10 SEQUENCE BLOCK

This chapter will introduce the sequence block instruction and the application.

10-1. Concept of the BLOCK

10-2. Call the BLOCK

10-3. Edit the instruction inside the BLOCK

10-4. Running form of the BLOCK

10-5. BLOCK instruction editing rules

10-6. BLOCK related instructions

10-7. BLOCK flag bit and register

10-8. Program example

Block instruction:

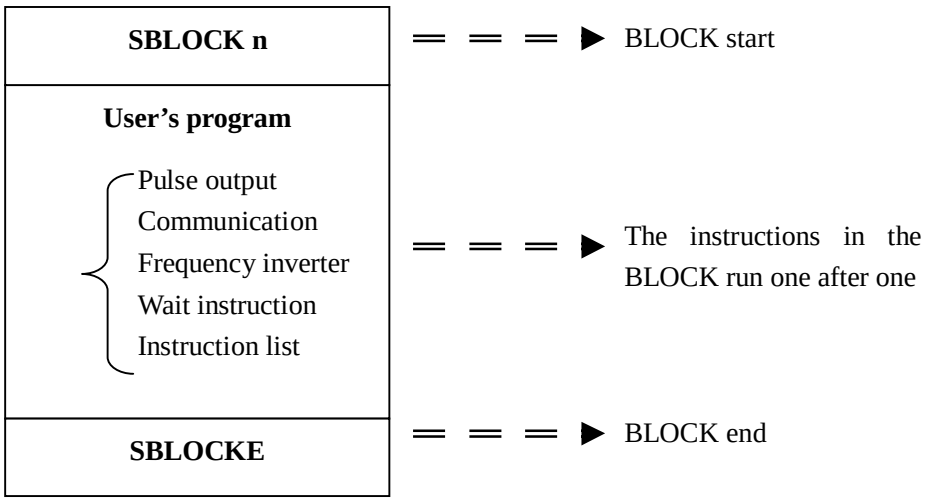
Instruction	Function	Ladder chart	Chapter
Block			
BSTOP	Stop the BLOCK		10-6-1
BGOON	Continue running the BLOCK		10-6-1

10-1．Concept of the BLOCK

10-1-1．BLOCK summarization

Sequence block, which is also called block, is a program block can realize certain function. Block is a special flow, all the instructions run in order; this is the difference from other flows. BLOCK starts from SBLOCK and ends by SBLOCKE, you can write program between them. If there are many pulse output instructions (or other instructions), they will run one after one according to the condition. After one pulse outputting over then the next pulse will output.

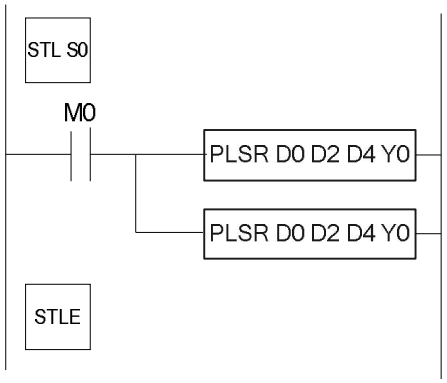
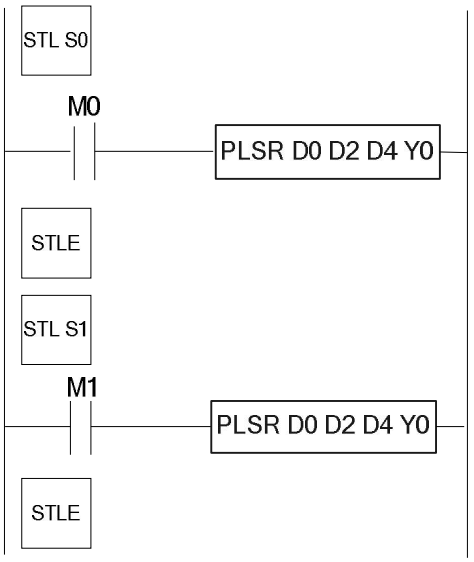
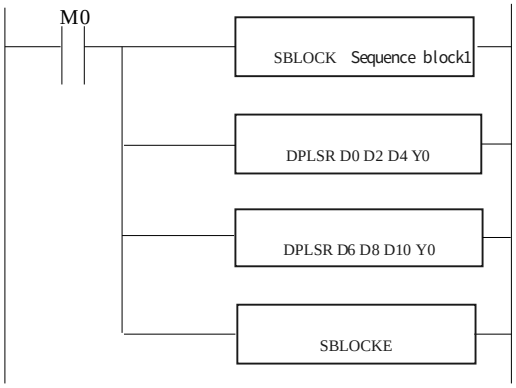
The construction of the block is as the following:



10-1-2. The reason to use BLOCK

To optimize the editing method of pulse and communication instruction in the process

In former program, XC series PLC can not support many pulse or communication instructions in one process, but BLOCK can support this and the instructions will run in sequence.

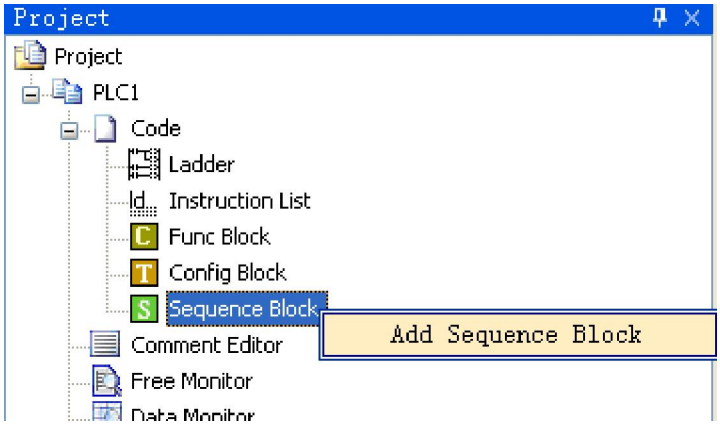
	Unavailable （×）	**Available （√）**
Former		
After using block		

10-2. Call the BLOCK

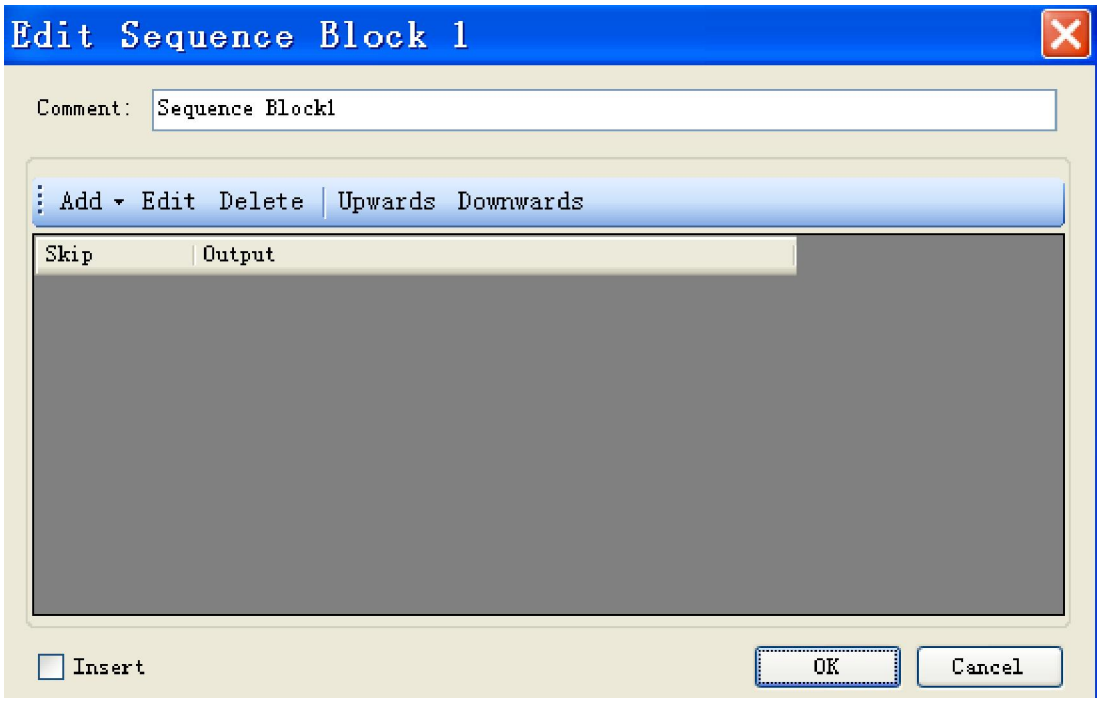
In one program file, it can call many BLOCK; the following is the method to add BLOCK in the program.

10-2-1. Add the BLOCK

Open XCPpro software, right click the sequence block in the project bar:

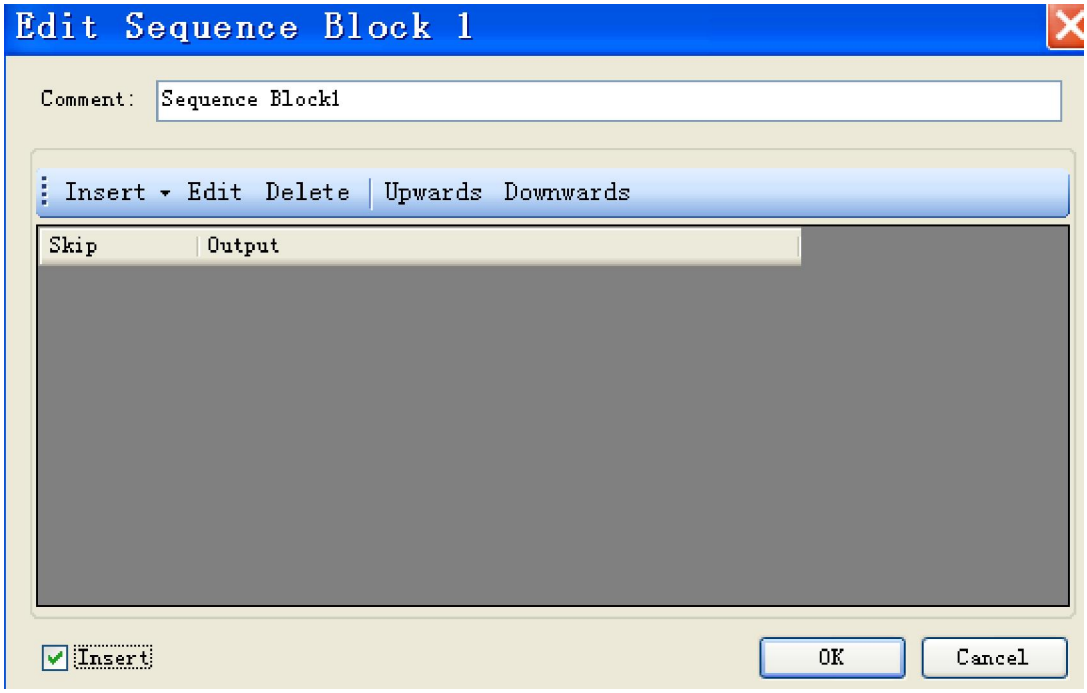


Click “add sequence block” will show below window:



You can edit the program in this window. Upwards and downwards are used to change the position of the instruction in the block.

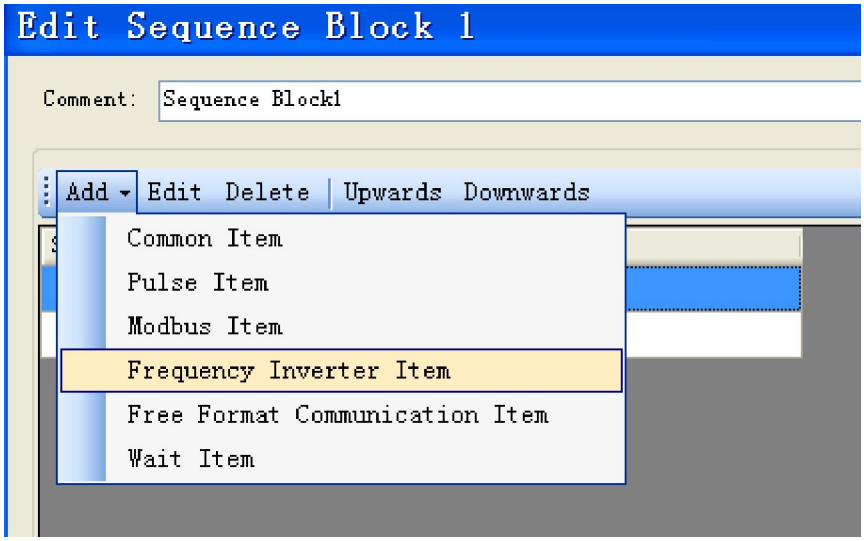
There is a “Insert” choice on the bottom left of the window, when selecting it, the add button will become insert:



The difference between insert and add:

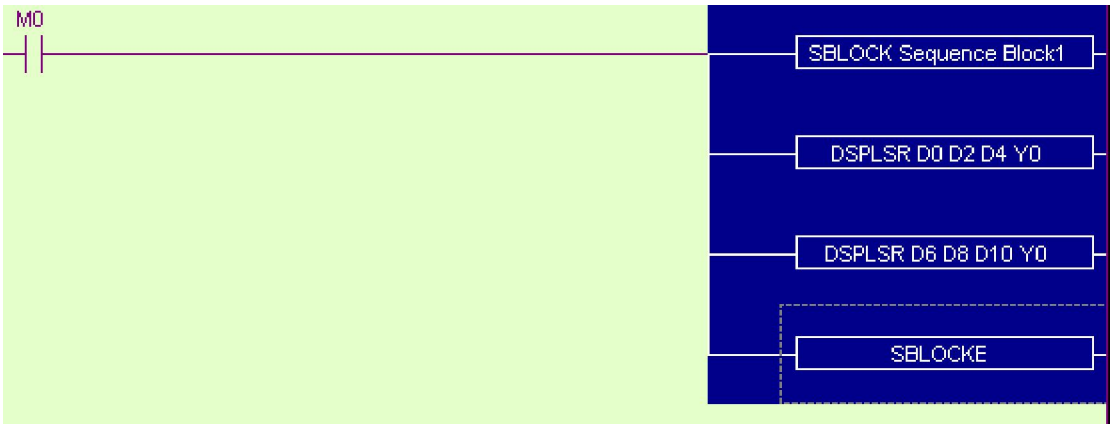
Add is to add instructions in the end of the block; insert can add instruction in any place in the block.

Click add button, you will see the instructions can be added in the block.

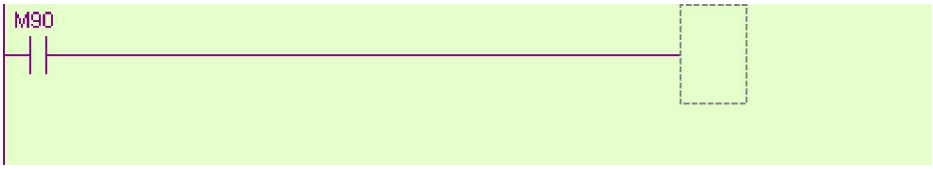


10-2-2. Move the BLOCK

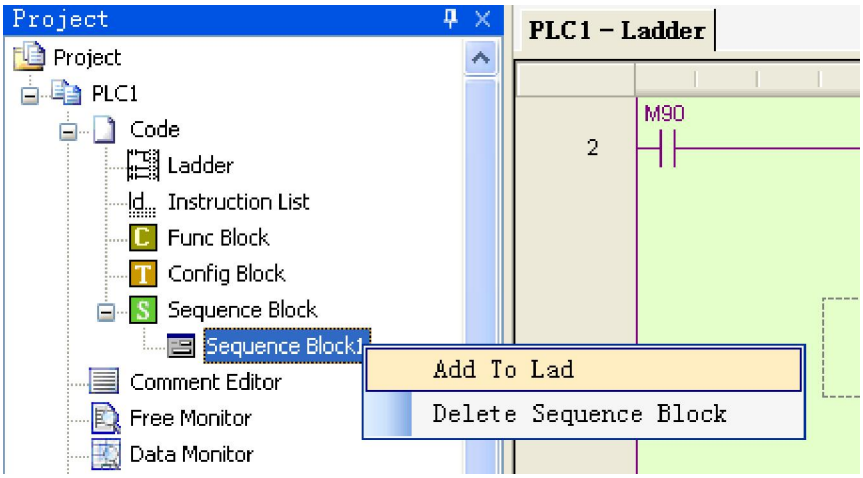
If you want to move the block to other position, you have to select the former block and delete it.



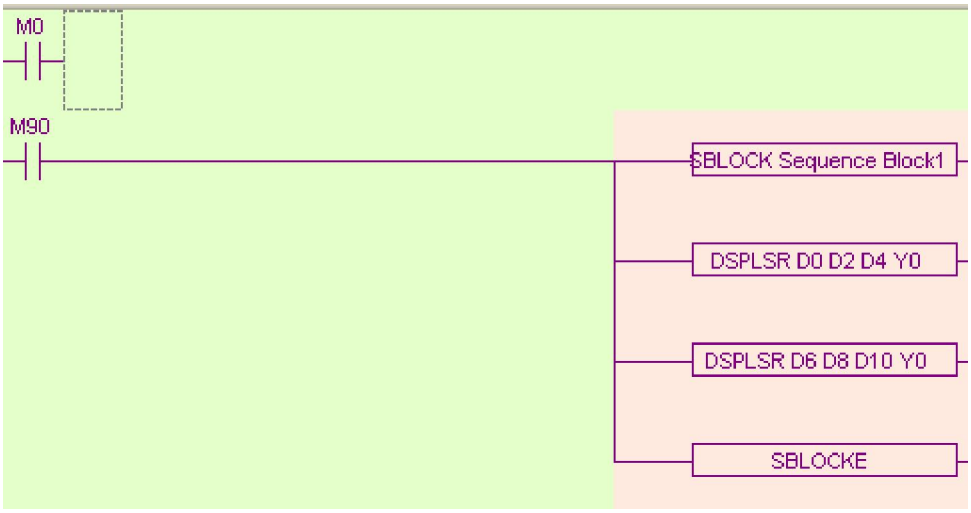
Then put the cursor in the place you want to move:



Right click the “add to lad” in the project bar:

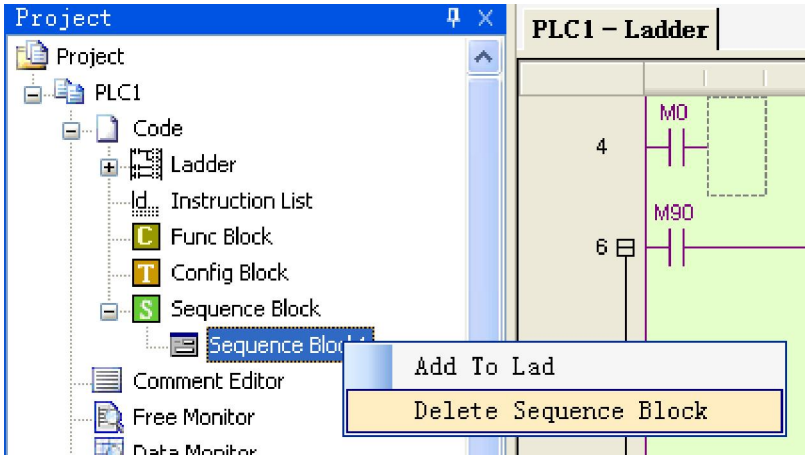


Now the block is moved to the new place:



10-2-3. Delete the BLOCK

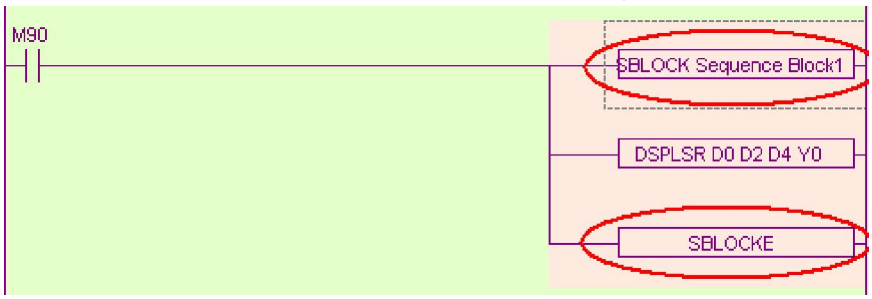
You can select the whole block and delete it. If you want to delete the block forever, please right click the block you want to delete in the project bar and select “delete sequence block”. After this operation, you can not call this block anymore.

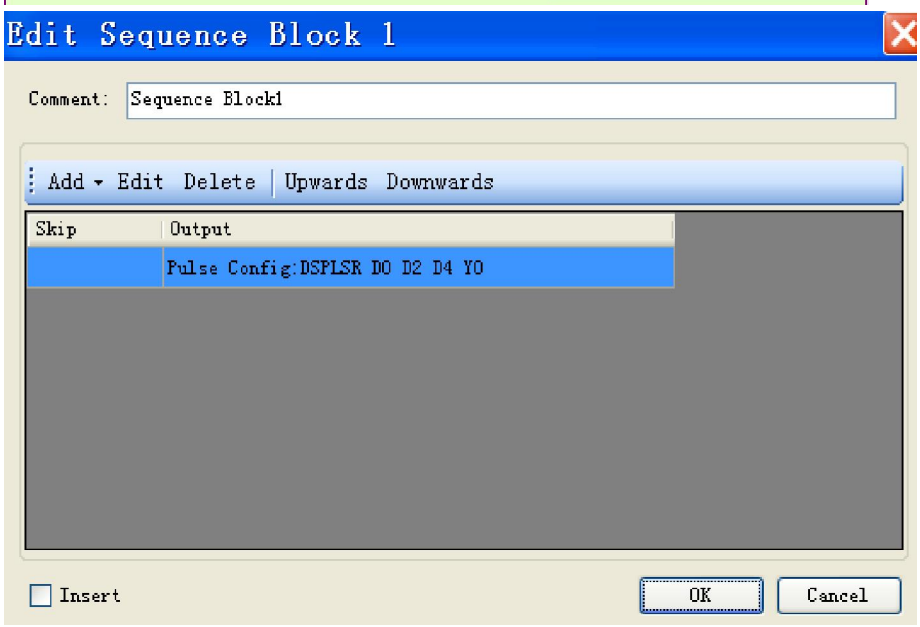


10-2-4. Modify the BLOCK

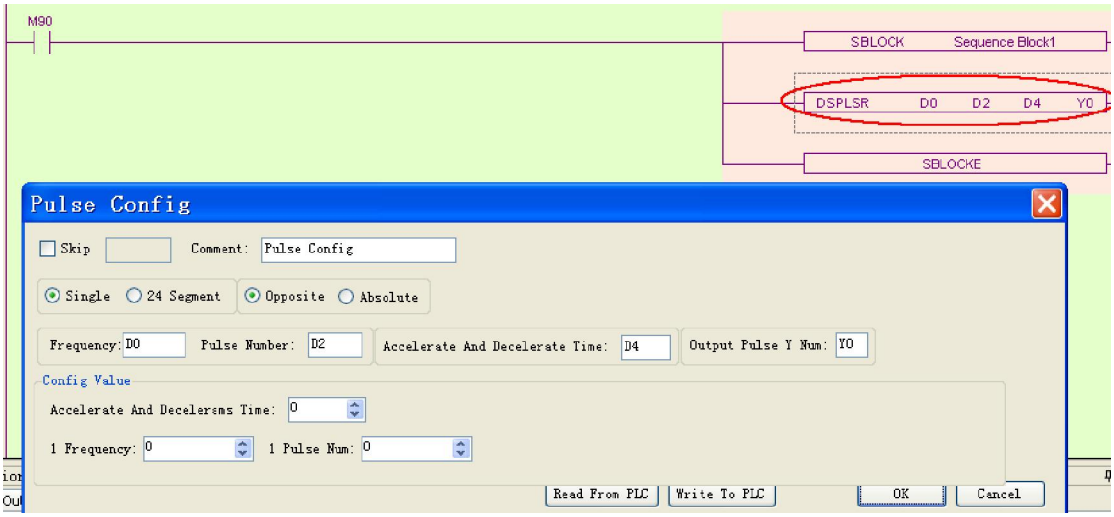
There are two methods to modify the block.

（A）double click the start or end instruction to modify all the instructions in the block.





（B）double click one instruction in the block to modify it:

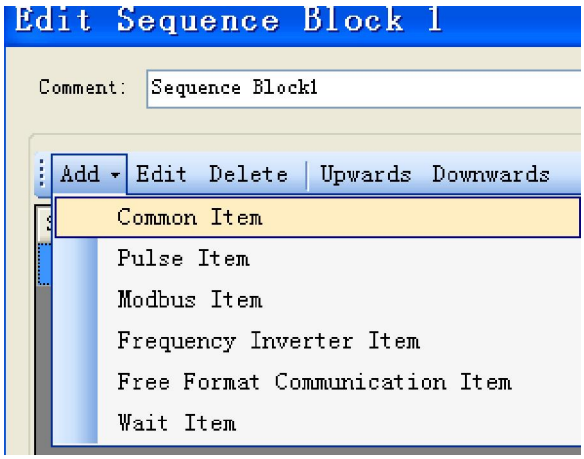


10-3．Edit the instruction inside the BLOCK

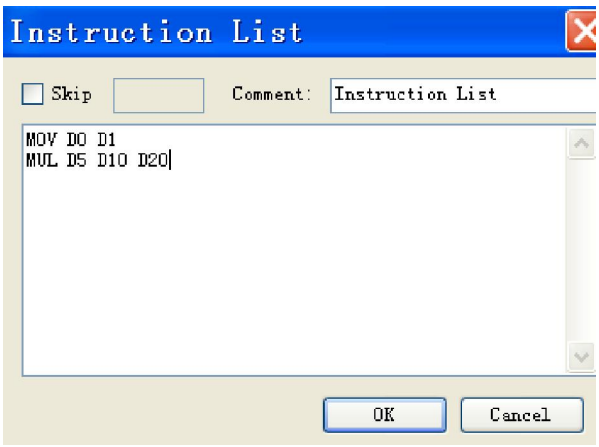
10-3-1．Common item

Use command to edit the program.

Open the block editing window, click add/common item:



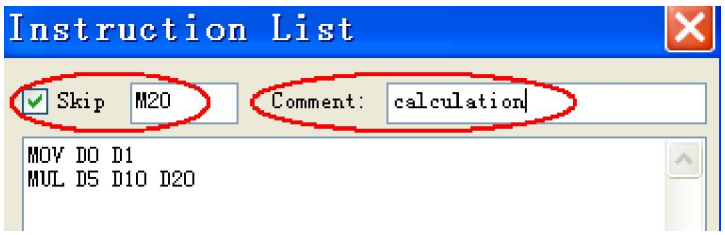
It will show the editing window:



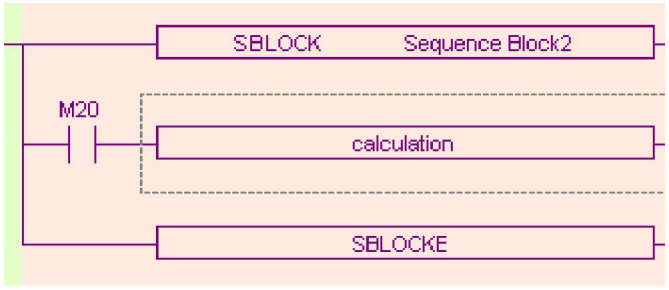
User can add instructions in this window.

SKIP condition: can control the stop and running of the instructions. When select skip and enter coil in it, if the coil is ON, the instructions will stop.

Comment: can modify the note for this instruction.

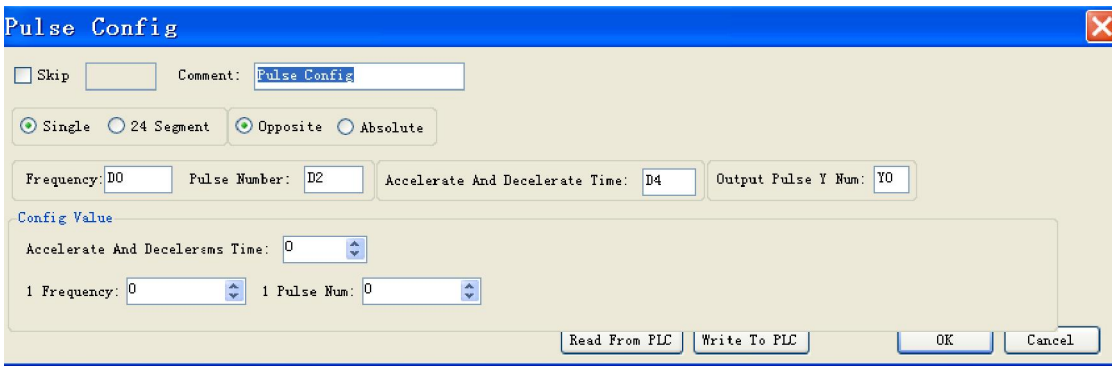


After setting, the block will be changed as the following:

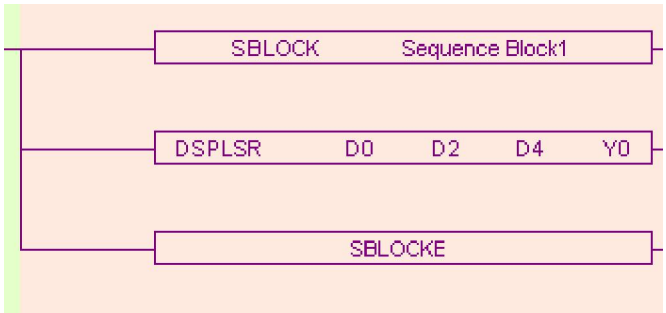


10-3-2. Pulse item

Open the pulse item window:



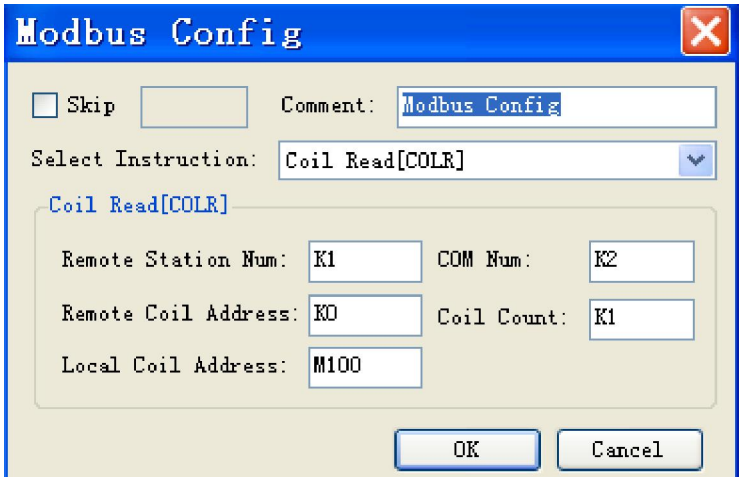
Set the pulse output frequency, numbers, output terminals, accelerate/decelerate time and so on. Then add the pulse instruction in the block:



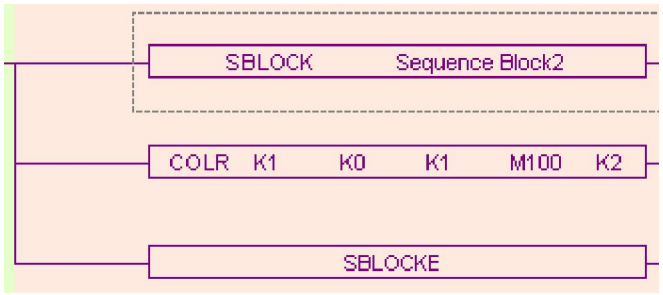
※1：the pulse output instructions are all 32bits.

10-3-3. Modbus item

Open the modbus item window:



Select the modbus instructions, set the address and com port, then software will build an instruction.



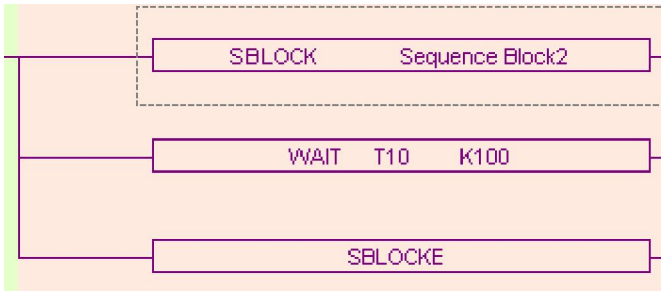
10-3-4. Wait item

There are two modes to wait.

（A）flag bit

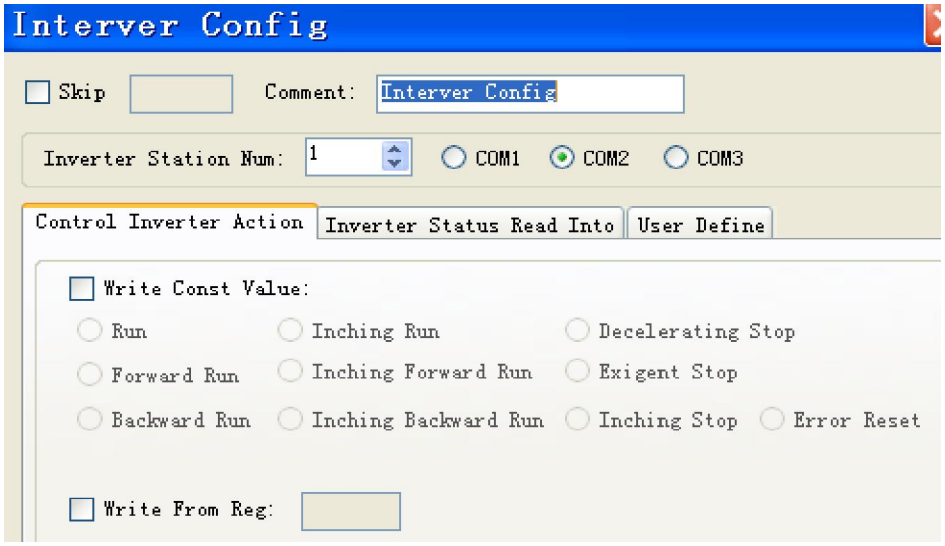
（B）timer wait

The ladder chart is as the following:



10-3-5. Frequency inverter item

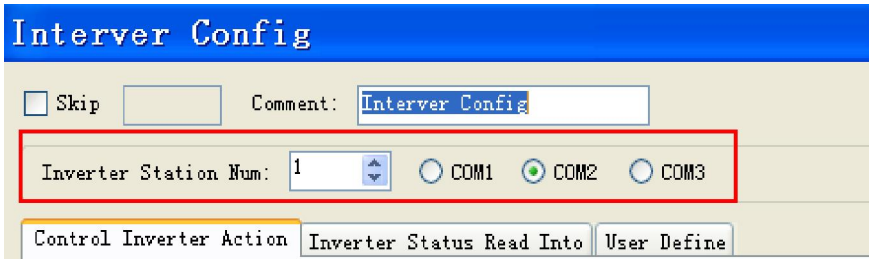
Users only have to set the parameters in below window, the PLC will communicate with the frequency inverter.



There are four areas in the window, the following will introduce one by one:

（A）Inverter station number and serial number

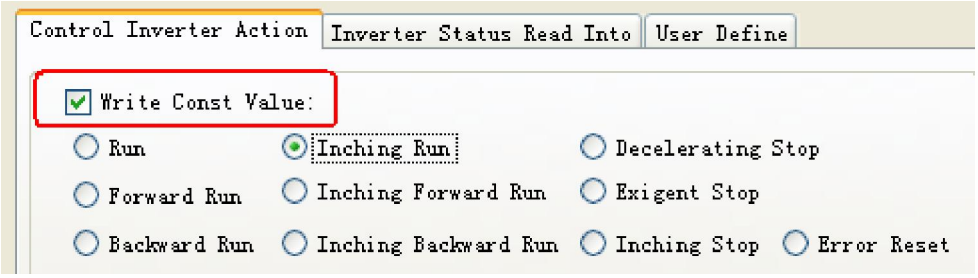
Set the station number of the frequency inverter and the PLC serial port:



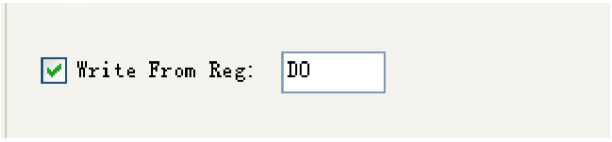
（B） Control inverter action

There are two modes to set parameters.

First one is write constant value:

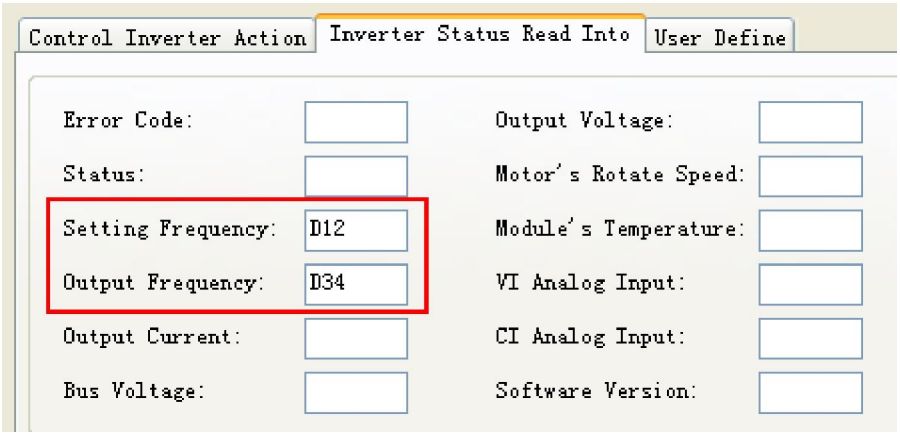


Second one is to set the parameters in register:



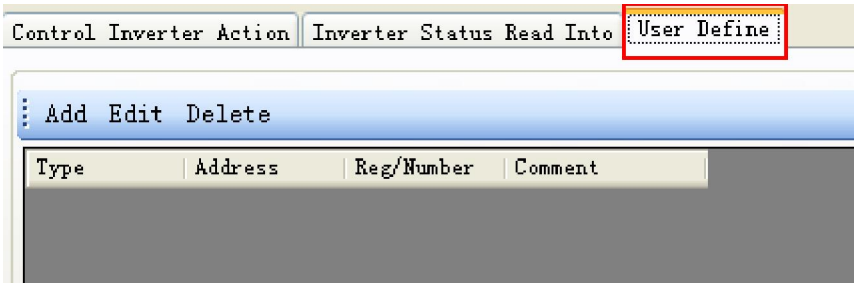
（C） Inverter status read into

To read the status from the frequency inverter to the PLC register.

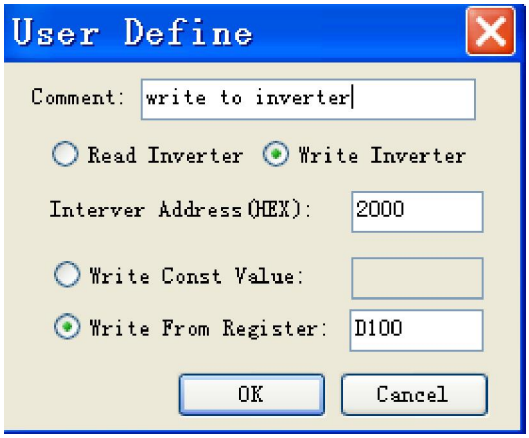


（D） User define

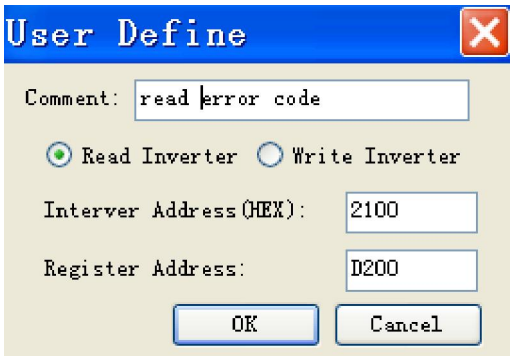
To write or read the frequency inverter address flexible.



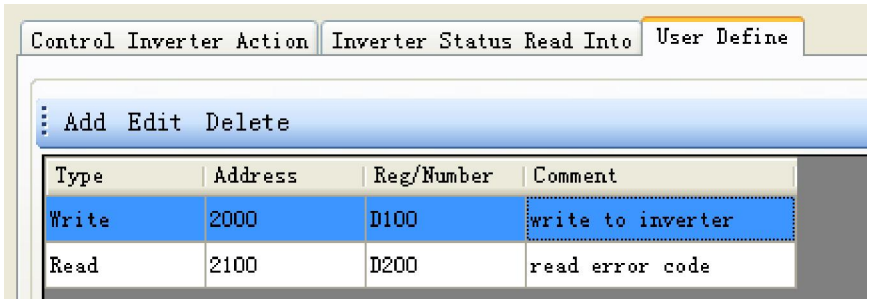
For example, add a writing inverter instruction:



Add a reading inverter instruction:



The result after adding:



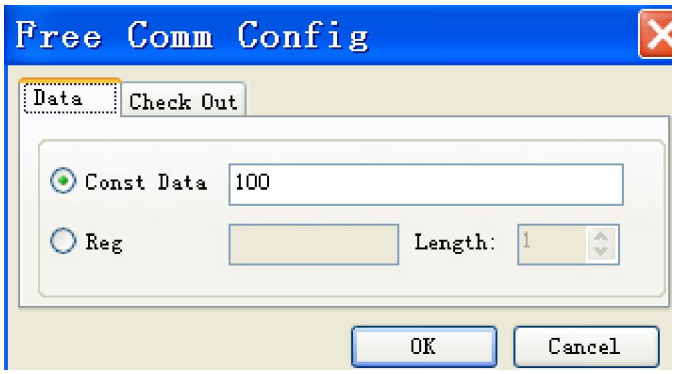
※1：Frequency inverter instructions will not expand in the block.

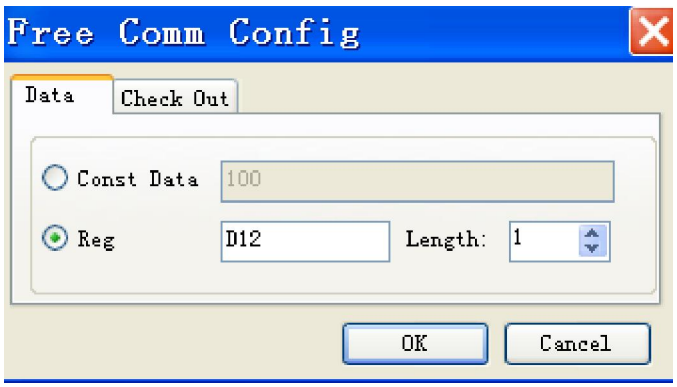
10-3-6. Free format communication item

Add free format communication instructions in the block.

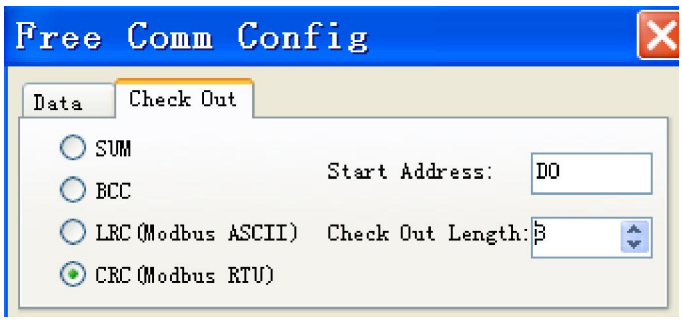
For example, select “send” instruction, first address set to D0, serial port is 2, 16 bits.

There are two methods to set the data. Const data is to set the value directly. Reg is to set the value via register.

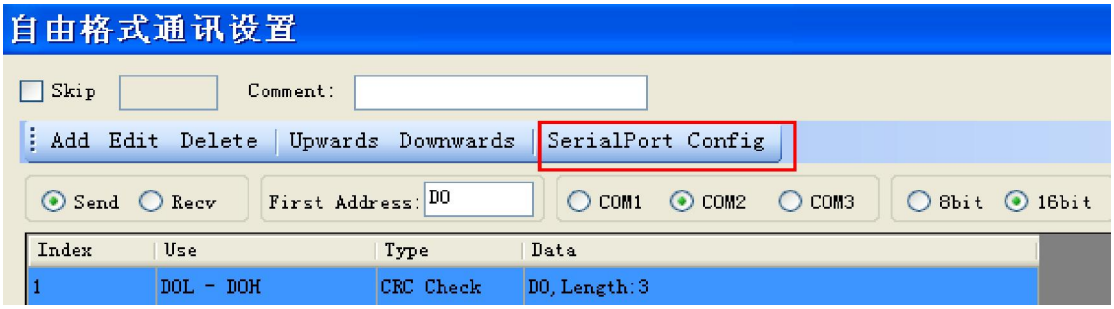




Change to check out tab, select the checking mode.



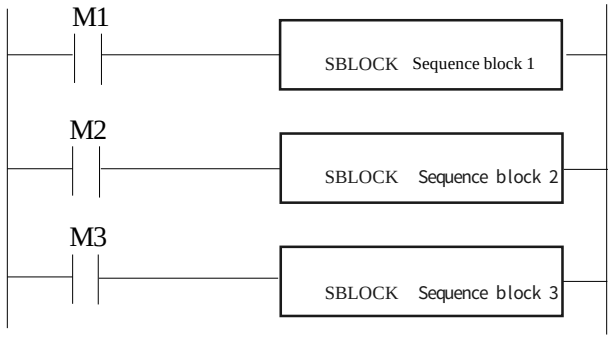
Besides, it needs to set the communication parameters. Click “serial port config”:

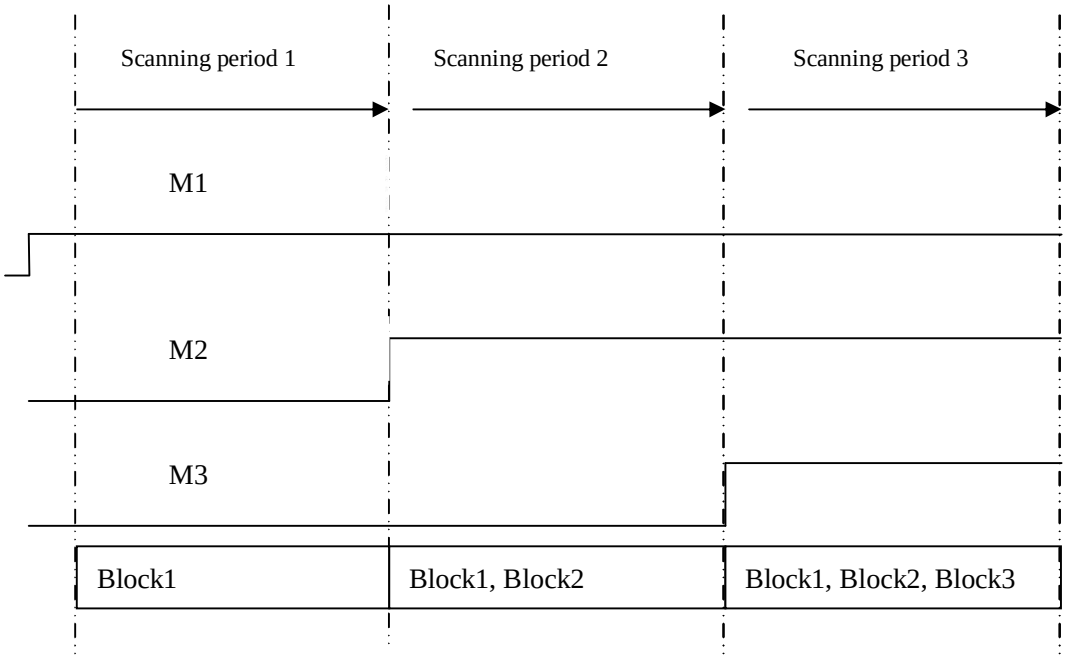


10-4. Running form of the BLOCK

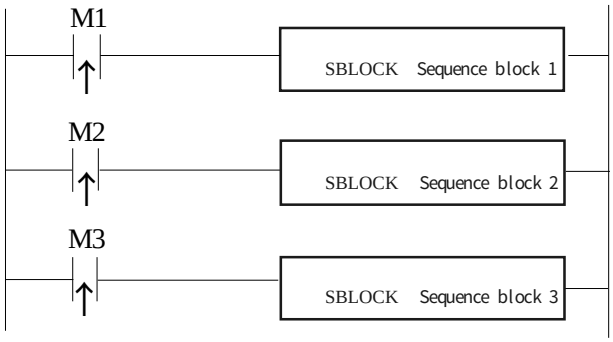
1、If there are many blocks, they run as the normal program. The block is running when the condition is ON.

（A）the condition is normal ON, normal OFF coil





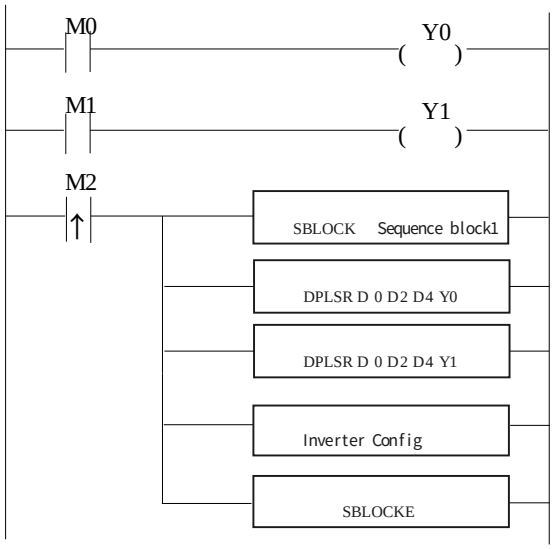
（B） the condition is rising or falling edge of pulse



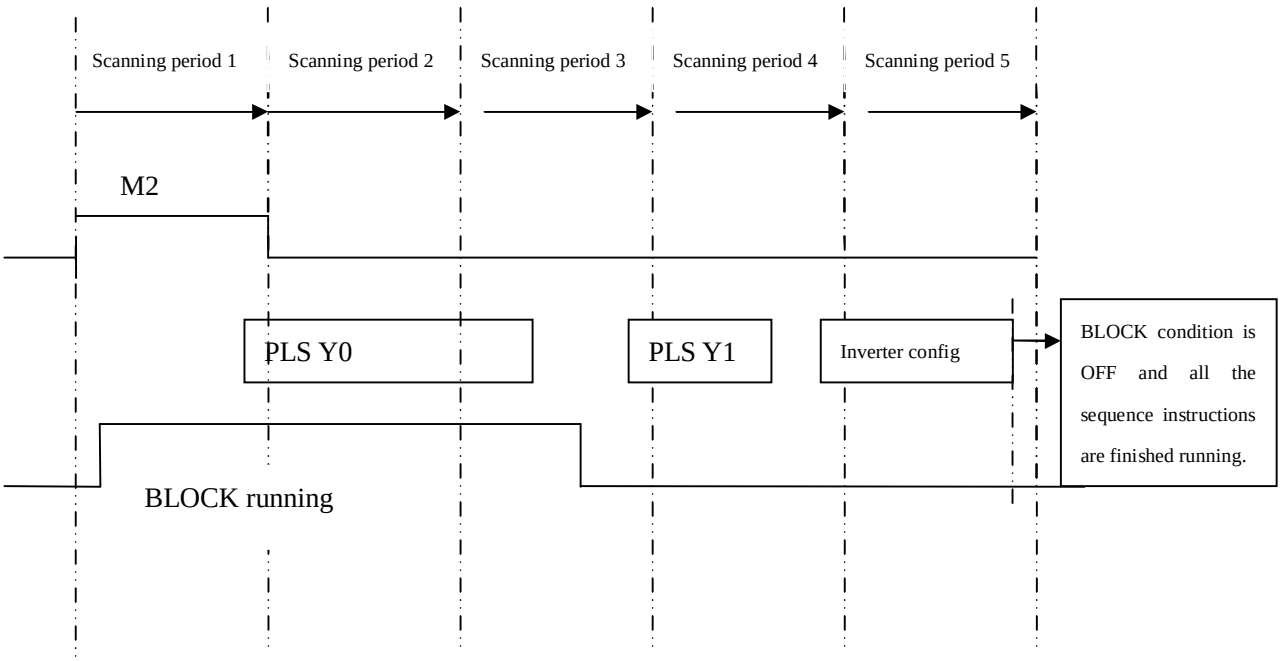
When M1, M2, M3 is from OFF to ON, all these blocks will run once.

2、The instructions in the block run in sequence according to the scanning time. They run one after another when the condition is ON.

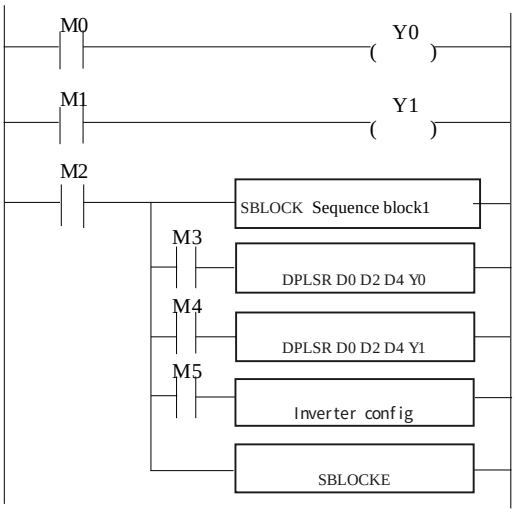
（A） Without SKIP condition



The instructions running sequence in block 1 is shown as below:



（B）With SKIP condition



Explanation:

A) When M2 is ON, block 1 is running.

B) All the instructions run in sequence in the block.

C) M3, M4, M5 are the sign of SKIP, when they are ON, this instruction will not run.

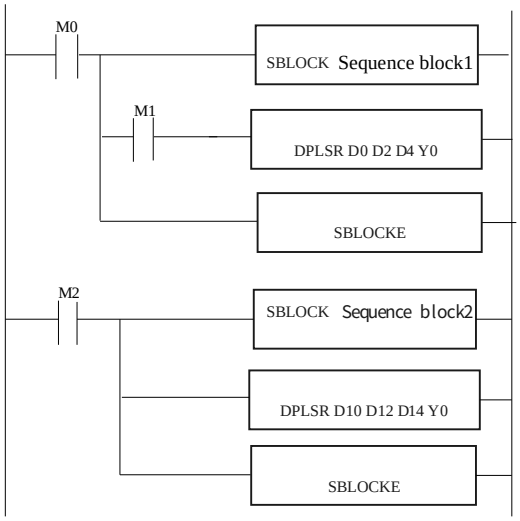
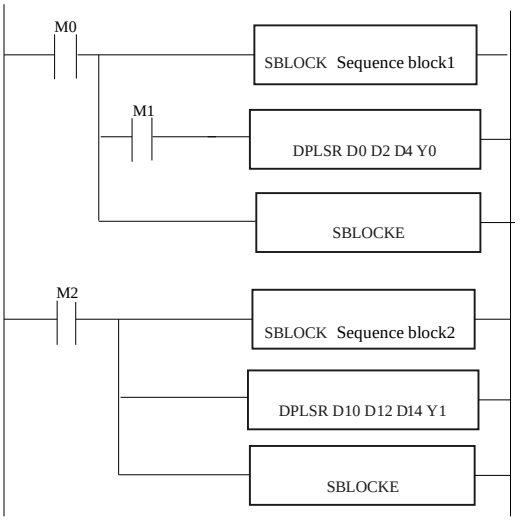
D) When M3 is OFF, if no other instructions use this Y0 pulse , DPLSR D0 D2 D4 Y0 will run; if not, the DPLSR D0 D2 D4 Y0 will run after it is released by other instructions.

E) After “DPLSR D0 D2 D4 Y0” is over, check M4. If M4 is OFF, check “DPLSR D0 D2 D4 Y1”, if M4 is ON, check M5. If M5 is OFF, “inverter config” will run.

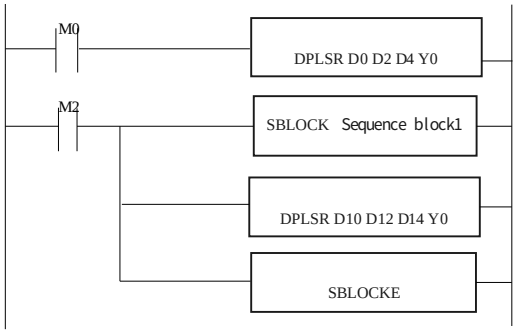
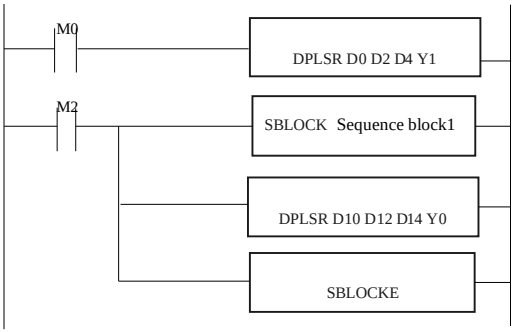
10-5．BLOCK instruction editing rules

In the BLOCK, the instruction editing should accord with some standards.

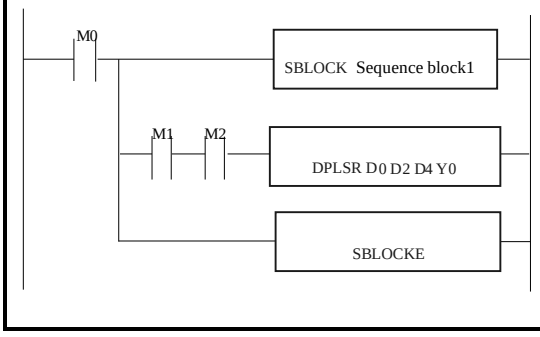
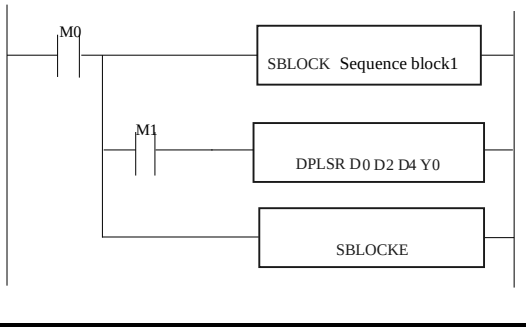
1、Do not use the same pulse output terminal in different BLOCK.

NO（×）	YES（√）
	

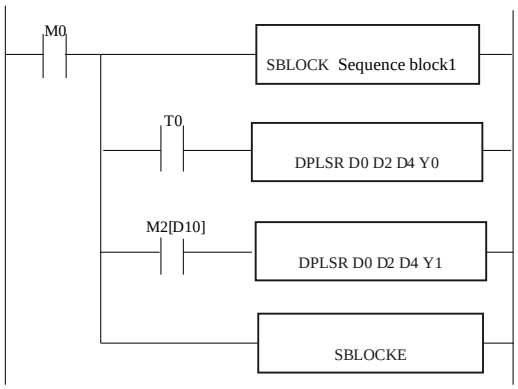
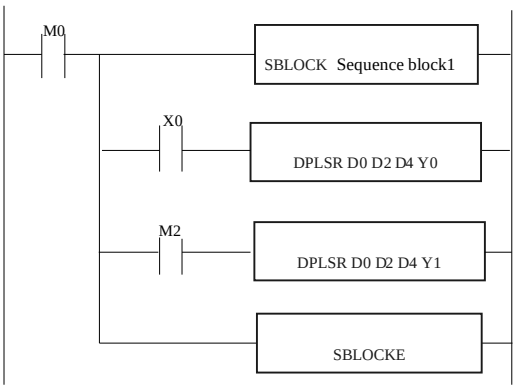
2、Do not use the same pulse output terminal in BLOCK and main program.

NO（×）	YES（√）
	

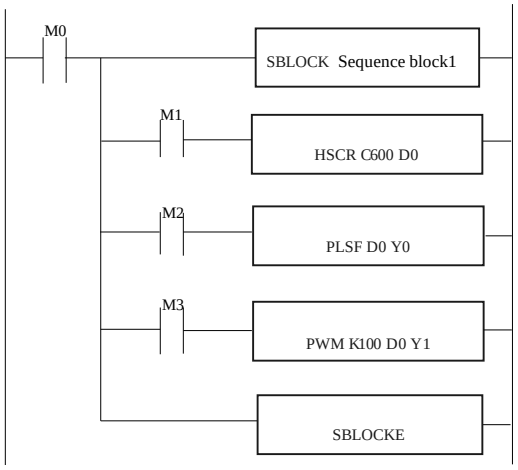
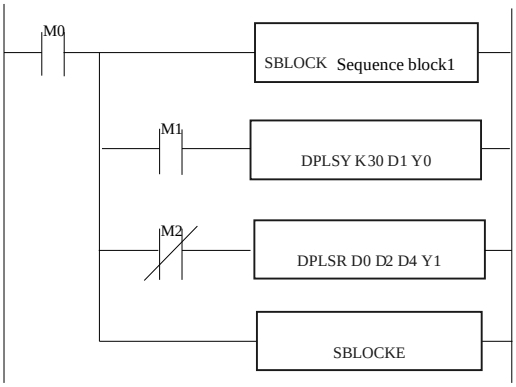
3、There only can be one SKIP condition for one BLOCK instruction.

NO（×）	YES（√）
	

4、The SKIP condition only can use M, X, can not use other coil or register.

NO（×）	YES（√）
	

5、The output instructions can not be HSC, PLSF, PWM, FRQM.

NO（×）	YES（√）
	

6、LabelKind type can not be used in the block

Sign P, I can not be used in block. Even they can be added in block, but they do not work in fact.

10-6．BLOCK related instructions

10-6-1．Instruction explanation

Ø stop running the BLOCK [BSTOP]

1、Summarization

Stop the instructions running in the block

[BSTOP]			
16 bits	BSTOP	32 bits	-
Condition	NO,NC coil and pulse edge	Suitable types	XC1、XC2、XC3、XC5、XCM
Hardware	V3.1i and above	Software	V3.1h and above

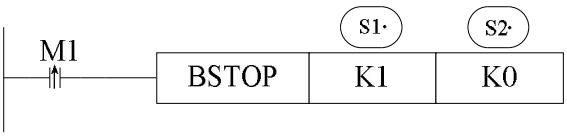
2、Operand

Operand	Function	Type
S1	The number of the BLOCK	16 bits, BIN
S2	The mode to stop the BLOCK	16 bits, BIN

3、Suitable component

Word component	Operand	Register									Constant	Module	
		D	FD	ED	TD	CD	DX	DY	DM	DS	K/H	ID	QD
	S1	●									●		
	S2										K		

Function



l S2 is the mode to stop BLOCK, operand K1, K2

K0: stop the BLOCK slowly, if the pulse is outputting, the BLOCK will stop after the pulse outputting is finished.

K1: stop the BLOCK immediately; stop all the instructions running in the BLOCK.

Ø Continue running the BLOCK[BGOON]

1、Summarization

This instruction is opposite to BSTOP. To continue running the BLOCK.

[BGOON]			
16 bits	BGOON	32 bits	-
Condition	Pulse edge	Suitable types	XC1、XC2、XC3、XC5、XCM
Hardware	V3.1i and above	Software	V3.1h and above

2、Operand

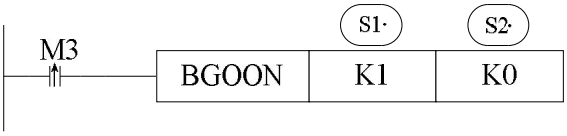
Operand	Function	Type
S1	The number of the BLOCK	16 bits, BIN
S2	The mode to continue running the BLOCK	16 bits, BIN

3、Suitable component

Word Component

Operand	Register									Constant	Module	
	D	FD	ED	TD	CD	DX	DY	DM	DS	K/H	ID	QD
S1	●									●		
S2										K		

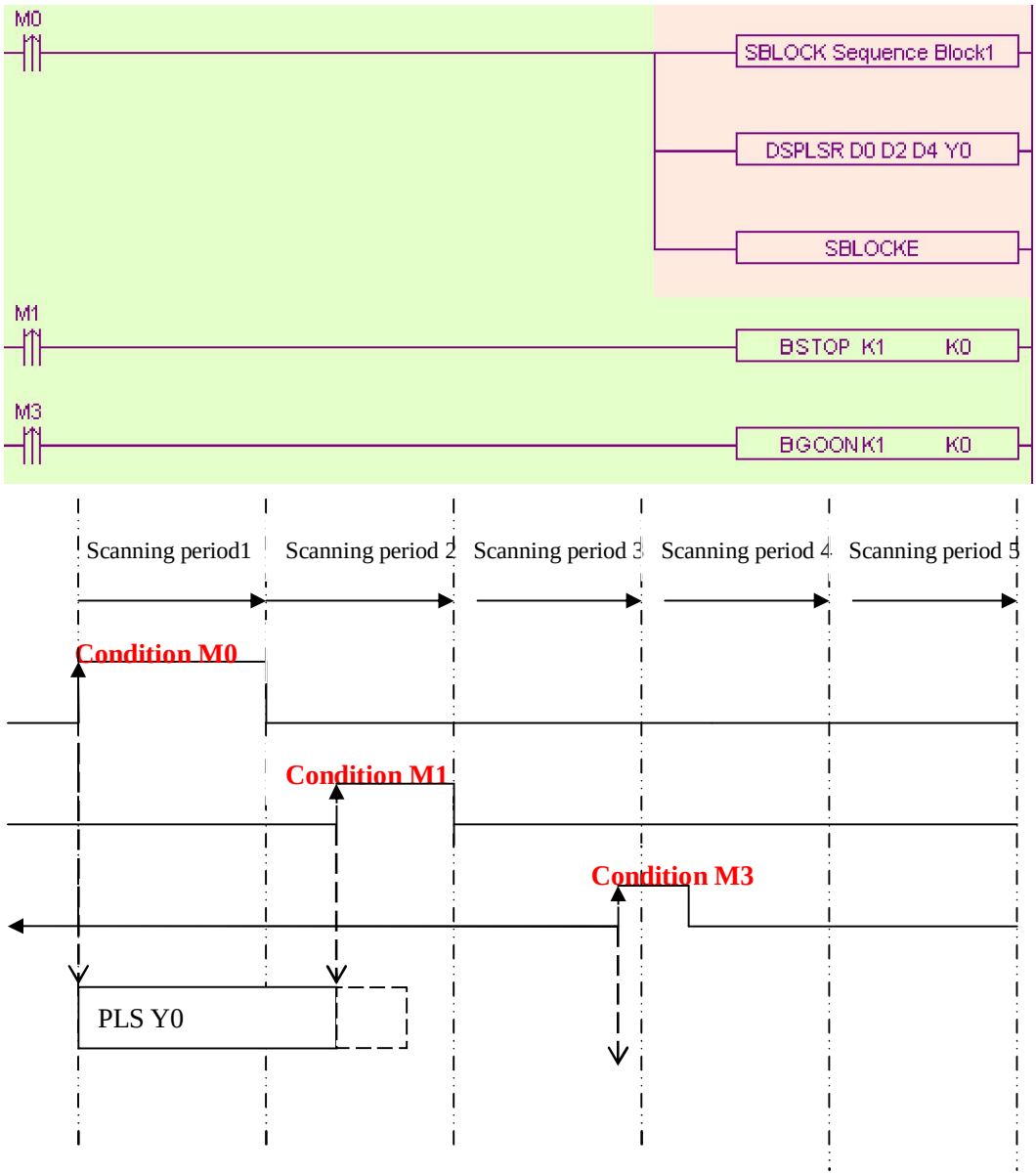
Function



l S2 is the mode to continue running the BLOCK. Operand: K0, K1. K0: continue running the instructions in the BLOCK. For example, if pulse outputting stopped last time, BGOON will continue outputting the rest pulse. K1: continue running the BLOCK, but abandon the instructions have not finished last time. Such as the pulse output instruction, if the pulse has not finished last time, BGOON will not continue outputting this pulse but go to the next instruction in the BLOCK.

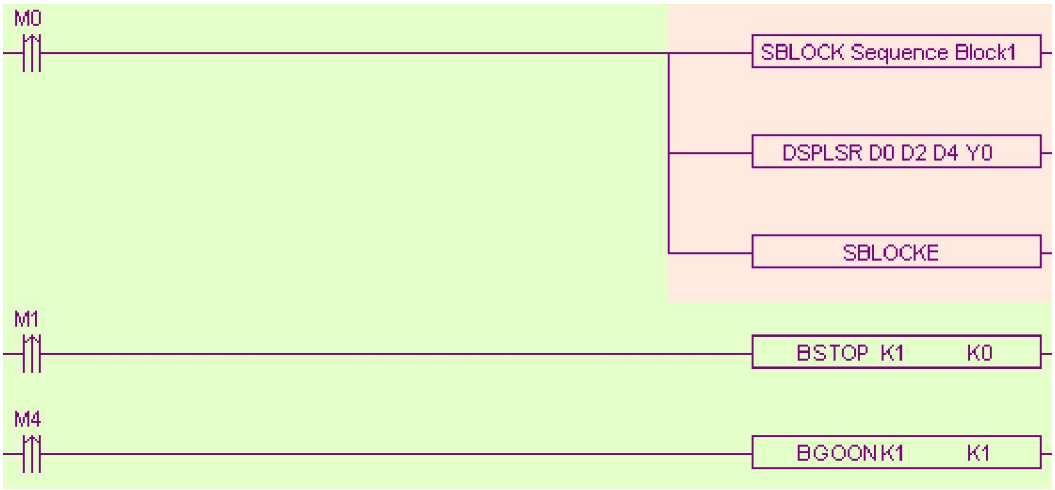
10-6-2. The timing sequence of the instructions

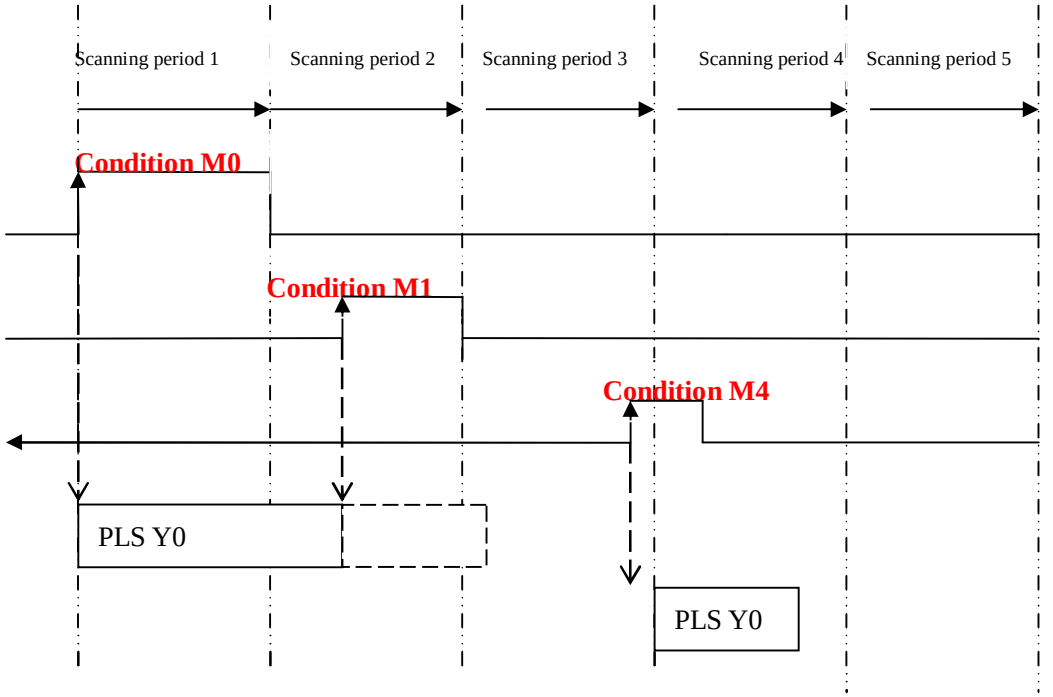
1、BSTOP（K1 K0）+BGOON（K1 K0）



When M0 is from OFF→ON, run “DSPLSR D0 D2 D4 Y0” in the BLOCK to output the pulse; when M1 is from OFF→ON, the BLOCK stops running, pulse outputting stops at once; when M3 is from OFF→ON, abandon the rest pulse.

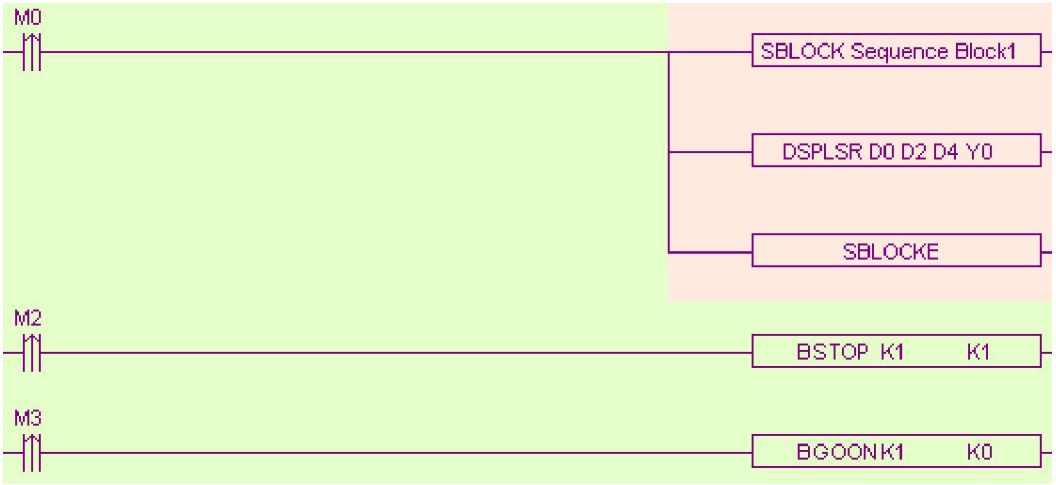
2、BSTOP（K1 K0）+BGOON（K1 K1）

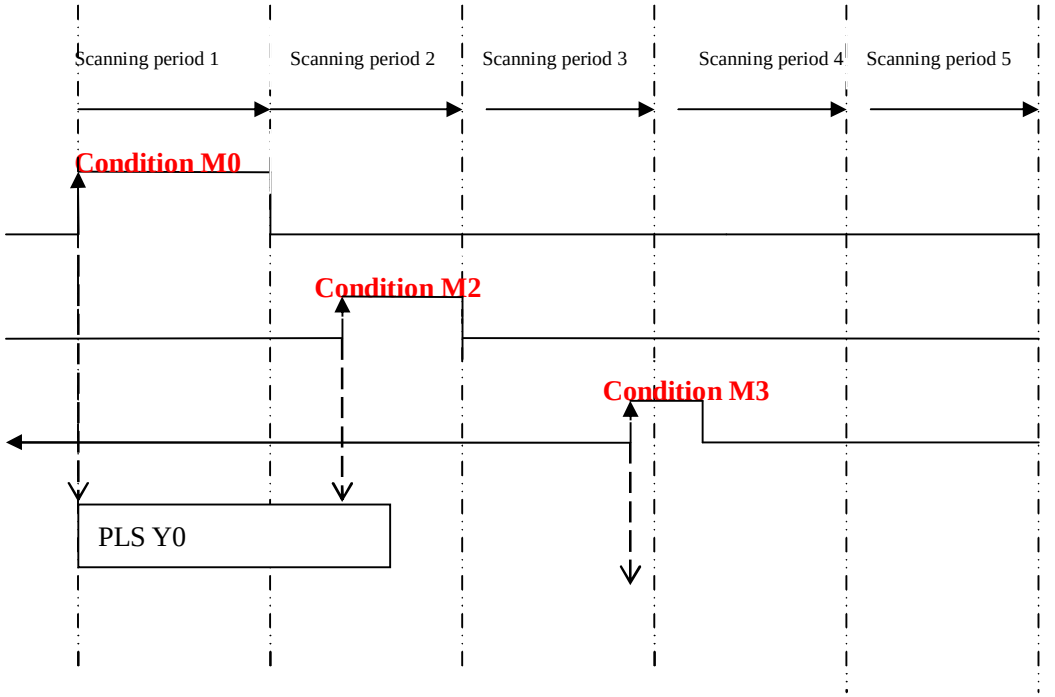




When M0 is from OFF→ON, run “DSPLSR D0 D2 D4 Y0” in the BLOCK to output the pulse; when M1 is from OFF→ON, the BLOCK stops running, the pulse outputting stops at once; when M4 is from OFF→ON, output the rest pulses.

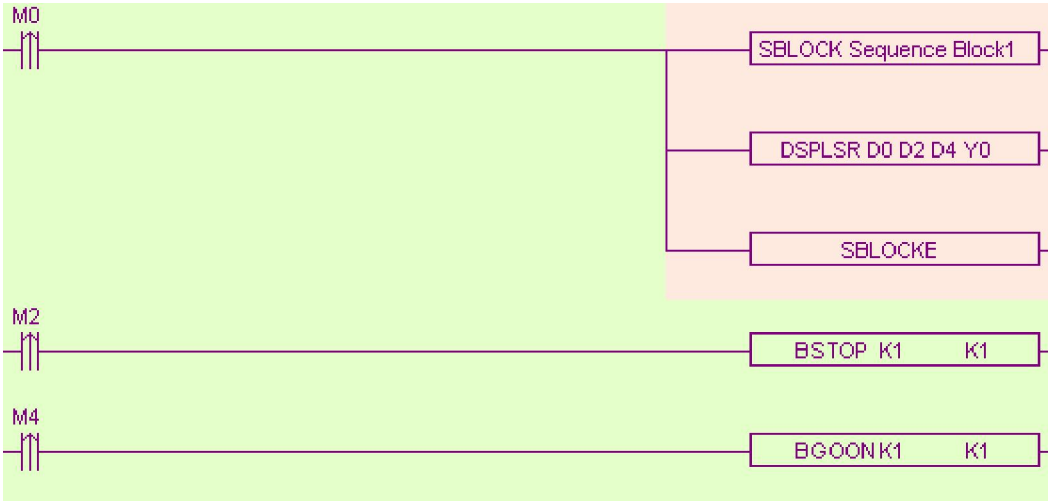
3、BSTOP（K1 K1）+BGOON（K1 K0）

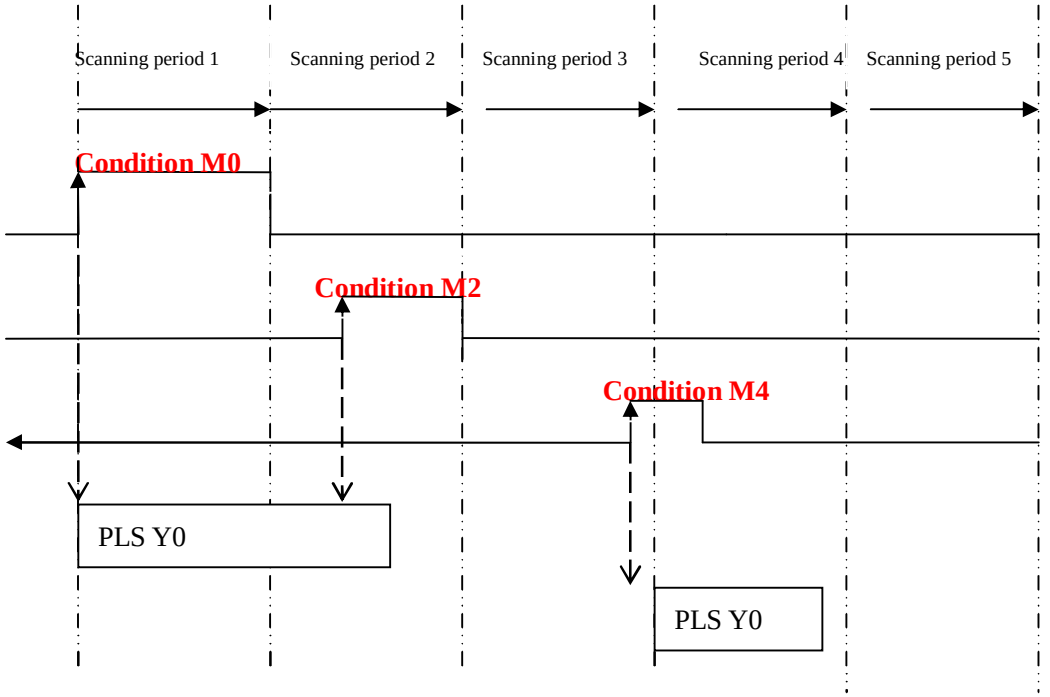




When M0 is from OFF→ON, run “DSPLSR D0 D2 D4 Y0” in the BLOCK to output the pulse; when M2 is from OFF→ON, stop the BLOCK, the pulse will stop slowly with slope, when M3 is from OFF→ON, abandon the rest pulses.

4、BSTOP（K1 K1）+BGOON（K1 K1）





When M0 is from OFF→ON, run “DSPLSR D0 D2 D4 Y0” in the BLOCK to output the pulse; when M2 is from OFF→ON, stop running the BLOCK, the pulse will stop slowly with slope; when M4 is from OFF→ON, output the rest pulses.

Please note that though the BSTOP stops the pulse with slope, there maybe still some pulses; in this case, if run BGOON K1 K1 again, it will output the rest of the pulses.

10-7．BLOCK flag bit and register

1、BLOCK flag bit:

Address	Function	Explanation
M8630		1: running 0: not running
M8631	BLOCK1 running flag	
M8632	BLOCK2 running flag	
…….	…….	
……..	…….	
M8730	BLOCK100 running flag	

2、BLOCK flag register

Address	Function	Explanation
D8630		BLOCK use this value when monitoring
D 8631	BLOCK1 current running instruction	
D8632	BLOCK2 current running instruction	
…….	…….	
……..	…….	
D8730	BLOCK10 current running instruction	

10-8. Program example

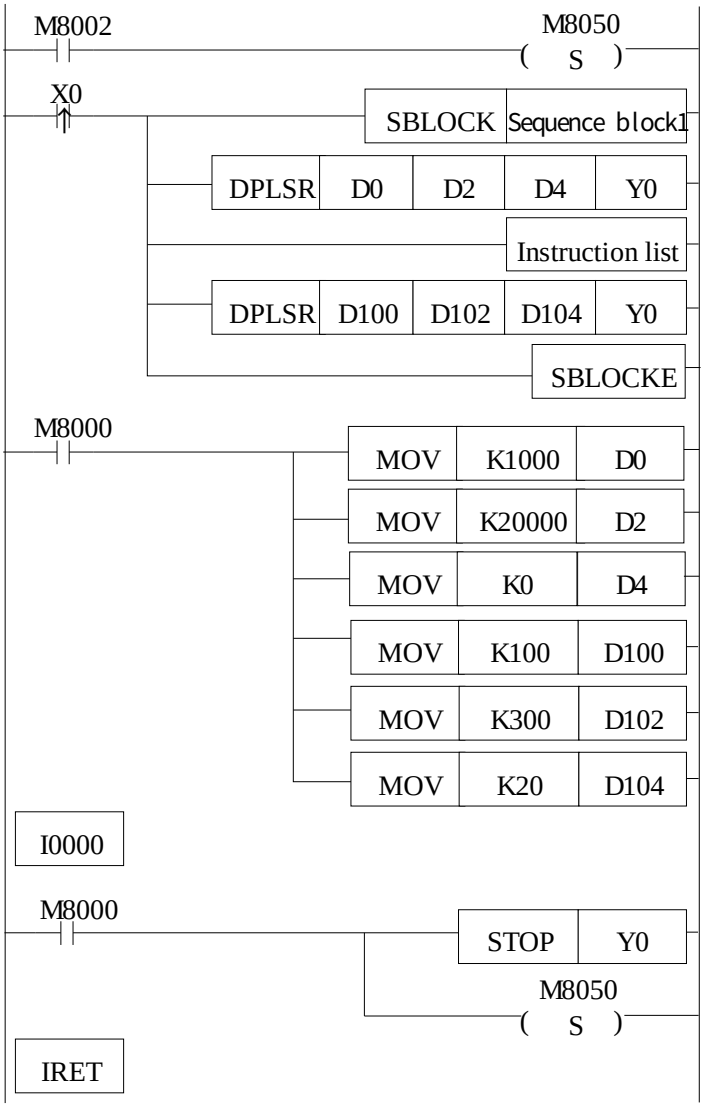
Example:

This example is used in the tracking system. The process is like this:

Output some pulses and prohibit the exterior interruption.

Continue outputting the pulse but at low speed, and open the exterior interruption. When checked the exterior cursor signal, stop the pulse outputting and machine running.

Ladder chart:



PLC power on, prohibit the exterior interruption

BLOCK starts

Output the pulses and move some distance

Reset M8050, open the exterior interruption

Output the pulses at low speed

BLOCK ends

The first pulse frequency

The first pulse numbers

Accelerate/decelerate time for the first pulse

The second pulse frequency

The second pulse numbers

Accelerate/decelerate time for the second pulse

The interruption starts

Stop outputting the pulse

Close the interruption

The interruption ends

The instruction list content:

RST M8050

Notes:

M8050: prohibit the exterior interruption