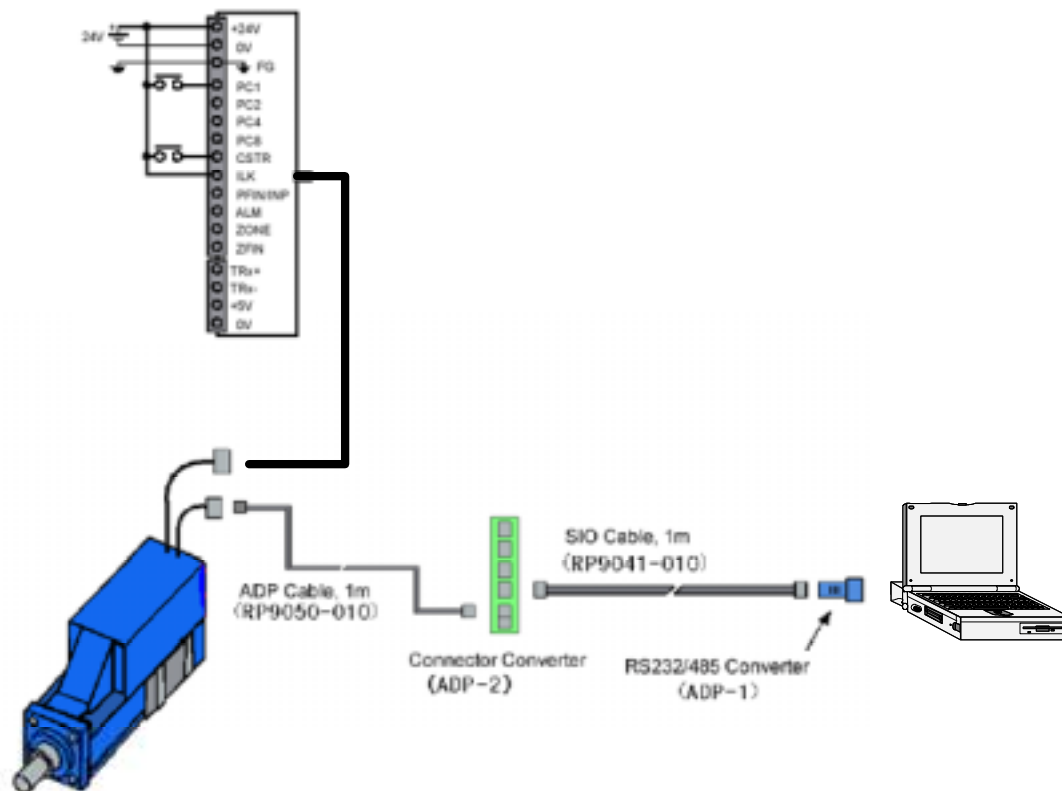


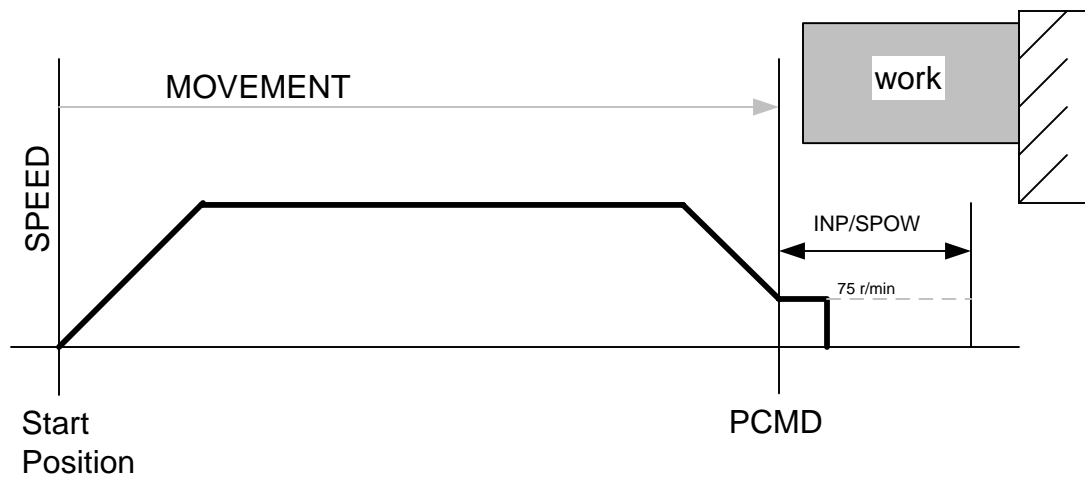
System Configuration



<u>Part #'s</u>	<u>Description</u>
SCN05-010-100-AS03	Mechatronics Cylinder
RP9050-010	ADP Cable
RP9100-030	Parallel Cable (power & I/O)
ADP-2	Connector Junction
ADP-1	RS232/RS485 Adapter
RP9041-010	SIO Cable
TBVST	Termi-BUS command output tool software
	Hyper-Terminal
	Terminal Block
	Laptop

Application Goal

The goal of this application is change the command position, pusher movement and pusher force command via serial communication. Whether it be a laptop or PLC.



Position Data for Point #01

PCMD = -6787 pulses	(command position)
INP = 1141 pulses	(pusher movement)
SPOW = 50%	(pusher force command)

With the use of Hyper-terminal or similar software, the following ascii strings are used to change PCMD, INP and SPOW. The ascii strings need to be entered in consecutive order as follows:

- | | |
|-------------------------|---|
| 1) STX0T400000400094ETX | This will write to add. H0400 (PCMD) |
| 2) STX0W4FFFFE57D008ETX | Change command position #1 to -6787 pulses |
| 3) STX0T400000403091ETX | This will write to add. H0403 (INP) |
| 4) STX0W400000475085ETX | Change pusher movement to 1141 pulses |
| 5) STX0T40000040608EETX | This will write to add. H0406 (SPOW) |
| 6) STX0W40000005A07FETX | Change pusher force command to 50% |
| 7) STX0V501010000093ETX | Write to memory for position area point #01 |

ASCII Character String Format

Format

Header	Station #	Function Character	Function Number	Operand+0	BCC	Delimiter
STX	0	T	4	0000 0400 0	94	ETX

Calculate Block Check Character (BCC)

Convert to Hexadecimal

STX	0	T	4	0	0	0	0	0	4	0	0	0	9	4	ETX
H02	H30	H54	H34	H30	H30	H30	H30	H30	H34	H30	H30	H30	H39	H34	H03

Add Hex values starting at Station # thru Operand+00

$$H30+ H54+ H34+ H30+ H30+ H30+ H30+ H30+ H34+ H30+ H30+ H30= H26C$$

Convert to binary

$$H26C = 0010\ 0110\ 1100$$

Delete bits 8 - 11

or

Since,

$$H6C = 0110\ 1100$$

$$H6C = 0110\ 1100 = 108$$

Invert bits

Negate 108 to -108

$$\text{NOT } (0110\ 1100) = 1001\ 0011$$

$$-108 = 1001\ 0100 = H94$$

Add binary 1

Thus BCC = 94 (hex)

$$\begin{array}{r} 1001\ 0011 \\ +0000\ 0001 \\ \hline 1001\ 0100 \end{array}$$

2's complement = 1001 0100

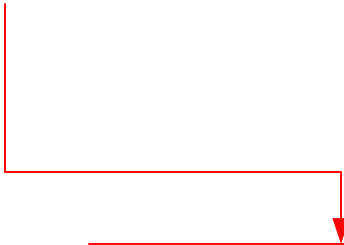
Convert to Hexadecimal

$$1001\ 0100 = H94$$

BCC = 94 (Hex)

T4 command

- This command writes to the virtual memory address area.



[STX] + [STAT#] + [T4] + [Memory Address] + [0] + [BCC] + [ETX]

[02] + [0] + [T4] + [0000 0400] + [0] + [94] + [03]

Memory Address H0400 (PCMD)

- Absolute position coordinate target positioning stop/relative movement stroke.
- Absolute coordinate position of deceleration completion target position for approach movement.
- Relative movement stroke from current position of deceleration completion target position for approach movement.

Memory Address H0403 (INP)

- In position width/push depth.

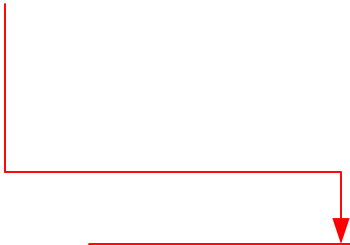
Memory Address H0406 (SPOW)

- Electrical current limitation value for positioning stop.
- Electrical current limitation value for push movement.

Memory Commands

W4 command

- This command writes data to the T4 memory address.

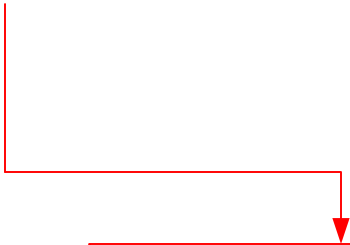


[STX] + [STAT#] + [W4] + [Command Data] + [0] + [BCC] + [ETX]

[02] + [0] + [W4] + [FFFF E57D] + [0] + [08] + [03]

V5 command

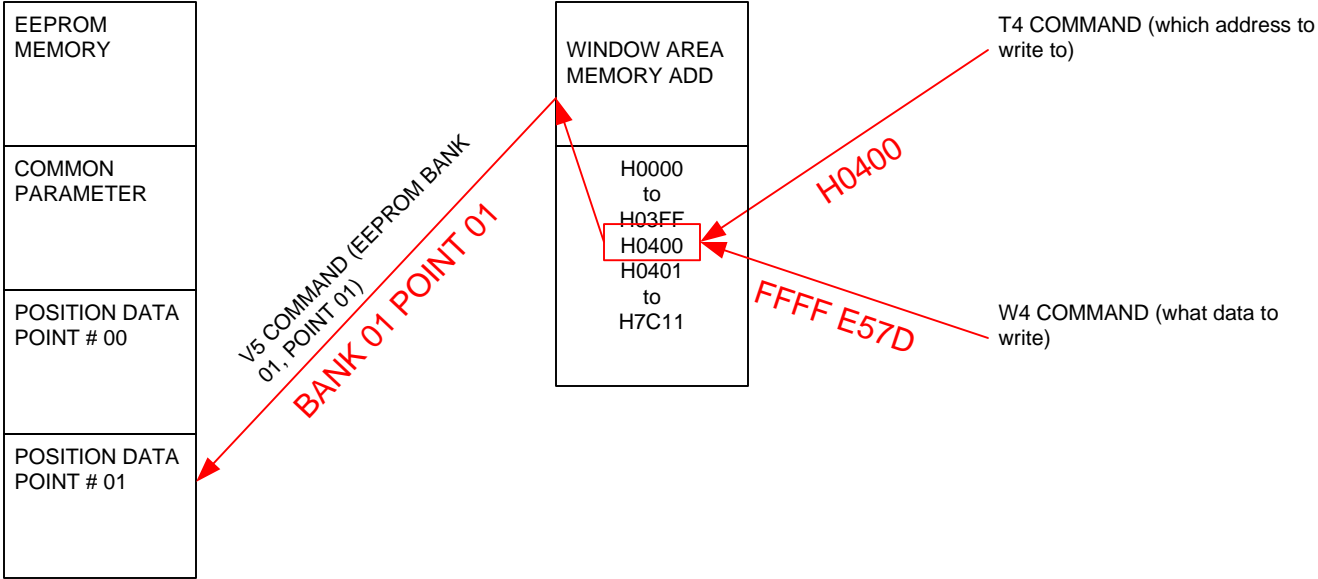
- This command transmit position data to EEPROM.



[STX] + [STAT#] + [V5] + [Bank #] + [Pos.#] + [00000] + [BCC] + [ETX]

[02] + [0] + [V5] + [01] + [01] + [00000] + [93] + [03]

Memory Transfer



ASCII Character Set

Some commands described in this manual require that you enter the decimal representation of an ASCII character. Table D-1 provides code translations from ASCII characters to decimal numbers. For example, the ASCII carriage return (CR) is decimal 13. This means that pressing Ctrl-M at your terminal generates decimal 13, which is interpreted as a CR.

Table D-1 ASCII Translation Table

Numeric Values		ASCII Character Name	Comment	Keyboard Entry
Decimal	Hex			
0	00	NUL	Null	Ctrl-@
1	01	SOH	Start of heading	Ctrl-A / ^A
2	02	STX	Start of text	Ctrl-B / ^B
3	03	ETX	Break/end of text	Ctrl-C / ^C
4	04	EOT	End of transmission	Ctrl-D
5	05	ENQ	Enquiry	Ctrl-E
6	06	ACK	Positive acknowledgment	Ctrl-F
7	07	BEL	Bell	Ctrl-G
8	08	BS	Backspace	Ctrl-H
9	09	HT	Horizontal tab	Ctrl-I
10	0A	LF	Line feed	Ctrl-J
11	0B	VT	Vertical tab	Ctrl-K
12	0C	FF	Form feed	Ctrl-L
13	0D	CR	Carriage return	Ctrl-M
14	0E	SO	Shift out	Ctrl-N
15	0F	SI	Shift in/XON (resume output)	Ctrl-O
16	10	DLE	Data link escape	Ctrl-P
17	11	DC1	Device control character 1	Ctrl-Q
18	12	DC2	Device control character 2	Ctrl-R
19	13	DC3	Device control character 3	Ctrl-S
20	14	DC4	Device control character 4	Ctrl-T
21	15	NAK	Negative Acknowledgment	Ctrl-U
22	16	SYN	Synchronous idle	Ctrl-V
23	17	ETB	End of transmission block	Ctrl-W

Numeric Values		ASCII Character Name	Comment	Keyboard Entry
Decimal	Hex			
24	18	CAN	Cancel	Ctrl-X
25	19	EM	End of medium	Ctrl-Y
26	1A	SUB	substitute/end of file	Ctrl-Z
27	1B	ESC	Escape	Ctrl-[
28	1C	FS	File separator	Ctrl-\
29	1D	GS	Group separator	Ctrl-]
30	1E	RS	Record separator	Ctrl-^
31	1F	US	Unit separator	Ctrl-_
32	20	SP	Space	Space
33	21			!
34	22			“
35	23			#
36	24			\$
37	25			%
38	26			&
39	27			‘
40	28			(
41	29)
42	2A			*
43	2B			+
44	2C			,
45	2D			-
46	2E			.
47	2F			/
48	30			0
49	31			1
50	32			2
51	33			3
52	34			4
53	35			5
54	36			6
55	37			7
56	38			8
57	39			9
58	3A			:
59	3B			;
60	3C			<
61	3D			=

Numeric Values		ASCII Character Name	Comment	Keyboard Entry
Decimal	Hex			
62	3E			>
63	3F			?
64	40			@
65	41			A
66	42			B
67	43			C
68	44			D
69	45			E
70	46			F
71	47			G
72	48			H
73	49			I
74	4A			J
75	4B			K
76	4C			L
77	4D			M
78	4E			N
79	4F			O
80	50			P
81	51			Q
82	52			R
83	53			S
84	54			T
85	55			U
86	56			V
87	57			W
88	58			X
89	59			Y
90	5A			Z
91	5B			[
92	5C			\
93	5D]
94	5E			^
95	5F			_
96	60			`
97	61			a
98	62			b
99	63			c

Numeric Values		ASCII Character Name	Comment	Keyboard Entry
Decimal	Hex			
100	64			d
101	65			e
102	66			f
103	67			g
104	68			h
105	69			i
106	6A			j
107	6B			k
108	6C			l
109	6D			m
110	6E			n
111	6F			o
112	70			p
113	71			q
114	72			r
115	73			s
116	74			t
117	75			u
118	76			v
119	77			w
120	78			x
121	79			y
122	7A			z
123	7B			{
124	7C			
125	7D			}
126	7E		Tilde	~
127	7F		Delete	Del