Servo Tech Middle East Co. Ltd.

Advanced User Guide

Programmable single axis stepper motor controller

Type

MSP 01

For stepper motors,

Up to 50 kHz output frequency (Pulse + Dir),
With 2 high priority INTERRUPT,
8 digital inputs (isolated) ,
8 transistor outputs,
220 VAC supply

MSP 01
Programmable single axis stepper motor controller

Main features:

1) two mode of operation , RUN mode and PROGRAMMING mode,

2) 50 kHz maximum output frequency,

3) up to 100 line capacity for programming,

4) ability for LOOP function,

5) Two higher priority INTERRUPT for digital inputs, JP0 & JP1,

6) 8 optically isolated digital inputs,

7) 8 digital outputs (non isolated),

8) 0~9999.99 mm for adjusting the maximum range of destination point,

9) ability for adjusting the scale of motion,

10) ability for adjusting linear acceleration and deceleration ramp up to 35 kHz/s,

11) 8 digit of 7-seg display for RUN mode and PROGRAMMING mode,

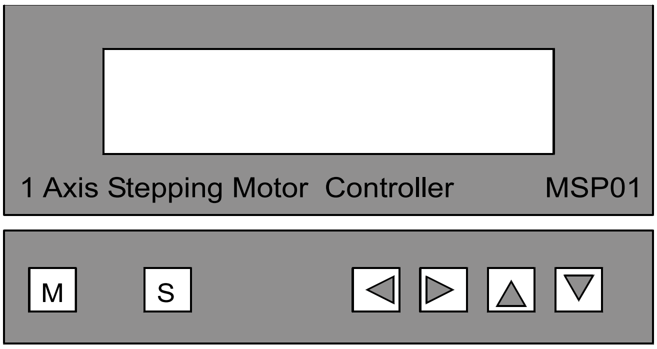
12) keypad for programming and adjusting parameters,

13) RUN / STOP command is controllable from keypad or from external inputs according to program,

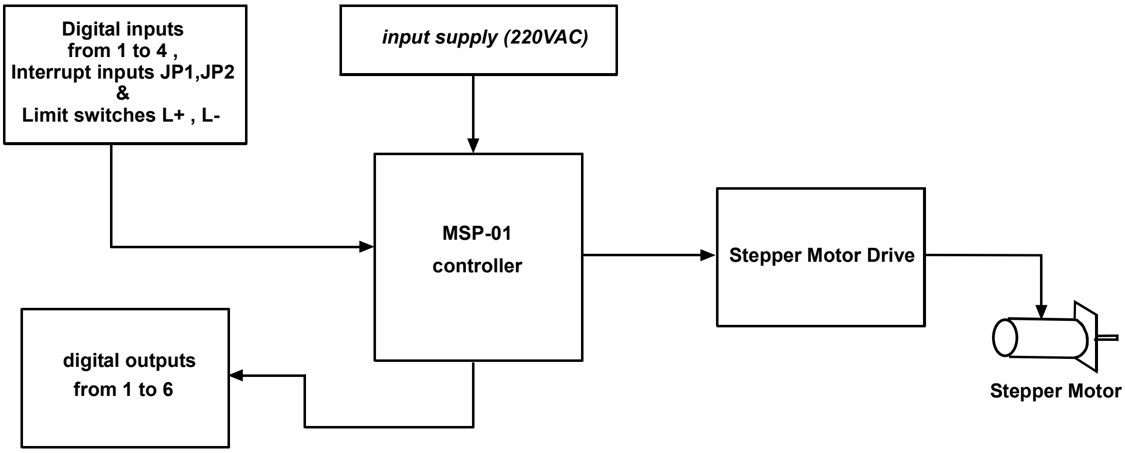
14) 220 VAC input supply (+/- 15%),

Front view:

1) 6 push button on the panel for programming or for controlling and adjustment of the parameters,
2) 8 digit for display,



Wiring schematic :



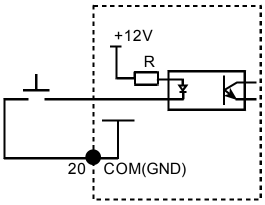
Terminal I/O Descriptions :

Terminal No.	Definition	Description
1	IN1	Digital input number 1
2	IN2	Digital input number 2
3	IN3	Digital input number 3
4	IN4	Digital input number 4
5	CWL	Clockwise limit switch input **(L+)**
6	CCWL	counterclockwise limit switch input **(L-)**
7	JP2	Interrupt input number 2 **(JP1)**
8	JP1	Interrupt input number 1 **(JP0)**
9	**220VAC**	**AC supply input**
10	**220VAC**	**AC supply input**
11	OUT1	Digital output number 1
12	OUT2	Digital output number 2
13	OUT3	Digital output number 3
14	OUT4	Digital output number 4
15	OUT5	Digital output number 5
16	OUT6	Digital output number 6
17	**CP**	**"Clock pulse (CP) "output to stepper drive**
18	**DIR**	**"Direction (Dir) " signal output to stepper drive**
19	+5V(300mA)	Auxiliary "+5VDC " output
20	GND	GND

Input & output circuitry :

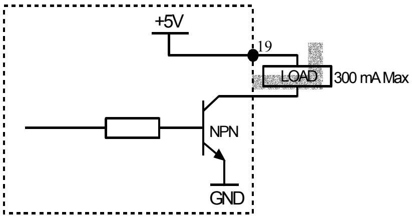
Input circuitry :

Please note to the voltage level of internal supply for digital inputs (+12VDC).
You can use a normally open (N.O.) , or normally close (N.C.) contact of a relay to apply the input.



Output circuitry :

Please note to the voltage level of the internal supply for digital output signals (+5VDC).
You can use a suitable relay (+5V coil excitation) to connect the outputs. If other voltages are required , you can use an opto coupler and related circuit to do that.



Modes of operation :

1) PROGRAM mode:
in order to enter the lines of program ,one by one , and/or to change and adjust the parameters of each line you can use the PROGRAM MODE.
If **M** key is pressed and held for 3 seconds the controller will switch to program mode.

2) RUN mode:
to run the program either externally or by keypad, you can use RUN mode.
When you power ON the controller , or when **M** key is pressed shortly in PROGRAM MODE, the controller is switched to RUN MODE.
In RUN MODE, the execution of program is viewed line by line when it is in operation.

Keypad operation :

1) ***M*** *key:* M

M key is used to enter the program mode.

Press and hold **M** key M for 3 seconds, you will enter the PROGRAM MODE.
Press it shortly , you will return back to RUN MODE.

2) ***S*** *key:* S

S key is used to define , adjust , edit and save a line of program or the value of a parameter.

When in PROGRAM MODE , you can press **S** key S on a line number to edit that line or to check its content.

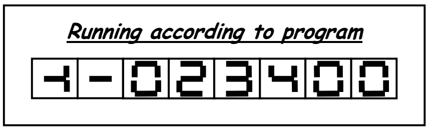
To SAVE the modifications , you can press **S** key S shortly after finishing the adjustments.

Hint! :

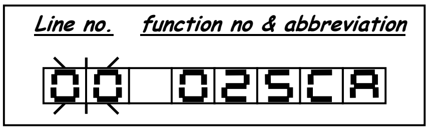
If **M** key is pressed instead of **S** key after modification of any parameter the controller will return back to RUN MODE without saving the whole modifications.

Programming procedure:

*At power ON , the controller is in **RUN MODE**.*

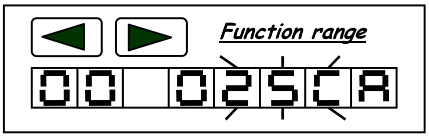


You can switch to PROGRAM MODE if you press **M** key M for 3 seconds.
When entering to program mode , you will see the first line of program with its line number as 00.

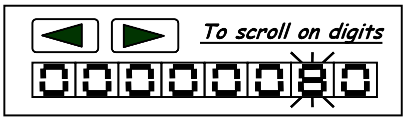


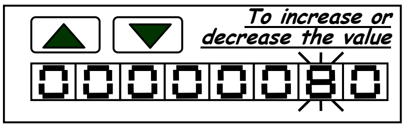
If you want to start programming for the first time , you can enter the desired function according to your program from the first line (00).

For this purpose you must press **S** key S first, to enter to the function range, and then you can use left arrow ◀ or right arrow ▶ key to select your desired function.



After selection the function you can press **S** key S shortly to **SAVE** its number of line in the program, and also to enter the content register of the function to define its value.

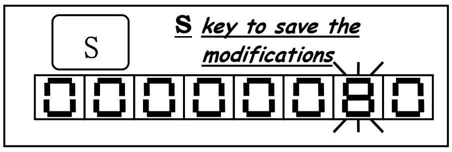




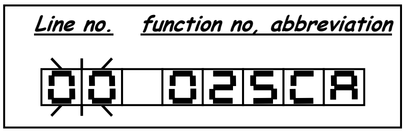
With up arrow ▲ or down arrow ▼ you can change the value of the content register for each parameter as you like.

You can use left arrow ◀ or right arrow ▶ to scroll between digits.

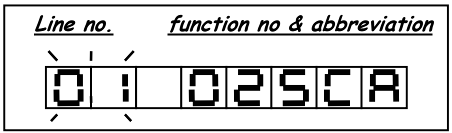
After adjusting the desired value you can **SAVE** it by pressing **S** key S shortly.



If you do that the function value is saved and the controller will return back to PROGRAM mode, on the current line number.



For programming the next line you can use up arrow ▲ or down arrow ▼ key to change the line number as required.



At the end of programming if you press **M** key M shortly, the controller will switch to RUN MODE and the program execution will start.

Editing a program:

For editing a program you can use **M** key M to enter to PROGRAM MODE,

(Press it for 3 seconds) after entering the program mode use down arrow ▼ or up arrow key ▲ to find the line number which you want to edit.

Afterwards you can press **S** key S one time shortly to edit the function or press **S** key one more time S to edit the value of that function.

After editing , press **S** key S one more time to **SAVE** the whole modifications and to return back to RUN MODE.

Function table:

Func. no.	name	display	Definition	range
00	End of prog.	**00End**	Last line of the program	-
01	Running frequency	**01SPE**	Output frequency to drive for next line of position command in Hz	**0-50,000**
02	Scale of motion	**02Sca**	Pulse number needed to rotate the motor to give a 1mm linear movement to the load	**0-65535**
03	Run the program	**03run**	Starts the motor to run in defined direction till the next STOP function (0 for CW, 1 for CCW)	-
04	Stop the program	**04Sto**	Stops the execution of program	-
05	Length of motion in CW	**O5Cw**	Defines the CW movement length in mm	**0-9999.99**
06	Length of motion in CCW	**06Ccw**	Defines the CCW movement length in mm	**0-9999.99**
07	Delay time	**07DEL**	Delay time as per need in program in ms	**1-65535**
08	High level at INPUT port	**08IN+**	Waits for HIGH level at input port (1-4) to activate the program line (for N.C. contacts at input)	**1-4**
09	Low level at INPUT port	**09IN-**	Waits for LOW level at input port (1-4) to activate the program line (for N.O. contacts at input)	**1-4**
10	High level at OUTPUT port	**10 OU+**	Makes the output port (1-6) to HIGH level on this line of program (port must be pulled up)	**1-6**
11	Low level at OUTPUT port	**11 OU-**	Makes the output port (1-6) to LOW level on this line of program (port must be pulled up)	**1-6**
12	-	-	Not used	-
13	Loop cycle number	**13Let**	Number of LOOP cycles to be repeated	**0-65535**
14	Decrement pointer	**14Dec**	Subtracts the down current value of LOOP counter by 1, will jump to a defined line if the counter get zero, if not zero continues with the next line	**0-99**
15	Jump unconditionally	**15Go**	Execution will jump to the defined line unconditionally	**0-99**
16	Start frequency	**16JF**	Start frequency of output clock pulse in Hz	**0-9999**
17	Ramp rate	**17rS**	Ramp rate of acceleration and deceleration of motion , linear and equal at both ends, in kHz/s	**0-50**
18	Interrupt no. 1	**18JP0**	Interrupts the program and jumps to the defined line if the corresponding input (JP1) is connected to GND (closed)	**0-99**
19	Interrupt no. 2	**19JP1**	Interrupts the program and jumps to the defined line if the corresponding input (JP2) is connected to GND (closed)	**0-99**

Example1:

write a program to rotate the stepper motor with a linear velocity of 10mm/s ,
for a length of 30cm. the resolution is 400 ppr, and the ball screw pitch is 5mm/rev.
after finishing the program ,the motor must wait at that position for 1 second and repeat again the procedure.
The acceleration rate should be 6 khz/s and the start frequency from 100hz.

Program 1:

V=10mm/s, in every revolution the motor will move 5mm, if this value is passed in 1 sec , the speed will get 5mm/s, to be at this velocity the motor must receive 400 pulse /s or 400hz, so for 10mm/s the frequency must be 2x400=800hz.
Scale = (pulse/rev) / (mm/rev) = (pulse/mm), So 400/5 = 80
Target position is 30cm , or 300mm, so the figure in the destination point will be 300x0.01=30000 (the resolution of measurement is 0.01mm as it is in the controller)

```
00  01SPE  800     :/ defines the running frequency as 800 hz
01  02SCa   80    :/  defines the scale ratio as 80
02  05Cw    30000     :/  defines the length of motion in Cw direction as 300 mm
03  07DEL  1000     :/ defines the delay time (1s)
04  15Go    00    :/  lead the program execution to the line number 00
05  00End   00    :/  end of main program
06  16JF    100    :/  defines the start frequency as 100 hz
07  17Rs    6    :/  defines the acceleration and deceleration rate as 6 kz/s
08  00End  00    :/ end of program
```

Note : The line numbers 06 and 07 are not visible on the display although you have entered them in the program.

Example 2:

Write a program for a **plastic cutting machine** that starts the motor to run when sensor no. 1 is detected (the sensor will close at detection point) and stops the motor when sensor no. 2 is detected (the sensor will close at detection point). The speed of motion must be 20 cm/s , the length of motion for 1 revolution of motor shaft is 5cm , and the motor will rotate 1 revolution for every 1000 pulses input to drive. The distance between every 2 mark on the plastic sheet is about 20cm. The acceleration rate to be 15 khz/s and the start frequency to be 50 hz.

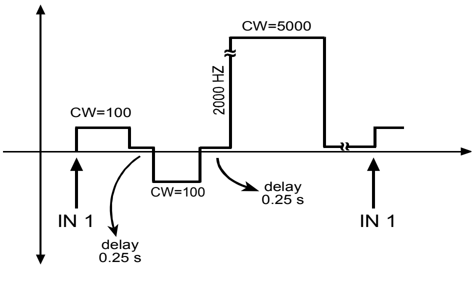
Program 2:

V=200mm/s, in every revolution the motor will move 50mm, if this value is passed in 1 sec , the speed will get 50mm/s, to be at this velocity the motor must receive 1000 pulse /s or 1000hz, so for 200mm/s the frequency must be (200/50)x1000 = 4x1000=4000hz.
Scale = (pulse/rev) / (mm/rev) = (pulse/mm), So 1000/50 = 20
Target position is about 20cm or 200mm, but the exact value is detected by the sensor no.2 at the **JP0** input port, so the stop point length is not defined by CW or CCW commands exactly , instead , it is managed by sensor input, *but for safety and better performance at stopping point we can adjust the rough stopping point at neighborhood of the mark position ,say about 22cm away from starting point, in this way we can have a deceleration effect at stop point although the exact stop point is managed by sensor detection momentarily.*
The inputs are normally open and will close to GND when detected.

```
00 09IN-   01     :/  wait until the input number 1 is detected ( CLOSES to gnd )
01 01SPE   4000     :/  defines the running frequency as 4000 hz
02 02SCa   20     :/ defines the scale ratio as 20
03 05Cw   44000     :/  starts the motor to RUN in CW direction for 22cm
04 15Go   00     :/  the execution of the program to line number 00
05 00End   00     :/  end of main program
06 18JP0   07     :/  waits for interrupt input no.1 at TERMINAL NO. 18 with higher
                        priority rather than the CW or CCW commands ( CLOSES to gnd )
07 04Sto   00     :/  stops the motor immediately
08 15Go    00     :/  the execution of the program to line number 00
09 00End   00     :/  end of main program
10  17Rs    15     :/  acceleration and deceleration rate as 15 kz/s
11  16JF   10     :/  defines the start frequency as 10 hz
12  00End  00     :/  end of program
```

Example 3 :

A machine for **cutting and clipping the wire head** uses a stepping motor.
The motor must rotate 36 deg in CW direction in 0.2 sec , then return back in the original position and finally rotates 5 turns in CW in 1 sec and stops till the next input command is activated again.
A delay of 0.25 sec is needed between direction change.
Drive is in half step mode, so 400 pulses are needed to run the motor for 1 complete turn.
Scale is assumed to be 10.



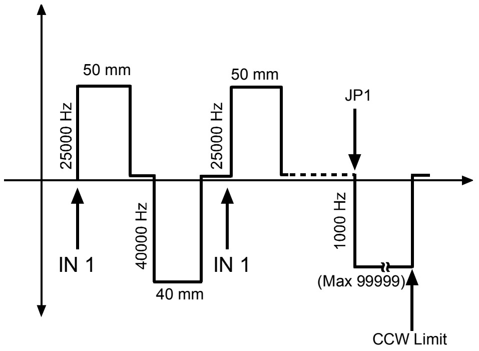
Program 3 :

Speed is 36 deg in 0.2 sec or 1 revolution in 2 sec., so in 1 second it is 0.5 turn or 200 pulse/sec , it means 200 Hz.
Scale is 10 ,
36 deg is 10% of a complete turn or 40 pulses.
So the length of motion will get (40 pulse) / (10 pulse/mm) = 4mm
5 turns is 2000 pulses or 200mm, accordingly.
The speed for 5 turns per second is 2000 pulses per second or 2000 Hz.

```
00 09IN-   01     :/  wait until the input number 1 is detected ( CLOSES to gnd )
01 01SPE   200     :/  defines the running frequency as 200 hz
02 02SCa   10     :/ defines the scale ratio as 10
03 05Cw   400     :/  starts the motor to RUN in CW direction for 4mm
04 07DEL   250     :/  wait for 0.25 s.
05 06Ccw   400     :/  returns for 4mm in CCW direction
06 07DEL   250     :/  wait for 0.25 s.
07 01SPE   2000     :/  speed to be 2000Hz
08 05Cw    20000     :/  starts the motor to RUN in CW direction for 200mm
09  15Go    00     :/  lead the execution of the program to line number 00
10  00End  00     :/  end of program
11  16JF    10     :/  starting frequency as 10 hz
12  17Rs    15     :/  acceleration and deceleration rate as 15 kz/s
13  00End  00     :/  end of program
```

Example 4 :

A mechanism driven by a stepper motor has the following algorithm :
After power ON it waits for IN1 to be detected (at IN1 is a normally closed or N.C. contact).
At this moment it starts to move in CW direction for 50mm in 1 sec.
If again IN1 is activated it moves in CCW direction for 40mm in 0.5 sec.
Meanwhile , if at any instance during work **INTERRUPT 1 (JP0)** is activated it will move in CCW direction until **CCW LIMIT (L-)** switch is detected, at the speed of 2mm/s.
The mechanism will move 2mm for 1 complete revolution of motor, and the motor needs 1000 pulses for 1 complete revolution.



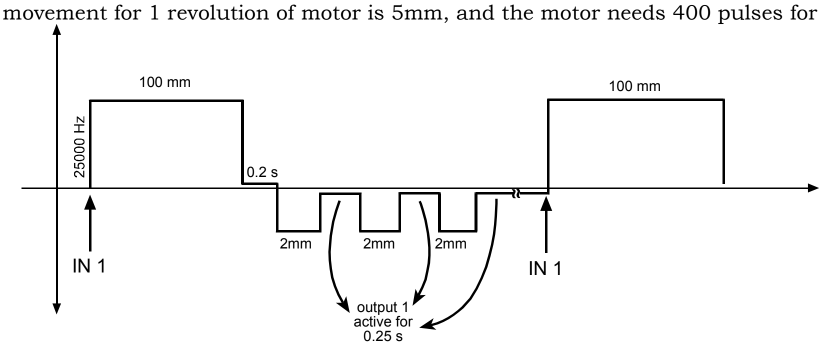
program 4 :

scale will get (1000 pulse/rev) / (2mm/rev) = 500 ,
for 50mm stroke it will need 25 complete turn and to move this value in 1 sec it needs 25000 pulse/s or 25000 Hz.
To move 40mm in 0.5 sec the speed should be ((40/2)x1000) / 0.5 = 40000 pulse/s or 40000 Hz.

```
00  01SPE   25000    :/  defines the running frequency as 25000 hz
01  02Sca   500     :/ defines the scale ratio as 500
02  08IN+    01     :/ wait until the input number 1 is detected ( OPENS from gnd )
03  05Cw   5000     :/  starts the motor to RUN in CW direction for 50mm
04  08IN+    01     :/ wait until the input number 1 is detected ( OPENS from gnd )
05  01SPE   40000    :/defines the running frequency as 40000 hz
06  06Ccw   4000     :/  returns for 40mm in CCW direction
07  15Go    00     :/  lead the execution of the program to line number 00
08  00End   00     :/  end of program
09  18JP0   10    :/  waits for interrupt input no.1 at TERMINAL NO. 18 with higher
                       priority rather than the CW or CCW commands ( CLOSES to gnd )
10  01SPE   1000     :/  defines the running frequency as 1000 hz
11  06Ccw   99999     :/  returns for 999.99mm in CCW direction ( actually the
                            dimension of the mechanism is shorter than 999.99mm , so the
                            side limit switches will be detected in this way )
12  15Go    00     :/  lead the execution of the program to line number 00
13  00End   00     :/  end of program
14  16JF    00    :/  starting frequency as 0 hz
15  17rS   20    :/  acceleration and deceleration rate as 20 kz/s
16  00End   00     :/  end of program
```

Example 5 :

A mechanism after power ON waits for IN1 to be detected (normally open N.O.).
If IN1 is activated the mechanism will move 100mm in CCW direction in 5 sec , after 0.2 sec delay it will move in CW direction for 2mm and at the end of motion the controller will give an output for 0.25 sec at OUTPUT 1.
After that, the mechanism will move for next 2mm and stop again, and output 1 will get ON for 0.25 sec.
This is repeated for 5 times totally. After finishing the procedure, the controller will wait for IN1 to start the whole program again.
The length of movement for 1 revolution of motor is 5mm, and the motor needs 400 pulses for that.



program 5 :

100mm is 20 complete turns of motor. It must be traveled in 5 sec. , so the speed will be (20x400) / 5 = 1600 pulse/s or 1600 Hz.
Scale will get 400 / 5 = 80 pulse/mm

```
00  01SPE   1600    :/  defines the running frequency as 1600 hz
01  02Sca    80     :/ defines the scale ratio as 80
02  09IN-    01     :/  wait until the input number 1 is detected ( CLOSES to gnd )
03  06Ccw   10000    :/  starts the motor to RUN in CCW direction for 100mm
04  07DEL   200    :/  wait for 0.2 s.
05  13Let    05    :/  start line of LOOP function for 5 times repeat
06  05Cw    200     :/  starts the motor to RUN in CW direction for 2mm
07  11Ou-   01    :/   will switch the output 1 to LOW or GND. Normally all the outputs
                        are at HIGH level ( VCC=5VDC ) , so in this way they are activated
                        to switch the load to GND.
08  07DEL   250    :/  wait for 0.25 s.
09  10Ou+  01     :/  will switch the output 1 to HIGH or VCC=5VDC
10  14DEC  02     :/ decrease the loop counter by ONE, if the result is ZERO , will jump
                      to the defined line number ( 02 here ), if not will continue with the
                      next line
11  15Go    06     :/  lead the execution of the program to line number 00
12  00End   00     :/  end of program
13  16JF     50    :/  starting frequency as 50 hz
14  17rS     10    :/  acceleration and deceleration rate as 10 kz/s
15  00End   00     :/  end of program
```