

NellyCOM Specification - V1.2

Nelevator System Control Commands

Revision History

Issue	Date	Modification
1.0	07 July 2016	Initial draft.
1.1	15 August 2016	Added status command.
1.2	27 November 2016	Status response referred to channels, but should be motors. Added note about channels and motors. Added locked status response for Nelevator software versions >= 1.4. Added stop motors command.

Serial Port Configuration

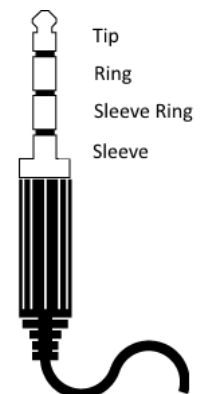
Baud Rate: 19,200
Parity: None
Data Bits: 8
Stop Bits: 1

Serial Port Connection

TSR 3.5mm 4 pole jack plug.

E.g. RS part [805-1652](#), Farnell part [2309467](#) or [Cliff Electronics](#) part FC68124.

Tip: +5V
Ring: TX, 5V logic levels (*RX on Nelevator*)
Sleeve Ring: RX, 5V logic levels (*TX on Nelevator*)
Sleeve: 0V



Data Frame Format

<SOH> <CMD> <DATA> n... <BCC> <EOT>

Command	Description
SOH	0x01 - Start of header / transmission.
CMD	Command to execute.
DATA	Data bytes (if any) related to the command.
BCC	Block Check Code of all previous bytes after (but not including) <SOH>, XORd together, but prior to any byte substitutions of the data (see below), i.e. to be performed on the raw data.
EOT	0x04 - End of transmission.
SUB	0x1A - Substitution - transparency byte.
SUB_XOR	0x20 - Value to XOR with data byte for substitution.

Byte Substitution

After the initial <SOH> byte, if any data bytes equate to **SOH**, **EOT** or **SUB**, then a SUB byte is to be sent first and then the data byte is sent after XORing with SUB_XOR.

This should also be done for the <BCC> byte.

To decode, discard the received SUB byte and then XOR the following byte with SUB_XOR to reconstitute the data byte.

Motors and Channels

Please note: Because of the way the system PCB is mounted on the Nelevator the motor and channel labels are swapped, i.e.:

- **Motor 1** = Channel 2
- **Motor 2** = Channel 1

In the following specification, please pay attention as to whether the specification is referring to a *motor* or a *channel* as they differ.

Commands

X – Stop Both Motors

<X>

Example Data String to Stop Both Motors

Command Format	Hexadecimal Data String
<SOH> + X + <BCC> + <EOT>	0x01 0x58 0x58 0x04

M - Move

<M> <Channel> <Mode> <Variable>

Command Options	Options / Description	Data Format
Channel	1, 2 = Channel selection	ASCII
Mode	T = Track select	ASCII
Variable	0 – 9 = Track number	8 bit byte

Example Data Strings to Move to Selected Positions

Channel 1:

Track No.	Command Format	Hexadecimal Data String
1	<SOH> + M1T + Chr(0) + <BCC> + <EOT>	0x01 0x4D 0x31 0x54 0x00 0x28 0x04
2	<SOH> + M1T + Chr(1) + <BCC> + <EOT>	0x01 0x4D 0x31 0x54 0x1A 0x21 0x29 0x04
3	<SOH> + M1T + Chr(2) + <BCC> + <EOT>	0x01 0x4D 0x31 0x54 0x02 0x2A 0x04
4	<SOH> + M1T + Chr(3) + <BCC> + <EOT>	0x01 0x4D 0x31 0x54 0x03 0x2B 0x04
5	<SOH> + M1T + Chr(4) + <BCC> + <EOT>	0x01 0x4D 0x31 0x54 0x1A 0x24 0x2C 0x04
6	<SOH> + M1T + Chr(5) + <BCC> + <EOT>	0x01 0x4D 0x31 0x54 0x05 0x2D 0x04
7	<SOH> + M1T + Chr(6) + <BCC> + <EOT>	0x01 0x4D 0x31 0x54 0x06 0x2E 0x04
8	<SOH> + M1T + Chr(7) + <BCC> + <EOT>	0x01 0x4D 0x31 0x54 0x07 0x2F 0x04
9	<SOH> + M1T + Chr(8) + <BCC> + <EOT>	0x01 0x4D 0x31 0x54 0x08 0x20 0x04
10	<SOH> + M1T + Chr(9) + <BCC> + <EOT>	0x01 0x4D 0x31 0x54 0x09 0x21 0x04

Channel 2:

Track No.	Command Format	Hexadecimal Data String
1	<SOH> + M2T + Chr(0) + <BCC> + <EOT>	0x01 0x4D 0x32 0x54 0x00 0x2B 0x04
2	<SOH> + M2T + Chr(1) + <BCC> + <EOT>	0x01 0x4D 0x32 0x54 0x1A 0x21 0x2A 0x04
3	<SOH> + M2T + Chr(2) + <BCC> + <EOT>	0x01 0x4D 0x32 0x54 0x02 0x29 0x04
4	<SOH> + M2T + Chr(3) + <BCC> + <EOT>	0x01 0x4D 0x32 0x54 0x03 0x28 0x04
5	<SOH> + M2T + Chr(4) + <BCC> + <EOT>	0x01 0x4D 0x32 0x54 0x1A 0x24 0x2F 0x04
6	<SOH> + M2T + Chr(5) + <BCC> + <EOT>	0x01 0x4D 0x32 0x54 0x05 0x2E 0x04
7	<SOH> + M2T + Chr(6) + <BCC> + <EOT>	0x01 0x4D 0x32 0x54 0x06 0x2D 0x04
8	<SOH> + M2T + Chr(7) + <BCC> + <EOT>	0x01 0x4D 0x32 0x54 0x07 0x2C 0x04
9	<SOH> + M2T + Chr(8) + <BCC> + <EOT>	0x01 0x4D 0x32 0x54 0x08 0x23 0x04
10	<SOH> + M2T + Chr(9) + <BCC> + <EOT>	0x01 0x4D 0x32 0x54 0x09 0x22 0x04

S – Status Data

Status Data Request

<S>

Note: To avoid overloading the Nelevator with continuous status requests it is advised that status requests are sent no more frequently than every half second (500 ms intervals). This rate should be more than sufficient for any feedback / control display requirements.

Example Data String to Request a Status Update

Command Format	Hexadecimal Data String
<SOH> + S + <BCC> + <EOT>	0x01 0x53 0x53 0x04

Status Data Response

<S> <M1 Status> <M1 Track> <M1 Target> <M2 Status> <M2 Track> <M2 Target>

- M1 = Motor 1 / Channel 2
- M2 = Motor 2 / Channel 1

Response Options	Options / Description	Data Format
Status	S = Stopping going up	ASCII
	s = Stopping going down	ASCII
	x = Stopped - OK	ASCII
	L = Locked motor, prevented from moving	ASCII
	Y = Stopped - Overshoot going up	ASCII
	y = Stopped - Overshoot going down	ASCII
	u = Moving up	ASCII
	d = Moving down	ASCII
	B = Braking going up	ASCII
	b = Braking going down	ASCII
	A = Accelerating going up	ASCII
	a = Accelerating going down	ASCII
	O = Over-current going up	ASCII
	o = Over-current going down	ASCII
	c = Calibration mode	ASCII
	l = Start speed incremented going up	ASCII
i = Start speed incremented going down	ASCII	
Track	0 – 9 = Current track number	8 bit byte
Target	0 – 9 = Target track number	8 bit byte