0 and the last number is n).

S2 ： Data table head address.

S3 ： Search data input address.

S4 ： Search result output address. Actual transmission address=head address +index value，index value is 0～(S1-1)，the actual transmission address cannot exceed the data table.

After searching, if search data is found, the table number where the data is stored will be output. The searched table number is output in this search result output address. This address requires memory of number of byte according to the format designation.

Output:

W=0，Search data found.

W=1，Search data not found.

5.28 ADD（Binary Addition）

Function:

This instruction performs binary addition between 1-, 2-byte data. In the operation result register, operation data is set besides the numerical data representing the operation results. The required number of bytes is necessary to store each Augean, the added, and the operation output data.

Format:

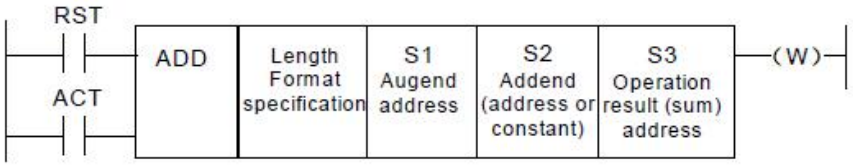


Fig. 5-30-1

Control conditions:

Reset (RST):

RST=0： Release reset.

RST=1： Reset. W=0.

Activation instruction (ACT)：

ACT＝0 ： Do not execute ADD. W does not changed.

ACT＝1 ： Execute ADD.

Parameter:

Length: Specifies data length(1 or 2 bytes) and the format for the addend(constant or address).

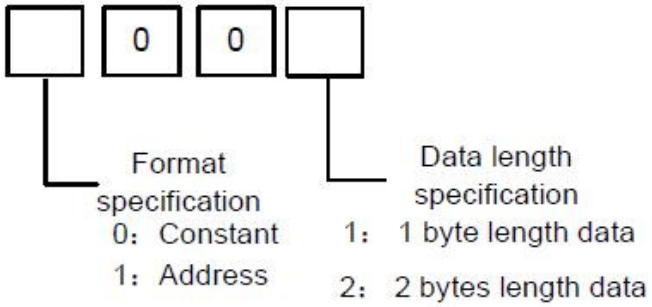


Fig. 5-30-2

S1 ： Address containing the augend

S2 ： Length specification determines the format of the addend.

S3 ： Specify the address to contain the result of output operation.

Output:

W=0： Operation correct.

W=1： Operation incorrect.

When W=1, the result of addition exceeds the specified data length.

5.29 SUB（Binary Subtraction）

Function:

This instruction executes the subtraction operation in the binary format of 1 or 2 byte s. In the operation result register, operation data is set besides the numerical data representing the operation. A required number of bytes is necessary to st ore the subtrahend, and the result.

Format:

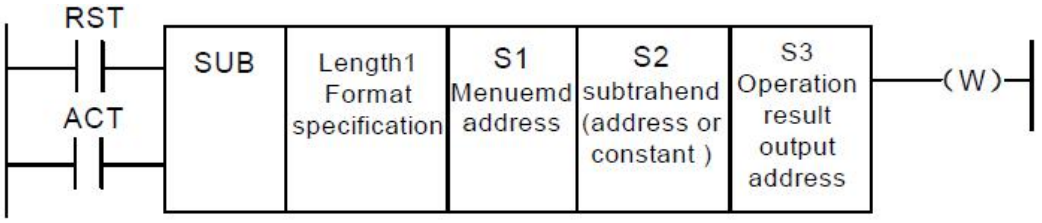


Fig. 5-31-1

Control conditions:

Reset (RST)：

RST= 0：Release reset.

RST= 1： Reset. W=0. Activation instruction (ACT)：

ACT＝0 ： Do not execute SUB. W does not change.

ACT＝ 1 ： Execute SUB.

Parameter:

Length :Specifies data length(1 or 2 bytes) and the format for the subtrahend(constant or address).

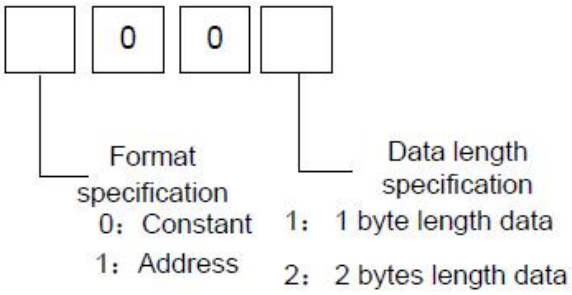


Fig. 5-31-2

S1 ： Address containing the minuend.

S2 ： Specification determines the Length.

S3 ： Specifies the a ddress to contain the result of operation.

Output:

W=0 ： Operation correct.

W=1 ： Operation incorrect.

When W =1, the result of subtraction exceeds the specified data length.

5.30 ANDF（Functional And）

Function:

The ANDF instruction ANDFs the contents of address A with a constant (or the contents of ad dress B), and stores the result at address C.

Format:

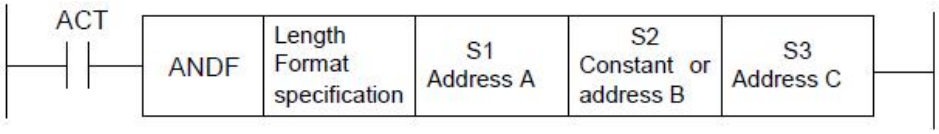


Fig. 5-32-1

Control conditions:

ACT＝0 ： The ANDF instruction is not executed.

ACT＝1 ： The AND F instruction is execute d.

Parameter:

Length ： Specify a data length (1 or 2 bytes), and an input data format (constant or address

specification).

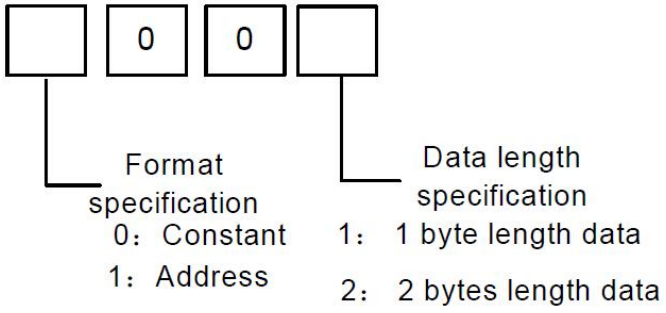


Fig. 5-32-2

S1 ： Input data to be ANDed. The data that is held starting at this address and has the data

length specified in Length format specification is treated as input data.

S2 ： Input data to be ANDed with. When address specification is selected in format

specification, the data that is held starting at this address and has the data length specified in Length format specification is treated as input data.

S3 ： Address used to store the result of an ANDF operation. The result of and ANDF operation is stored starting at this address, and has the data length specified in Length format specification.

Example:

When address A and address B hold the following data:

Addres s A	1	1	1	0	0	0	1	1

Addres s B	0	1	0	1	0	1	0	1

The result of the ANDF operation is as follows:

Addres s C	0	1	0	0	0	0	0	1

5.31 ORF（Functional OR）

Function:

The ORF instruction ORFs the contents of address A with a constant (or the contents of address B), and stores the result at address C.

Format:

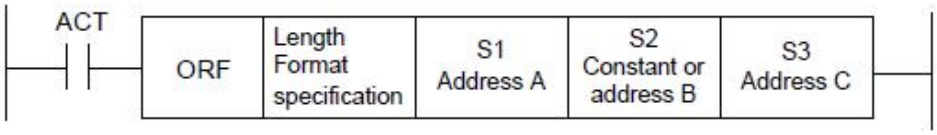


Fig. 5-33-1

Control conditions:

ACT＝0 ： The ORF instruction is not executed.

ACT＝1 ： The ORF instruction is executed.

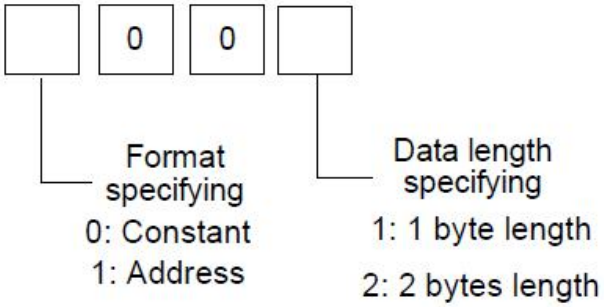


Fig. 5-33-2

Parameter:

Length: Specify a data length(1 or 2 bytes), and an input data format(constant or address specification).

S1 : Specify the input data t o ORed. The data that is held starting at this

address and has the data length specified in Length format specification is treated as input data.

S2 : Input data to be OR ed with. When address specification is selected in format

specification, the data that is held starting at t his address and has the data length

specified in Length for mat specification is treated as input data.

S3: Ad dress used to store the result of an ORF operation. Result of an ORF operation is stored star ting at this address, and has the data length specified in format specification.

Example:

When address A and address B hold the following data:

Address A	1	1	1	0	0	0	1	1
Addres s B	0	1	0	1	0	1	0	1

The result of the ANDF operation is as follows:

Addres s C	1	1	1	1	0	1	1	1

5.32 NOT（Logical Not）

Function:

The NOT instruction inverts each bit of the contents of address A, and stores the result at address B.

Format:

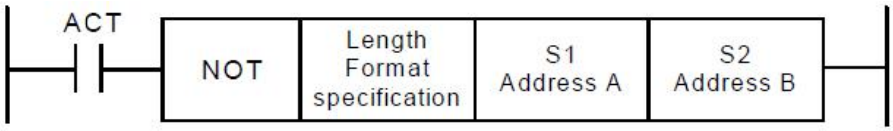


Fig. 5-34-1

Control condition:

ACT＝0: The NOT instruction is not executed.

ACT＝1: The NOT instruction is executed.

Parameter:

Length ： Specifies a data length (1 or 2 bytes).

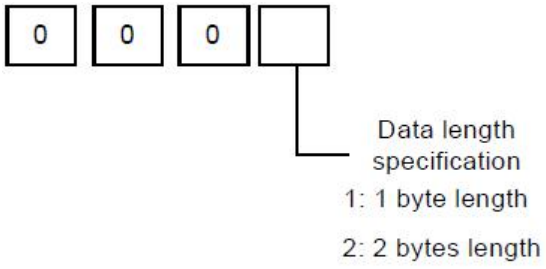


Fig. 5-34-2

S1 ： Input data to be inverted bit by bit. The data that is held starting at this address an d has the data length specified in Length format specification is treated as input data.

S2 ： Address used to output the result of a NOT operation. The result of a NOT operation is stored starting at this address. An d has the data length specified in Length format specification.

Example:

When address A holds the following data:

Address A	1	1	1	0	0	0	1	1

The result of the NOT operation is as follows:

Address B	0	0	0	1	1	1	0	0

5.33 EOR（Exclusive OR）

Function:

The EOR instruction exclusive-ORs the contents of address A with a constant (or the contents of address B), and stores the result at address C.

Format:

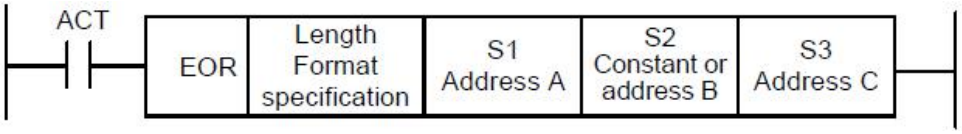


Fig. 5-35-1

Control conditions:

ACT＝0 ： The EOR instruction is not executed.

ACT＝1 ： The EOR instruction is executed.

Parameter:

Length ：Specify a data length (1 or 2 bytes) and an input data format(constant or address specification).

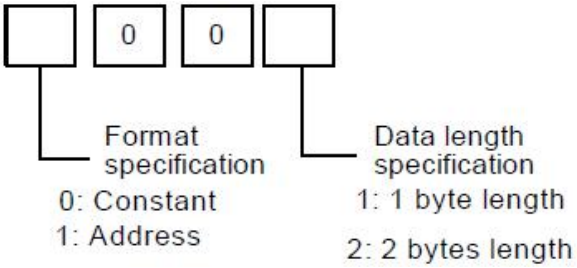


Fig. 5-35-2

S1： Input data to be exclusive-ORed. The data that is held starting at this address and has the data length specified in Length format specification is treated as input data.

S2： Input data to be exclusive-ORed with. When address specification is selected report that specification, the data that is held starting at this address and has the data length specified in Length format specification is treated as input data.

S3： Address used to store the result of an exclusive EOR operation. The result of an exclusive EOR operation is stored starting at this address, and has the data length specified in Length format specification.

Example:

When address A and address B hold the following data:

Address A	1	1	1	0	0	0	1	1

Addres s B	0	1	0	1	0	1	0	1

The result of the ANDF operation is as follows:

Addres s C	1	0	1	1	0	1	1	0

Chapter 6 Ladder Writing Limit

1. Sequence program must have END1 and END2 which are taken as the end marks of 1st level and 2nd level sequence part, and END1 must be before END2.

2. They only support the parallel output and do not support the multi-level output.

3. The result output address in all basic instructions and output function instruction are not set the following addresses:

1）Counter preset address DC, timer preset address DT.

2）K0~K5 address are occupied by the system, and the user can’t define them.

3）G63, R510, R511 address are occupied by the system. and the user can’t define them.

4）X, F address on IO input window.

4. Such case like vertical line overhanging, node disconnected, horizontal through line paralleling to the node network will result in the nodes or network that can' t be executed, so alarm will be issued by the system.

5. Star network, in which there is no direct connection between the vertical lines of different lines in a column, and a line in the middle isn't jointed with a vertical line. So alarm will be issued because the case can't be processed by the system.

6. The upward convex is not allowed in the network. That is there is a parallel network above he nodes of a line, and no line can be connected to this network. So alarm will be issued

The followings are the phrasing error, and an alarm occurs.

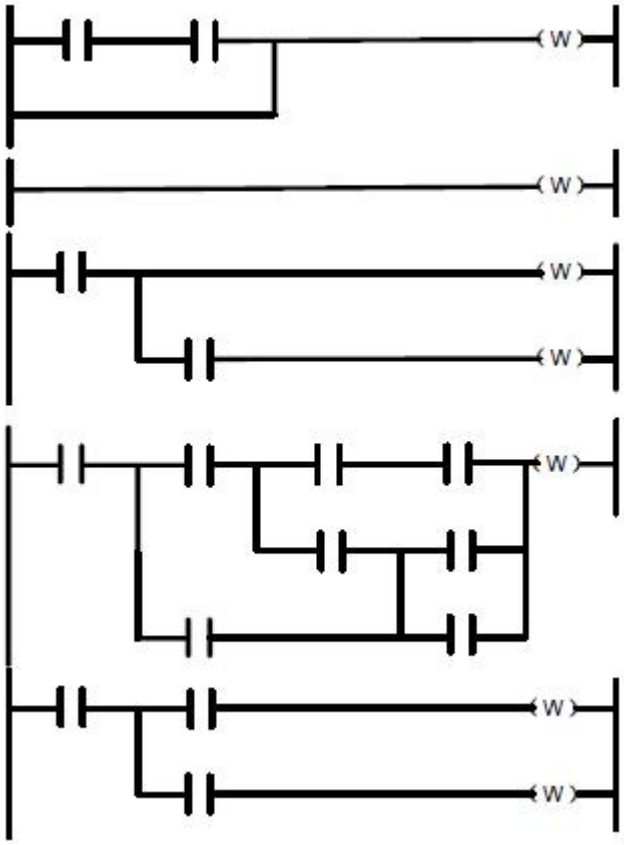


Fig. 6-1

Chapter 7 PLC Window Display

7.1 Automatic Operation When PLC Power ON

PLC starts after power on: it will make use of R510.0 to conduct the net contain it during the first period, and then set R510.0 to “0” which can't be output, the value of the keep relay gets from the PLC last output before stop running.

Note: The keys during < > are the panel; the ones in【】are the right menu title; ⊞reports there is the sub-menu in the menu; all operations in PCL are executed during MDI mode and only view and search can be executed during other modes.

7.2 INFO Window Display

7.2.1 PLC INFO Window

1. Press PLC key on panel /F1 Rapid Function key on after press PLC key to enter PLC info window as Fig.7-1 display

Fig. 7-2-1

2.Press “Page Down” on PLC info window, it will show as following:

Fig. 7-2-2

(1) When the system is turned on, current running plc ladder is set by parameter. After the system determines to load some ladder which is running (the operation maybe appear the danger, it is enabled after the system restarts). If the format is incorrect, the ladder is deleted to recreated, and the user needs to specify the running ladder No. carefully. Names of all ladder files must be with “ladderXX.grp” (XX is the serial number) , otherwise, the system does not identify the files. The file format is determined by the system, and the user cannot modify the file outside of the system, otherwise, the file maybe be deleted or cannot be identified

(2) Selecting ladder. Move the cursor or input “LX”/“LXX”（X/XX is number）to specify the file name, the system checks whether “X”/“XX”is the known file number after “Enter” is pressed, if the system has not checked it, it creates an ladder with the name “ladder0X.grp” or “ladderXX.grp” . The system automatically creates “END 1” and “END2” when the file is created, which can make the user continuously operate the ladder file (it cannot be switched after the file is opened , and the instruction list is always empty). To get safety, after one ladder is opened to executed edit and when it is saved and an other file is opened, the system automatically saves the current file. Before the system saves the file, it executes the ladder syntax check. When the system finds the syntax errors, it does not save the ladder diagram.

(3)The file head includes the basic information of file, such as row number, step number. The step information is the new one when it is converted. The user can delete the ladder which is not opened and is not running, which must be executed orderly. After the user opens the ladder which is not running, the system stops refreshing the ladder network in formation to avoid the mistake. The running ladder can execute only the two operations including save and copy, in order that when the user copies the file to other ladder files , and edits the currently running ladder, the user firstly interrupts its running state. When the cursor stops in the background edit file the user can press “Change” key to open Info to modify the file ground (including ladder version number, adaptive machine tool, ladder maintain personnel).

7.2.2 PLCGRA Window

Press PLC/F2 Rapid Function key on PLC info window, it will show as following:

Fig. 7-2-2-1

Contents and operations on PLCGRA window :

PLCGRA[l adder01]： current ladder name.

1/1173：current line position specified by the cursor during the ladder the cursor.

RUN: operation state o f ladder, including RUN /run, STOP/stop, DEB

UG/debug. Diagram: ladder program.

DATA: Display input data. Press <P>key, it will shift to sequence number.

Input：display the input data. Press ↓ key in the panel to view the input data.

MEA：Commentaries of element positioned by the cursor.

MDI mode: current ope ration mode

Press the Page Up/Page Down, four Direction keys to search, view and modify the elements .

7.2.3 PLCPAR Window

Press PLC/F3 Rapid Function key on PLC info window, it can enter PLC Parameter screen, as following shown:

Fig. 7-2-3-1

Contents and operations on PLCPAR window:

RUN ： operation state of ladder

ADDR： keep relay address

Bit0~Bit7 ： bit status of keep relay address

1：the address maintains the state before power OFF

0：the address resets to default state after power OFF

Input: display the input data.

MDI mode: current operation mode(note: the relative parameter of PLCPAR can be modified only in MDI mode).

Press the Page Up/Page Down, four Direction keys to search, view and modify the elements.

7.2.4 PLCDGN Window

Press PLC/F4 Rapid Function key on PLC info window, it can enter PLC Diagnosis screen, as following shown:

Fig. 7-2-4-1

Contents and operations on PLCDGN window :

RUN：operation state of ladder.

ADDR：address of diagnosis number.

Bit0~Bit7：bit number state of diagnosis address.

1：the signal is connected；0：the signal is not connected. Input: display the input data.

MDI mode: current operation mode.

Press the Page Up/Page Down, four Direction keys to search the corresponding diagnosis number. Generally, only the searching can be operated in the window, the I/O window of P LC enters t he signal debug mode only when the user gets the authority to set K0.1 to 1. At the moment, the user can modify the X, Y signal.

7.2.5 PLC Trace Window

Press F4 Rapid Function key,and then press Right key on PLCDGN window, and then press F3 rapid function key ,which can enter PLC Trace screen, as following

shown:

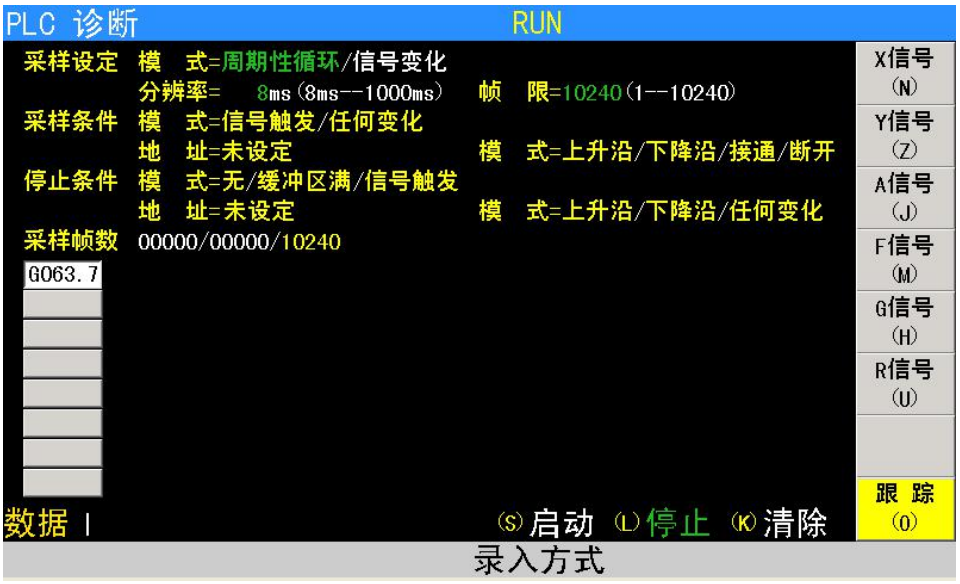


Fig. 7-2-5-1

The content and operation of the P LCTRA “SETTING” window

(1)Mode:

-- Periodic cycle: periodic sampling for each time.

-- Signal change: sampling for the change of current signal.

(2)Resolution:

Input the sampling resolution, the default value is the least resolution (8ms) , its range is (8ms --1000ms).

The input value uses t he multiple of the 8ms.

(3) Time limit:

When the sampling mode is set to “periodic cycle ”, then display this parameter. Input the performance time of the trace. The numerical range of the “periodic cycle” is determined by the value of the “ resolution ” or the specified signal address quantity, and its range is displayed at the right side.

(4) Frame limit:

When the sample m ode is set to “signal change”, then display this parameter. Input the sampling quantity, and its range is displayed a t the right side.

(5) Stop condition:

- - without: Do not stop tracing.

- - Buffer area full: It stops tracing when the buffer area is full.

- - Signal trigger: It stops tracing by the signal trigger.

Trigger setting: This parameter is enabled when the “stop condition” is selected to the “signal trigger”.

1. Address: The input signal address is regarded as trigger stopping. (R address can not be

used for trigger stopping)

2. Mode; It determines that what kind trigger mode is used to stop tracing.

Rising edge: The tracing is automatically stopped by the rising edge of the trigger signal.

Falling edge: The tracing is automatically stopped by the falling edge of the trigger signal.

Any change: The tracing is automatically stopped by the rising or falling edge of the trigger signal.

(6) Sampling condition: This parameter is enabled when the sampling mode is set to “signal change”,which is determined the sampling condition.

- - Signal trigger: The specified mode changes when the signal specified by the trigger address which is set by the sampling condition, collect the signal.

- - Any change: Any change occurs when the signal specified by the trigger address which is set by the sampling condition.

Trigger setting: When the sampling mode is set to “signal change”, and then the sampling condition is set to “signal trigger”, this parameter is enabled.

1. Address: The input signal address, instead of using the R address, is treated as the sampling of the trigger signal.

2. Mode: The trigger mode inputs the specified trigger signal.

Rising edge: The rising edge sampling of the trigger signal specifies the signal state.

Falling edge: The falling edge sampling of the trigger signal specifies the signal state.

Any change: Specify the signal state by the rising or falling edge sampling of the trigger signal.

Switch on: Sample the specified signal state when the trigger signal is switched on.

Switch off: Sample the specified signal state when the trigger signal is switched off.

(1)Sampling mode: Display the current sampling mode of the system.

(2)Period: Display the current sampling period of the system, that is, resolution

(3)Time: This parameter displays when the “sampling mode” selects the “periodic cycle”.

-- Format display when tracing: the current timing is at the left side, and the max. allowance timing is at the right side.

-- Format display when stopping: the most right side timing is placed at the right side; the timing of trace stopping is placed at the middle side, and the max. allowance timing is placed at the right side.

(4) Setting address: Move the cursor by the  and  , the signal address that will be traced is inputted inside the , it can be trace d 15 signals at the same time. Any address can be inputted. As for the R address, the previous 3 positions can be inputted the address before 256; the 4t h and 5gh position can be inputted 2 addresses after 255.

(5) (S) start: The signal trace can be performed pressing <S> key on the panel after the trace parameter is set correctly.

(T) Stop: Stop the signal trace after controlling the <T> key on the panel.

(K) Clear: the value under the cursor pressing the <K> key on the panel.

Chapter 8 PLC Programming Operation

8.1 General

Operations of GHPLC are completed in its corresponding window. All modifying the ladder can be executed after getting the authority above the system debugging level.

Operations are completed in two main windows:

1)PLC Ladder Graphic Window: Press the soft key [PLC] twice or press [PLC] & [F2] to enter PLCGRA classification window as Fig.8-1-1： PLCGRA window includes: basic codes, functional codes, instruction table and edit command.

Fig. 8-1-1

2)PLC Ladder Parameter Window: Press the soft key [PLC] or [F3] to enter PLCPAR window as Fig8-1-2, PLCPAR window includes: KPAR, TMR, DATA, CTR, MDEC.Modifying the parameter, PLC running control, entering I/O debug mode can be executed after getting the authority above the system debugging level.

Fig. 8-1-2

8.2 Basic Code

Press [F1] key under Fig. 8-1-1 window to enter the basic instruction operation window as Fig. 8-2-1.

Fig. 8-2-1

Basic codes are divided into 7 kinds of graph to display:

【 ⊣⊢ 】 ： norm ally open contact

【 ⊣/⊢ 】 ： norm ally-closed contact

【 -( )- 】 ： output coil

【—o( ) 】:output coil reverse

【—— 】: level conduct line

【 .....| 】: vertical conduct line

【 ⋰| 】: delete vertical conduct line

Auxiliary soft key:

【 ▤ 】 ： Page up page

【 ▤ 】 ： Page down page

【Return 】 ： return to the previous menu

8.3 Ladder Operation Explanation

Adding an element: position the cursor to the required position, press the corresponding menu to input the element name, press <EOB> to confirm the addition after it is displayed behind the data. If the current position has element, the new element replaces the previous one.

Inserting an element: position the cursor to the required, press <INSERT> to insert an empty position, and then add the new element as the above method. The cursor can insert orderly.

Deleting an element: press to delete the current element

Adding a vertical conductive line: press 【 ⋰| 】soft key to add one vertical

conductive line under the lower-right of current cursor position.

Deleting a vertical conductive line: press 【 】 to delete one vertical conductive line under the lower-right of current cursor position.

Adding a horizontal conductive line: press 【——】 to add one horizontal conductive line before the cursor position, if the current position has element, the horizontal conductive line replace the element.

Inserting a line: position the cursor to the any line of target line, press , and then press , insert the blank line at the place above of the specified line by cursor, and the sequence line will orderly move down one line.

Deleting a line: position the cursor to the target line, press , and then press to delete the current line, and the sequence line will orderly move up one line.

Deleting a block: position the cursor to the initial position which will be deleted, Input the address number of target block’s coil, and last press <Delete>

Search: directly input the required element name, press【 】 to search up and press 【 】 to search down after the data on screen is displayed

Ladder programming example:

1.Position the cursor to the initial position of programming, press 【 -| |- 】 soft key and there is norm ally-open contact symbol at the cursor position, directly input the element name X0.5 and press <EOB> and X000.5 appears at the current cursor position.

2.Right move the cursor, press 【 -|/|- 】, and there is there is normally-open contact symbol at the cursor position, directly input the element name F021.3 and press < EOB> and F021.3 appears at the current cursor position.

3.Position the cursor to the initial position of next line, press【 -| |- 】soft key, there is there is norm ally-open contact symbol at the cursor position, directly input the element name G008.4 and press <EOB> and G008.4 appears at the current cursor position.Press【 -|/|- 】, and there is there is normally-open contact symbol at the cursor position, directly input the element name F001.1 and press < EOB> and F001.1 appears at the current cursor position.

4.Right move the cursor, press 【——】, and draw a horizontal conductive line at the current cursor position.

5.Move up the cursor, press 【 】,and draw a vertical conductive line at right-down position of current cursor.

6.Press 【 —( ) 】an the system automatic create the output coil, namely the necessary horizontal conductive line. Directly input the element name G8.4, press <EOB>and G008.4 appears at the current cursor position.

The programmed ladder is as Fig. 2-3-1:

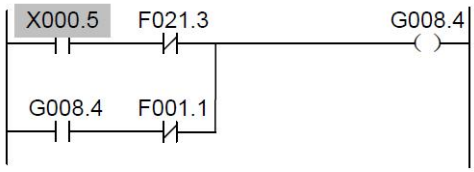


Fig. 2-3-1 Ladder example

Note: The green element in the ladder is turned on no matter that it is normally-open and normally-closed or outputs the coil, and the white indicates it is turned off (owing to the printing, the dark stands it i s turned off, and the light stands it is turned on.)

8.4 Function Code

Press【FUNC】in Fig. 8-1-1 to enter the function instruction operation window as Fig. 8-4-1.

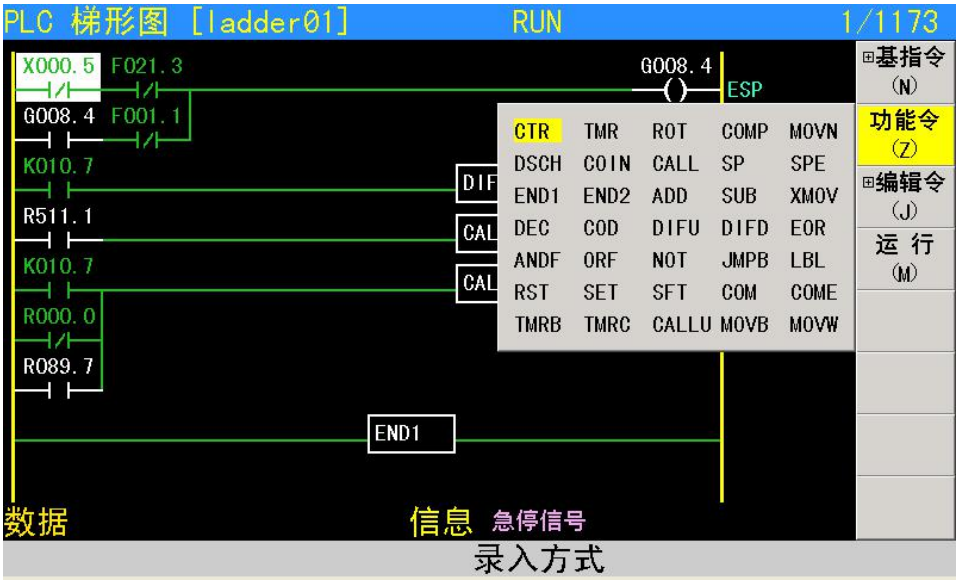


Fig. 8-4-1

There are 35 PLC function instructions in the function instruction list. For the format and use of function instruction, see Programming, and the operation compiling of functional codes are consistent with those of Section 2.3 Ladder Operation.

8.5 Instruction List

Press【PLCINS】in Fig. 8-1-1 to enter the function instruction operation window as Fig. 8-5-1.

Fig. 8-5-1

Inserting a line: position the cursor to the any line of target line, press <Insert>,

insert the blank line at the place above of the specified line by cursor, and the sequence line will orderly move down one line.

Deleting a line: position the cursor to the target line, press <Delete> , and then press to delete the current line, and the sequence line will orderly move up one line.

8.6 PLC Run/STOP

【STOP】: PLC Stop
【RUN】: PLC Run

8.7 PLC Operation Step

PLC operation steps:

1. Press<Setting> key and input the machine manufacturer level password in 【PASSWORD】window.
2. press<PLC>key, then press 【KPAR】in 【PLCPAR】 window to enter the viewing and setting window of keep relay, operate PLC by modifying the related bit of K000, K001. (Example: setting K000.7 to 1 can modify the ladder after saving). see Appendix K INSTRUCTION List in chapter four Connection for the definition of related bit
3. In 【PLCGRA】window, press 【INSTRUCTION】to press【STOP】, the system stops the running ladder(if the modified ladder is not the current, the step can be omitted).
4. In【PLCGRA】window, complete PLC programming by executing【B. INST】, 【FUNCTION INST】,【EDIT INST】 etc . Press <SAVE>key, the data field prompts “ Ladder Saved!”, it means the saving is successful. The corresponding PLC alarms during saving when PLC is mistaken, please check the PLC program.
5. In 【 PLCGRA 】 window, press <CHANGE> on the panel then the data field will prompt Ⅱ “CHANGING……”, and prompts “CHANGE SUCCEEDED!” after the change finishes.
6. In【PLCGRA】window, click <INSTRUCTION LIST>, then click 【DOWNLOAD】 , and the data column prompts "DOWNLOADING……", and "DOWNLOAD SUCCEEDED!" after the download finishes. The ladder is changed into instruction list and downloaded to CNC, then automatically operates.

Chapter 9 PLC Address, Parameter Setting

The addresses and parameters, such as the counter, timer, data list and nonvolatile relay may be used in the PLC; the viewing and setting of these addresses and parameter should be performed in the corresponding window. Press the [PLCPAR] soft key again in the PLCPAR window, then enter the PLC address and parameter setting windows, refer to the Fig. 9-1, which includes the nonvolatile relay, timer, data list, counter, F address corresponding with the M function. It is used for checking and setting these addresses, parameters and data list. (User can set it after the debugging password is input and gained an authority).

Fig. 9-1

9.1 Nonvolatile/Hold Relay

Press the [KPAR] soft key in the Fig. 9-1, then enter the checking and setting windows of the nonvolatile replay, refer to the Fig. 9-1-1.

Fig. 9-1-1

The content and operation of the nonvolatile replay window:

ADDR: Nonvolatile relay address.

Bit0~Bit7: Bit number state of the nonvolatile replay address.

1: This address remain s the state before pow er off after t he power is turned off;

0: This address resets on default state after the power is turned off.

After modification, press <Download> key on the panel to download the set value to the CNC run. After the save is successful, the system displays “KPAR downloading successful”; the system displays: “downloading fail” when the save is incorrect; the “illegal downloading parameter” displays without downloading conditions.

The search and positioning can be performed by Page-Up, Page-Down and four direction keys on the panel; checking and modifying of the nonvolatile relay address can be performed.

9.2 Timer

Press the [TMR] soft key in the Fig. 9-1, the n enter the checking and setting windows of the timer, refer to the Fig. 9-2-1.

Fig. 9-2-1

NO: Timer serial number; Do not change.

ADDR.: Timer address; Do not change.

CURT: Current value of the timer; Do not change.

SET: Presetting value of the timer; it can be changed after K000.0 (PLC parameter modification permission) is set to 1.

After modification, press <Download> key on the panel to download the set value to the C NC run. After the save is successful, the system displays “TMR downloading successful”; the system displays: “downloading fail” when the save is incorrect; the “illegal downloading parameter” display s without downloading conditions.

The search and positioning can be performed by Page-Up, Page-Down and four direction keys on the panel; checking and modifying of the nonvolatile relay address can be performed.

9.3 Data List

Press the [DATA] soft key in the Fig. 9-1, then enter the checking and setting window of the data list, refer to the Fig. 9-3-1.

Fig. 9-3-1

The content and operation of the nonvolatile replay window:

ADDR: Data address,cannot alter.

DATA: Setting value of data table, can alter.

After modification, press <Download> key on the panel to download the set value to the C NC run. After the save is successful, the system displays “TMR downloading successful”; the system displays: “downloading fail” when the save is incorrect; the “illegal downloading parameter” display s without downloading conditions.

The search and positioning can be performed by Page-Up, Page-Down and four direction keys on the panel; checking and modifying of the nonvolatile relay address can be performed.

9.4 Counter

Press the [CTR] soft key in the Fig. 9-1, then enter the checking and setting window of the counter, refer to the Fig. 9-4-1.

Fig. 9-4-1

ADDR.: Counter address; Do not change.

CURT: Current value of the timer; Do not change.

SET: Presetting value of the timer; it can be changed after K000.0 (PLC parameter modification permission) is set to 1.

After modification, press <Download> key on the panel to download the set value to the C NC run. After the save is successful, the system displays “TMR downloading successful”; the system displays: “downloading fail” when the save is incorrect; the “illegal downloading parameter” display s without downloading conditions.

The search and positioning can be performed by Page-Up, Page-Down and four direction keys on the panel; checking and modifying of the nonvolatile relay address can be performed.

Chapter 10 Ladder Edit Software

10.1 Summary

At present, GH series CNC system supports the compilation software of the configured GSK ladder diagram.

The compilation software of GH series ladder diagram is a ladder diagram editor on the PC machine of the GH series CNC system, which mainly offers the functions such as the edit, conversion, debugging and printing of the GH series ladder diagram. This software can be used in the Windows 98, Windows Me, Windows 2000, Windows XP and Windows 2003.

10.2 Software Introduction

10.2.1 Software Start

The compilation software of the GH ladder diagram is a green one without being installed,which can run software directly by press Ladder_V2.0. Exe twice. The Ladder01 file is the standard ladder diagram of the system. When the Ladder01 ladder diagram in the LadFile folder is opened in the software, the window is shown below:

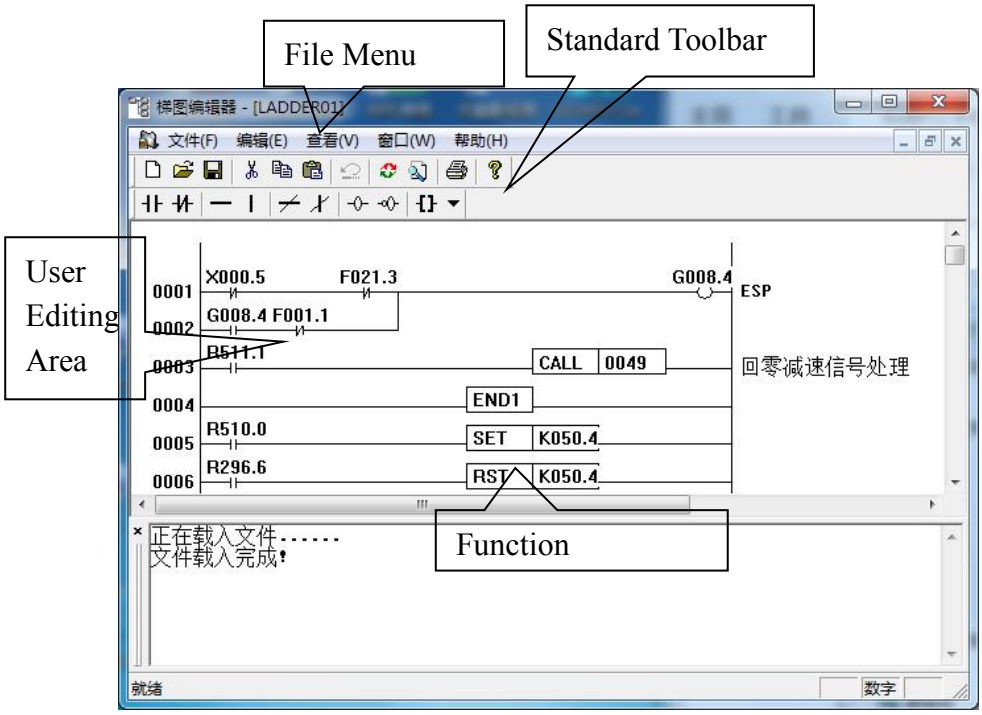


Fig. 10-2-1

10.2.2 Function Introduction

- **File menu**

The file menu includes some program files, namely, the new, open and save, which can be produced some functions, such as the performed ladder diagram file or binary system file, printing, printing preview and printing setup and the recently opened file list.

Note: In the "ladder diagram editing" dialog box, each volume of the "ladder diagram

version number", "suitable machine" and "ultimate modifier", can be indicated by English, instead of Chinese, otherwise, the error may occur after transferring.

■ **Edit menu**

The edit menu includes some functions such as the cutting, copy, pasting, searching, conversion and editing etc.

- View menu

Control the display and concealing of the tool bar, state bar, output and instruction list windows.

- Window menu

Control the selection and layout of each window.

10.3 Software Operation

10.3.1Toolbar

There are two toolbar of the main view frame, which are related with the ladder diagram compilation.

10.3.1.1 Main Toolbar

 New ladder diagram file

 Open the ladder diagram file

 Save the ladder diagram file

 Cut the selected content to the clipboard

 Copy the selected content to the clipboard

 Paste content from the clipboard

 Ladder diagram conversion

 Component search

 Print the ladder

 About the dialog box

10.3.1.2 Editing Toolbar

 Insert the normally opened contact

Insert the normally closed contact

Insert the horizontal break-over line

Insert the vertical break-over line (place at lower right corner of cursor)

Delete single cell or horizontal break-over line

Delete the vertical break-over line at lower right corner of component

Insert the in put coil

Insert the output coil reverse

Function code button: There are two methods in Edit function code:

1. Pop up the drawing menu by click the small arrow at the right, and then select the function codes.

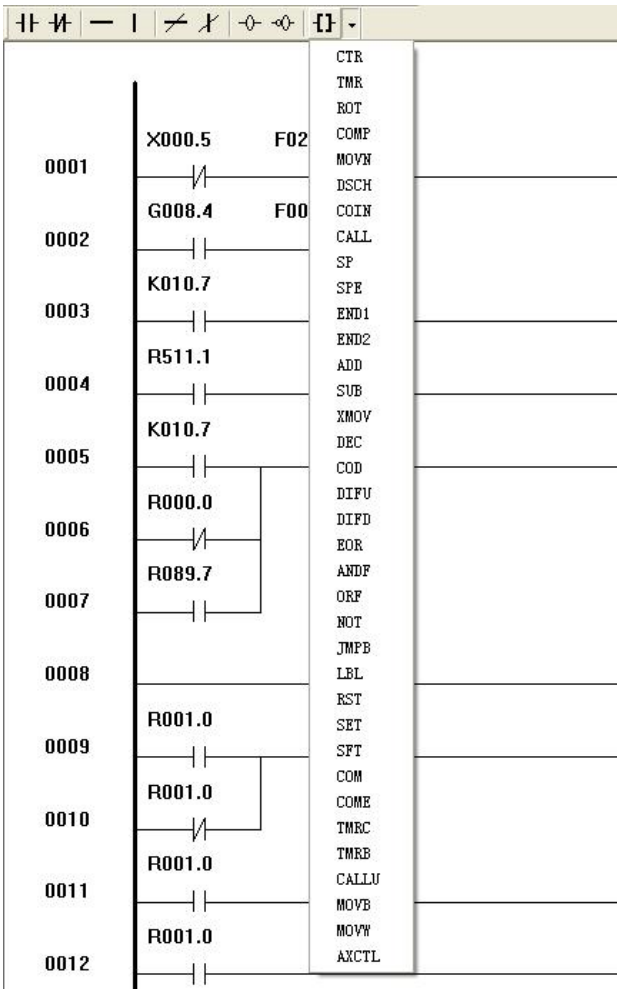


Fig. 10-3-1-2-1

2. Or click the button icon, set the function code in the function code selection dialog.

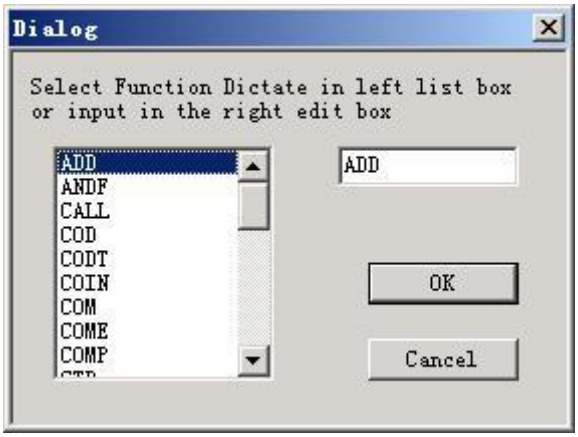


Fig. 10-3-2-1-2

10.3.2 Selecting a Figure

In the editing view of the ladder diagram, the black rectangle shadow means cursor, click the left key of the mouse in the figure editing area between two bus cables, and select the position where the figure unit needs to be edited. Refer to the following figure:

Fig.10-3-2-1

When the b lock is selected, press the mouse left key at the beginning position of the block, then drag to the end. The selected area indicates by the rectangle with dotted line before re leasing the left key.

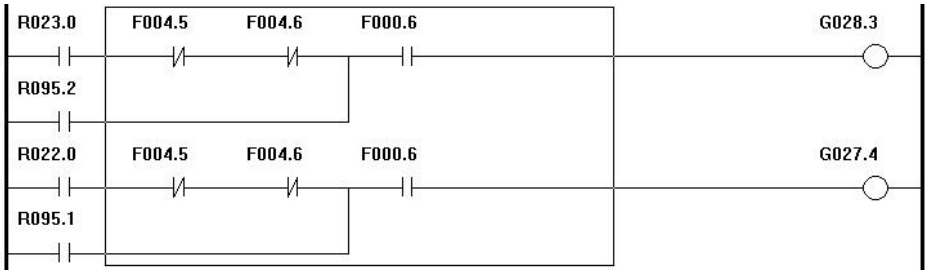


Fig. 10-3-2-2

The inverted color of t he whole ladder diagram after releasing, that is, the ladder diagram within this rage is selected, and the next operation can be perform ed. For example, cutting, deletion and copy etc..

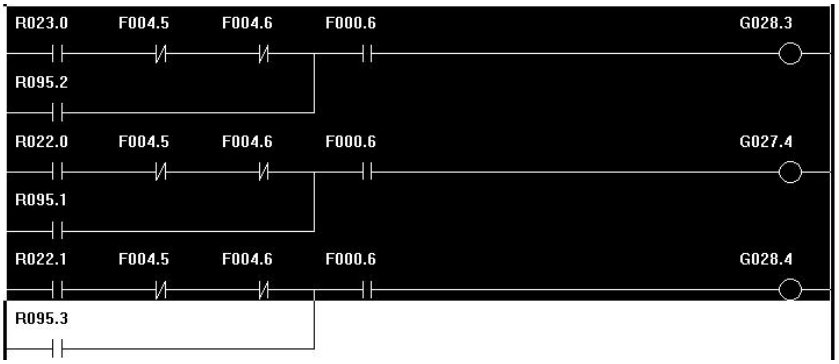


Fig. 10-3-2-3

10.3.3 Editing Figure

10.3.3.1 Cutting

There are three ways for carrying out this operation after the ladder diagram area to be edited is selected:

1. Select the cutting after springing the environment menu by clicking the right key of the mouse;

2.Select the Edit [Alt+E]--- Cutting [T] of the main menu;

3.Shortcut key [Ctrl+X].

The cut content is placed to the clipboard, which is copied to the ladder diagram by t he paste operation.

10.3.3.2 Copy

There are three ways for carrying out this operation aft er the ladder diagram area to be copied is selected:

1. Select the copy after springing out the environment menu by clicking the right key of the mouse;

2.Select the Edit [Alt+E]--- Copy [C] of the ma in menu;

3.Shortcut key [Ctrl+C].

The select ed content after copying is put to the clipboard, which is copied to the ladder diagram by the paste operation.

10.3.3.3 Pasting

There are three ways for carrying out this operation after the ladder diagram area to be pasted is selected:

1.Select the pasting after springing out the environment menu by clicking the right key of the mouse;

2.Select the Edit [Alt+E]--- Pasting [P] of the main menu;

3.Shortcut key [Ctrl+V].

10.3.3.4 Deletion

There are three ways for carrying out this operation after the ladder diagram area to be deleted is selected:

1.Select the basis code ---- Deletion node after springing out the environment menu by clicking the right key of the mouse once;

2.Click the [Deletion node] button on the editing bar;

3.Shortcut key [Delete];

10.3.3.5 Insertion Line

There are three ways for carrying out this operation after moving the cursor to the position to be inserted the ladder diagram line:

1.Select the insert after springing out the environment menu by clicking the right key of the mouse;

2.Select the Edit [Alt+E]---Insertion line [I] of the main menu;

3.Shortcut key [Insert];

10.3.3.6 Deletion Line

There are three ways for carrying out this operation after moving the cursor to the position to be deleted the ladder diagram line:

1.Select the insert after springing out the environment menu by clicking the right key of the mouse;

2.Select the Edit [Alt+E]--- Deletion line [D] of the main menu;

3.Shortcut key [Ctrl+Delete];

10.3.3.7 Conversion

There are three ways for carrying out this operation after the ladder diagram of the current editing window is converted into the instruction list program:

1.Select the Edit [Alt+E]--- Conversion [V] of the main menu;

2.Click once the [Ladder diagram conversion button on the editing bar;

3.Shortcut key [F7];

10.3.4 Ladder Diagram Note

10.3.4.1 The Line Note of the Ladder Diagram

Click the left key of the mouse twice out of the bus area at the ladder diagram right; input the notes in the editing frame.

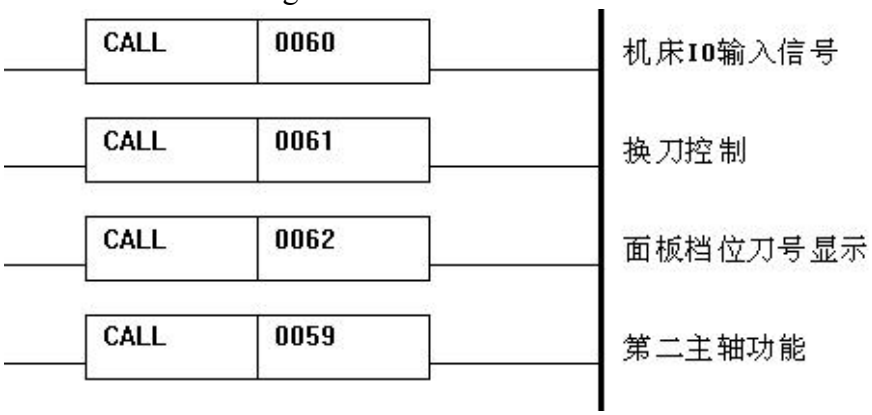


Fig. 10-3-4-1-1

10.3.4.2 Notes of the Ladder Diagram’s Component

There are two ways for carrying out this operation after moving the cursor to the position to be modified the ladder diagram component:

1. Click the right key of the mouse after the component is selected; select the modified notes [M] in the springing environment menu;

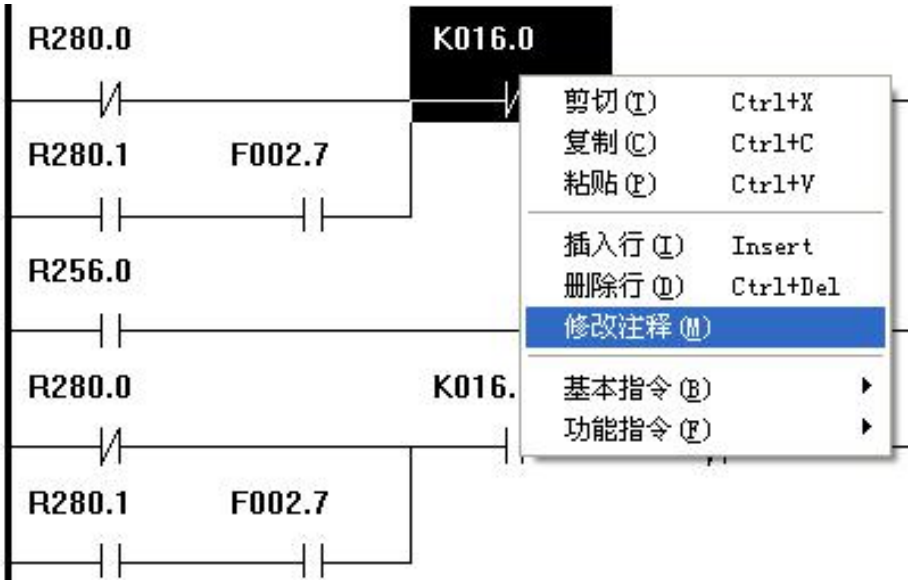


Fig. 10-3-4-2-1

2. Select the edit [Alt+ E]----Note modification [M] of the main menu.

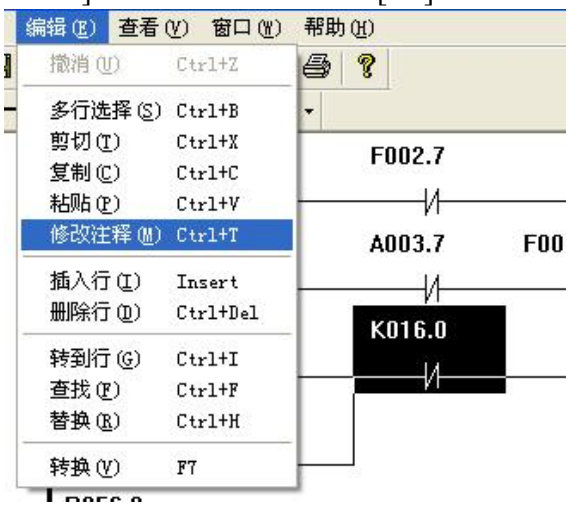


Fig. 10-3-4-2-2

3. Shortcut key [Ctrl+T].

Input the notes in the springing dialog box; save it by clicking the OK button.

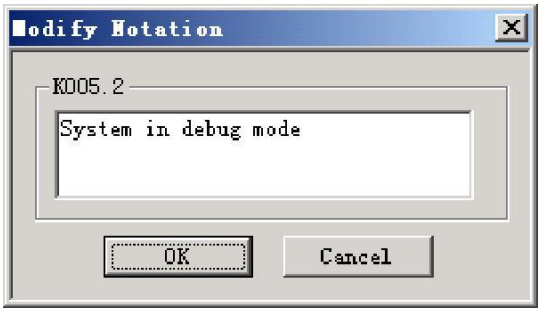


Fig.10-3-4-2-3

The notes saved will be display ed the output window under the screen when the component is selected each time, refer to the following figure:

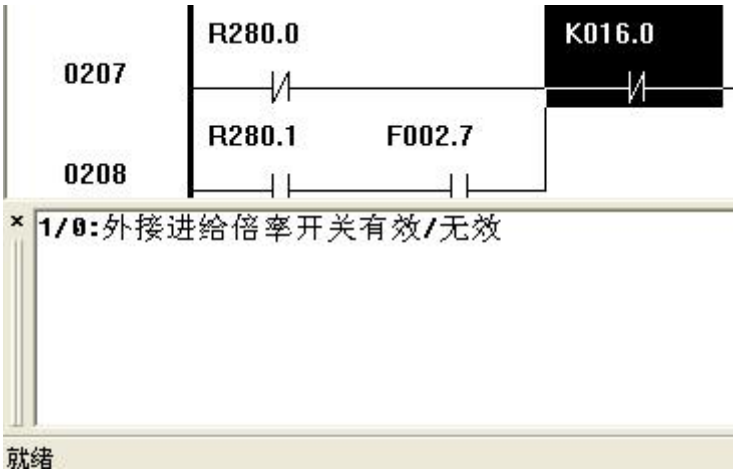


Fig. 10-3-4-2-4

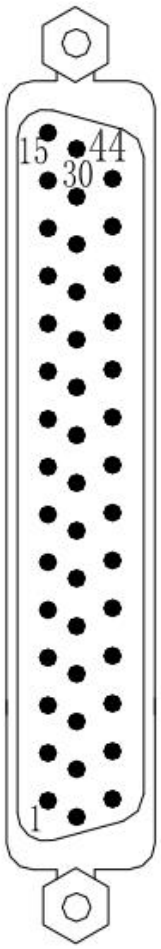
10.3.5 Export

The ladder diagram file should be converted w hen it is edited and saved, which can be generated the performed file after converting, and then transfer to the CNC using the serial -port communication

software or U disk that it is performed by the PLC from the CNC system.

Appendix I IO Defines of GH-1000TC

1.1 Input X Address_CN61 Plug(Standard Ladder)



Input Port
Male DB44
(CN61)

Pin	Add.	Symbol	Explanation
21-24	0V	Power	0V_Power
18-20 25-28	Null	Null	Null
1	X0.0	SAGT	Detecting Input_Protective Door
2	X0.1	SP	External Pause
3	X0.2	DIQP	Control Input_Chuck
4	X0.3	DECX	Input_X-Axis homing Switch
5	X0.4	DITW	Control Input_Tailstock
6	X0.5	ESP	Emergency Stop
7	X0.6	LIMU	Input_Cancel over-travel
8	X0.7	T05	Tool Position Bit5
9	X1.0	T06	Tool Position Bit6
10	X1.1	T07	Tool Position Bit7
11	X1.2	T08	Tool Position Bit8
12	X1.3	DECZ	Input_Z-Axis homing Switch
13	X1.4	ST	External Cycle Start
14	X1.5	M41I	Position Input of 1st Gear
15	X1.6	M42I	Position Input of 2nd Gear
16	X1.7	T01	Tool Position Bit1
29	X2.0	T02	Tool Position Bit2
30	X2.1	T03	Tool Position Bit3
31	X2.2	T04	Tool Position Bit4
32	X2.3	DECY	Input_Y-Axis homing Switch
33	X2.4	DEC4	Input_4th-Axis homing Switch
34	X2.5	DEC5	Input_5th Axis homing Switch
35	X2.6	TCP	Turret Clamp Position Signal
36	X2.7	AEY/BDT	Skip
37	X3.0	LMIX	X-Limitation Input
38	X3.1	LMIY	Y-Limitation Input
39	X3.2	LMIZ	Z-Limitation Input
40	X3.3	WQPJ	Position Input of Chuck Loose
41	X3.4	NQPJ	Position Input of Chuck Tighten
42	X3.5	SKIP	G31 SKIP
43	X3.6	AEX	G36 Skip
44	X3.7	AEZ	G37 Skip

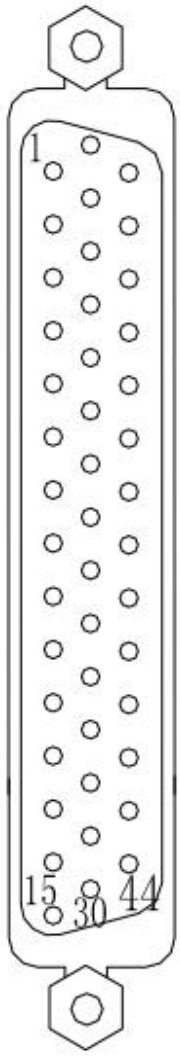
1.2 X-Address on MDI Panel

Address is from X20 to X25, total is 6 words,which are corresponding to keys on panel one by one. User cannot alter their defines.

Table of Address & Panel Keys

KEY	Address	KEY	Address
EDIT	X20.0	RAPID	X23.0
AUTO	X20.1	Z-FEED-	X23.1
MDI	X20.2	JOG	X23.2
MACHINE ZERO	X20.3	S.STOP	X23.3
MPG	X20.4	-	X23.4
MANUAL	X20.5	-	X23.5
PROGRAM ZERO	X20.6	-	X23.6
Y-MPG/FEED+	X20.7	-	X23.7
X-MPG/FEED+	X21.0	COOLING	X24.0
4TH-FEED-	X21.1	LUBRICATE	X24.1
CS	X21.2	User2	X24.2
S.CW	X21.3	X1	X24.3
S.OVERRIDE+	X21.4	X10	X24.4
G00.OVERRIDE+	X21.5	X100	X24.5
F.OVERRIDE+	X21.6	X1000	X24.6
CYCLE START	X21.7	4^{th} MPG/FEED+	X24.7
SINGLE	X22.0	X-FEED-	X25.0
SKIP	X22.1	Y-FEED-	X25.1
MACHINE LOCK	X22.2	T.CHANGE	X25.2
M.S.T LOCK	X22.3	S.CCW	X25.3
DNC	X22.4	S.OVERRIDE-	X25.4
OPTIONAL STOP	X22.5	G00.OVERRIDE-	X25.5
User1	X22.6	F.OVERRIDE-	X25.6
Z-MPG/FEED+	X22.7	FEED HOLD	X25.7

1.3 Output Y-Address_CN62 Plug(Standard Ladder)



Output Port
Female DB44
CN61 Plug

PIN	ADD.	Symbol	Explanation
17-19 26-28	0V	Power	0V_Power
20-25	+24V	Power	+24V_Power
1	Y0.0	COOL	Coolant Output
2	Y0.1	M32	Lubricate Output
3	Y0.2		Remain
4	Y0.3	M03	Spindle-Clockwise Rotation
5	Y0.4	M04	Spindle-Counter Clockwise Rotation
6	Y0.5	M05	Spindle-Stop
7	Y0.6	SCLP	Spindle Clamp
8	Y0.7	SPZD	Spindle Brake
9	Y1.0	S1/M41	Spindle Mechanical Gear1 Output
10	Y1.1	S2/M42	Spindle Mechanical Gear2 Output
11	Y1.2	S3/M43	Spindle Mechanical Gear3 Output
12	Y1.3	S4/M44	Spindle Mechanical Gear4 Output
13	Y1.4	DOQPJ	Chuck Tighten Output
14	Y1.5	DOQPS	Chuck Loose Output
15	Y1.6	TL+	Turret CW Rotation
16	Y1.7	TL-	Turret CCW Rotation
29	Y2.0	TZD	Turret Brake
30	Y2.1	INDXS	Turret Pre-Indexing Coil Output
31	Y2.2	CLPY	Yellow-Indicator
32	Y2.3	CLPG	Green-Indicator
33	Y2.4	CLPR	Red-Indicator
34	Y2.5	DOTWJ	Forward-Tailstock
35	Y2.6	DOTWS	Backward-Tailstock
36	Y2.7		Remain
37	Y3.0		Remain
38	Y3.1		Remain
39	Y3.2		Remain
40	Y3.3		Remain
41	Y3.4		Remain
42	Y3.5		Remain
43	Y3.6		Remain
44	Y3.7		Remain

1.4 Y-Address on MDI Panel

Key Indicator Output	Address	Key Indicator Output	Address
EDIT Indicator	Y20.0	RAPID Indicator	Y23.0
AUTO Indicator	Y20.1	-	Y23.1
MDI Indicator	Y20.2	JOG Indicator	Y23.2
MACHINE ZERO Indicator	Y20.3	S.STOP Indicator	Y23.3
MPG Indicator	Y20.4	-	Y23.4
MANUAL Indicator	Y20.5	-	Y23.5
PROGRAM ZERO Indicator	Y20.6	-	Y23.6
Y-MPG/FEED+ Indicator	Y20.7	-	Y23.7
X-MPG/FEED+ Indicator	Y21.0	COOLING Indicator	Y24.0
-	Y21.1	LUBRICATING Indicator	Y24.1
CS Indicator	Y21.2	USER2 Indicator	Y24.2
S.CW Indicator	Y21.3	X1 Indicator	Y24.3
-	Y21.4	X10 Indicator	Y24.4
-	Y21.5	X100 Indicator	Y24.5
-	Y21.6	X1000 Indicator	Y24.6
CYCLE START Indicator	Y21.7	4TH-MPG/FEED+ Indicator	Y24.7
SINGLE Indicator	Y22.0	X-FEED- Indicator	Y25.0
SKIP Indicator	Y22.1	Y-FEED- Indicator	Y25.1
MACHINE LOCK Indicator	Y22.2	T.CHANGE Indicator	Y25.2
M>S.T LOCK Indicator	Y22.3	S.CCW Indicator	Y25.3
DNC Indicator	Y22.4	-	Y25.4
OPTIONAL STOP Indicator	Y22.5	-	Y25.5
USER1 Indicator	Y22.6	-	Y25.6
Z-MPG/FEED+ Indicator	Y22.7	FEED.HOLD Indicator	Y25.7

1.5 F Signal

F0000 OP SA STL SPL *** *** *** ***
F0008 Auto run
F0007 Servo ready
F0006 Cycle start alarm
F0005 Dwell alarm
F0004 System Run Remark

F0010 MA *** TAP ENB DEN SAR RST AL
F0018 CNC ready
F0016 Tapping
F0015 Spindle enable
F0014 allocation end
F0013 Spindle rev arrive
F0012 RESET
F0011 alarm

F0020 MDRN CUT MSTOP SRNMV THRD *** RPDO ***
F0028 Dry run
F0027 Cutting
F0026 optional STOP detection
F0025 Program start
F0024 Threading
F0022 Rapid cutting

F0030 *** MEDT MMEM MRMT MMDI MJ MH MINC
F0037 Memory edit select detection signal
F0036 Auto run select detection signal
F0035 DNC select decetion signal
F0034 MDI mode select decetion signal
F0033 JOG feeding detection
F0032 Handwheel feed detection signal
F0031 Increment feed detection

F0040 *** MPST MREF MAFL MSBK *** MMLK MBDT
F0047 The start point of return program detection signal
F0046 Manual return reference point decetion signal
F0045 MST lock decetion signal
F0044 Single block decetion signal
F0042 All axis machine lock decetion signal
F0041 Skip optional program decetion signal

F0070 *** *** *** *** TF SF *** MF
F0074 tool function strobe signal
F0073 spindle speed strobe signal
F0071 MST strobe signal

F0080 *** *** *** *** *** *** *** ENTF
F0081 Signal_Input Single Mode on POS Screen

F0090 DM00 DM01 DM02 DM30 *** *** *** ***
F0098 M00 decode signal
F0097 M01 decode signal
F0096 M02 decode signal
F0095 M30 decode signal

F0100 MB07 MB06 MB05 MB04 MB03 MB02 MB01 MB00
F0108 MST code MB07
F0107 MST code MB06
F0106 MST code MB05
F0105 MST code MB04
F0104 MST code MB03
F0103 MST code MB02
F0102 MST code MB01
F0101 MST code MB00

F0110 *** *** *** *** *** *** *** ***
F0111 (Y0006) M89I0_OUT
F0112 (Y0007) M89I1_OUT
F0113 (Y0008) M89I2_OUT
F0114 (Y0011) M89I3_OUT
F0115 (Y0012) M89I4_OUT
F0116 (Y0013) M89I5_OUT
F0117 (Y0014) M89I6_OUT
F0118 (Y0028) M89I7_OUT

F0120 *** *** *** *** *** *** *** ***
F0121 (Y0032) Chip Remove Output
F0122 (Y0033) Working Indicator

F0140 *** *** *** *** *** *** DRUN PDBG
F0141 PLC in debug mode
F0142 switching mode inhibit signal

F0150 *** *** *** EN5T EN4T ENY *** ***

F0155 5th axis select
F0154 4th axis select
F0153 Y axis select

F0180 AR07 AR06 AR05 AR04 AR03 AR02 AR01 AR00
F0181 spindle actual speed AR00
F0182 spindle actual speed AR01
F0183 spindle actual speed AR02
F0184 spindle actual speed AR03
F0185 spindle actual speed AR04
F0186 spindle actual speed AR05
F0187 spindle actual speed AR06
F0188 spindle actual speed AR07

F0190 AR15 AR14 AR13 AR12 AR11 AR10 AR09 AR08
F0191 spindle actual speed AR08
F0192 spindle actual speed AR09
F0193 spindle actual speed AR10
F0194 spindle actual speed AR11
F0195 spindle actual speed AR12
F0196 spindle actual speed AR13
F0197 spindle actual speed AR14
F0198 spindle actual speed AR15

F0200 *** *** *** *** *** *** BCLP BUCLP
F0201 4TH axis index table unlock signal
F0202 4TH axis index table lock signal

F0210 *** MST MSP *** MESP *** *** ***
F0214 shield external links emergency stop
F0216 shield external links Dwell
F0217 shield external links Cycle start

F0220 SB07 SB06 SB05 SB04 SB03 SB02 SB01 SB00
F0228 spindle speed code signal SB07
F0227 spindle speed code signal SB06
F0226 spindle speed code signal SB05
F0225 spindle speed code signal SB04
F0224 spindle speed code signal SB03
F0223 spindle speed code signal SB02
F0222 spindle speed code signal SB01
F0221 spindle speed code signal SB00

F0260 TB07 TB06 TB05 TB04 TB03 TB02 TB01 TB00
F0268 tool function code signal TB07
F0267 tool function code signal TB06
F0266 tool function code signal TB05
F0265 tool function code signal TB04
F0264 tool function code signal TB03
F0263 tool function code signal TB02
F0262 tool function code signal TB01
F0261 tool function code signal TB00

F0270 Y signal output
F0280 X signal output high level select
F0290 X signal output low level select

F0300 R08O R07O R06O R05O R04O R03O R02O R01O
F0308 S12 bit code signal R08O
F0307 S12 bit code signal R07O
F0306 S12 bit code signal R06O
F0305 S12 bit code signal R05O
F0304 S12 bit code signal R04O
F0303 S12 bit code signal R03O
F0302 S12 bit code signal R02O
F0301 S12 bit code signal R01O

F0310 *** *** *** *** R12O R11O R10O R09O
F0314 S12 bit code signal R12O
F0313 S12 bit code signal R11O
F0312 S12 bit code signal R10O
F0311 S12 bit code signal R09O

F0320 X1000 X100 X10 X1 *** *** *** ***
F0328 step X1000 soft key
F0327 step X100 soft key
F0326 step X10 soft key
F0325 step X1 soft key

F0330 *** *** *** *** *** G88T G84T RTAP
F0331 Rigid tapping mode signal
F0332 G84 rigid tapping mode signal
F0333 G88 rigid tapping mode signal

F0340 SSTOP SCW Z- Z+ X- X+ *** ***
F0343 X+ soft key

F0344 X- soft key
F0345 Z+ soft key
F0346 Z- soft key
F0347 spindle forward soft key
F0348 spindle stop soft key

F0350 SCCW MSTOP AFLO BDTO SBKO MLKO DRNO QFAST
F0358 spindle anti revolution soft key
F0357 optional STOP soft key
F0356 MST lock soft key
F0355 Skip block soft key
F0354 single block soft key
F0353 MT lock soft key
F0352 Dry run soft key
F0351 rapid move soft key

F0360 S- S+ FAST- FAST+ *** *** FEED- FEED+
F0361 feed overrid increase soft key
F0362 feed overrid decrease soft key
F0365 rapid overrid increase soft key
F0366 rapid overrid decrease soft key
F0367 spindle overrid increase soft key
F0368 spindle overrid decrease soft key

F0370 *** *** *** ZP5 ZP4 ZP3 ZP2 ZP1
F0375 Return reference point end signal ZP5
F0374 Return reference point end signal ZP4
F0373 Return reference point end signal ZP3
F0372 Return reference point end signal ZP2
F0371 Return reference point end signal ZP1

F0380 *** *** *** MV5 MV4 MV3 MV2 MV1
F0385 axis move signal MV5
F0384 axis move signal MV4
F0383 axis move signal MV3
F0382 axis move signal MV2
F0381 axis move signal MV1

F0390 *** *** *** MVD5 MVD4 MVD3 MVD2 MVD1
F0395 axis move direction signal MVD5
F0394 axis move direction signal MVD4
F0393 axis move direction signal MVD3
F0392 axis move direction signal MVD2

F0391 axis move direction signal MVD1

F0400 *** *** *** ZRF5 ZRF4 ZRF3 ZRF2 ZRF1
F0405 Reference setup signal ZRF5
F0404 Reference setup signal ZRF4
F0403 Reference setup signal ZRF3
F0402 Reference setup signal ZRF2
F0401 Reference setup signal ZRF1

F0410 *** *** *** ZP15 ZP14 ZP13 ZP12 ZP11
F0411 X 1st reference return end signal
F0412 Z 1st reference return end signal
F0413 Y 1st reference return end signal
F0414 4TH 1st reference return end signal
F0415 5TH 1st reference return end signal

F0420 *** *** *** PRO5 PRO4 PRO3 PRO2 PRO1
F0425 Return program zero point end signal PRO5
F0424 Return program zero point end signal PRO4
F0423 Return program zero point end signal PRO3
F0422 Return program zero point end signal PRO2
F0421 Return program zero point end signal PRO1

F0430 *** *** *** *** *** *** *** MSPHD
F0431 spindle JOG detection signal

F0440 *** *** *** SIMSPL *** *** FSCSL ***
F0442 CS contour control switching end signal
F0443 2nd spindle CS contour control switching end signal
F0445 Analog spindle valid

F0450 *** *** *** *** *** *** *** ***
F0460 *** *** *** *** *** *** *** ***
F0470 Total tool position number
F0478 tool position number 7 bit
F0477 tool position number 6 bit
F0476 tool position number 5 bit
F0475 tool position number 4 bit
F0474 tool position number 3 bit
F0473 tool position number 2 bit
F0472 tool position number 1 bit
F0471 tool position number 0 bit

F0490 *** *** *** *** *** *** *** ***

F0500 *** *** *** *** SYS4 SYS3 SYS2 SYS1

F0501 1000T system panel selection

F0502 1000T-V system panel selection

F0503 580T system panel selection

F0504 828T system panel selection

F0510 *** *** *** VAL5 VAL4 VALY VALZ VALX

F0515 5 direction select

F0514 4 direction select

F0513 Y direction select

F0512 Z direction select

F0511 X direction select

F0520 *** *** *** *** *** *** *** ***

F0530 *** *** *** *** *** *** *** ***

F0540 UO07 UO06 UO05 UO04 UO03 UO02 UO01 UO00

F0548 Macro output signal UO07

F0547 Macro output signal UO06

F0546 Macro output signal UO05

F0545 Macro output signal UO04

F0544 Macro output signal UO03

F0543 Macro output signal UO02

F0542 Macro output signal UO01

F0541 Macro output signal UO00

F0550 UO15 UO14 UO13 UO12 UO11 UO10 UO09 UO08

F0558 Macro output signal UO15

F0557 Macro output signal UO14

F0556 Macro output signal UO13

F0555 Macro output signal UO12

F0554 Macro output signal UO11

F0553 Macro output signal UO10

F0552 Macro output signal UO09

F0551 Macro output signal UO08

F0570 *** *** *** ZP25 ZP24 ZP23 ZP22 ZP21

F0571 X second ref. return end signal

F0572 Z second ref. return end signal

F0573 Y second ref. return end signal

F0574 4TH second ref. return end signal

F0575 5TH second ref. return end signal

F0580 *** *** *** ZP35 ZP34 ZP33 ZP32 ZP31
F0581 X third ref. return end signal
F0582 Z third ref. return end signal
F0583 Y third ref. return end signal
F0584 4TH third ref. return end
F0585 5TH third ref. return end

F0590 *** *** *** ZP45 ZP44 ZP43 ZP42 ZP41
F0591 X fourth ref. return end signal
F0592 Z fourth ref. return end signal
F0593 Y fourth ref. return end signal
F0594 4TH fourth ref. return end signal
F0595 5TH fourth ref. return end signal

F0600 *** *** *** *** *** *** *** TLIFE
F0601 The life of all tool in a group have arrived

F0610 *** *** *** *** *** *** ESEND ***
F0612 The required number of part arrived signal

1.6 G Signal

G0040 *** *** *** *** FIN *** *** ***
G0044 MST end signal

G0050 LEDT AFL *** LAXIS *** *** *** ***
G0058 Edit lock signal
G0057 MST lock signal
G0055 All axis interlocking signal

G0060 *** SKIPP *** OVC *** ABSM MSTOP SRN
G0067 Skip signal
G0065 cancel feed rate signal
G0063 Manual absolute value signal
G0062 optional stop signal
G0061 Program restart signal

G0070 *** *** *** *** *** ST *** ***
G0073 Cycle start signal

G0080 *** RRW SP ESP *** *** *** ***
G0087 Reset and cursor return signal
G0086 feed hold signal
G0085 Emergency stop signal

G0090 *** *** *** *** *** M12 M32 COOL
G0093 Chuck clamp signal
G0092 Ubrication signal
G0091 Cooling signal

G0100 JV07 JV06 JV05 JV04 JV03 JV02 JV01 JV00
G0108 Manual move override signal JV07
G0107 Manual move override signal JV06
G0106 Manual move override signal JV05
G0105 Manual move override signalJV04
G0104 Manual move override signal JV03
G0103 Manual move override signal JV02
G0102 Manual move override signal JV01
G0101 Manual move override signal JV00

G0120 FV07 FV06 FV05 FV04 FV03 FV02 FV01 FV00
G0128 feedrate override signal FV07
G0127 feedrate override signal FV06

G0126 feedrate override signal FV05
G0125 feedrate override signal FV04
G0124 feedrate override signal FV03
G0123 feedrate override signal FV02
G0122 feedrate override signal FV01
G0121 feedrate override signal FV00

G0140 RV08 RV07 RV06 RV05 RV04 RV03 RV02 RV01
G0148 rapid feed override signal RV08
G0147 rapid feed override signal RV07
G0146 rapid feed override signal RV06
G0145 rapid feed override signal RV05
G0144 rapid feed override signal RV04
G0143 rapid feed override signal RV03
G0142 rapid feed override signal RV02
G0141 rapid feed override signal RV01

G0150 *** *** *** *** *** *** *** ***
G0160 *** *** *** *** SAR *** *** ***
G0164 spindle speed reached signal

G0170 *** *** *** *** DECA DECZ DECY DECX
G0174 4TH axis zero return dec detection
G0173 Y axis zero return dec detection
G0172 Z axis zero return dec detection
G0171 X axis zero return dec detection

G0180 *** *** *** *** H4TH HY HZ HX
G0184 4TH handwheel feed selection signal
G0183 Y handwheel feed selection signal
G0182 Z handwheel feed selection signal
G0181 X handwheel feed selection signal

G0190 RT *** MP2 MP1 *** *** *** ***
G0198 Manual rapid feed selection signal
G0196 Handwheel override signal MP2
G0195 Handwheel override signal MP1

G0200 *** *** *** *** *** *** *** ***
G0210 SOV7 SOV6 SOV5 SOV4 SOV3 SOV2 SOV1 SOV0
G0218 spindle speed override signal SOV7
G0217 spindle speed override signal SOV6
G0216 spindle speed override signal SOV5

G0215 spindle speed override signal SOV4
G0214 spindle speed override signal SOV3
G0213 spindle speed override signal SOV2
G0212 spindle speed override signal SOV1
G0211 spindle speed override signal SOV0

G0250 *** *** SRRB SFVB *** *** SWS2 SWS1
G0256 spindle reverse signal
G0255 spindle forward signal
G0252 Multi-spindles 2nd spindle selection signal
G0251 Multi-spindles 2nd spindle selection signal

G0260 CON CON2 *** *** *** *** CLV2 CLV1
G0268 CS cotour control switch signal
G0267 2nd spindle CS cotour control switch signal
G0262 2nd spindle analog voltage close signal
G0261 1st spindle analog voltage close signal

G0270 *** *** *** *** +J4 +J3 +J2 +J1
G0274 Feed axis and direction selection signal +J4
G0273 Feed axis and direction selection signal +J3
G0272 Feed axis and direction selection signal +J2
G0271 Feed axis and direction selection signal +J1

G0280 *** *** *** *** -J4 -J3 -J2 -J1
G0284 Feed axis and direction selection signal -J4
G0283 Feed axis and direction selection signal -J3
G0282 Feed axis and direction selection signal -J2
G0281 Feed axis and direction selection signal -J1

G0290 *** *** *** *** *** *** *** ***
G0300 *** *** *** *** +L4 +L3 +L2 +L1
G0304 Axis overtravel signal +L4
G0303 Axis overtravel signal +L3
G0302 Axis overtravel signal +L2
G0301 Axis overtravel signal +L1

G0310 *** *** *** *** -L4 -L3 -L2 -L1
G0314 Axis overtravel signal -L4
G0313 Axis overtravel signal -L3
G0312 Axis overtravel signal -L2
G0311 Axis overtravel signal -L1

G0320 *** *** *** *** *** *** *** LIMG
G0321 Axis overtravel relieve signal

G0330 *** *** *** *** *** *** *** ***
G0340 *** *** *** *** *** *** *** MCON
G0341 Shift Signal_4th As S-axis
G0342 Shift Signal_4th As S-axis

G0350 *** *** *** *** *** *** *** ***
G0360 BEUCL BECLP *** *** *** *** *** SPHD
G0368 Indexing table release complete signal
G0367 Indexing table clamp complete signal
G0361 Spindle inching function signal

G0370 NT07 NT06 NT05 NT04 NT03 NT02 NT01 NT00
G0378 current tool number NT07
G0377 current tool number NT06
G0376 current tool number NT05
G0375 current tool number NT04
G0374 current tool number NT03
G0373 current tool number NT02
G0372 current tool number NT01
G0371 current tool number NT00

G0430 ZRN *** DNC1 *** *** MD4 MD2 MD1
G0438 current operate mode select4
G0436 DNC run selection signal
G0433 current operate mode select3
G0432 current operate mode select2
G0431 current operate mode select1

G0440 HDT *** *** *** *** *** MLK BDT
G0448 Manual sequence tool change signal
G0442 MT lock signal?PLC->CNC?
G0441 PRG skip signal (PLC->CNC)

G0460 DRN *** *** *** KEY1 *** SBK ***
G0468 Dry run signal
G0464 Memory protection signal
G0462 Single block signal (PLC->CNC)
G0470 *** *** *** *** *** *** *** ***
G0480 *** *** *** *** *** *** GR2 GR1
G0482 Gear selection signal

G0481 Gear selection signal

G0540 UI07 UI06 UI05 UI04 UI03 UI02 UI01 UI00
G0548 Macro input UI07
G0547 Macro input UI06
G0546 Macro input UI05
G0545 Macro input UI04
G0544 Macro input UI03
G0543 Macro input UI02
G0542 Macro input UI01
G0541 Macro input UI00

G0550 UI15 UI14 UI13 UI12 UI11 UI10 UI09 UI08
G0558 Macro input UI15
G0557 Macro input UI14
G0556 Macro input UI13
G0555 Macro input UI12
G0554 Macro input UI11
G0553 Macro input UI10
G0552 Macro input UI09
G0551 Macro input UI08

G0610 *** *** *** *** *** *** *** RGTAP
G0611 Rigid tapping signal

Appendix II IO Defines of GH-1000MC

2.1 Input X Address_CN61 Plug(Standard Ladder)

Pin	Add.	Define	Explain
21~24	0V	Power	0V_Power supply
18~20	NULL	NULL	NULL
25~28	+24V	Power	+24V_Power supply
1	X0.0	SAGT	Signal of guard gate detection
2	X0.1	SP	External connection interrupt input
3	X0.2	THAN	Input for Manual Clamp/Loose Tool
4	X0.3	DECX	Signal of X axis deceleration
5	X0.4	G31	Jump signal of G31
6	X0.5	ESP	Signal of urgent stop
7	X0.6	LIMU	Signal of lubricating detection input
8	X0.7	PRES	Signal of pressure detection input
9	X1.0	TOPE	Position Signal of Spindle Loose Tool
10	X1.1	TCLO	Position Signal of Spindle Tighten Tool
11	X1.2	TZER	Signal of knife library return to zero key
12	X1.3	DECZ	Key Input for Homing of magazine
13	X1.4	ST	Signal of external cycle start
14	X1.5	M41I	Signal of Automatic shift 1 stop
15	X1.6	M42I	Signal of Automatic shift 2 stop
16	X1.7	HOLD	HOLD
29	X2.0	TFRX	Position Signal of Magazine Forward
30	X2.1	TBAX	Position Signal of Magazine Backward
31	X2.2	TCUX	Signal of Magazine counter switch
32	X2.3	DECY	Y axis deceleration signal
33	X2.4	DEC4	Counting Signal
34	X2.5	TZEX	Signal of magazine return to zero
35	X2.6	TRSW	Tool detection switch signal
36	X2.7	TMSW	Spindle tool detection switch signal
37	X3.0	LMIX	X axis hard line over range input //M82/M83 K0 detection signal
38	X3.1	LMIY	Y axis hard line over range input //M82/M83 K0 detection signal
39	X3.2	LMIZ	Z axis hard line over range input //M82/M83 K0 detection signal
40	X3.3	TCW	Magazine reversal key signal//M82/M83 K3 detection signal
41	X3.4	TCCW	Magazine Forward key signal//M82/M83 K4 detection signal
42	X3.5	M82K5	M85/M83 K5 detection signal
43	X3.6	TFRX	Magazine forward key signal//5th Deceleration signal //M82/M83 K6 detection signal
44	X3.7	M82K7	Magazine backward key signal//M82/M83 K7 detection signal

2.2 X-Address on MDI Panel

Address is from X20 to X25, total is 6 words,which are corresponding to keys on panel one by one. User cannot alter their defines.

Table of Address & Panel Keys

KEY	Address	KEY	Address
EDIT	X20.0	RAPID	X23.0
AUTO	X20.1	Z-FEED-	X23.1
MDI	X20.2	JOG	X23.2
MACHINE ZERO	X20.3	S.STOP	X23.3
MPG	X20.4	-	X23.4
MANUAL	X20.5	-	X23.5
DNC	X20.6	-	X23.6
4TH-MPG/FEED+	X20.7	-	X23.7
Z-MPG/FEED+	X21.0	COOLING	X24.0
Y-FEED-	X21.1	LUBRICATE	X24.1
ORIENTATION	X21.2	X1	X24.2
S.CCW	X21.3	X10	X24.3
S.OVERRIDE+	X21.4	X100	X24.4
G00.OVERRIDE+	X21.5	X1000	X24.5
F.OVERRIDE+	X21.6	-	X24.6
CYCLE START	X21.7	5TH MPG/FEED+	X24.7
SINGLE	X22.0	X-FEED-	X25.0
SKIP	X22.1	4TH-FEED-	X25.1
MACHINE LOCK	X22.2	T.CHANGE	X25.2
M.S.T LOCK	X22.3	S.CW	X25.3
JOG	X22.4	S.OVERRIDE-	X25.4
OPTIONAL STOP	X22.5	G00.OVERRIDE-	X25.5
User1	X22.6	F.OVERRIDE-	X25.6
Z-MPG/FEED+	X22.7	FEED HOLD	X25.7

2.3 Output Y-Address_CN62 Plug(Standard Ladder)

PIN	Add.	Define	Explain
17~19 26~28	0V	Power	0V_Power supply
20~25	+24V	Power	+24V_Power supply
1	Y0.0	Cool	Cooling output （M8/M9）
2	Y0.1	M32	Lubrication output（M32/M33）
3	Y0.2	TCLA	Loosening and clamping of cutting tools(M16/M17)
4	Y0.3	M03	Counter clockwise rotation of the principal axis(foreward)
5	Y0.4	M04	Counter anti-clockwise rotation of the principal axis(reversal ）
6	Y0.5	M05	Principal axis stops
7	Y0.6	HOLD	Hold
8	Y0.7	SPZD	Principal axis braking
9	Y1.0	S1/M41	Principal axis mechanical gear 1
10	Y1.1	S2/M42	Principal axis mechanical gear 2
11	Y1.2	S3/M43	Principal axis mechanical gear 3
12	Y1.3	S4/M44	Principal axis mechanical gear 4
13	Y1.4	K1	K1(M70/M71)
14	Y1.5	M10	Blow(M10/M11)
15	Y1.6	K2	K2(M72/M73)
16	Y1.7	K3	K3(M74/M75)
29	Y2.0	TLP	Principal axis cutting tools loosen indicator light
30	Y2.1	Tool brake	Band-type brake open
31	Y2.2	CLPY	The yellow light of the three lights
32	Y2.3	CLPG	The green light of the three lights
33	Y2.4	CLPR	The red light of the three lights
34	Y2.5	HOLD	Hold
35	Y2.6	HOLD	Hold
36	Y2.7	HOLD	Hold
37	Y3.0	K4	K4 (M76/M77）
38	Y3.1	TCCY	Cutting tools foreward
39	Y3.2	TCWY	Cutting tools reversal
40	Y3.3	TFRY	Cutting tools forward
41	Y3.4	TBAY	Cutting tools retreat
42	Y3.5	TBAL	Cutting tools retreating indicator light
43	Y3.6	HOLD	Hold
44	Y3.7	HOLD	Hold

2.4 Y-Address on MDI Panel

Key Indicator Output	Address	Key Indicator Output	Address
EDIT Indicator	Y20.0	RAPID Indicator	Y23.0
AUTO Indicator	Y20.1	-	Y23.1
MDI Indicator	Y20.2	JOG Indicator	Y23.2
MACHINE ZERO Indicator	Y20.3	S.STOP Indicator	Y23.3
MPG Indicator	Y20.4	-	Y23.4
MANUAL Indicator	Y20.5	-	Y23.5
PROGRAM ZERO Indicator	Y20.6	-	Y23.6
4TH-MPG/FEED+ Indicator	Y20.7	-	Y23.7
Z-MPG/FEED+ Indicator	Y21.0	COOLING Indicator	Y24.0
-	Y21.1	LUBRICATING Indicator	Y24.1
ORIENTATION Indicator	Y21.2	X1 Indicator	Y24.2
S.CCW Indicator	Y21.3	X10 Indicator	Y24.3
-	Y21.4	X100 Indicator	Y24.4
-	Y21.5	X1000 Indicator	Y24.5
-	Y21.6	5TH-Homing End Indicator	Y24.6
CYCLE START Indicator	Y21.7	Y-MPG/FEED+ Indicator	Y24.7
SINGLE Indicator	Y22.0	X-FEED- Indicator	Y25.0
SKIP Indicator	Y22.1	Y-FEED- Indicator	Y25.1
MACHINE LOCK Indicator	Y22.2	T.CHANGE Indicator	Y25.2
M>S.T LOCK Indicator	Y22.3	S.CW Indicator	Y25.3
DNC Indicator	Y22.4	-	Y25.4
OPTIONAL STOP Indicator	Y22.5	-	Y25.5
5TH MPG/FEED+ Indicator	Y22.6	-	Y25.6
X-MPG/FEED+ Indicator	Y22.7	FEED.HOLD Indicator	Y25.7

2.5 F Signal (GH-1000MC)

F0000 OP SA STL SPL *** *** *** ***
F0008 Auto run
F0007 Servo ready
F0006 Cycle start alarm
F0005 Dwell alarm

F0010 MA *** TAP ENB DEN SAR RST AL
F0018 CNC ready
F0016 Tapping
F0015 Spindle enable
F0014 allocation end
F0013 Spindle rev arrive
F0012 RESET
F0011 alarm

F0020 MDRN CUT MSTOP SRNMV THRD *** RPDO ***
F0028 Dry run
F0027 Cutting
F0026 optional STOP detection
F0025 Program start
F0024 Threading
F0022 Rapid cutting

F0030 *** MEDT MMEM MDNC MMDI MJ MH MINC
F0037 Memory edit
F0036 Auto detection
F0035 DNC comfirming
F0034 MDI mode
F0033 JOG cutting detection
F0032 Handwheel feed detection
F0031 single step feed detection

F0040 *** *** MREF MAFL MSBK MABSM MMLK MBDT
F0046 Manual reference return
F0045 MST lock
F0044 Single block
F0043 Manual absolute value detection
F0042 All axis machine lock
F0041 Skip optional program

F0050 *** *** *** *** *** *** *** POM

F0052 Retraction Singal of Rigid Tapping
F0051 Reboot Signal

F0070 *** *** *** *** TF SF *** MF
F0074 tool function strobe
F0073 spindle speedgate
F0071 MST strobe

F0080 TLIN *** *** *** *** *** SCHK PRGI
F0081 Block Record Signal
F0082 Syntax being checked signal
F0088 Linear magazine Selection Signal

F0090 DM00 DM01 DM02 DM30 *** *** *** RCT
F0098 M00 code
F0097 M01 code
F0096 M02 code
F0095 M30 code
F0090 Tool Changing

F0100 MB07 MB06 MB05 MB04 MB03 MB02 MB01 MB00
F0108 MST code MB07
F0107 MST code MB06
F0106 MST code MB05
F0105 MST code MB04
F0104 MST code MB03
F0103 MST code MB02
F0102 MST code MB01
F0101 MST code MB00

F0110 N07 N06 N05 N04 N03 N02 N01 N00
F0111 Remain [0] Key
F0112 Remain [1] Key
F0113 Remain [2] Key
F0114 Remain [3] Key
F0115 Remain [4] Key
F0116 Remain [5] Key
F0117 Remain [6] Key
F0118 Remain [7] Key

F0120 *** *** *** *** *** *** N09 N08
F0121 Remain [8] Key
F0122 Remain [9] Key

F0140 *** *** *** *** *** *** DRUN PDBG
F0141 PLC in debug
F0142 switching mode inhibit signal

F0150 *** *** *** EN5T EN4T ENZ *** ***
F0155 5 axis select
F0154 4 axis select
F0153 Z axis select

F0160 *** *** *** *** ZP4 ZP3 ZP2 ZP1
F0161 X Reference return end
F0162 Y Reference return end
F0163 Z Reference return end
F0164 4TH Reference return end

F0180 AR07 AR06 AR05 AR04 AR03 AR02 AR01 AR00
F0181 spindle actual velocity AR00
F0182 spindle actual velocity AR01
F0183 spindle actual velocity AR02
F0184 spindle actual velocity AR03
F0185 spindle actual velocity AR04
F0186 spindle actual velocity AR05
F0187 spindle actual velocity AR06
F0188 spindle actual velocity AR07

F0190 AR15 AR14 AR13 AR12 AR11 AR10 AR09 AR08
F0191 spindle actual velocity AR08
F0192 spindle actual velocity AR09
F0193 spindle actual velocity AR10
F0194 spindle actual velocity AR11
F0195 spindle actual velocity AR12
F0196 spindle actual velocity AR13
F0197 spindle actual velocity AR14
F0198 spindle actual velocity AR15

F0200 *** *** *** *** *** *** BCLP BUCLP
F0201 4TH axis index table unlock signal
F0202 4TH axis index table lock signal

F0210 *** MST MSP *** MESP *** *** ***
F0214 shield external links emergency stop
F0216 shield external links Dwell

F0217 shield external links Cycle start

F0220 SB07 SB06 SB05 SB04 SB03 SB02 SB01 SB00
F0228 spindle speed code signal SB07
F0227 spindle speed code signal SB06
F0226 spindle speed code signal SB05
F0225 spindle speed code signal SB04
F0224 spindle speed code signal SB03
F0223 spindle speed code signal SB02
F0222 spindle speed code signal SB01
F0221 spindle speed code signal SB00

F0230 *** *** *** *** *** *** *** ***
F0231 Activate Tool Setter
F0232 Disable Tool Setter

F0260 TB07 TB06 TB05 TB04 TB03 TB02 TB01 TB00
F0268 tool function code signal TB07
F0267 tool function code signal TB06
F0266 tool function code signal TB05
F0265 tool function code signal TB04
F0264 tool function code signal TB03
F0263 tool function code signal TB02
F0262 tool function code signal TB01
F0261 tool function code signal TB00

F0300 R08O R07O R06O R05O R04O R03O R02O R01O
F0308 S12 location code signal R08O
F0307 S12 location code signal R07O
F0306 S12 location code signal R06O
F0305 S12 location code signal R05O
F0304 S12 location code signal R04O
F0303 S12 location code signal R03O
F0302 S12 location code signal R02O
F0301 S12 location code signal R01O

F0310 *** *** *** *** R12O R11O R10O R09O
F0314 S12 location code signal R12O
F0313 S12 location code signal R11O
F0312 S12 location code signal R10O
F0311 S12 location code signal R09O

F0320 X1000 X100 X10 X1 *** *** RGSPM RGSPP

F0328 step X1000soft key
F0327 step X100soft key
F0326 step X10soft key
F0325 step X1soft key
F0322 spindle reverse in rigid tapping
F0321 .spindle forward in rigid tapping

F0330 MTAP DTAP *** *** *** *** *** RTAP
F0337 G63 tapping mode signal
F0336 rigid tapping executing
F0331 tapping mode

F0340 SSTOP SCW Z- Z+ Y- Y+ X- X+
F0341 X+soft key
F0342 X-soft key
F0343 Y+soft key
F0344 Y-soft key
F0345 Z+soft key
F0346 Z-soft key
F0347 spindle forward soft key
F0348 spindle stop soft key

F0350 SCCW MSTOP AFLO BDTO SBKO MLKO DRNO QFAST
F0358 spindle anti revolution soft key
F0357 optional STOP soft key
F0356 MST lock soft key
F0355 Skip optional program soft key
F0354 single block soft key
F0353 MT lock soft key
F0352 Dry run soft key
F0351 rapid move soft key

F0360 S- S+ FAST- FAST+ *** *** FEED- FEED+
F0361 feedrate increase soft key
F0362 feedrate decrease soft key
F0365 rapid overrid increase soft key
F0366 rapid overrid decrease soft key
F0367 spindle overrid increase soft key
F0368 spindle overrid decrease soft key

F0380 *** *** *** MV5 MV4 MV3 MV2 MV1
F0385 axis move signal MV5
F0384 axis move signal MV4

F0383 axis move signal MV3
F0382 axis move signal MV2
F0381 axis move signal MV1

F0390 *** *** *** MVD5 MVD4 MVD3 MVD2 MVD1
F0395 axis move direction signal MVD5
F0394 axis move direction signal MVD4
F0393 axis move direction signal MVD3
F0392 axis move direction signal MVD2
F0391 axis move direction signal MVD1

F0400 *** *** *** ZRF5 ZRF4 ZRF3 ZRF2 ZRF1
F0405 Reference setup ZRF5
F0404 Reference setup ZRF4
F0403 Reference setup ZRF3
F0402 Reference setup ZRF2
F0401 Reference setup ZRF1

F0410 *** *** *** ZP15 ZP14 ZP13 ZP12 ZP11
F0411 X 1st reference return end signal
F0412 Y 1st reference return end signal
F0413 Z 1st reference return end signal
F0414 4TH 1st reference return end signal
F0415 5TH 1st reference return end signal

F0430 *** *** *** *** *** *** *** MSPHD
F0431 spindle JOG detection
F0440 *** *** *** SIMSPL *** *** FSCSL ***
F0442 change CS contour control end
F0445 Analog spindle valid

F0450 *** *** *** *** *** AQ3 AQ2 AQ1
F0451 Z Second ref.area check signal
F0452 Z third ref.area check signal
F0453 Z fourth ref.area check signal

F0460 *** *** *** *** *** *** *** ***
F0490 *** *** *** *** *** *** *** ***
F0500 *** *** *** *** *** *** *** ***
F0510 *** *** *** VAL5 VAL4 VALY VALZ VALX
F0515 5 direction select
F0514 4 direction select
F0513 Z direction select

F0512 Y direction select
F0511 X direction select

F0520 *** *** *** *** *** *** *** ***
F0530 *** *** *** *** *** *** *** ***
F0540 UO07 UO06 UO05 UO04 UO03 UO02 UO01 UO00
F0548 Macro output UO07
F0547 Macro output UO06
F0546 Macro output UO05
F0545 Macro output UO04
F0544 Macro output UO03
F0543 Macro output UO02
F0542 Macro output UO01
F0541 Macro output UO00
F0550 UO15 UO14 UO13 UO12 UO11 UO10 UO09 UO08
F0558 Macro output UO15
F0557 Macro output UO14
F0556 Macro output UO13
F0555 Macro output UO12
F0554 Macro output UO11
F0553 Macro output UO10
F0552 Macro output UO09
F0551 Macro output UO08

F0570 *** *** *** ZP25 ZP24 ZP23 ZP22 ZP21
F0571 X second ref. return end
F0572 Y second ref. return end
F0573 Z second ref. return end
F0574 4TH second ref. return end
F0575 5TH second ref. return end

F0580 *** *** *** ZP35 ZP34 ZP33 ZP32 ZP31
F0581 X third ref. return end
F0582 Y third ref. return end
F0583 Z third ref. return end
F0584 4TH third ref. return end
F0585 5TH third ref. return end

F0590 *** *** *** ZP45 ZP44 ZP43 ZP42 ZP41
F0591 X fourth ref. return end
F0592 Y fourth ref. return end
F0593 Z fourth ref. return end
F0594 4TH fourth ref. return end

F0595 5TH fourth ref. return end

F0610 *** *** *** *** *** *** ESEND ***
F0612 Workpieces reached

2.6 G Signal_GH-1000MC

G0040 *** *** *** *** FIN *** *** ***
G0044 MST end

G0050 LEDT AFL *** LAXIS *** *** TCMP ***
G0058 edit lock signal
G0057 MST lock signal
G0055 all axis interlocking
G0052 Singal of Tool Setter is ON

G0060 *** SKIPP *** OVC *** ABSM MSTOP SRN
G0067 Skip
G0065 cancel feedrate
G0063 Manual absolute value signal
G0062 optional stop
G0061 PRG restart

G0070 *** *** *** *** *** ST *** ***
G0073 cycle start

G0080 *** *** SP ESP *** *** *** ***
G0086 feed hold signal
G0085 Emergency stop

G0090 *** *** *** *** *** M12 M32 COOL
G0093 0/1:spindle tool release/clamp signal
G0092 LUBRICANT
G0091 COOLANT

G0100 JV07 JV06 JV05 JV04 JV03 JV02 JV01 JV00
G0108 Manual move override JV07
G0107 Manual move override JV06
G0106 Manual move override JV05
G0105 Manual move override JV04
G0104 Manual move override JV03
G0103 Manual move override JV02
G0102 Manual move override JV01
G0101 Manual move override JV00

G0120 FV07 FV06 FV05 FV04 FV03 FV02 FV01 FV00
G0128 feedrate signal FV07
G0127 feedrate signal FV06

G0126 feedrate signal FV05
G0125 feedrate signal FV04
G0124 feedrate signal FV03
G0123 feedrate signal FV02
G0122 feedrate signal FV01
G0121 feedrate signal FV00

G0140 RV08 RV07 RV06 RV05 RV04 RV03 RV02 RV01
G0148 rapid feedrate signal RV08
G0147 rapid feedrate signal RV07
G0146 rapid feedrate signal RV06
G0145 rapid feedrate signal RV05
G0144 rapid feedrate signal RV04
G0143 rapid feedrate signal RV03
G0142 rapid feedrate signal RV02
G0141 rapid feedrate signal RV01

G0150 *** *** *** *** *** *** *** ***
G0151 M82 Application of Cxx

G0160 *** *** *** *** SAR *** *** ***
G0164 spindle speed reached

G0170 *** *** *** *** DECA DECZ DECY DECX
G0174 4TH axis zero return dec detection
G0173 Z axis zero return dec detection
G0172 Y axis zero return dec detection
G0171 X axis zero return dec detection

G0180 *** *** *** H5TH H4TH HY HZ HX
G0185 5TH handwheel
G0184 4TH Handwheel
G0183 Z Handwheel
G0182 Y Handwheel
G0181 X Handwheel

G0190 RT *** MP2 MP1 *** *** *** ***
G0198 Manual rapid feed select
G0196 Handwheel override MP2
G0195 Handwheel override MP1

G0200 *** *** *** *** *** *** *** ***
G0210 SOV7 SOV6 SOV5 SOV4 SOV3 SOV2 SOV1 SOV0

G0218 spindle speed override signal SOV7
G0217 spindle speed override signal SOV6
G0216 spindle speed override signal SOV5
G0215 spindle speed override signal SOV4
G0214 spindle speed override signal SOV3
G0213 spindle speed override signal SOV2
G0212 spindle speed override signal SOV1
G0211 spindle speed override signal SOV0

G0240 MRDYA *** *** *** *** *** *** ***
G0248 MT ready

G0250 *** *** SRRB SFVB *** *** *** ***
G0256 spindle reverse signal
G0255 spindle forward signal

G0260 CON CON2 *** *** *** *** CLV2 CLV1
G0268 CS cotour controlsignal change
G0267 second spindle CS cotour controlsignal change
G0262 Analog second spindle bottery voltage close
G0261 Analog first spindle bottery voltage close

G0270 *** *** *** +J5 +J4 +J3 +J2 +J1
G0275 feed axis and direction select+J5
G0274 feed axis and direction select+J4
G0273 feed axis and direction select+J3
G0272 feed axis and direction select+J2
G0271 feed axis and direction select+J1

G0280 *** *** *** -J5 -J4 -J3 -J2 -J1
G0285 feed axis and direction select-J5
G0284 feed axis and direction select-J4
G0283 feed axis and direction select-J3
G0282 feed axis and direction select-J2
G0281 feed axis and direction select-J1

G0290 *** *** *** *** *** *** *** ***
G0300 *** *** *** +L5 +L4 +L3 +L2 +L1
G0305 axis Overtravel+L5
G0304 axis Overtravel+L4
G0303 axis Overtravel+L3
G0302 axis Overtravel+L2
G0301 axis Overtravel+L1

G0310 *** *** *** -L5 -L4 -L3 -L2 -L1
G0315 axis Overtravel-L5
G0314 axis Overtravel-L4
G0313 axis Overtravel-L3
G0312 axis Overtravel-L2
G0311 axis Overtravel-L1

G0320 *** *** *** *** *** G54 *** LIMG
G0326 Del 160720
G0325 Del 160720
G0324 Del 160720
G0323 Zero-Point Setting Function of G54
G0321 axis Overtravel cancel

G0330 *** *** *** *** *** *** *** ***
G0331 Start Setting Tool(O9200)(20lines)
G0332 File Selection in Tool Setter(O9201)
G0323 File Selection in Tool Setter(O9202)
G0324 File Selection in Tool Setter(O9203)
G0325 File Selection in Tool Setter(O9204)
G0326 File Selection in Tool Setter(O9205)
G0327 File Selection in Tool Setter(O9206)
G0328 File Selection in Tool Setter(O9207)

G0340 G84Z *** *** *** *** *** 5CS ***
G0258 CS Shift when 5th* As Analog SP.(0:S 1:C)
G0348 Reverse_Z-DIR when Tapping(0:NO 1:YES)

G0350 *** *** *** *** *** *** *** ***
G0360 BEUCL BECLP *** *** *** *** *** SPHD
G0368 index table release
G0367 index table clamp
G0361 spindle inching function signal

G0370 NT07 NT06 NT05 NT04 NT03 NT02 NT01 NT00
G0378 current tool number NT07
G0377 current tool number NT06
G0376 current tool number NT05
G0375 current tool number NT04
G0374 current tool number NT03
G0373 current tool number NT02
G0372 current tool number NT01
G0371 current tool number NT00

G0380 ST07 ST06 ST05 ST04 ST03 ST02 ST01 ST00
G0388 current tool number ST07
G0387 current tool number ST06
G0386 current tool number ST05
G0385 current tool number ST04
G0384 current tool number ST03
G0383 current tool number ST02
G0382 current tool number ST01
G0381 current tool numbe ST00

G0390 *** *** *** EN5T EN4T ENZ ENY ENX
G0395 PLC Control Signal of 5th_Enable
G0394 PLC Control Signal of 4th_Enable
G0393 PLC Control Signal of Z_Enable
G0392 PLC Control Signal of Y_Enable
G0391 PLC Control Signal of X_Enable

G0400 *** SALM *** AL5T AL4T ALZ ALY ALX
G0407 PLC Control Signal of SP_ALM
G0405 PLC Control Signal of 5TH_ALM
G0404 PLC Control Signal of 4TH_ALM
G0403 PLC Control Signal of Z_ALM
G0402 PLC Control Signal of Y_ALM
G0401 PLC Control Signal of X_ALM

G0430 ZRN *** DNC1 *** *** MD4 MD2 MD1
G0438 current operate mode select4
G0436 DNC run selection signal
G0433 current operate mode select3
G0432 current operate mode select2
G0431 current operate mode select1

G0440 HDT *** *** *** *** *** MLK BDT
G0448 Manual sequence tool change
G0442 MT lock?PLC->CNC?
G0441 PRG skip(PLC->CNC)

G0460 DRN *** *** *** KEY1 *** SBK ***
G0468 Dry run
G0464 Memory protection
G0462 Single block(PLC->CNC)

G0470 *** *** *** *** *** *** *** ***

G0480 *** *** *** *** *** *** GR2 GR1
G0482 Gear select
G0481 Gear select

G0530 CDZ SMZ *** *** *** *** *** ***
G0538 Chamfer signal
G0537 error-detecting

G0540 UI07 UI06 UI05 UI04 UI03 UI02 UI01 UI00
G0548 Macro input UI07
G0547 Macro input UI06
G0546 Macro input UI05
G0545 Macro input UI04
G0544 Macro input UI03
G0543 Macro input UI02
G0542 Macro input UI01
G0541 Macro input UI00

G0550 UI15 UI14 UI13 UI12 UI11 UI10 UI09 UI08
G0558 Macro input UI15
G0557 Macro input UI14
G0556 Macro input UI13
G0555 Macro input UI12
G0554 Macro input UI11
G0553 Macro input UI10
G0552 Macro input UI09
G0551 Macro input UI08

G0560 *** *** *** *** HZP14 HZP13 HZP12 HZP11
G0561 First ref. X Pos check enable
G0562 First ref. Y Pos check enable
G0563 First ref. Z Pos check enable
G0564 First ref. Th4 Pos check enable

G0570 *** *** *** *** HZP24 HZP23 HZP22 HZP21
G0571 Second ref. X Pos check enable
G0572 Second ref. Y Pos check enable
G0573 Second ref. Z Pos check enable
G0574 Second ref. 4TH Pos check enable

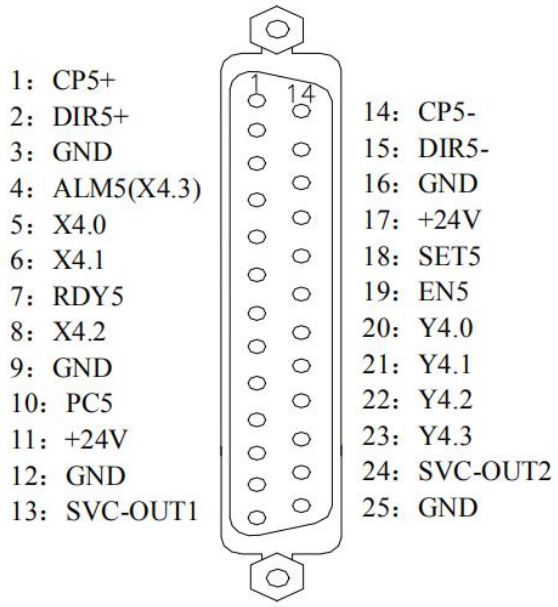
G0580 *** *** *** *** HZP34 HZP33 HZP32 HZP31
G0581 Third ref. X Pos check enable
G0582 Third ref. Y Pos check enable

G0583 Third ref. Z Pos check enable
G0584 Third ref. 4TH Pos check enable

G0590 *** *** *** *** HZP44 HZP43 HZP42 HZP41
G0591 Fourth ref. X Pos check enable
G0592 Fourth ref. Y Pos check enable
G0593 Fourth ref. Z Pos check enable
G0594 Fourth ref. 4TH Pos check enable

G0610 *** *** *** *** *** *** *** RGTAP
G0611 Rigid tapping

Appendix III IO Defines of Spindle



Pin	Address	Definition
3, 9, 16	0V	Supply power ： 0V
11,17	+24V	Supply power ： +24V
1	CP5+	spindles Pulse positive signal
2	DIR5+	Spindle Direction positive signal
4	ALM5	spindles pulse driving alarm
5	X4.0	spindle switching position mode in place signal
6	X4.1	spindles speed reach
7	RDY5	spindles ready signal
8	X4.2	spindles positioning in place signal
9	GND	0V end of supply power
10	PC5	spindles zero point signal
11	+24V	+24V end of supply power
12	GND	Analog voltage 0V end of spindles
13	SVC-OUT1	Analog output voltage 1 of spindles
14	CP5 -	Spindles pulse negative signal
15	DIRD -	spindles negative signal
19	EN5	Enable signal
20	Y4.0	spindle position control output （tapping）
21	Y4.1	Hold
22	Y4.2	Directional output of spindle
23	Y4.3	Hold
24	SVC-OUT2	Analog output voltage 2 of spindle
25	GND	Analog voltage 0V end of spindle