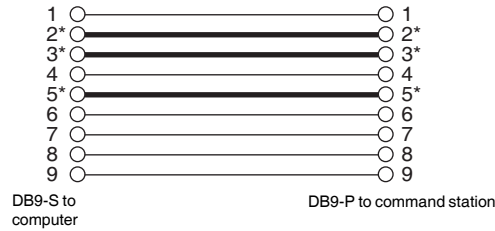


# RS232 SERIAL COMPUTER INTERFACE

## CABLE

You can use a Male 9 pin D connected to a Female 9 pin D with ribbon cable, or use a 9 pin D extension cable.



Note: The signals marked with an "\*" are the only ones that are necessary.

## USB TO SERIAL ADAPTER CABLE

It is possible to use a USB to Serial Adapter cable with your system. Most adapters are known to work.

## COMMUNICATIONS PARAMETERS

The factory default baud rate is 9600 baud. You can use the SET CMD STA menu to change the baud rate to 19200 or 38400. The rest of the communications parameters are 8 bits, No parity, and One stop bit. If you are using a terminal program on a PC, set it to echo characters. The characters sent to the Command Station are not echoed.

## ASCII COMMAND SET

The ASCII command set is no longer supported except for programming track related commands. These commands are no longer listed as they will be deleted in the next revision of the system software. Use the binary command set.

# BINARY COMMAND SET

The RS-232 port binary commands are designed to work in a computer friendly way.

Command format is: <command number> <data> <data> ...

Commands range from 0x80 to 0xBF

Commands and formats supported:

Commands 0xAD to 0xBF are not used and return '0'

Errors returned:

'0'= command not supported

'1'= loco address out of range

'2'= cab address out of range

'3'= data out of range

'4'= byte count out of range

'!'= command completed successfully

For a complete description of Binary Commands see: [www.ncecorporation.com/pdf/bincmds.pdf](http://www.ncecorporation.com/pdf/bincmds.pdf)

CMD FORMAT	DESCRIPTION	# OF BYTES RETURNED	POSSIBLE RESPONSES
0x80	NOP, dummy instruction	(1)	!
0x81 xx xx yy	assign loco xxxx to cab cc	(1)	!, 1, 2
0x82	read clock	(2)	<hours><minutes>
0x83	Clock stop	(1)	!
0x84	Clock start	(1)	!
0x85 xx xx	Set clock hr./min	(1)	!, 3
0x86 xx	Set clock 12/24	(1)	!, 3
0x87 xx	Set clock ratio	(1)	!, 3
0x88 xxxx	Dequeue packet by loco addr	(1)	!, 1, 2
0x89	Enable main trk, kill prog	(1)	!
0x8A yy	Return status of AIU yy	(4)	<current hi byte> <current lo byte> <change hi byte> <change lo byte>
0x8B	Kill main trk, enable prog	(1)	!
0x8C	dummy inst. returns"!" followed CR/LF	(3)	!0x0D, 0x0A
0x8D xxxx mm	Set speed mode of loco xxxx to mode mm, 1=14, 2=28, 3=128	(1)	!, 1, 3 <speed mode, 0 to 3>
0x8E aaaa nn <16 data bytes>	Write nn bytes, start at aaaa Must have 16 data bytes, pad them out to 16 if necessary	(1)	!, 4

CMD FORMAT	DESCRIPTION	# OF BYTES RETURNED	POSSIBLE RESPONSES
0x8F aaaa	Read 16 bytes, start at aaaa	(16)	16 bytes
0x90 cc xx...	Send 16 char message to Cab ccLCD line 3. xx = 16 ASCII char	(1)	!, 2
0x91 cc xx	Send 16 char message to cab cc LCD line 4. xx=16 ASCII	(1)	!, 2
0x92 cc xx	Send 8 char message to cab cc LCD line 2 right xx=8 char	(1)	!, 2
0x93 ss <3 byte packet>	Queue 3 byte packet to temp_Q send ss times	(1)	!
0x94 ss <4 byte packet>	Queue 4 byte packet to temp_Q send ss times	(1)	!
0x95 ss <5 byte packet>	Queue 5 byte packet to temp_Q send ss times	(1)	!
0x96 ss <6 byte packet>	Queue 6 byte packet to temp_Q send ss times	(1)	!
0x97 aaaa xx	Write 1 byte to aaaa	(1)	!
0x98 aaaa xx xx	Write 2 bytes to aaaa	(1)	!
0x99 aaaa<4 data bytes>	Write 4 bytes to aaaa	(1)	!
0x9A aaaa<8 data bytes>	Write 8 bytes to aaaa	(1)	!
0x9B yy	Return status of AIU yy (short form of command 0x8A)	(2)	<current hi byte> <current lo byte>
0x9C xx	Execute macro number xx	(1)	!, 0, 3
0x9D aaaa	Read 1 byte from aaaa	(1)	1 byte
0x9E	Enter programming track mode	(1)	!=success 3=short circuit
0x9F	Exit programming track mode	(1)	!=success
0xA0 aaaa xx	Program CV aa with data xx in paged mode	(1)	!=success 0=program track no
0xA1 aaaa	Read CV aaaa in paged mode Note: cv data followed by ! for OK. 0xFF followed by 3 for can't read CV	(2)	!, 0, 3
0xA2<4 data bytes>	Locomotive control command See Page 77	(1)	!, 1

CMD FORMAT	DESCRIPTION	# OF BYTES RETURNED	POSSIBLE RESPONSES
0xA3 <3 byte packet>	Queue 3 byte packet to TRK_Q (replaces any packet with same address if it exists)	(1)	!, 1
0xA4 <4 byte packet> 0xA4 xx xx...	Queue 4 byte packet to TRK_Q (replaces any packet with same address if it exists)	(1)	!, 1
0xA5 <5 byte packet> 0xA5 xx xx...	Queue 5 byte packet to TRK_Q (replaces any packet with same address if it exists)	(1)	!, 1
0xA6 rr dd	Program register rr with dd	(1)	!=success 0=program track no
0xA7 rr	Read register rr. Note: cv data followed by ! for OK. 0xFF followed by 3 for can't read CV	(2)	!, 3 0=program track no
0xA8 aaaa dd	Program CV aaaa with dd in direct mode.	(1)	!=success 0=program track no
0xA9 aaaa	Read CV aaaa in direct mode. Note: cv data followed by ! for OK. 0xFF followed by 3 for can't read CV	(2)	!, 3 0=program track no
0xAA	Return software revision number. Format: VV.MM.mmm	(3)	3 data bytes
0xAB	Perform soft reset of command station (like cycling power)	(0)	Returns nothing
0xAC	Perform hard reset of command station. Reset to factory defaults (Note: will change baud rate to 9600)	(0)	Returns nothing
0xAD <4 data bytes>	Accy/signal and macro commands See Page 77	(1)	!, 1
<p>NOTE: A single byte of 0 will be returned if not in programming mode for commands 0x9F, 0xA0, 0xA1 and 0xA6 through 0xA9.</p> <p>Errors returned:           '0'=Command not supported                               '1'=Loco/Accy/Signal address out of range                               '2'=Cab address or OP code out of range                               '3'=Data out of range                               '4'=Byte count out of range                               '!'=Command completed successfully</p> <p>Enter:           BIN_BUFF contains completed binary command.</p> <p>Exit:            Action performed based on command                   RS232FLAGS bits 0 and 1 cleared to 0                   BINSVPTR set to BIN_BUFF                   BIN_RXCOUNT and BIN_RXEXPECTED cleared to 0</p>			

## 0xA2 sends speed or function packets to a locomotive.

Command Format: 0xA2 <addr\_h> <addr\_l> <op\_1> <data\_1>

Addr\_h and Addr\_l are the loco address in DCC format.  
If a long address is in use, bits 6 and 7 of the high byte are set.

Example: Long address 3 = 0xc0 0x03  
Short address 3 = 0x00 0x03

op_1	data_1	Operation description
01	0-7f	Reverse 28 speed command
02	0-7f	Forward 28 speed command
03	0-7f	Reverse 128 speed command
04	0-7f	Forward 128 speed command
05	0	Estop reverse command
06	0	Estop forward command
07	0-1f	Function group 1 (same format as DCC packet for FG1)
08	0-0f	Function group 2 (same format as DCC packet for FG2)
09	0-0f	Function group 3 (same format as DCC packet for FG3)
0a	0-7f	Set reverse consist address for lead loco
0b	0-7f	Set forward consist address for lead loco
0c	0-7f	Set reverse consist address for rear loco
0d	0-7f	Set forward consist address for rear loco
0e	0-7f	Set reverse consist address for additional loco
0f	0-7f	Set forward consist address for additional loco
10	0	Del loco from consist
11	0	Kill consist
12	0-9	Set momentum
13	0-7f	No action, always returns success
14	0-7f	No action, always returns success
15	0-ff	Functions 13-20 control (bit 0=F13, bit 7=F20)
16	0-ff	Functions 21-28 control (bit 0=F21, bit 7=F28)
17	0-3f	Assign this loco to cab number in data_1
18-7f		reserved reserved

Returns: ! = success  
1 = bad loco address

## 0xAD sends accessory or signal packets

Command Format: 0xAD <addr\_h> <addr\_l> <op\_1> <data\_1>

The address range is 1-2047  
Addr\_h and Addr\_l are the accessory/signal address (NOT in DCC format).

Example: Accessory Address 513 = 0x02 0x01 (hi byte first)  
Accessory Address 6 = 0x00 0x06 (hi byte first)

NOTE: accy/signal address 0 is not a valid address  
address 2044 is the broadcast address

Op_1	Data_1	Operation description
01	0-255	NCE macro number 0-255
02	0-255	Duplicate of Op_1 command
03	0	Accessory Normal direction (ON)
04	0	Accessory Reverse direction (OFF)
05	0-1f	Signal Aspect 0-31
05-7f	reserved	reserved

Returns: ! = success  
1 = bad accy address