CTSC-200 Series Motion control Module User Manual

SHENZHEN CO-TRUST TECHNOLOGY CO.,LTD.

Releasing Date: 06/2013

Version: V1.10

Copyright

Disclaimer

Only qualified personnel should be allowed to install and work on CTSC-200 Series Motion Control module. Because the user not strictly in accordance with the operating requirements, causing serious consequences, and CO-TRUST is not responsible for.

Preface

Brief Introduction

CTSC-200 Series Motion control module is organized according to the following topics:

- Overview of EM253 Motion control module: Introduces the appearance and features of EM231-7WA module;
- Specifications of EM253 Motion control module: Mainly introduces the specification of Weight module EM231-7WA;
- Using the motion_ctrl_lib: Mainly introduces the specification of EM231_7WA_lib;
- Application Example: Mainly introduces an application example of EM253 module.
- Applications: Introduces the application of in detail;
- Appendix: Offers ordering info to users;

Suitable Personnel

This manual provides information about installing and programming CTSC-200 Series Motion control module and is designed for engineers, installers, maintainer and electricians who have a general knowledge of programmable logic controllers.

Service on the Internet

In addition to this manual, we offer related products info and technical service online on the internet at: http://www.co-trust.com

Safety Instructions

Please read and understand the following instructions or precautions carefully before using CTSC-200 Series Motion control module, and must be complied with the installation and debugging preventive measures and operating procedures.

The following DANGER and CAUTION symbols are used according to the level of dangers possibly occurring if you fail to observe the instructions or precautions indicated.

 **DANGER**	Danger indicates an imminently hazardous situation which, if not Operate according to the requirement, will result in personal injury".
 **CAUTION**	Caution indicates a potentially hazardous situation which, if not operate according to the requirement, may result in mild or moderate injury and equipment damage.
 **TIP**	Remark indicates a necessary additional remarks to the operating descriptions.

CONTENTS

1 Overview of EM253 Motion Control Module

EM253 module belongs to CTSC-200 series PLC, which is used in motion control field.

EM253 module has two independent high-speed counter MC253_HSC0 and MC253_HSC1 (Single-phase 200KHz, double-phase 200KHz), and 2*200KHz high-speed pulse output, which support instructions MC253_PTP/SPEED_CTL/PWM of motion_ctrl_module_lib.

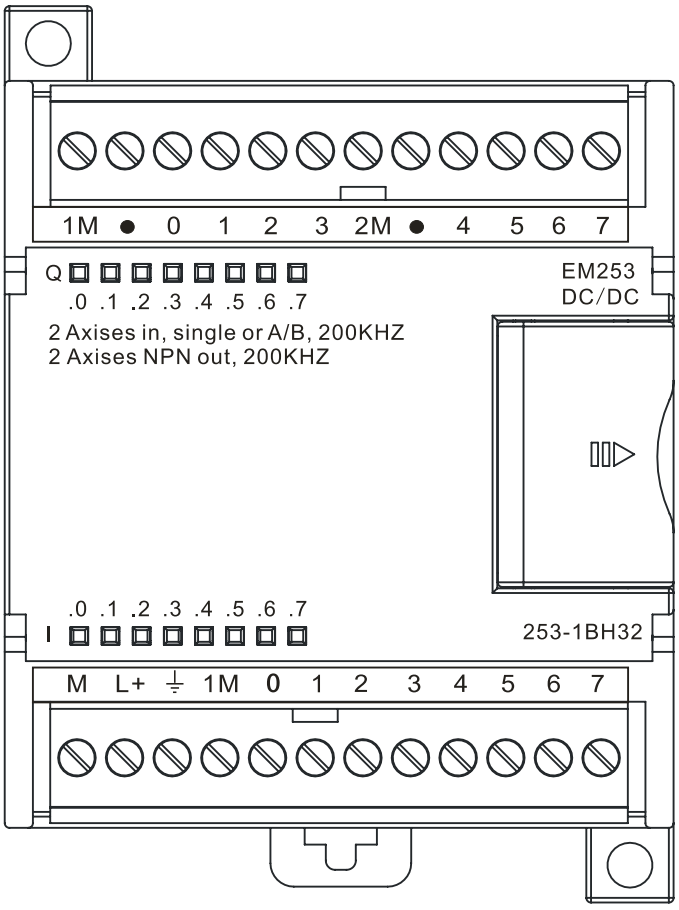


Figure 1-1 Appearance of EM253 motion control module

Tip

- Micro/WIN of Siemens supports motion_ctrl_module_lib;
- EM253 module supports the connection with CTSC-200 series CPU (one CPU could connect 5 EM253 module at most), but CTSC-100 CPU and Siemens CPU are incompatible with EM253 module.
- V3.98 and above firmware version of CTSC-200 CPU supports EM253 module.

2 Specifications of EM253 Motion Control Module

2.1 Technical specifications

Table 2-1 Digital input features

General		24VDC input
Number of integrated digital inputs		8 inputs
Input type		Sink/Source(IEC Type 1 / sink)
Rated Voltage		24 VDC at 5 mA, nominal
Permissible max. continuous voltage		30 VDC
Surge Voltage		35 VDC for 0.5 s
Logic 1 signal (min.)		15.6 VDC at 2.72mA (I0.0, I0.1, I0.2, I0.4, I0.5, I0.6) 12.8VDC at 2.55mA (I0.3, I0.7)
Logic 0 signal (max.)		15.4VDC at 2.69mA (I0.0, I0.1, I0.2, I0.4, I0.5, I0.6) 12.6 VDC at 2.51mA (I0.3, I0.7)
Input delay		< 1.1us (I0.0, I0.1, I0.2, I0.4, I0.5, I0.6) < 1ms (I0.3, I0.7)
Connection of 2 wire proximity sensor(Bero) Permissible leakage current (max.)		1mA
Isolation (field to logic) Optical (galvanic) Isolation groups		Yes 500 VAC for 1 minute See wiring diagram
High-speed input rate High speed counter logic1=16～26 VDC		200KHz (Single / Double-phase) (I0.0, I0.4) 200KHz (A/B phase) (I0.0 and I0.1, I0.4 and I0.5) (Input waveform duty ratio 40%~60%)
Inputs on simultaneously		All inputs
Cable Length (max.)	Shielded	500m standard input, 50m high-speed counter input
	Unshielded	300m standard input
High-speed input, max. turning frequency supported by counter		50KHz

Table 2-2 Digital output features

General	Transistor output
Number of integrated digital outputs	8 outputs
Output type	Solid State-MOSFET
Rated Voltage	24 VDC
Output voltage range	5 to 28.8 VDC
Surge Current (max.)	8A for 100ms
Logic 1 signal (max.)	0.5V
Logic 0 signal (min.)	VCC-0.5V
Rated Current per point (max.)	0.5A
Rated current per common (max.)	2.0A
Leakage current (max.)	10μA
Lamp load (max.)	3.5W
Inductive clamp voltage	L+ minus 48 VDC, 1W consumption
On state resistance (contact)	0.3 Ω typical (0.6 Ω max.)
Isolation Optical (galvanic, field to logic) Logic to contact Resistance (logic to contact) Isolation groups	 500 VAC for 1 minute -- -- See wiring diagram
Output Delay (max.) OFF to ON ON to OFF	 0.2μs (Q0.0, Q0.1, Q0.2, Q0.3) , 50μs (Q0.4, Q0.5, Q0.6, Q0.7) 0.2μs (Q0.0, Q0.1, Q0.2, Q0.3) , 130μs (Q0.4, Q0.5, Q0.6, Q0.7) DO output instruction is executed with some delays due to communication transmission. If CPU connects one EM253 module, the instruction delay time from enabling to execution is 780us; If CPU connects 5 EM253 module, the delay time from Roebling to execution is about 930us.
Pulse frequency (Max.)	200KHz (Q0.0, Q0.2)
Mechanical Life cycle	/
Contacts Lifetime	/
Output on simultaneously	All at 55℃
Connecting two outputs in parallel	Yes, only outputs in same group
Cable Length (max.) Shielded Unshielded	 500m 150m

2.2 Wiring Specification

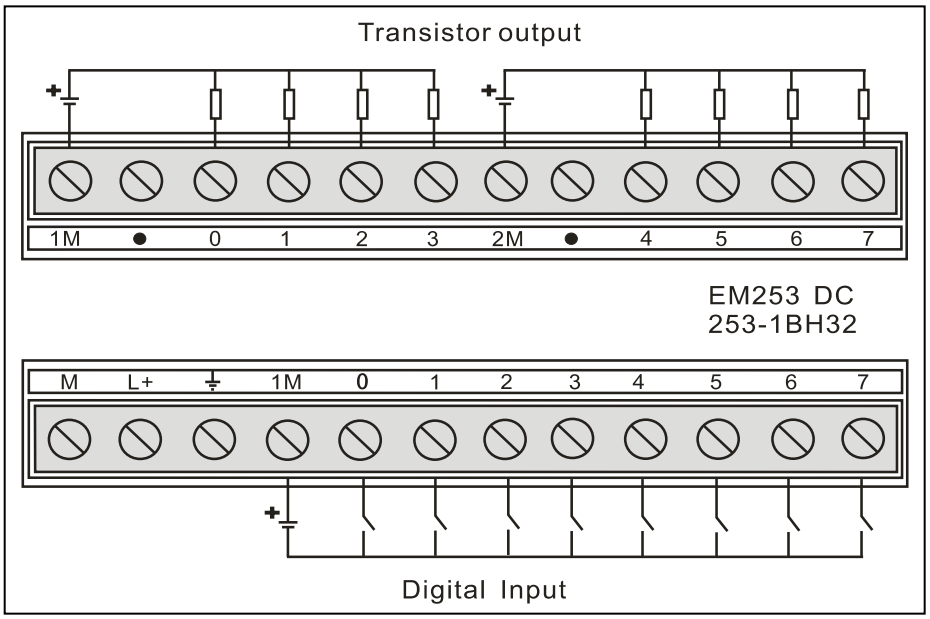


Figure 2-1 Wiring diagram of EM253 module

Table 2-3 Definition of I/O function

I0.0	I0.1	I0.2	I0.3		I0.4	I0.5	I0.6	I0.7
MC253_HSC0 pulse input	MC253_HSC0 external direction signal	MC253_HSC0 reset signal	Motion axis 0 emergency stop signal		MC253_HSC1 pulse input	MC253_HSC1 external direction signal	MC253_HSC1 reset signal	Motion axis 1 emergency stop signal
Q0.0	**Q0.1**	**Q0.2**	**Q0.3**		**Q0.4**	**Q0.5**	**Q0.6**	**Q0.7**
Axis 0 pulse output	Axis 0 direction signal	Axis 1 pulse output	Axis 1 direction signal		Common outputs			

2.3 Installation Specification

Installation Diagram

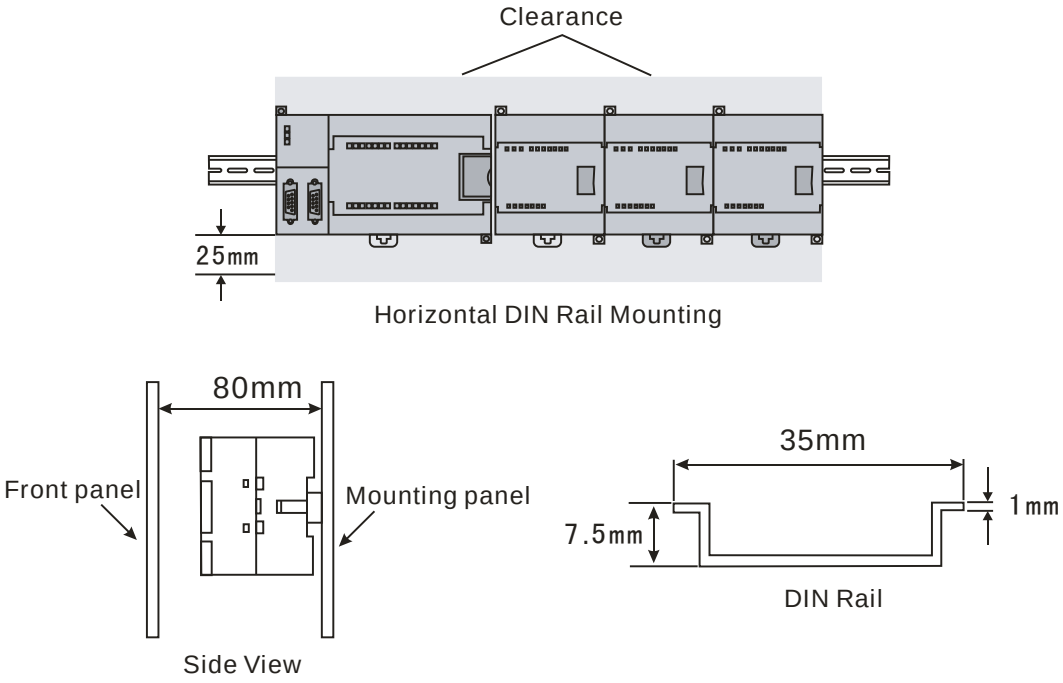
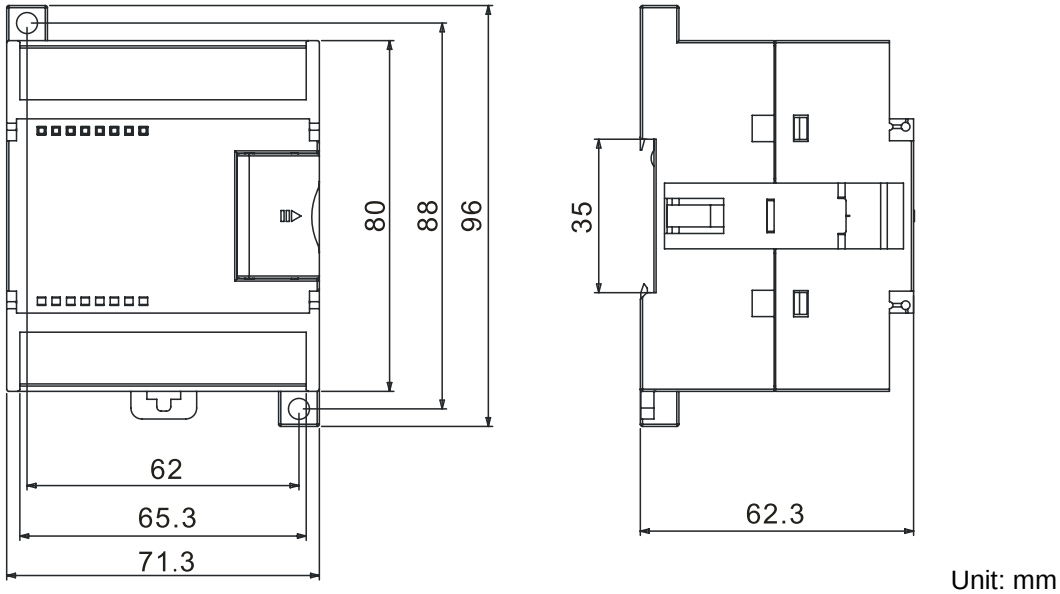


Figure 2-2 Installation Diagram

Installation dimension

EM253 module has mounting holes, which could be easily installed on the panels. The installation dimension show as following figure 2-3:



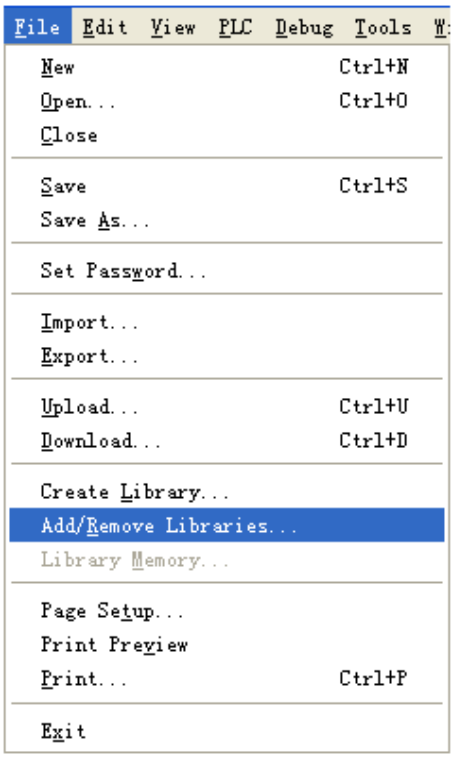
Unit: mm

Figure 2-3 Installation dimension diagram

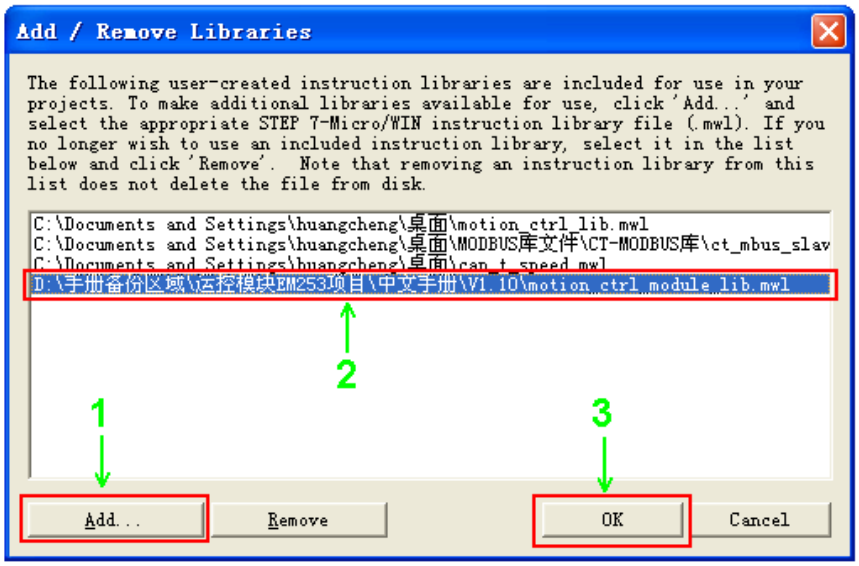
3 Using motion_ctrl_module_lib

3.1 Add library file

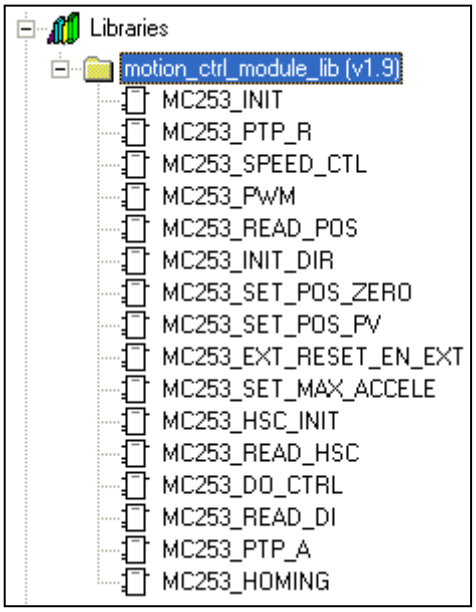
Select the "File" > "Add/ Remove Library" menu command, and click "Add..." to add your library to the project.



Click "Add..." from below dialog, navigate to the directory where you saved your library file, select the "motion_ctrl_module_lib.mwl " file and click "OK".



Once you add your library, look at the Libraries node in the instruction tree. You should see the new library "motion_ctrl_module_lib" has been added to the tree.



3.2 Instruction list of "motion_ctrl_module_lib"

Table 3-1 Instruction list of "motion_ctrl_module_lib"

Instruction Name	Function
MC253_INIT_DIR	Configurate motor direction instruction
MC253_READ_POS	Read position instruction
MC253_PTP_R	Single shaft relative motion instruction
MC253_SPEED_CTL	Speed control instruction
MC253_SET_POS_ZERO	Software reset instruction
MC253_SET_POS_PV	Setting target position instruction
MC253_EXT_RESET_EN_EXT	External reset coordinate enabling instruction
MC253_SET_MAX_ACCELE	Set maximum acceleration instruction
MC253_PWM	Pulse width modulation instruction
MC253_INIT	Motion control module initialization instruction Remark: This instruction should be executed at the beginning of the program.
MC253_DO_CTRL	Control module output instruction
MC253_READ_DI	Read module input state instruction
MC253_HSC_INIT	Setting module high speed counter instruction
MC253_READ_HSC	Read module high speed counter state instruction
MC253_PTP_A	Single shaft absolute motion instruction
MC253_HOMING	Homing instruction

3.3 Instruction details for "motion_ctrl_module_lib"

For programming by Micro/Win , instruction format of EM253 module is the same as CPU instructions, the difference is the library , the prefix for CPU instructions are MC, and the prefix of EM253 instructions are MC253.

The axis No. of CPU instruction is 0~3, and instructions of module are numbered starting from 0, and it can be last, system automatically allocates for each module. eg. Axis 0~1 of module corresponds to axis 0 and axis 1 of the first module; Axis 2~3 of module corresponds to axis 0 and axis 1 of the second module, and so on. If beyond the range of module, which would be regarded as parameter error, such as, if one module(with 2 axis only), but module assigns axis 2, and axis 2 cannot be found, so this would not be used.

Likewise, high-speed counter of module and CPU are not numbered unified, you must call module instruction while using high-speed counter of module, high-speed counter of module are numbered starting from 0, and it can be last, system automatically allocates for each module until module is under-allocated. eg. High-speed counter of module instruction marks from 0~1, which corresponds to HSC0 and HSC1 of the first module; High-speed counter of module instruction 2~3 corresponds to HSC0 and HSC1 of the second module.

※Matters need to pay special attention

1, Initialization and Library Memory

It's required to call MC253_INIT in system program for initialization as you are using EM253 module (You can only use SM0.1 to call MC253_INIT for one time). Which need pay attention is that the EM253 module requires V memory, users can allocate memory by "Library Memory" menu command, and users could not use the memory which has been occupied by the library. The memory occupied by different number of EM253 module are detailed as below table:

EM253 module number connected by CPU	Memory occupied by library
1	218 bytes
2	236 bytes
3	254 bytes
4	272 bytes
5	290 bytes

2、Clear Analog input Filter

Be sure to clear filter function for the corresponding channel of EM253 module, operation step: System Block→ Input Filters→Analog→Clear the analog channel, then download the system block to PLC. If not clear filter function for EM253 module, the library would not function properly. (Viewing PLC info, the corresponding channel for the 4 input/4 output analog module is the channel for EM253 module)

3, Communication status bit

All instructions, except MC253_INIT, added communication status bit, if alarm was reported by this bit, please check the connection, or check whether the analog filter for EM253 module channel is cleared, etc.

4、Execution sequence of Pulse output instructions

Coaxial pulse output instruction (Including MC253_PTP_R, MC253_SPEED_CTL, MC253_PWM) has 3 data buffers at most, which means 3 Coaxial pulse output instructions could be executed at one time. When multiple Coaxial pulse output instructions with one axis, and the buffer is not full (less than 3 instructions were enabled), that the latter enabling instruction can be executed in time, so the execution sequence of instructions is the same as the enabling sequence of instructions, which is not necessarily related to the position sequence of instructions.

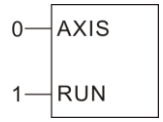
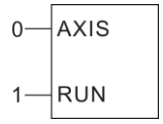
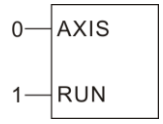
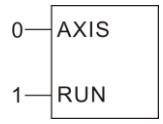
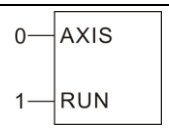
If the data buffers were full (more than 3 instructions were enabled), that the latter enabling instructions cannot be executed in time, which would be executed according to the position sequence after data buffer is released, so the execution sequence of instructions is related to position sequence of instructions, rather than enabling sequence. Refer to Example 1 for details.

You can select to enable all these instructions at one time if the motion trail you are desired are in the same order of instruction position sequence (See as Example 1).

If the motion trail you are desired are not in the same order with instruction position sequence, you have to enable less than 3 instructions for one time (See as Example 2).

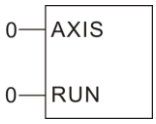
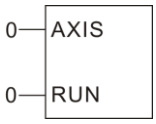
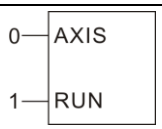
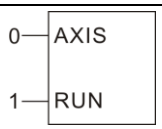
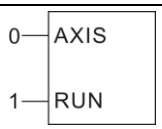
Example 1: Enabling all instructions at once

The instructions were executed according to the enabling sequence (Instruction A/B/C) when instructions in buffer are less than 3. See as following table: enabling 5 instructions at once, the enabling sequence is: Instruction A→Instruction C→Instruction B→Instruction E→Instruction D, so that the first 3 instructions were executed according to the enabling sequence, execution sequence: Instruction A→Instruction C→Instruction B:

Enabling Sequence	Position Sequence		Execution Sequence
1		Instruction A	1
3		Instruction B	3
2		Instruction C	2
5		Instruction D	--
4		Instruction E	--

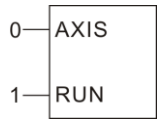
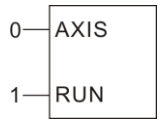
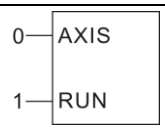
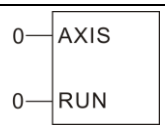
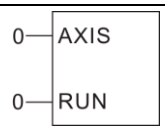
When instructions in buffer are more than 3, the rest of the instructions (Instruction D/E) were executed according to the position sequence. After instruction A is finished and released the buffer, instruction D obtains the released buffer of instruction A first; As instruction C is finished and released the buffer, instruction E obtains the released buffer of instruction C.

Instruction enabling sequence: Instruction A→Instruction C→Instruction B→Instruction E→Instruction D, see as following table, the execution sequence: Instruction A→Instruction C→Instruction B→Instruction D→Instruction E:

Enabling Sequence	Position Sequence		Execution Sequence
1		Instruction A	1
3		Instruction B	3
2		Instruction C	2
5		Instruction D	4
4		Instruction E	5

Example 2: Enabling three instructions at first, and enabling the rest after the buffer were released

The instructions were executed according to the enabling sequence (Instruction A/B/C) when instructions in buffer are less than 3. See as following table: First enable 3 instructions, the enabling sequence is: Instruction A→Instruction C→Instruction B, so that the 3 instructions were executed according to the enabling sequence, execution sequence: Instruction A→Instruction C→Instruction B:

Enabling Sequence	Position Sequence		Execution Sequence
1		Instruction A	1
3		Instruction B	3
2		Instruction C	2
--		Instruction D	--
--		Instruction E	--

After instruction A is finished and released the buffer, enable instruction E at first, then

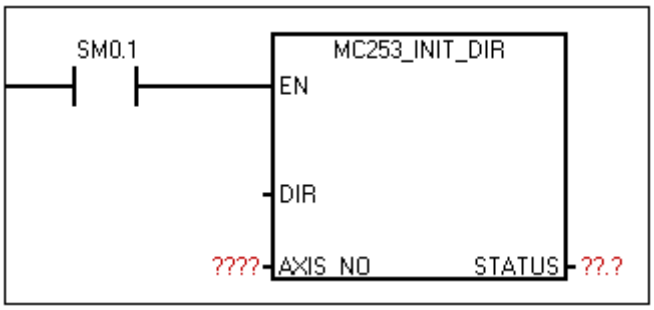
instruction E obtains the released buffer of instruction A; As instruction C is finished and released buffer, enable instruction D, so instruction D obtains the released buffer of instruction C.

Instruction enabling sequence: Instruction A→Instruction C→Instruction B→Instruction E →Instruction D, see as following table, the execution sequence is: Instruction A→ Instruction C→Instruction B→Instruction E→Instruction D:

Enabling Sequence	Position Sequence		Execution Sequence
1	0—AXIS 0—RUN	Instruction A	1
3	0—AXIS 0—RUN	Instruction B	3
2	0—AXIS 1—RUN	Instruction C	2
5	0—AXIS 1—RUN	Instruction D	5
4	0—AXIS 1—RUN	Instruction E	4

3.3.1 Configurate motor direction instruction

① Instruction Name: MC253_INIT_DIR



② Function: Configure the direction of the motor

Remark: This instruction is only executed once at the first scan cycle when CPU powered up.

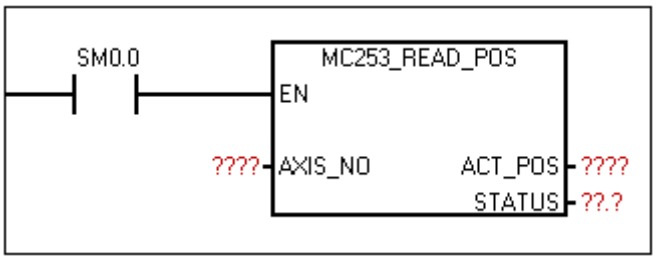
③ Parameters

Name	Input /Output	Description	Data type	Range	Remark
DIR	IN	Configure the effective level when direction signal is positive. DIR=1, the corresponding	Bool	0~1	Default: 1, Motor rotates positively,

		axis outputs '1', and the motor rotates positively. DIR=0, the corresponding axis outputs '0', and the motor reverses.			when the direction axis output is 1.
AXIS_NO	IN	Set axis No.(Every EM253 module has two axes, axis range is up to the number of EM253 module)	Byte	0~255	
STATUS	OUT	Flag bit of communication status: 1: Communication timeout	Bool	0~1	

3.3.2 Read position instruction

① Instruction Name: MC253_READ_POS



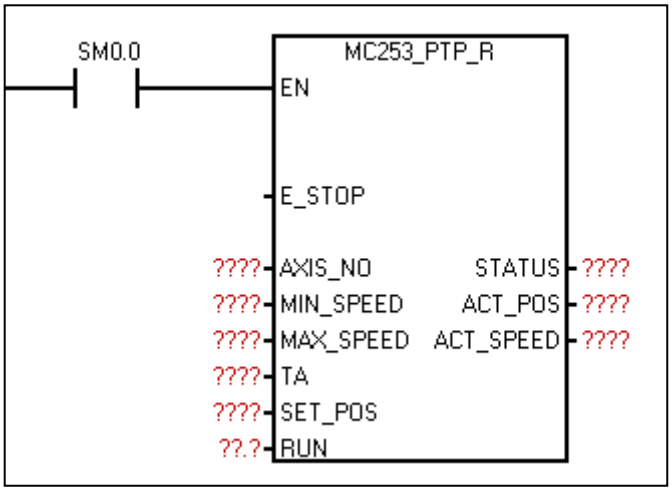
② Function: Read the absolute coordinate per axis. Once origin coordinate is set, this value would execute algebraic calculus according to the output pulse and direction: output 1 pulse as rotating positively: +1, output 1 pulse as rotating negatively: -1. Finally you can get an absolute coordinate which is regarded the setting point as origin point.

③ Parameters

Name	Input /Output	Description	Data type	Range	Remark
AXIS_NO	IN	Set axis No. (Every EM253 module has two axes, axis range is up to the number of EM253 module)	Byte	0~255	
ACT_POS	OUT	Absolute coordinate of present axis (1 pulse represents 1 coordinate)	Dint	-2147483648 ~ +2147483647	No fault status output, be sure to set correct axis No..
STATUS	OUT	Flag bit of communication status: 1→Communication timeout	Bool	0~1	

3.3.3 Single shaft relative motion instruction

① Instruction Name: MC253_PTP_R



② Function: Used as the point to pint control for single shaft (Single shaft fixed-length drive). Fixed pulses can be outputted if call once, which is set by max. Speed, min. Speed and TA/TD, the output pulses would accelerate to the max. Speed at the starting time, as the pulses is going to be finished, the pulse frequency would slow down, so as to avoid the vibration or locking caused by great inertia at the starting or stopping of the machine.

③ Parameters

Name	Input /Output	Description	Data type	Range	Remark
E_STOP	IN	Emergency stop bit: 1: Valid 0: Invalid	Bool	0/1	1. Only Run ==1 & E_Stop ==0 that it is running. 2. E_STOP =1, RUN internal resets.
AXIS_NO	IN	Set axis No.(Every EM253 module has two axes, axis range is up to the number of EM253 module)	Byte	0~255	It could not be modified during running process.
MIN_SPEED	IN	Min. Speed, which is the start or stop speed. Unit: Hz	Dword	500~200000	1. Min. Speed should be less than the max. speed. 2. It can be modified during
MAX_SPEED	IN	Max. Speed, which is the max. Running Speed. Unit: Hz	Dword	500~200000	

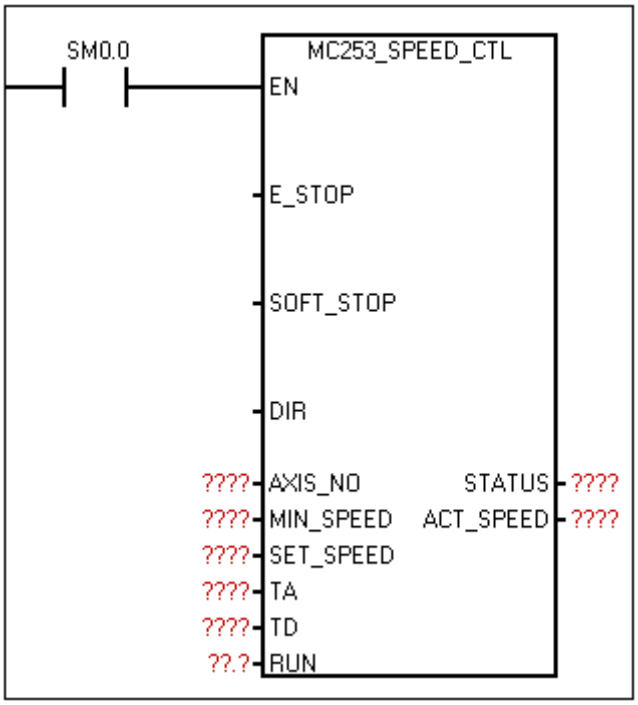
					running process.
TA	IN	Acceleration/ Deceleration time. Unit: ms	Dword	0~10000 (Refer to **additional comment 1**)	It can be modified during running process. (TA setting refer to additional comment 2)
SET_POS	IN	Output pulses, has both positive and negative value. Positive pulses represent the positive direction of X axis, negative pulses represent the negative direction of X axis.	Dint	-2147483648 ~ +2147483647	It can be modified during running process. As the new value is greater than the outputted pulses, so the final output pulses would be subjected to the new value. If new value is less than the outputted pulses, the outputting pulses would be stopped immediately.

RUN	IN/OUT	Enabling bit for running: 1: Valid 0: Invalid	Bool	0/1	1. Only RUN ==1 & E_STOP ==0 that it is running. 2. After running is completed, RUN internal reset. 3. RUN internal reset while E_STOP =1
STATUS	OUT	Output status bit: 7 6 5 4 3 2 1 0 Bit0: Error flag of parameter configuration 1—Configuration error. 0—Configuration is correct. Bit1: Running flag 1—It is running, pulse is being outputting, and instruction has not been completed. 0—Does not run, public resources are occupied by other instructions so that this instruction has not been executed. Or this parameter has been completed. Bit2: Finish flag 1—Finished, instruction has been executed. 0—Unfinished, instruction has not been executed or instruction is been performing but not finished. Bit3: Busy flag 1—Busy flag is valid, the axis is occupied by other instructions.	Byte	0~255	Bit0: 1. Only judge for axis and TA configuration. 2. MIN_SPEED/ MAX_SPEED would not report errors, and could set to a proximal value automatically.

		0—Busy flag is invalid, instruction is performing or has been finished. Bit4: Emergency stop flag (Refer to **additional comment3**) 1—Emergency stop flag is valid, the axis is disabled for external conditions. 0—Emergency stop flag is invalid. Bit5~Bit6: Reserved Bit7: Flag bit of communication status 1—Communication timeout 0—No timeout			
ACT_POS	OUT	Current relative coordinates or the output pulses of this instruction	Dint	-2147483648 ~ +2147483647	
ACT_SPEED	OUT	Present actual running speed.	Dword	500~200000	

3.3.4 Speed control instruction

① Instruction Name: MC253_SPEED_CTL



② Function: To control the frequency of signal shaft output pulses, pulse frequency(rate) could be changed at any time. Once stop command is received, it would slow to a stop automatically. If emergency stop command is received, it would stop outputting pulses without deceleration.

③ Parameters

Name	Input /Output	Description	Data type	Range	Remark
RUN	IN	Enabling bit for running: 1: Valid 0: Invalid	Bool	0/1	1. Only RUN ==1 & E_STOP ==0 & SOFT_STOP ==0 that it is running. 2. After running is completed, RUN internal reset. 3. RUN internal reset while E_STOP =1.
E_STOP	IN	Emergency stop bit. 1: Valid, 0: Invalid Once emergency stop	Bool	0/1	1. Only RUN ==1 & E_Stop ==0 &

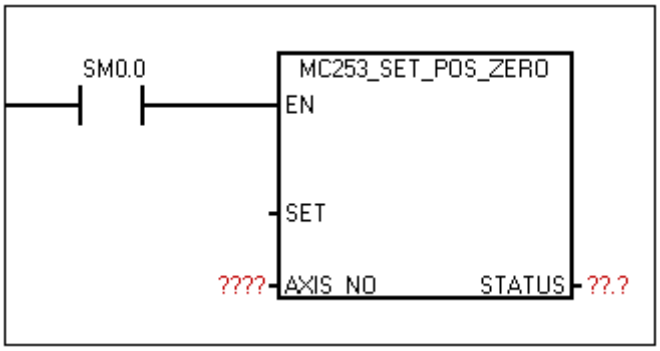
		command is received, it would stop outputting pulses without deceleration.			Soft_Stop==0 that it is running. 2. RUN internal reset when E_STOP = 1.
SOFT_STOP	IN	Soft stop bit. 1: Valid, 0: Invalid Output pulse would slow to a stop while a valid stop command is received.	Bool	0/1	Only RUN ==1 & E_Stop==0 & SOFT_STOP ==0 that it is running.
DIR	IN	Direction of pulse	Bool	0/1	It can be modified during running process.
AXIS_NO	IN	Set axis No. (Every EM253 module has two axes, axis range is up to the number of EM253 module)	Byte	0~255	It could not be modified during running process.
MIN_SPEED	IN	Min. Speed, which is the start or stop speed. Unit: Hz	Dword	0~200000	1.When SET_SPEED =0, there are no pulses outputted; When MIN_SPEED ≠ 0 & <92, module defaults to 92; If SET_SPEED is less than MIN_SPEED, the value of MIN_SPEED would default to the value of SET_SPEED. 2.It can be modified during
SET_SPEED	IN	Setting speed, output pulse would accelerate or decelerate to this speed before stop command is received.	Dword	0~200000	

<table>
<tr><td></td><td></td><td></td><td></td><td></td><td>running process.</td></tr>
<tr><td>TA</td><td>IN</td><td>Acceleration time, the time from min. speed to a setting speed.
Unit: ms</td><td>Dword</td><td>0~10000
(Refer to additional comment1)</td><td rowspan="2">It can be modified during running process. (TA setting refer to additional comment2).</td></tr>
<tr><td>TD</td><td>IN</td><td>Deceleration time, the time from setting speed to min. speed.
Unit: ms</td><td>Dword</td><td>0~10000
(Refer to additional comment1)</td></tr>
<tr><td>STATUS</td><td>OUT</td><td>Output status bit:
<table><tr><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr></table>Bit0: Error flag of parameter configuration
1—Configuration error
0—Configuration is correct.
Bit1: Running flag
1—It is running, pulse is being outputting, and instruction has not been completed.
0—Does not run, public resources are occupied by other instructions so that this instruction has not been executed. Or this parameter has been completed.
Bit2: Finish flag
1—Finished, instruction has been executed.
0—Unfinished, instruction has not been executed or instruction is been performing but not finished.
Bit3: Busy flag
1—Busy flag is valid, the axis is occupied by other instructions.
0—Busy flag is invalid, instruction is performing or has been</td><td>Byte</td><td>0~255</td><td>Bit0:
1.Only judge for axis and TA/TD configuration.
2.MIN_SPEED/SET_SPEED would not report errors, and could set to a proximal value automatically.</td></tr>
</table>

		finished. Bit4: Emergency stop flag (Refer to **additional comment3**) 1—Emergency stop flag is valid, the axis is disabled for external conditions. 0—Emergency stop flag is invalid. Bit5~Bit6: Reserved Bit7: Flag bit of communication status 1—Communication timeout 0—No timeout			
ACT _SPEED	OUT	Outputs as present speed (Frequency)	Dword	500~200000	The value would be some deviation with actual speed, the deviation is less than 5K, it's related to TA and SET_SPEED.

3.3.5 Software reset instruction

① Instruction name: MC253_SET_POS_ZERO



② Function: Reset absolute coordinates.

Tip

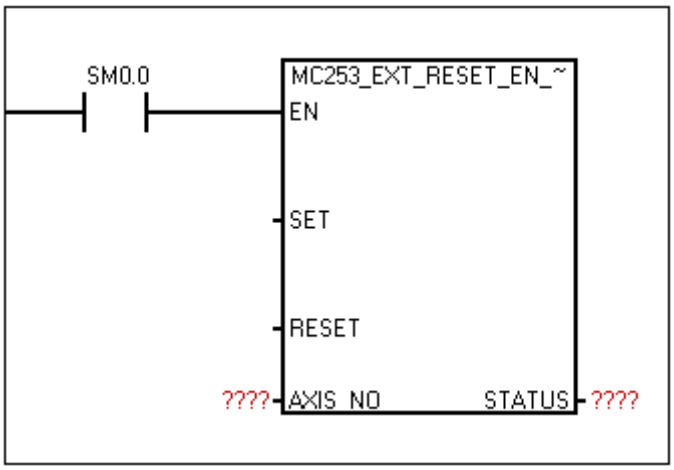
Call this instruction as machine is moving to a position, which means set the origin point of the axis to this position. Every time you call "Read absolute coordinate" that the coordinate of the point could obtain.

③ Parameters

Name	Input /Output	Description	Data type	Range	Remark
SET	IN	Enable bit of zero clearing. Rising edge of "SET", to set the absolute coordinate to 0. "SET" sets to 0 first and then sets to 1 whenever it's called.	Bool	0~1	
AXIS_NO	IN	Set axis No. (Every EM253 module has two axes, axis range is up to the number of EM253 module)	Byte	0~255	
STATUS	OUT	Flag bit of communication status: 1: Communication timeout	Bool	0~1	

3.3.6 External reset coordinate enabling instruction

① Instruction name: MC253_EXT_RESET_EN_EXT



② Function: Calling this instruction is to select to enable external IO reset absolute coordinate or not.

Correspondence between Axis No. and External reset signal:

Axis 0 ——I0.2 (MC253_HSC0)

Axis 1 ——I0.6 (MC253_HSC1)

③ Parameters

Name	Input /Output	Description	Data type	Range	Remark
SET	IN	Rising edge of "SET", to enable the external reset signal. "SET" resets first and then sets to 1 whenever it's called.	Bool	0~1	
RESET	IN	Rising edge of "RESET", to disable the external reset signal. "RESET" resets first and then sets to 1 whenever it's called.	Bool	0~1	
AXIS_NO	IN	Set axis No. (Every EM253 module has two axes, axis range is up to the number of EM253 module)	Byte	0~255	
STATUS	OUT	Status bit: 7 6 5 4 3 2 1 0 Bit0: Flag bit of reset status 1—Reset is finished 0—Reset has not been finished yet Bit1~Bit6: Reserved Bit7: Flag bit of communication status 1—Communication timeout 0—No timeout	Byte	0~255	

④ Instruction for use

Assume axis 0 call this instruction. After external reset signal is enabled by SET rising edge, if "Effective reset signal" is detected by I0.2, system reset absolute coordinate for axis 0, and STATUS is set to 1, which indicates the reset is completed. After external reset signal is inhibited by RESET rising edge, system would not reset the absolute coordinate for axis 0 even though "Effective reset signal" is detected by I0.2, and STATUS is cleared, which indicates reset has not been finished.

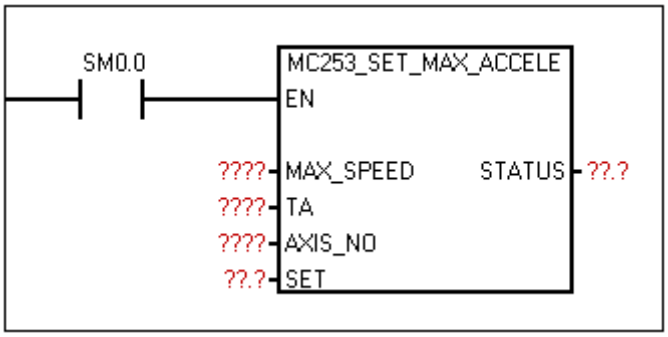
Tip

"Effective reset signal", there is an external IO corresponds to reset signal per axis, the effective level is set by the corresponding register. eg. Axis 0 corresponds to I0.2, effective reset level of axis 0 can be set by CONTROL bit 0 of MC253_HSC_INIT of MC253_HSC0, when sets to 0, the effective reset signal of axis 0 is the high level status

of I0.2; When sets to 1, effective reset signal of axis 0 is the low level status of I0.2; All the setting can be valid only when high-speed counter (axis 0 corresponds to MC253_HSC0) has been enabled, otherwise system would take high level as effective reset signal by default. Such as axis 0, high level of I0.2 is the effective reset signal. Other axes are the same, Correspondence between Axis No. and External reset signal refer to ② of "3.3.6 External reset coordinate enabling instruction".

3.3.7 Set maximum acceleration instruction

① Instruction name: MC253_SET_MAX_ACCELE



② Function: Set max. Acceleration (= MAX_SPEED/TA) (TA≠0)

If this instruction has not been called, it's considered that the max. Acceleration has not been set.

③ Parameters

Name	Input /Output	Description	Data type	Range	Remark
MAX_SPEED	IN	Max. Speed for long axis, which is the max. Running Speed. Unit: Hz	Dword	0~200000	It can be modified during running process.
TA	IN	Acceleration/ Deceleration time. Unit: ms	Dword	0~10000 (Refer to **additional comment 1**)	It can be modified during running process. If TA=0, which is considered no max. acceleration.
AXIS_NO	IN	Set axis No.(Every EM253 module has two axes, axis	Byte	0~255	

		range is up to the number of EM253 module)			
SET	IN	As above parameters were set, the settings would be valid by rising edge of "SET".	Bool	0~1	
STATUS	OUT	Flag bit of communication status: 1: Communication timeout	Bool	0~1	

④ Instructions for use

If set X axis TA to 0, or X axis has not called this instruction, so X axis is considered not to set max. Acceleration, otherwise it has set max. Acceleration, so MAX_ACCELE=MAX_SPEED/TA. The meaning is detailed as follows:

■ Set an suitable acceleration to limit the speed for instructions in the axis

eg. for PTP, set AXIS_NO=0, MIN_SPEED=1000, MAX_SPEED=11000,TA=500, so in theory, acceleration for PTP is 20Hz/ms (= (MAX_SPEED-MIN_SPEED) /TA) ; If axis 0 has called MC253_SET_MAX_ACCELE, to set the max. Acceleration to 15Hz/ms, so the actual acceleration of PTP is 15Hz/ms (MC253_LINE_R and MC253_CIRCLE_R are the same)

■ Instruction in the axis is going to obtain the max. acceleration

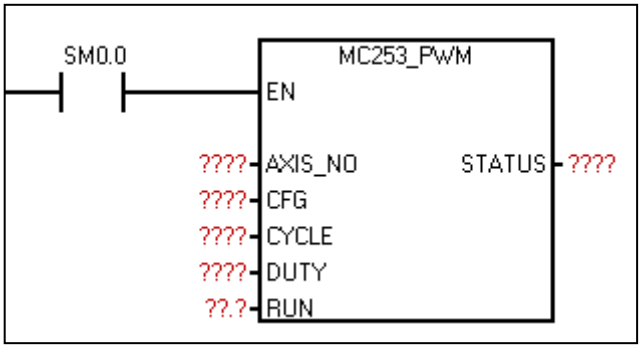
For example, if PTP instruction is running as max. acceleration, first you can call MC253_SET_MAX_ACCELE to set max. Acceleration (TA of MC253_SET_MAX_ACCELE cannot be 0, otherwise the max. Acceleration would not be obtained), and also need set TA of PTP to 0. If max. Acceleration is not set and TA of PTP =0, the parameter of PTP would report error (MC253_LINE_R and MC253_CIRCLE_R are the same)

Tip

- ❖ For double axis instruction "MC253_LINE_R" and "MC253_CIRCLE_R", if both axis were set the max. Acceleration, so the smaller one would be regarded as the max. Acceleration of the double axis system; If only one axis set the max. Acceleration, so it is the max. Acceleration of the double axis system; If both axis were not set the max. Acceleration, so the double axis system has no limit for speed.
- ❖ After the parameters like MAX_SPEED, TA, AXIS_NO of this instruction were confirmed, they would be effective after the rising edge of SET is entered.

3.3.8 Pulse width modulation instruction

① Instruction name: MC253_PWM



② Function: It can output different cycle and duty pulses by setting cycle and duty parameters.

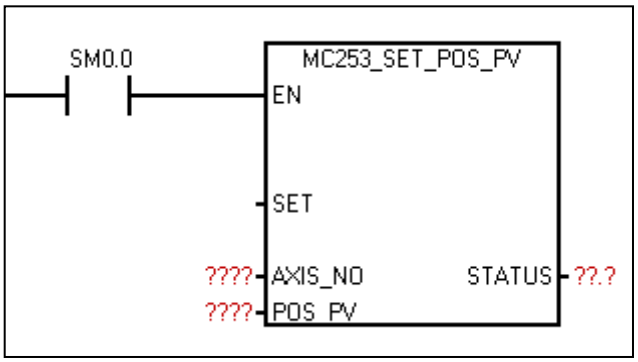
③ Parameters

Name	Input /Output	Description	Data type	Range	Remark
AXIS_NO	IN	Set axis No.(Every EM253 module has two axes, axis range is up to the number of EM253 module)	Byte	0~255	It could not be modified during running process.
CFG	IN	Base time unit 0: 1us, 1: 0.5ms	Byte	0~1	
CYCLE	IN	Pulse cycle	Word	2~65535	
DUTY	IN	Pulse duty cycle	Word	0~65535	
RUN	IN	Enable running	Bool	0~1	
STATUS	OUT	Output status bit: 7 6 5 4 3 2 1 0 Bit0: Error flag of parameter configuration 1—Configuration error 0—Configuration is correct. Bit1: Running flag 1—It is running, pulse is being outputting, and instruction has not been completed. 0—Does not run, public resources are occupied by other instructions so that this instruction has not been executed. Or this parameter has been completed. Bit2: Finish flag 1—Finished, instruction has been executed.	Byte	0~255	Bit0: 1. Only judge for axis configuration. 2. CYCLE / DUTY would not report errors, and could set to a proximal value automatically.

		0—Unfinished, instruction has not been executed or instruction is been performing but not finished. Bit3: Busy flag 1—Busy flag is valid, the axis is occupied by other instructions. 0—Busy flag is invalid, instruction is performing or has been finished. Bit4: Emergency stop flag (Refer to **additional comment 3)** 1—Emergency stop flag is valid, the axis is disabled for external conditions. 0—Emergency stop flag is invalid. Bit5~Bit6: Reserved Bit7: Flag bit of communication status 1—Communication timeout 0—No timeout			

3.3.9 Setting target position instruction

① Instruction name: MC253_SET_POS_PV



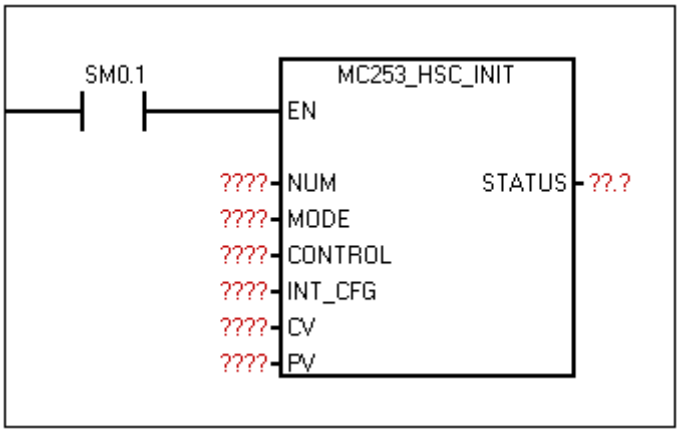
② Function: To write the absolute position of machine to the module. eg. If machine power off at certain position, it can save the current position of the machine, and this position would be written back to module till power on next time, so the absolute position of module is the same as actual starting position of machine, that machine no need to back to origin point; If the position is just the origin point, this instruction has the same effect with MC253_SET_POS_ZERO.

③ Parameters

Name	Input /Output	Description	Data type	Range	Remark
AXIS_NO	IN	Set axis No.(Every EM253 module has two axes, axis range is up to the number of EM253 module)	Byte	0~255	
SET	IN	Rising edge of "SET", to enable the external reset signal. "SET" resets first and then sets to 1 whenever it's called.	Bool	0~1	
POS_PV	IN	Set target position, has both positive and negative value. Output positive pulses represent the positive direction of X axis, negative pulses represent the negative direction of X axis.	Dint	-2147483648 ~ +2147483647	
STATUS	OUT	Flag bit of communication status: 1—Communication timeout	Bool	0~1	

3.3.10 Setting module high speed counter instruction

① Instruction name: MC253_HSC_INIT



② Function: Configure high-speed counter.

Remark: This instruction is only executed once at the first scan cycle when CPU powered up.

③ Parameters

Name	Input	Description	Data	Range	Remark

	/Output		type		
NUM	IN	High speed counter label (Every EM253 module has two HSC, label range is up to the number of EM253 module)	Byte	0~255	
MODE	IN	Count mode	Byte	0~12	See attached table.
CONTROL	IN	Control byte: 7 6 5 4 3 2 1 0 Bit0: Control byte of reset effective level 1—Low level 0—High level Bit1: Reserved Bit2: Selection of quadrature counter rate 1—1x rate 0—4x rate Bit3: Direction control byte 1—Increase counter 0—Decrease counter Bit4: Update direction 1—Update direction 0—No update Bit5: Update preset value 1—Write new preset value 0—No update Bit6: Update current value 1—Write new current value 0—No update Bit7: Valid byte 1—Valid 0—Invalid	Byte	0~255	
INT_CFG	IN	Interrupt configuration (Reserved)	Byte	Reserved	Reserved
CV	IN	Current value	Dword	-2147483648 ~ +2147483647	
PV	IN	Preset Value	Dword	-2147483648 ~ +214748364	

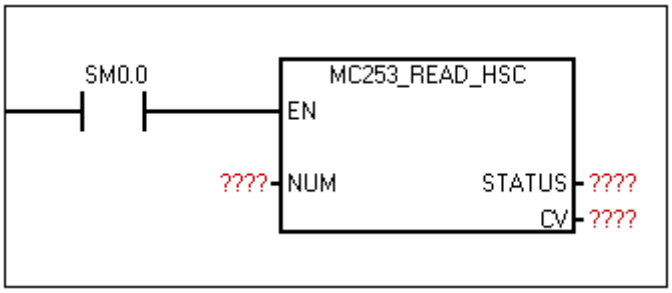
				7	
STATUS	OUT	Flag bit of communication status: 1—Communication timeout	Bool	0~1	

Attached table

Mode	Description	Inputs		
	MC253_HSC0	I0.0	I0.1	I0.2
	MC253_HSC1	I0.4	I0.5	I0.6
0	Single-phase counter with internal direction control	Clock		
1		Clock		Reset
3	Single-phase counter with external direction control	Clock	Direction	
4		Clock	Direction	Reset
9	A/B phase quadrature counter	Clock A	Clock B	
10		Clock A	Clock B	Reset

3.3.11 Read module high speed counter state instruction

① Instruction name: MC253_READ_HSC



② Function: Read the counter status and current value of module's high-speed counter.

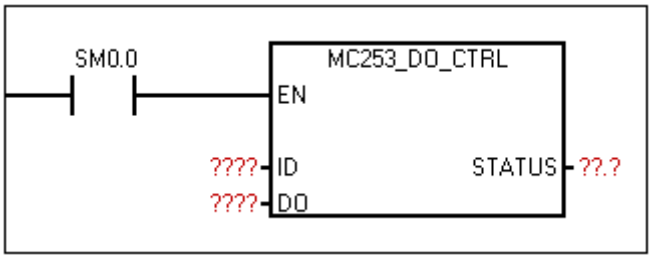
③ Parameters

Name	Input /Output	Description	Data type	Range	Remark
NUM	IN	High speed counter label (Every EM253 module has two HSC, label range is up to the number of EM253 module)	Byte	0~255	
STATUS	OUT	Status byte: 7 6 5 4 3 2 1 0 Bit0: Flag bit of communication status: 1—Communication timeout Bit1~Bit4: Reserved Bit5: Counter direction byte 1—Increase counter	Byte	0~255	

		Bit6: Current value is equal to preset value 1—Equal to Bit7: Current value is greater than preset value 1—Greater than			
CV	OUT	Current Value	Dword	-2147483648 ~ +2147483647	

3.3.12 Control module output instruction

① Instruction name:MC253_DO_CTRL



② Function: Control the output value of EM253 module.

③ Parameters

Name	Input /Output	Description	Data type	Range	Remark
ID	IN	ID No. of motion control module	Byte	0~255	It's fit for motion control module only.
DO	IN	Output value of module	Byte	0~255	DO corresponds to 8 Q outputs, that is bit0 of DO corresponds to Q0.0, and so on.
STATUS	OUT	Flag bit of communication status: 1: Communication timeout	Bool	0~1	

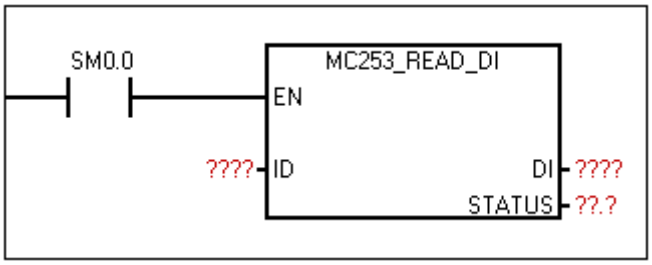
Tip

- ID No. of motion control module is internal unified. For example, if PLC connects 3 modules, module 0 is motion control module, module 1 is not motion control module, module 2 is motion control module, so the ID No. for module 0 is ID0, module 2 is ID1.

- If CPU connects one EM253 module, due to transmission delay, the instruction delay time from enabling to execution is 780us; If CPU connects 5 EM253 module, the delay time from enabling to execution is about 930us.

3.3.13 Read module input state instruction

① Instruction name: MC253_READ_DI



② Function: Read the input value of motion control module.

③ Parameters

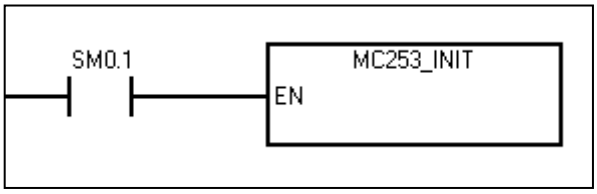
Name	Input /Output	Description	Data type	Range	Remark
ID	IN	ID No. of motion control module	Byte	0~255	It's fit for motion control module only.
DI	OUT	Input value of module	Byte	0~255	DI corresponds to 8 I inputs, that is bit0 of DI corresponds to I0.0, and so on.
STATUS	OUT	Flag bit of communication status: 1—Communication timeout	Bool	0~1	

Tip

- ID No. of motion control module is internal unified. For example, if PLC connects 3 modules, module 0 is motion control module, module 1 is not motion control module, module 2 is motion control module, so the ID No. for module 0 is ID0, module 2 is ID1.
- When some axis call motion control instructions, and instructions conform to running conditions, the axis port outputs pulse or direction according to the instruction during running process (The corresponding ports are not controlled by MC253_DO_CTRL), the port would restore to common IO function after running is completed (MC253_DO_CTRL can be received); Users should be care to use MC253_DO_CTRL, so as not to effect the motion control requirement.

3.3.14 Motion control module initialization instruction

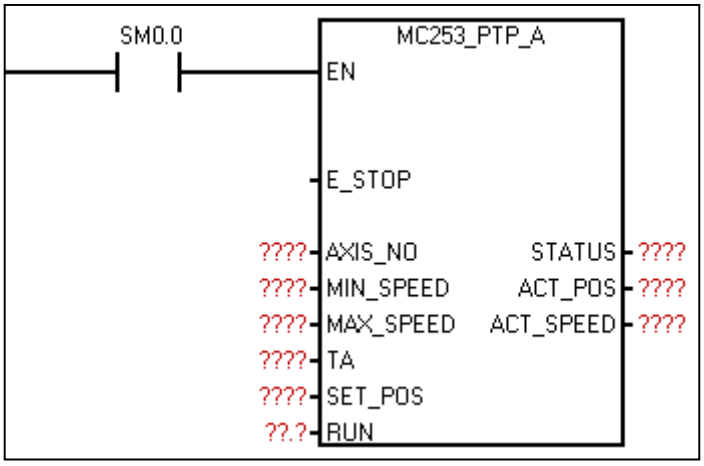
① Instruction name: MC253_INIT



② Function: To initialize the system control variable of motion control module (Refer to "※Matters need to pay special attention 1" of chapter 3.3) .

3.3.15 Single shaft absolute motion instruction

① Instruction name: MC253_PTP_A



② Function: Used as the point to pint control for single shaft (fixed point rather than fixed-length). Output pulses to the specified number on the basis of original pulses if call once, which is set by max. Speed, min. Speed and TA/TD, the output pulses would accelerate to the max. Speed at the starting time, as the pulses is going to be finished, the pulse frequency would slow down, so as to avoid the vibration or locking caused by great inertia at the starting or stopping of the machine.

③ Parameters

Name	Input /Output	Description	Data type	Range	Remark
E_STOP	IN	Emergency stop bit. 1: Valid 0: Invalid	Bool	0/1	1. Only Run =1 and E_Stop=0 that it is running. 2. RUN internal reset when E_STOP=1.
AXIS_NO	IN	Set axis No.	Byte	0~255	It could not be modified during running process.

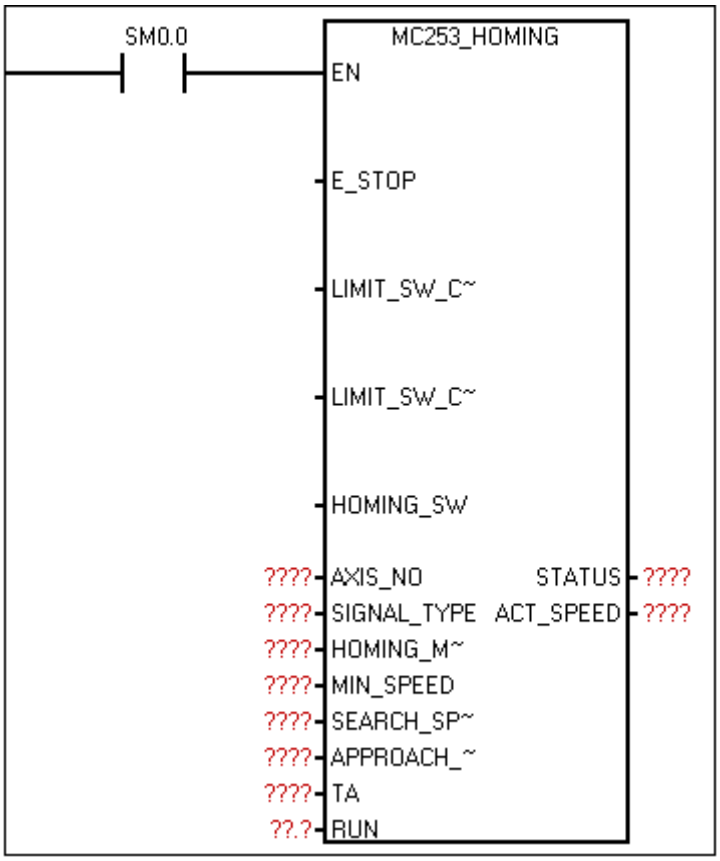
MIN_SPEED	IN	Min. Speed, which is the start or stop speed. Unit: Hz	Dword	500~200000	1. Min. speed must be less than the Max. Speed. 2. It can be modified during running process.
MAX_SPEED	IN	Max. Speed, which is the max. Running Speed. Unit: Hz	Dword	500~200000	
TA	IN	Acceleration / Deceleration Time Unit: ms	Dword	0~10000 (Refer to **additional comment1**)	It can be modified during running process (TA setting refer to **additional comment 2**)
SET_POS	IN	Output pulses, has both positive and negative value. Positive pulses represent the positive direction of X axis, negative pulses represent the negative direction of X axis (Absolute coordinate).	Dint	-2147483648 ~ +2147483647	It can be modified during running process. As the new value is greater than the outputted pulses, so the final output pulses would be subjected to the new value. If new value is less than the outputted pulses, the outputting pulses would be stopped immediately.
RUN	IN/OUT	Enabling bit for running: 1: Valid 0: Invalid	Bool	0/1	1. Only RUN=1 & E_STOP=0 that it is running. 2. RUN internal reset when running is completed. 3. RUN internal reset when E_STOP = 1.

STATUS	OUT	Output status bytes: 7 6 5 4 3 2 1 0 Bit0: Error flag of parameter configuration 1—Configuration error 0—Configuration is correct. Bit1: Running flag 1—It is running, pulse is being outputting, and instruction has not been completed. 0—Does not run, public resources are occupied by other instructions so that this instruction has not been executed. Or this parameter has been completed. Bit2: Finish flag 1—Finished, instruction has been executed. 0—Unfinished, instruction has not been executed or instruction is been performing but not finished. Bit3: Busy flag 1—Busy flag is valid, the axis is occupied by other instructions. 0—Busy flag is invalid, instruction is performing or has been finished. Bit4: Emergency stop flag (Refer to additional comment3) 1—Emergency stop flag is valid, the axis is disabled for external conditions. 0—Emergency stop flag is invalid. Bit5~Bit6: Reserved Bit7: Flag bit of communication status 1—Communication timeout 0—No timeout	Byte	0~255	Bit0 : 1. Only judge for axis parameters. 2. MIN_SPEED/ MAX_SPEED/TA would not report errors, and could set to a proximal value automatically.
ACT_POS	OUT	Current absolute coordinates	Dint	-2147483648 ~ +2147483647	

ACT_SPEED	OUT	Present actual running speed	Dword	500~200000	The value would be some deviation with actual speed, the deviation is less than 5K, it's related to TA and SET_SPEED.

3.3.16 Homing instruction

① Instruction name: MC253_HOMING



② Function: Search the origin point of device by setting homing mode parameters.

Correspondence between axis No. and external reset IO signal:

Axis 0 → I0.2 (MC253_HSC0)

Axis 1 → I0.6 (MC253_HSC1)

If homing mode refers to origin switch (Mode 3 or 4), please connect origin switch to above points.

③ Parameters

Name	Input/ Output	Description	Data type	Range	Remark
E_STOP	IN	Emergency stope bit: 1: Valid, 0: Invalid	Bool	0~1	1. Only RUN =1 and E_STOP =0 that it is running. 2. RUN internal

					reset when E_STOP = 1.
LIMIT_SW_CCW	IN	CCW limit switch	Bool	0~1	
LIMIT_SW-CW	IN	CW limit switch	Bool	0~1	
HOMING_SW	IN	Origin switch	Bool	0~1	
AXIS_NO	IN	Axis No.	Byte	0~255	It could not be modified during running process.
SIGNAL_TYPE	IN	7 6 5 4 3 2 1 0 Bit0: CCW limit switch signal 0—High level 1—Low level Bit1: CW limit switch signal 0—High level 1—Low level Bit2: Origin switch signal 0—High level 1—Low level	Byte	0~255	
HOMING_MODE	IN	Homing mode	Byte	1~14	Refer to additional comment4 for details.
MIN_SPEED	IN	Min. Speed Unit: Hz	Dword	0~200000	1. Min. Speed should set to less than max. speed. 2. It can be modified during running process. 3. Searching speed should not be too high, approaching speed should be low.
SEARCH_SPEED	IN	Searching speed Unit: Hz	Dword	0~200000	
APPROACH_SPEED	IN	Approaching speed Unit: Hz	Dword	0~200000	
TA	IN	Acceleration/Deceleration time Unit: ms	Dword	0~10000 (Refer to additional comment 1)	It can be modified during running process. (TA setting refer to additional comment2)
RUN	IN	Enabling bit for running: 1: Valid	Bool	0~1	1. Only RUN =1 and E_STOP =0 that it is running. 2. RUN internal reset when running is completed. 3. RUN internal reset when E_STOP=1.

STATUS	OUT	Output status byte: 7 6 5 4 3 2 1 0 Bit0: Error flag of parameter configuration 1—Configuration error 0—Configuration is correct. Bit1: Running flag 1—It is running 0—Does not run Bit2: Finish flag 1—Finished, instruction has been finished. 0—Unfinished, instruction has not been executed. Bit3: Busy flag 1—Busy flag is valid, the axis is occupied by other instructions. 0—Busy flag is invalid, instruction is performing or has been finished. Bit4: Emergency stop flag (Refer to additional comment 3) 1—Emergency stop flag is valid, the axis is disabled for external conditions. 0—Emergency stop flag is invalid. Bit5: Find the origin point or note 1—Origin point has been found. 0—Origin point has not been found. Bit6: Reserved	Byte	0~255	Bit0 : 1. Only judge for over range of axis parameters and Homing mode. 2. Other parameters would not report errors, and could set to a proximal value automatically.

		Bit7: Flag bit of communication status 1—Communication timeout 0—No timeout			
ACT_SPEED	OUT	Current speed	Dword	0~200000	The value represents the speed of long axis, which would be some deviation with actual value, the deviation is less than 5K, it's related to TA and SET_SPEED.

④ Remark

Program uses scanning way to detect switch signal, so changing switch signal would not be handled in time, which would be some delay.

If homing speed (Contain searching speed and approaching speed) are too high, the delay time would be longer, which would result in inaccurate homing.

Additional Comment

1. If TA≠0, Acceleration= (MAX_SPEED-MIN_SPEED) /TA (TA would subject to max. Acceleration when the max. Acceleration has been set).

If TA=0, set max. Acceleration by MC253_SET_MAX_ACCELE, if the max. Acceleration has not been set, the parameter would report error. So as TD. For instructions of double axis, if both axis were set max. Acceleration, the lesser one would be regarded as Acceleration of the system.

2. In theory, instruction acceleration=【(MAX_SPEED - MIN_SPEED) / TA】, If the instruction acceleration is too small (less than 1), the acceleration would default to 1 in the internal of instruction. Users should set reasonable value for TA/TD according to the requirements.

3. There is "module emergency stop signal" in module input port, as this signal is detected by module, which would inhibit pulse output, and report alarm at the status bit of pulse output instruction (MC253_PTP_R, MC253_SPEED_CTL, MC253_PWM)

Correspondence between Axis No. and emergency stop signal:

Axis 0 → I0.3

Axis 1 → I0.7

4. Homing Function

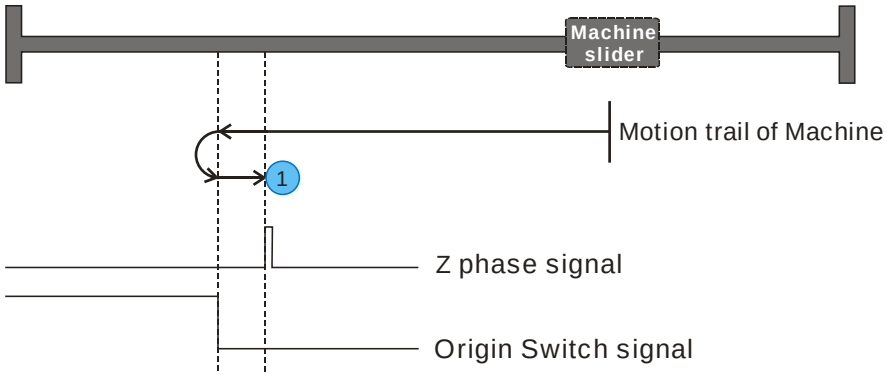
There are total 14 homing modes, users could select the modes according to its accuracy requirements.

Mode	Definition
1	Refer to both the negative origin switch and Z phase signal

2	Refer to both the positive origin switch and Z phase signal
3	Refer to the negative origin switch only
4	Refer to the positive origin switch only
5	Refer to Z phase signal only (Homing towards negative direction)
6	Refer to Z phase signal only (Homing towards positive direction)
7	Refer to origin switch, positive limit switch, Z phase signal (Gather Z phase signal at the left of the left edge of the positive origin switch)
8	Refer to origin switch, positive limit switch, Z phase signal (Gather Z phase signal at the right of the left edge of the positive origin switch)
9	Refer to origin switch, positive limit switch, Z phase signal (Gather Z phase signal at the left of the right edge of the positive origin switch)
10	Refer to origin switch, positive limit switch, Z phase signal (Gather Z phase signal at the right of the right edge of the positive origin switch)
11	Refer to origin switch, negative limit switch, Z phase signal (Gather Z phase signal at the right of the right edge of the positive origin switch)
12	Refer to origin switch, negative limit switch, Z phase signal (Gather Z phase signal at the left of the right edge of the positive origin switch)
13	Refer to origin switch, negative limit switch, Z phase signal (Gather Z phase signal at the right of the left edge of the positive origin switch)
14	Refer to origin switch, negative limit switch, Z phase signal (Gather Z phase signal at the left of the left edge of the positive origin switch)

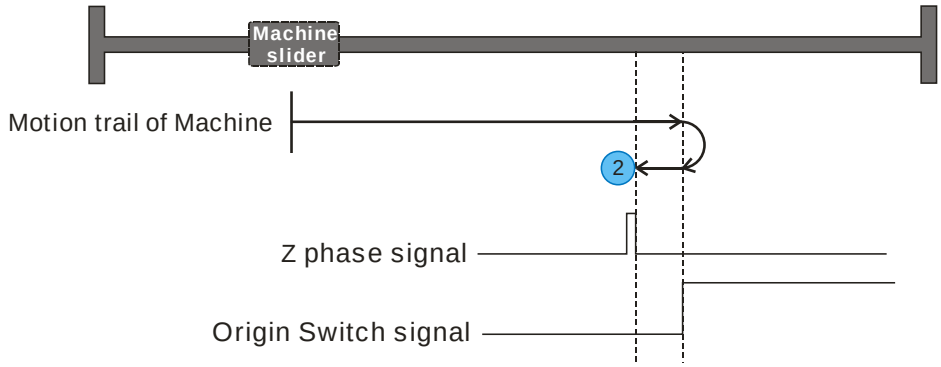
No matter where the machine is, the origin point which the servo is searching for is unique after the Origin switch, Positive limit switch or Negative switch had been fixed. In following diagram, the symbol "|" represents the machine initial position, "X" represents the origin point.

Homing Mode 1: Refer to both the negative origin switch and Z phase signal



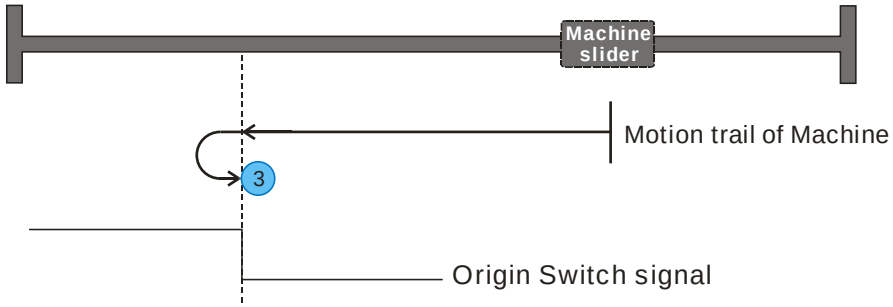
Origin switch is located in the negative direction of the machine. The machine is moving in the direction of origin switch signal, it would slow to a stop while the origin switch signal is detected, then exit from the origin switch signal, turn back to search the next Z phase signal and mark the Z phase signal as the origin point, finally the motor stop immediately .

Homing Mode 2: Refer to both the positive origin switch and Z phase signal



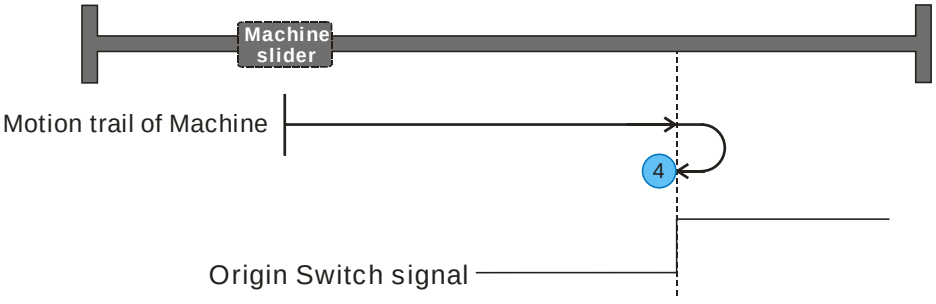
Origin switch signal is located in the positive direction of the machine. The machine is moving in the direction of origin switch signal, it would slow to a stop while the origin switch signal is detected, then exit from the origin switch signal, turn back to search the next Z phase signal and mark the Z phase signal as the origin point, finally the motor stop immediately.

Homing Mode 3: Refer to the negative origin switch only



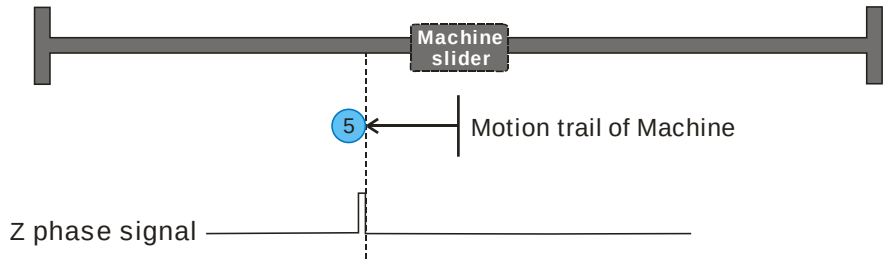
Origin switch signal is located in the negative direction of the machine. The machine is moving in the direction of origin switch signal, it would slow to a stop while the origin switch signal is detected, then exit from the origin switch signal, turn back to search the falling edge of origin switch signal and mark it as the origin point, finally the motor stop immediately.

Homing Mode 4: Refer to the positive origin switch only



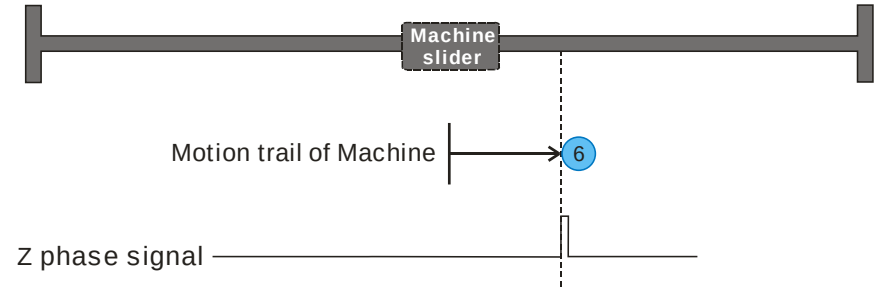
Origin switch signal is located in the positive direction of the machine. The machine is moving in the direction of origin switch signal, it would slow to a stop while the origin switch signal is detected, then exit from the origin switch signal, turn back to search the falling edge of origin switch signal and mark it as the origin point, finally the motor stop immediately.

Homing Mode 5: Refer to Z phase signal only (Homing towards negative direction)



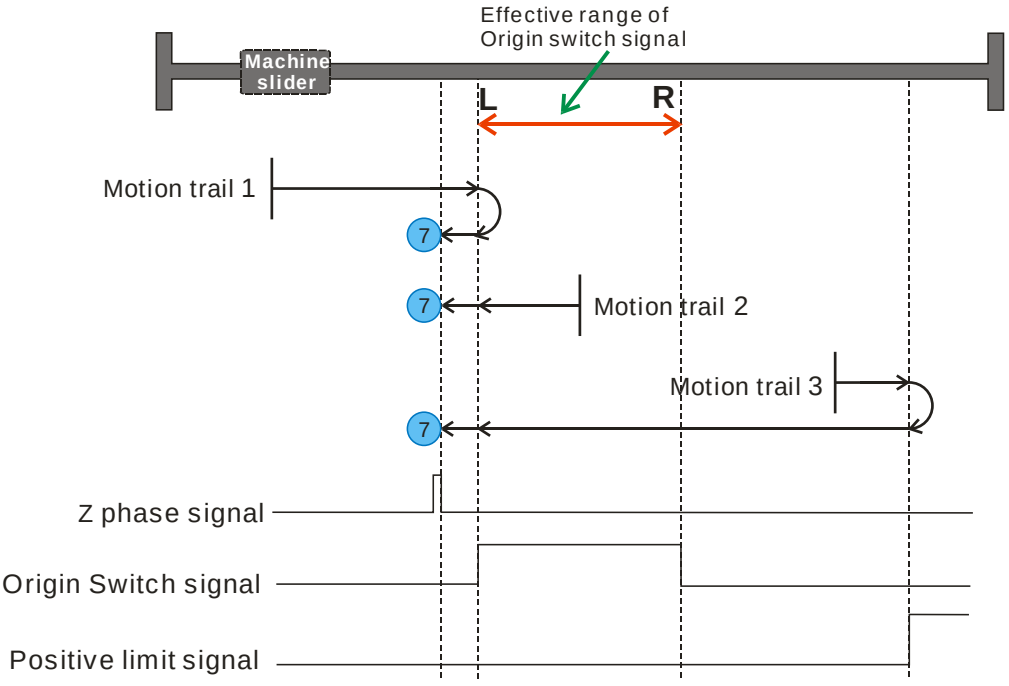
Motor is moving from current position to the negative direction, to find the next Z phase signal and it as the origin point.

Homing Mode 6: Refer to Z phase signal only (Homing towards positive direction)



Motor is moving from current position to the positive direction, to find the next Z phase signal and it as the origin point.

Homing Mode 7: Refer to origin switch, positive limit switch, Z phase signal (Gather Z phase signal at the left of the left edge of the positive origin switch)

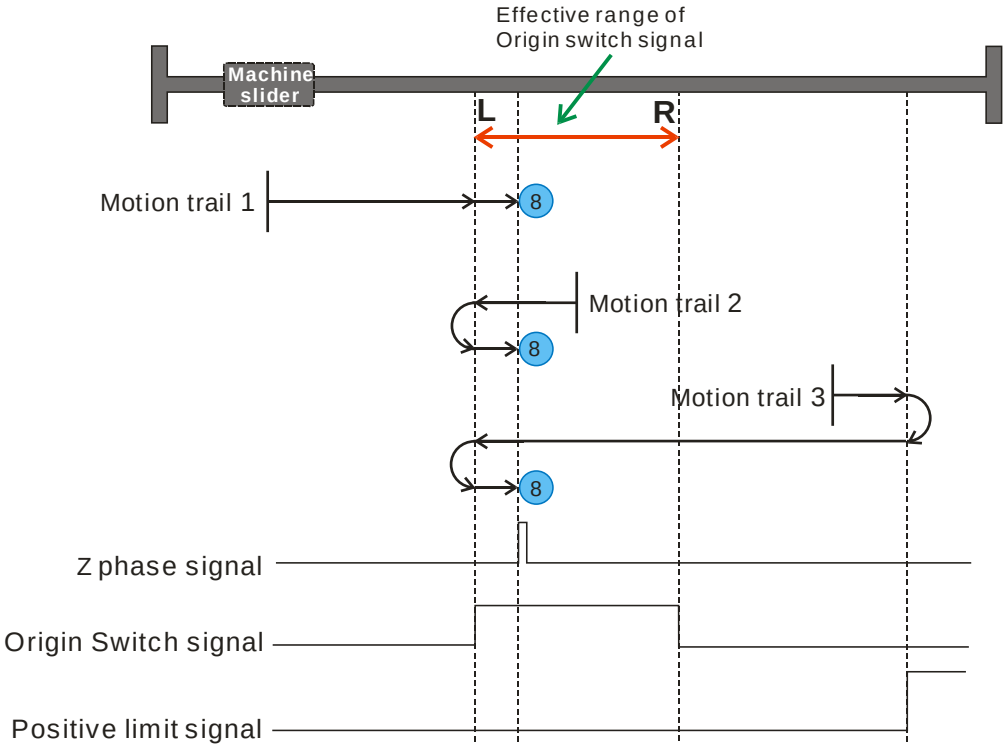


Viewing from above diagram, machine is moving to the positive limit signal, Z phase signal is located at the left of the left edge of the positive origin switch signal, which is outside the range of origin switch signal.

When machine is within the range of origin switch signal (Motion trail 2), it could search the

origin point signal in negative direction; As the machine is out of the range (Motion trail 1 & 3), it moves towards the direction of limit signal, which can find the origin point according to the motion trail (the detected sequence of origin switch signal and positive limit signal).

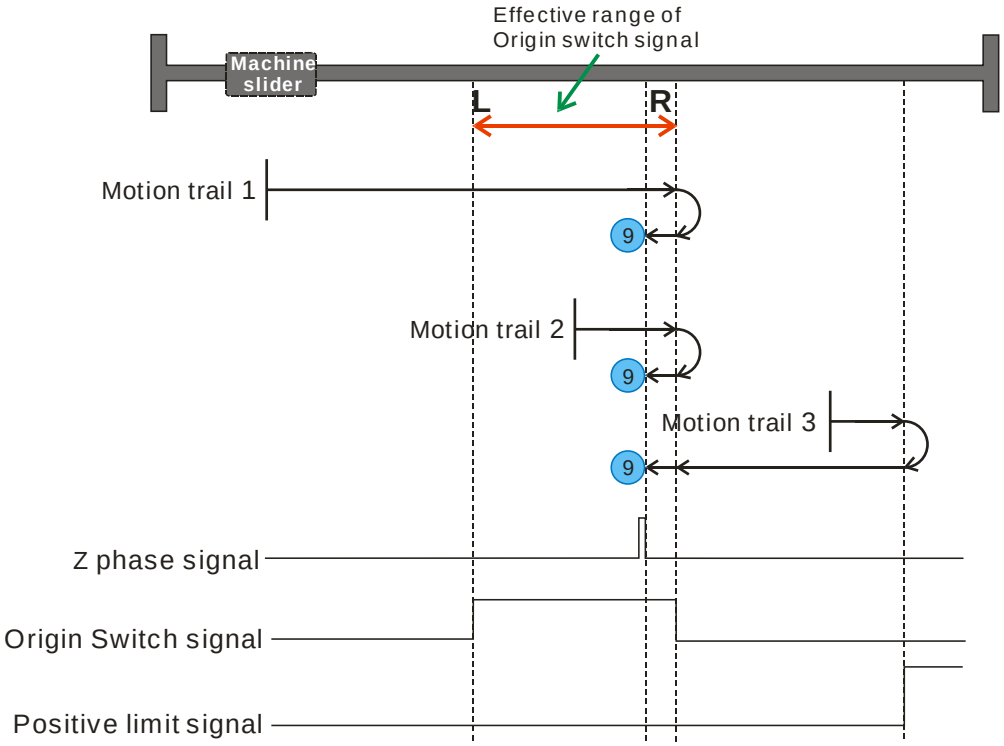
Homing Mode 8: Refer to origin switch, positive limit switch, Z phase signal (Gather Z phase signal at the right of the left edge of the positive origin switch)



Viewing from above diagram, machine is moving to the positive limit signal, Z phase signal is located at the right of the left edge of the positive origin switch signal, which is within the range of origin switch signal.

When machine is within the range of origin switch signal (Motion trail 2), it could search the origin point in negative direction; As the machine is out of the range (Motion trail 1 & 3), it moves towards the direction of limit signal, which can find the origin point according to the motion trail (the detected sequence of origin switch signal and positive limit signal).

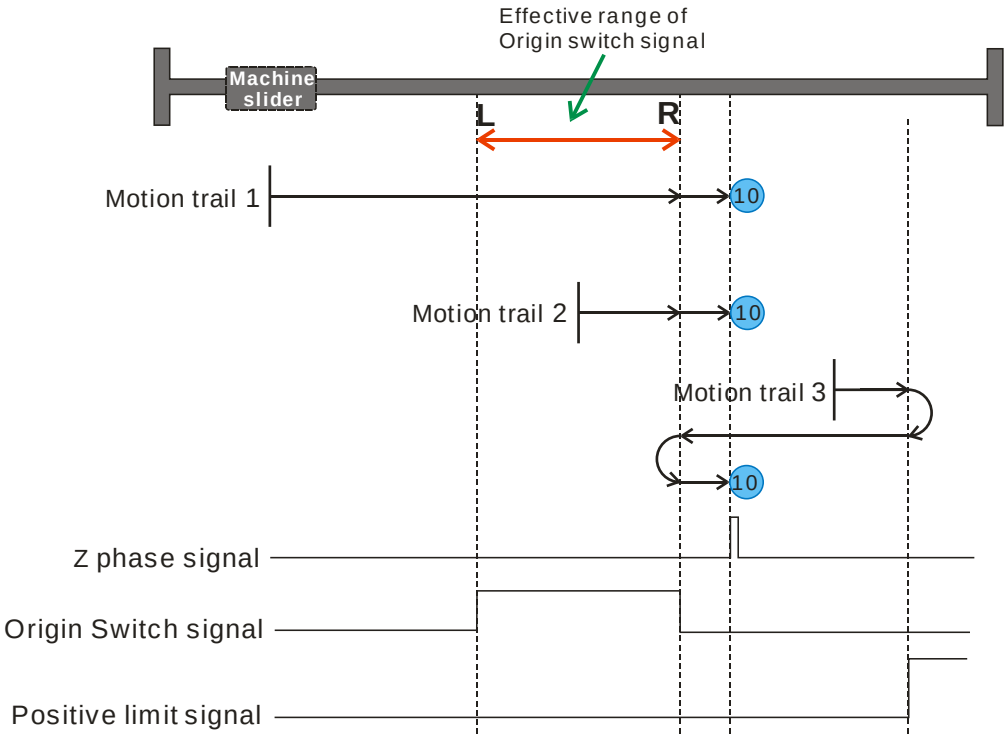
Homing Mode 9: Refer to origin switch, positive limit switch, Z phase signal (Gather Z phase signal at the left of the right edge of the positive origin switch)



Viewing from above diagram, machine is moving to the positive limit signal, Z phase signal is located at the left of the right edge of the positive origin switch signal, which is within the range of origin switch signal.

When machine is within the range of origin switch signal (Motion trail 2), it could search the origin point in positive direction; As the machine is out of the range (Motion trail 1 & 3), it moves towards the direction of limit switch signal, which can find the origin point according to the motion trail (the detected sequence of origin switch signal and positive limit signal).

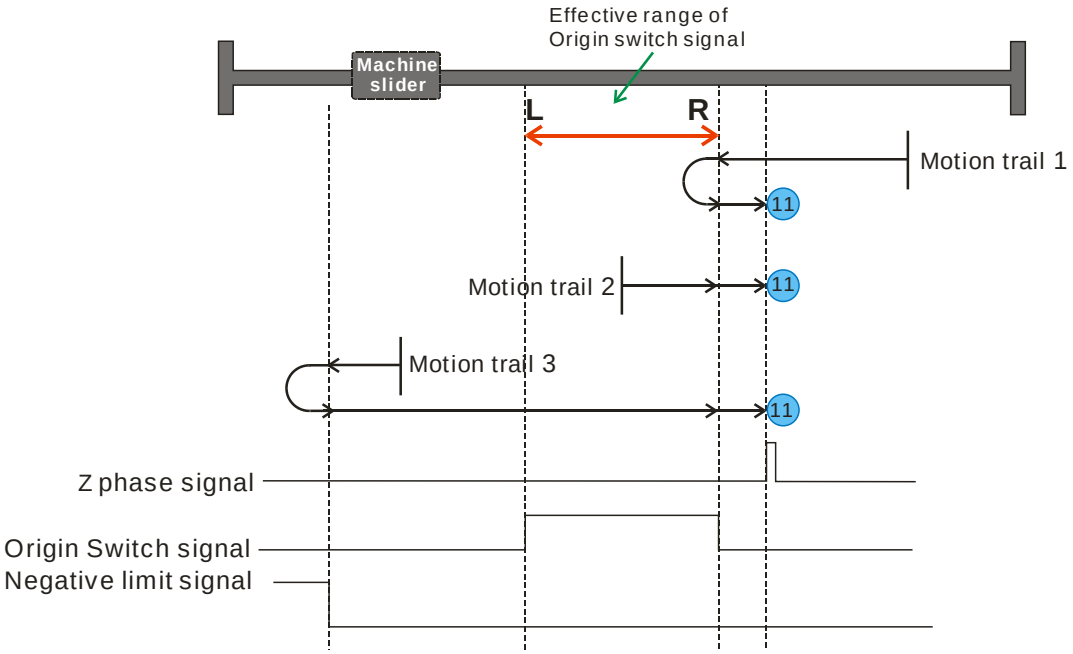
Homing Mode 10: Refer to origin switch, positive limit switch, Z phase signal (Gather Z phase signal at the right of the right edge of the positive origin switch)



Viewing from above diagram, machine is moving to the positive limit signal, Z phase signal is located at the right of the right edge of the positive origin switch signal, which is outside the range of origin switch signal.

When machine is within the range of origin switch signal (Motion trail 2), it could search the origin point in positive direction; As the machine is out of the range (Motion trail 1 & 3), it moves towards the direction of limit signal, which can find the origin point according to the motion trail (the detected sequence of origin switch signal and positive limit signal).

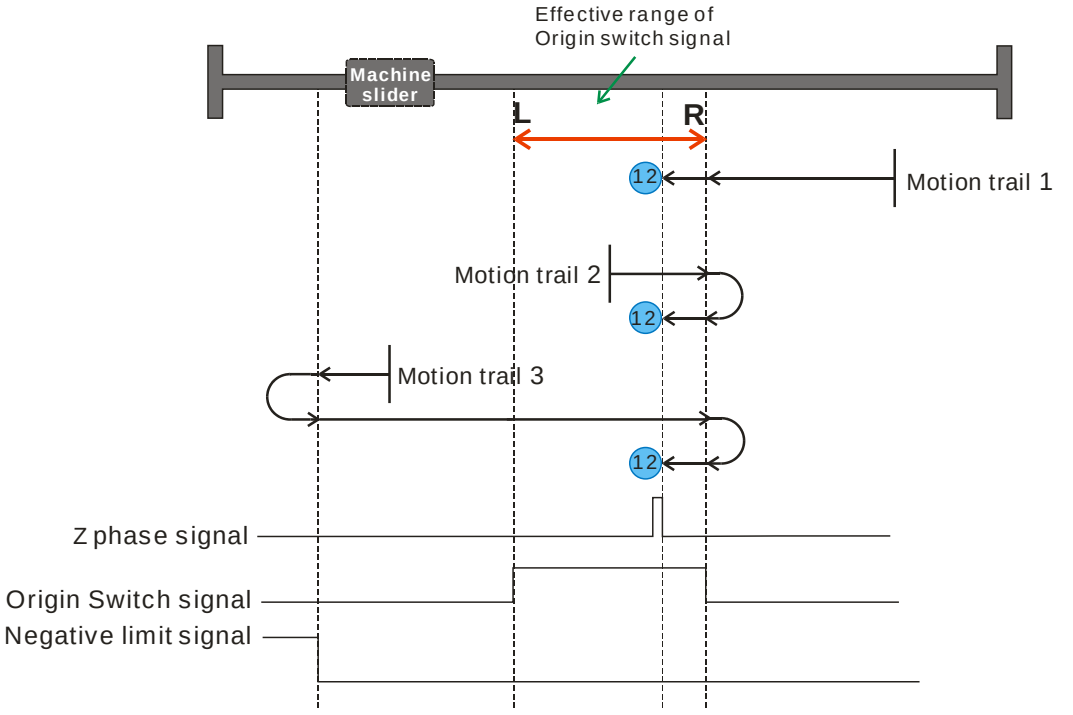
Homing Mode 11: Refer to origin switch, negative limit switch, Z phase signal (Gather Z phase signal at the right of the right edge of the positive origin switch)



Viewing from above diagram, machine is moving to the negative limit signal, Z phase signal is located at the right of the right edge of the positive origin switch signal, which is outside the range of origin switch signal.

When machine is within the range of origin switch signal (Motion trail 2), it could search the origin point in positive direction; As the machine is out of the range (Motion trail 1 & 3), it moves towards the direction of limit signal, which can find the origin point according to the motion trail (the detected sequence of origin switch signal and negative limit signal).

Homing Mode 12: Refer to origin switch, negative limit switch, Z phase signal (Gather Z phase signal at the left of the right edge of the positive origin switch)

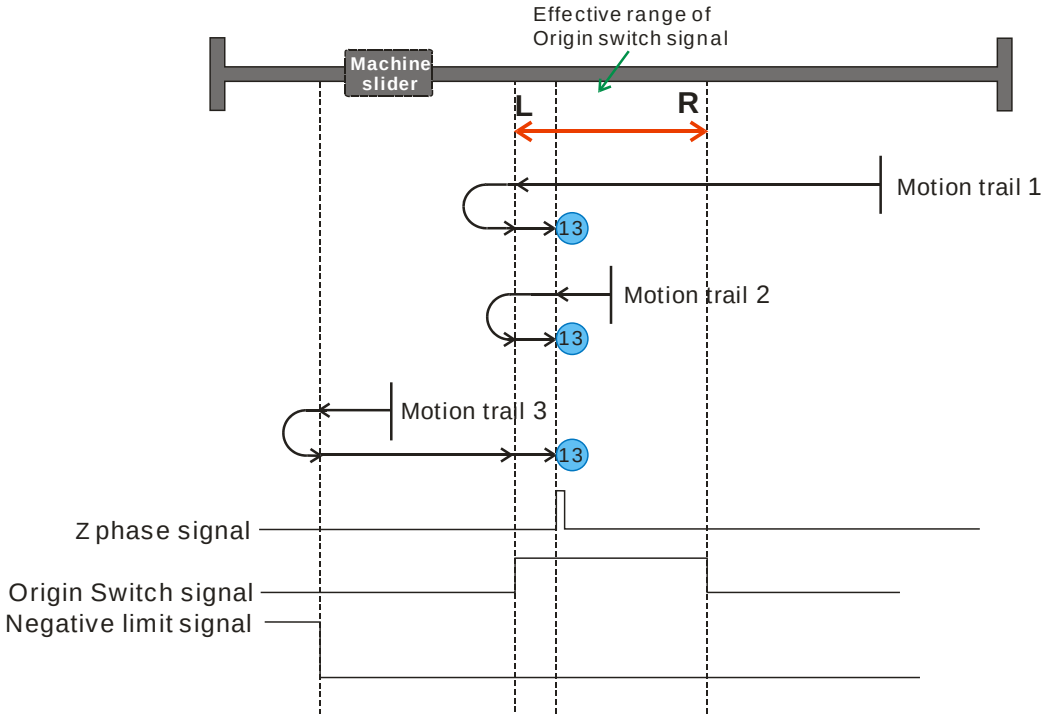


Viewing from above diagram, machine is moving to the negative limit signal, Z phase

signal is located at the left of the right edge of the positive origin switch signal, which is within the range of origin switch signal.

When machine is within the range of origin switch signal (Motion trail 2), it could search the origin point in positive direction; As the machine is out of the range (Motion trail 1 & 3), it moves towards the direction of limit signal, which can find the origin point according to the motion trail (the detected sequence of origin switch signal and negative limit signal).

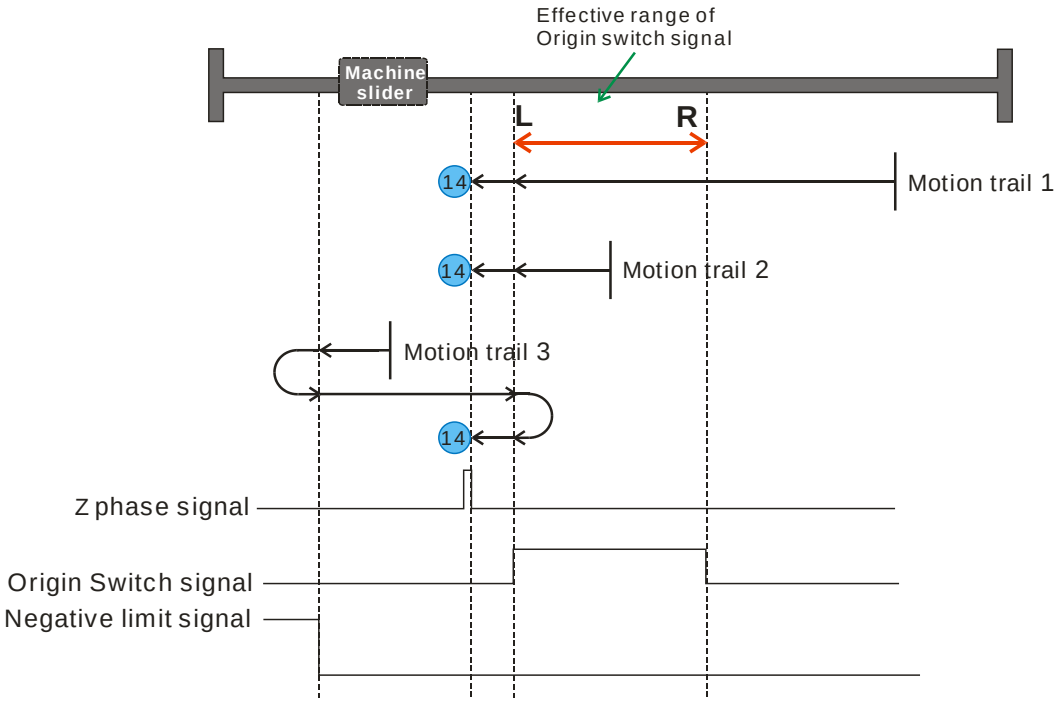
Homing Mode 13: Refer to origin switch, negative limit switch, Z phase signal (Gather Z phase signal at the right of the left edge of the positive origin switch)



Viewing from above diagram, machine is moving to the negative limit signal, Z phase signal is located at the right of the left edge of the positive origin switch signal, which is within the range of origin switch signal.

When machine is within the range of origin switch signal (Motion trail 2), it could search the origin point in negative direction; As the machine is out of the range (Motion trail 1 & 3), it moves towards the direction of limit signal, which can find the origin point according to the motion trail (the detected sequence of origin switch signal and negative limit signal).

Homing Mode 14: Refer to origin switch, negative limit switch, Z phase signal (Gather Z phase signal at the left of the left edge of the positive origin switch)



Viewing from above diagram, machine is moving to the negative limit signal, Z phase signal is located at the left of the left edge of the positive origin switch signal, which is outside the range of origin switch signal.

When machine is within the range of origin switch signal (Motion trail 2), it could search the origin point in negative direction; As the machine is out of the range (Motion trail 1 & 3), it moves towards the direction of limit signal, which can find the origin point according to the motion trail (the detected sequence of origin switch signal and negative limit signal).

4 Application Example

Example of Mix motion control instruction

Controlling stepping motor to carry out back and forth movement between point A and B, subdivision: 1000, Screw lead: 5mm, displacement L between A and B is 2000mm.

【System Description】

In this example, CPU 226H connects one EM253 module, the Axis 0 of EM253 module is used to do the point to point movement. Here we call instruction MC253_PTP_R to set control parameters.

I0.2 is the reset point of A (A is the over-travel inhibit input, set as the origin point of machine)

I1.0 is to input system emergency stop.

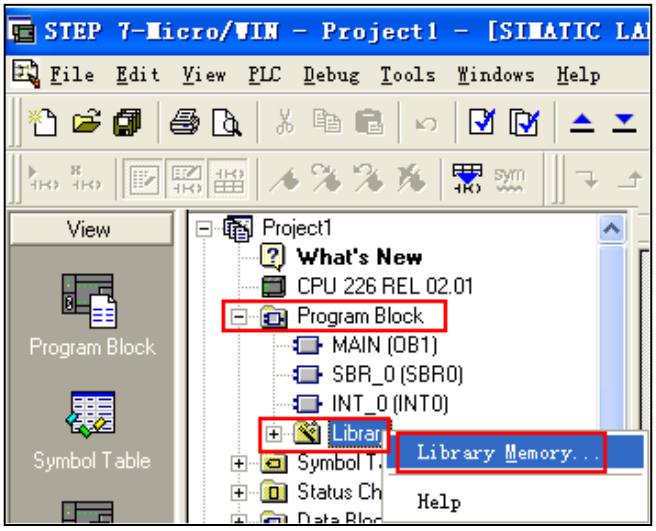
Q0.0 is pulse output, Q0.1 is direction output.

【Application Program】

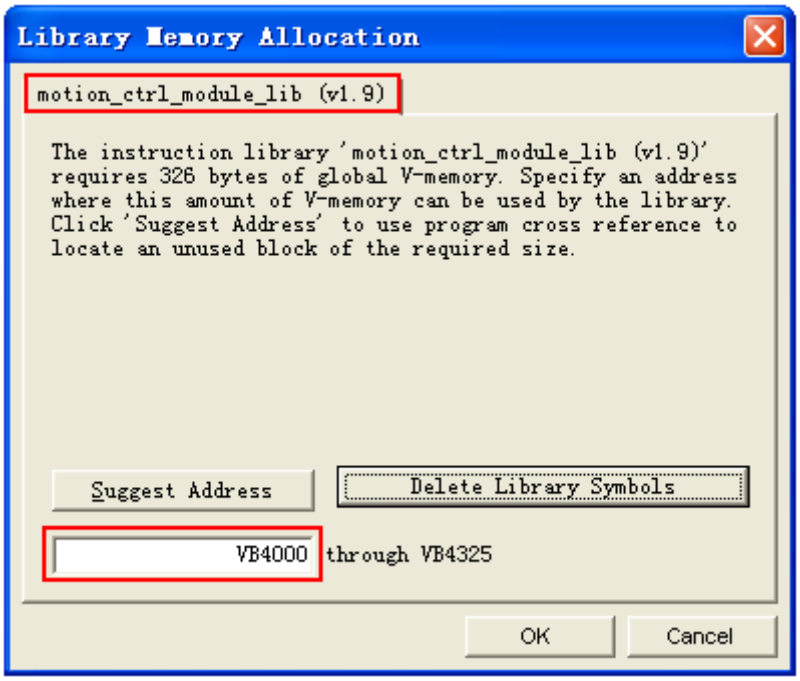
Please mind the following items while creating a new project:

1) Library memory allocation

Click Program Block from tool view, and right click the Library folder in the Instruction Tree and select the Library Memory command form the pop up menu.



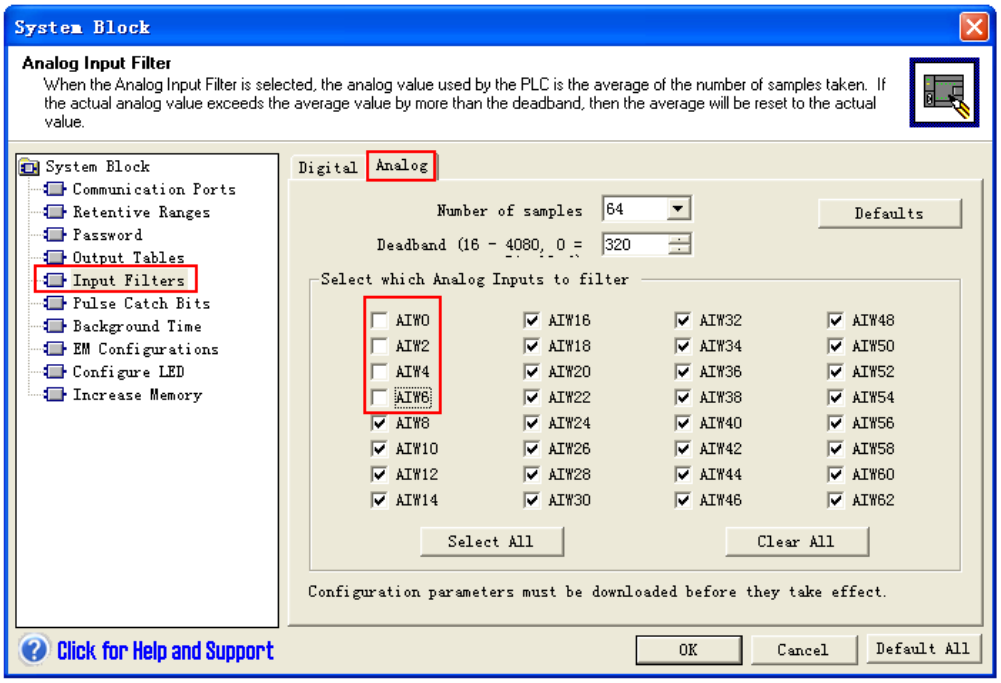
Choose the library "motion_ctrl_module_lib" from Library Memory Allocation dialog, and input the initial address in the Suggest Address:



Please refer to "Matters need to pay special attention 1" of chapter 3.3 for details.

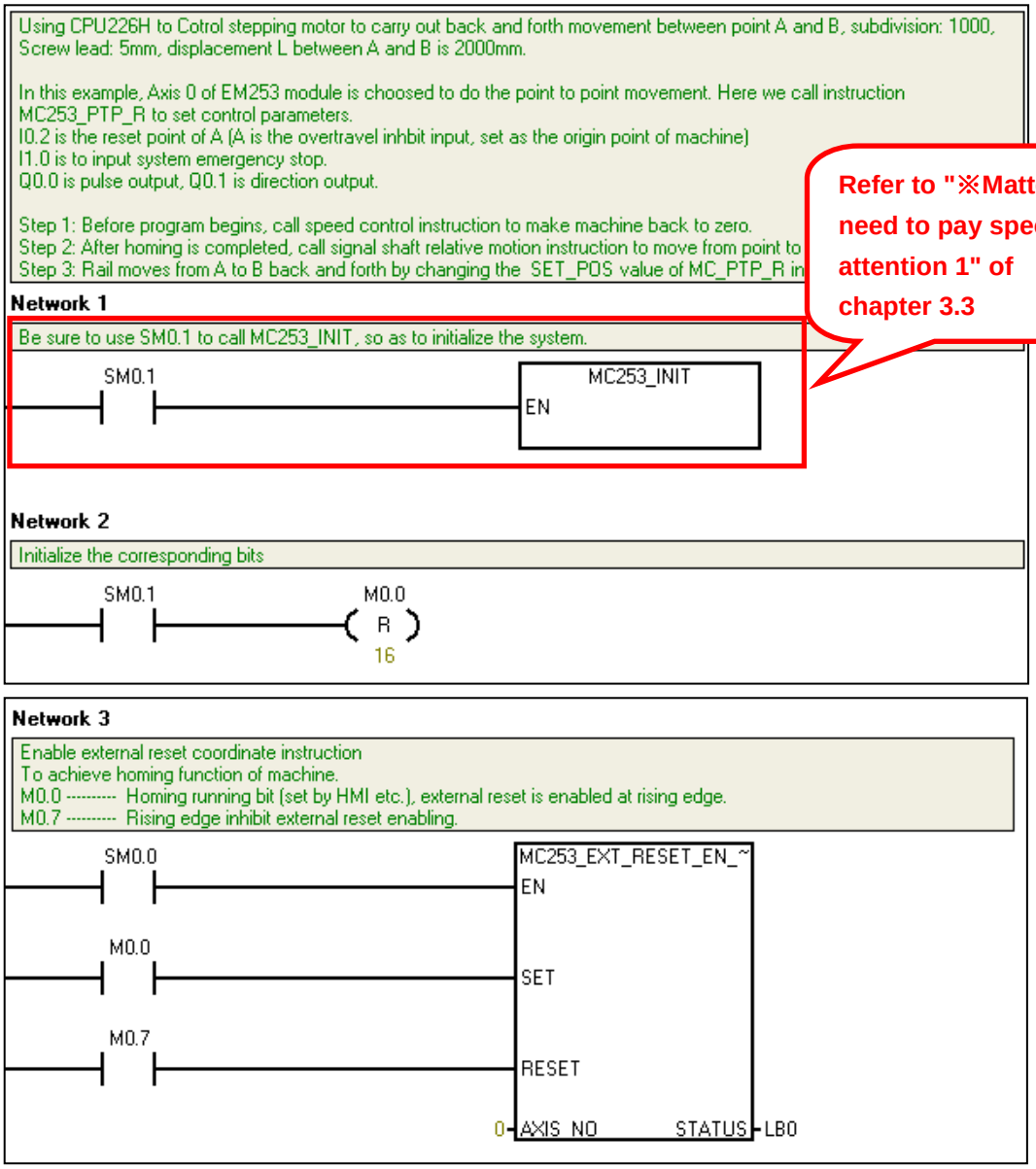
2) Clear analog input filter for the corresponding channel of EM253 module

Click System Block from tool view, pop up the following widow, and click "Input Filters" from the left view of System block to choose analog input filter, finally clear AIW0, AIW2, AIW4 and AIW6.



Please refer to "Matters need to pay special attention 2" of chapter 3.3 for details.

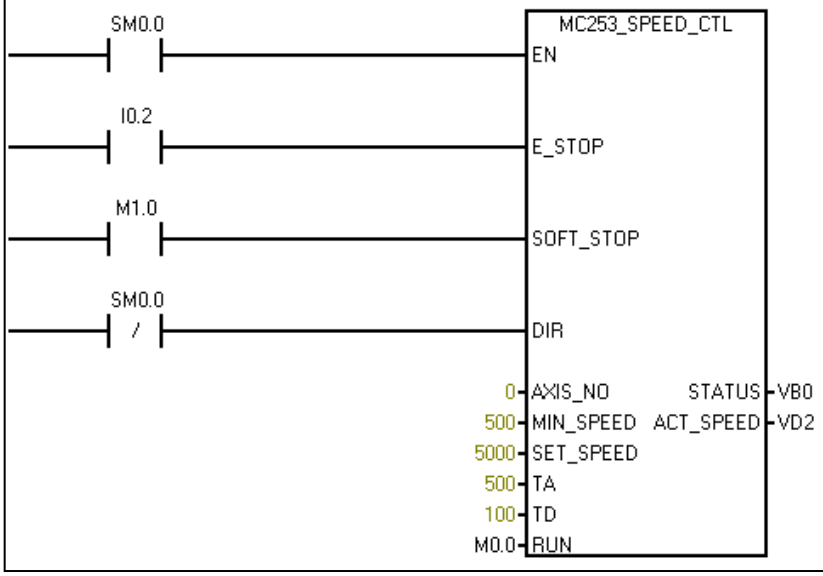
【Program Block】



Refer to "※Matters need to pay special attention 1" of chapter 3.3

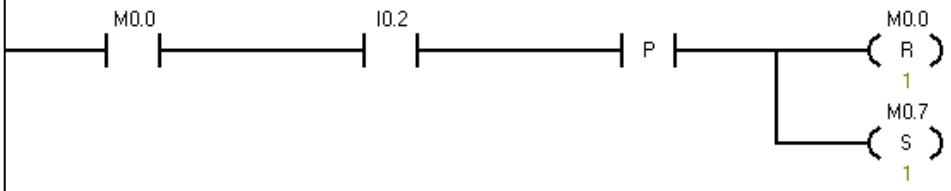
Network 4

Speed control instruction
Control axis 0 to rotate negatively, back to the origin point of machine (A is the initial point)
M0.0 ---------- Enabling bit for running;
I0.2 ---------- Overtravel inhibt switch, used as emergency stop switch for instruction, 即回到零点时关闭速度控制指令;
M1.0 ---------- Soft stop bit;
Axis No. is 0;
Start/Stop speed is 500Hz;
SET_Speed= 5000Hz(Running speed after TA is completed) ;
TA=500ms;
TD=100ms;
Output status word in VB0;
Current speed (Frequency) in VD2;



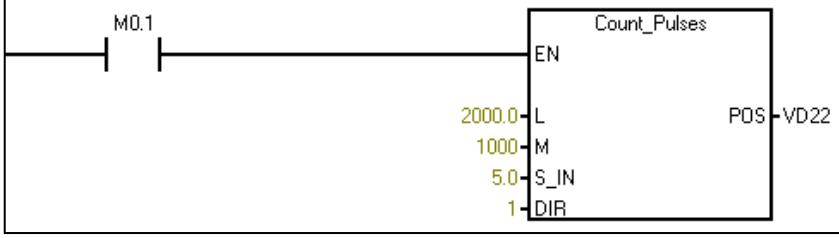
Network 5

Reset the corresponding bits after homing is completed.



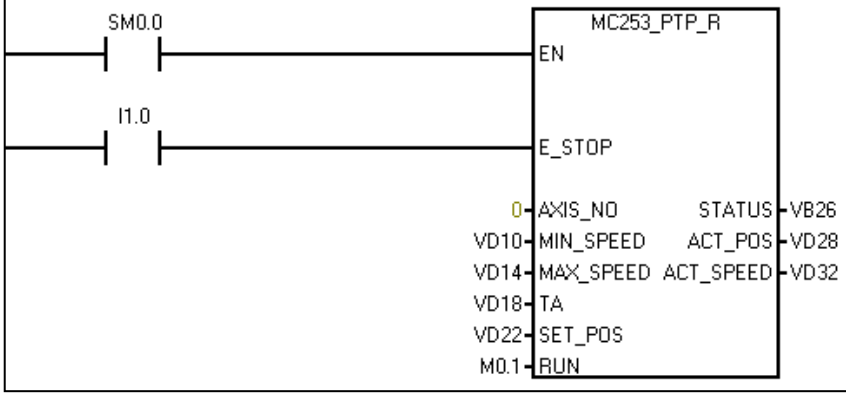
Network 6

M0.1---------- Enabling bit of runing from A to B
Displacement from A to B: 2000mm.
Subdivision of stepping motor: 1000
Screw lead: 5mm;
DIR =1 (1---rotaing positively、 0---rotating negatively)
Pulses in VD22



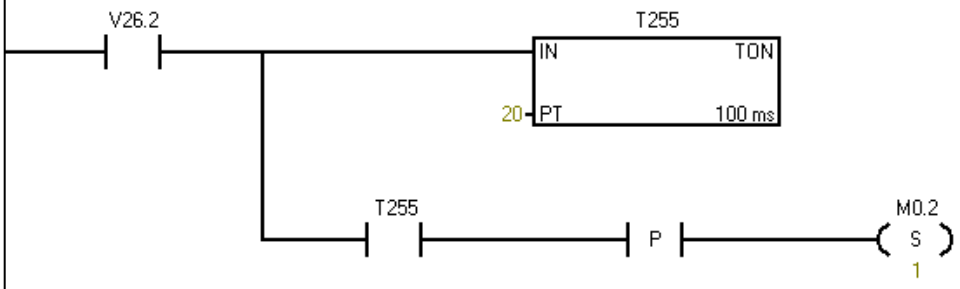
Network 7

Single shaft relative motion control instruction
To achieve the movement from A to B.
I1.0---------- Emergency stop bit;
Axis No. is 0
VD10 ----------Start/Stop speed;
VD14 ---------- Running speed after acceleration is completed;
VD18 ---------- Acceleration/Deceleration time;
VD22 ----------Output pulses;
M0.1 ---------- Enabling bit for running;
Output status byte-------- VB26;
Pulses has been outputted---- VD28;
Current actual running speed---- VD32;



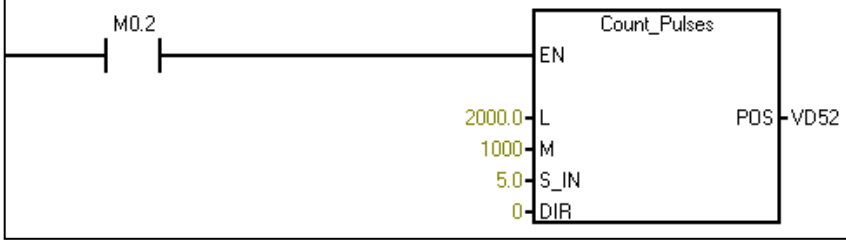
Network 8

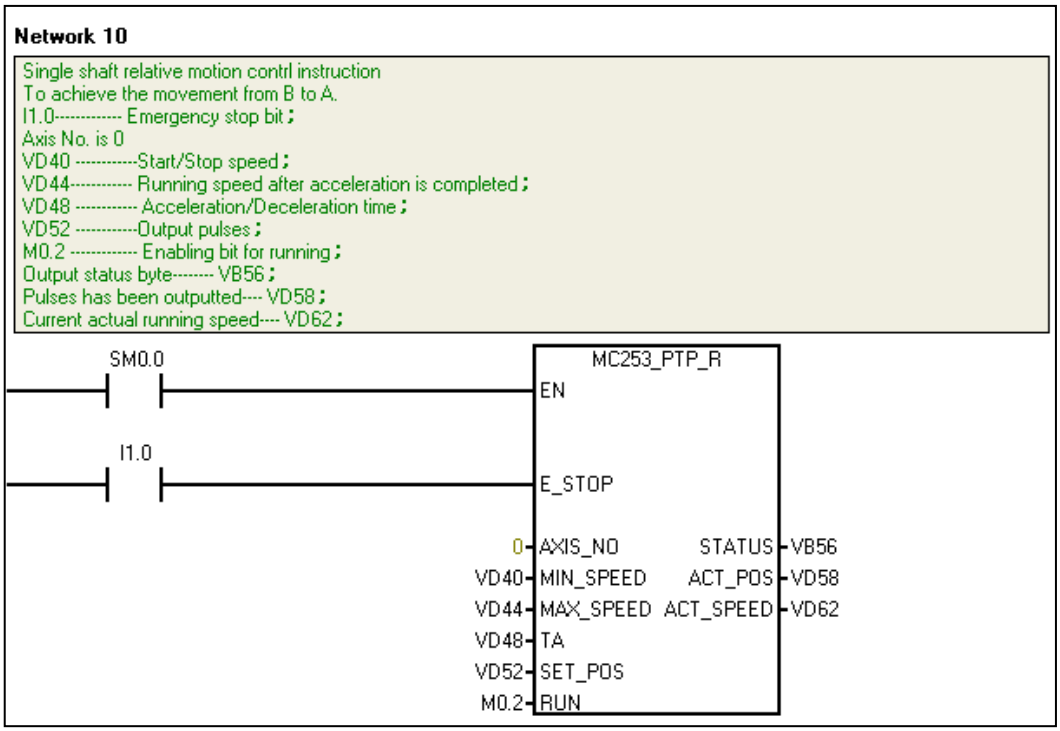
After output is completed, timing as 2 seconds to make M0.2 rotating negatively

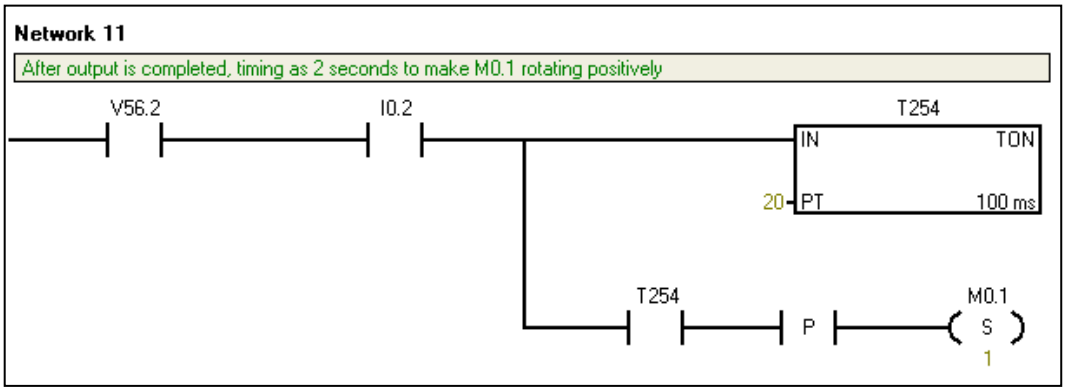


Network 9

Displacement from B to A: 2000mm.
Subdivision of stepping motor: 1000
Screw lead: 5mm;
DIR =0（1---rotaing positively、0---rotating negatively）
Pulses in VD52







Remark: Specific demo please refer to attachment "MC253-DEMO.mwp".

Appendix

A Ordering info

Table A-1 Order info

Specification	Order No.
EM253 motion control module 8 inputs /8 outputs, 24VDC (2 HSC, Single/Double phase 200KHz, 2*200KHz high-speed pulse output)	CTS7 253-1BH32

SHENZHEN CO-TRUST TECHNOLOGY CO.,LTD.

Address: Room 209 & 210, 2/F, ShenZhen IC design & application industrial park, South of Chaguang Road, Xili Town, Nanshan District, Shenzhen, CHINA.

Hotline: 400-700-4858

E-mail: sales@co-trust.com

http: //www.co-trust.com

It is to be noted that all these parameters are subjected to change without notice.