New Scale Technologies

SMALL, PRECISE, SMART ... IN MOTION

New Scale Pathway Software Guide

Revision N
August, 2021

Contact information

New Scale Technologies, Inc.
121 Victor Heights Parkway
Victor NY, 14564

Phone: (585)-924-4450
Fax: (585)-924-4468

Sales: sales@newscaletech.com
Service: nstservice@newscaletech.com
Web: www.newscaletech.com

Pathway: See **Report Software or Hardware Problems** tab on the **About** dialog.

Information in this document is current through August 2021

Contents

1: Getting Started....7
About this guide....8
Safety precautions....8
How the motor is controlled....8
Open- versus closed-loop control modes....9
Setting up the software....10
Installing the software....10
Starting the software....13
Selecting the controller port(s)....15
Establishing the PC to controller connection....17
Disconnecting a motor from the controller....19
Connecting to one or more M3-PoE-3:1 Units....19
Network Congestion and Communication Timeouts....22

2: Using a Motor/Sensor via the New Scale Pathway Software....23
Moving the motor with the Pathway manual controls (open-loop mode)....24
Using open-loop controls....26
Moving the motor with ASCII commands....27
Viewing the NSE-5310 sensor details....28
I^2C Communication Details....30
Reset Position Accumulator....30
Resolution....31
Record NSE-5310 Data....31
Temp Offset....32
Pole Position & Field Strength Monitor....32
Moving the motor with scripts....34
Creating and editing a script....34
Saving a script....36
Executing a script....36
Opening a script....37
Confirming Motor Operating Frequency and Q....41
Frequency Response of Direct Drive Motors....41
Frequency Response of Tank Circuit Driven Motors....41
Using the Pathway to Generate a Frequency Response Curve....42
Conditions that Affect Sweep Results....44

3: Setting Motor (Axis) Attributes....47
Viewing and editing motor EEPROM parameters....48
Viewing and editing open-loop parameters....49
Viewing and editing closed-loop parameters....50

Enabling travel limits....53
Enabling stall detection....54
Viewing and editing motor drive settings....55

4: Using Additional Options....59
Selecting Visible Tabs....59
Displaying motor (axis) position....60
Generating a Motion Performance Log Plot....60
Displaying the Performance Log Viewer....61
Example Performance Log....63
Other Features of the Performance Viewer....64
Updating Firmware....66
Difference if the Selected USB device is an M3 USB Interface....68
Reporting Problems with Pathway S/W and/or Controller(s)....69

Appendix A: ASCII Command Reference....73
Overview....73
Command conventions....73
Using the command integrity prefix (optional)....73
Enabling and disabling the command prefix....74
Command prefix syntax....75
Example....75
01: Establishing computer control....75
02: Releasing computer control....76
03: Halting the motor....76
04: Running the motor....76
Examples....77
05: Moving the motor in timed steps....77
Example....77
06: Moving the motor using closed-loop steps....78
Example....78
07: Clearing the encoder count....78
08: Moving to target....79
Example....79
09: Setting the open-loop speed....79
Example....80
10: Viewing closed-loop status and position....80
Motor status values....80
11: Writing a hex byte to the motor EEPROM....82
12: Reading a hex byte from the motor EEPROM....82
13: Writing ASCII to the motor EEPROM....83
14: Reading ASCII from the motor EEPROM....83
15: Updating firmware....84
18: Selecting or toggling the multiplexer controller axis (motor)....84
19: Reading motor flags from the controller....85
20: Viewing and selecting the open- or closed-loop drive mode....85

23: Illegal command format 85
24: Illegal command 86
40: Setting the closed-loop speed 86
Example 86
41: Setting position error thresholds and stall detection 87
Example 87
42: Detecting the zero-reference mark 88
43: Viewing and setting closed-loop PID coefficients in motor EEPROM 88
Example 88
44: Viewing and setting closed-loop encoder details 89
Example 89
45: Viewing and setting the minimum closed-loop drive amplitude 90
Example 90
46: Viewing and setting forward and reverse travel limits 90
Example 91
47: Viewing, activating and deactivating forward and reverse travel limits 91
Example 91
50: Viewing and activating motor burst or amplitude mode 91
51: Viewing and setting the forward and reverse speed ratio 92
Integer fraction examples 92
52: Viewing time interval units 93
54: Viewing and setting the port baud rate 93
Examples 94
87: Viewing and Starting Motor Freq. Calibration 94
Examples 95
A9, AA: Viewing and Starting a Performance Log 96

Tables

Figure 1-1. The Main window, Manual Control tab 14
Figure 1-2. The Setup Connection window..... 15
Figure 2-1. The Main window, Motor Control tab 24
Figure 2-2. The NSE-5310 Tab..... 28
Figure 2-3. The Script Control tab 34
Figure 3-1. The Main window, Axis Attributes tab..... 48
Figure 3-2. Voltage versus Time effect on motor parameters 55
Figure 4-1. The display Options tab..... 59
Figure 4-1. The Display Only tab..... 60

Figures

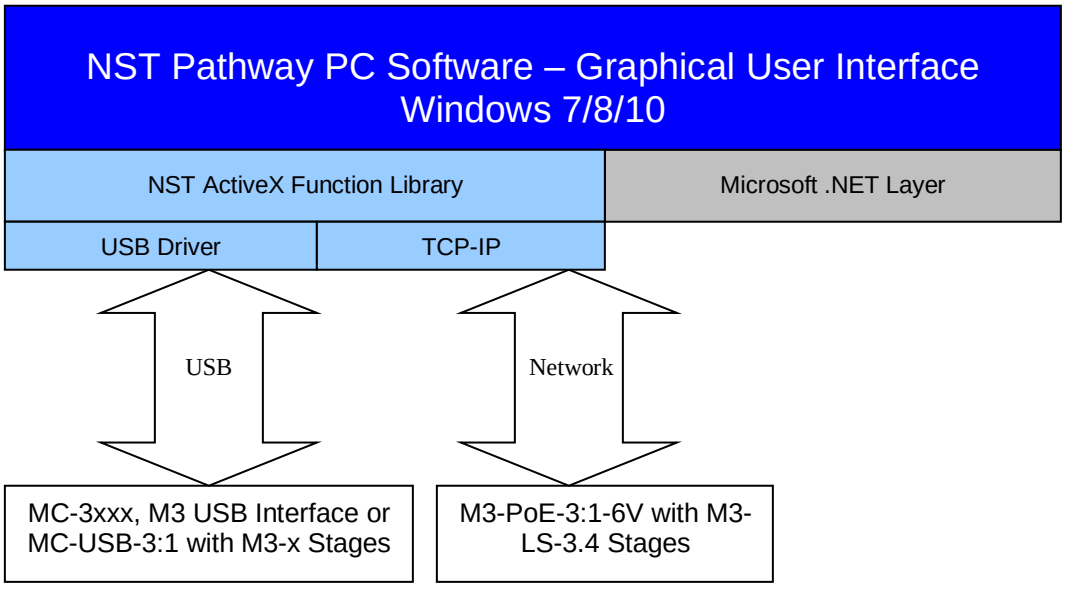
Figure 1-1. The Main window, Manual Control tab 14
Figure 1-2. The Setup Connection window..... 15
Figure 2-1. The Main window, Motor Control tab 24
Figure 2-2. The NSE-5310 Tab..... 28
Figure 2-3. The Script Control tab 34
Figure 3-1. The Main window, Axis Attributes tab..... 48
Figure 3-2. Voltage versus Time effect on motor parameters 55
Figure 4-1. The display Options tab..... 59
Figure 4-1. The Display Only tab..... 60

1: Getting Started

The New Scale Pathway software is designed to make it easy to evaluate NST's line of ultrasonic piezo motor modules as well as the NSE-5310 magnetic sensor.

It provides a graphical user interface (GUI) that lists the available motors and allows you to individually select and operate each motor. It also provides a scripting tool so that operations for one or more motors may be automated.

The GUI communicates with the motor controllers via an ActiveX function library. This library can also be used directly from your own application.



We suggest that you take a few moments to review this chapter before you begin using the kit.

About this guide

This guide is organized as follows:

- **Chapter 1, Introduction**: Describes the various operating modes, environmental and safety requirements and how to install the New Scale Pathway software and prepare for motor and/or sensor evaluation.
- **Chapter 2, Evaluating the Motor**: Explains how to evaluate the motor using manual controls, manual commands, scripts and the optional handset.
- **Chapter 3, Setting Axis Attributes**: Defines how to set motor parameters for motor use in your OEM product.
- **Chapter 4: Using Additional Options**: Outlines various procedures such as displaying the motor position and updating firmware.
- **Appendices**: Provides a list of ASCII commands.

Safety precautions

- Do not apply side loads to the motor shaft. Always apply loads parallel to the motor shaft.
- Do not touch the motor while it is running.
- The motor contains high voltage. Do not touch connections and components when power is on.
- Do not allow the motor to overrun its travel limits when moving backwards. The screw may disengage from the motor shaft.

How the motor is controlled

The New Scale Controllers, such as the MC-3300 Dual Axis Controller, provide the electrical signals necessary to drive one or two motors. They also serve as a reference design when you integrate motor electronics into your OEM system boards or ASICs. New Scale also offers single-chip ASIC drivers and provides integrated motion modules, as well as OEM design assistance.

You move the motor using either of the following items:

- An **optional handset** that plugs directly into the controller. With the handset, you can run the motor back and forth without a hard limit, but you cannot control precision movements or evaluate motor load.

If you want to use your own software to evaluate the motor, you can integrate the ActiveX function library to handle communication.

- The **New Scale Pathway Software** supplied with this kit, is installed on your computer and communicates to the controller via a USB, or in some cases an ethernet connection. This software uses an ActiveX function library to build & issue commands to the controller.

  The software enables you to control precision motor movements using three methods as described in Chapter 2:

  – **Manual control**: By clicking buttons on the main screen.

  – **ASCII commands**: By entering commands in a command line (see Appendix A). These are the native controller commands which are generated and issued by the NST ActiveX Function library.

  – **Scripts**: By creating command sequences, which can be repeated and saved for use in an integrated system.

Open- versus closed-loop control modes

When you move the motor, you will commonly see the terms **open loop** and **closed loop**. It is important to understand how these terms are used as you assess the motor.

- **Open-loop mode.** In open-loop mode, you can calibrate the motor's average distance moved in response to a number of drive bursts.

  While the SQUIGGLE motor can move in very fine increments, the distance moved in response to a single drive signal burst depends on the load and the motor's internal friction. Any variation in these parameters along the length of travel will affect the motor speed. If you need to know the precise speed or position of the motor, closed-loop operation is recommended.

- **Closed-loop mode.** In closed-loop mode, a position sensor is used to detect the actual position of the motor shaft and feed the information back to the controller. The controller then compares the actual position to the desired position, and moves the motor to

correct any error. This allows the motor to reach a precisely-controlled position.

Similarly, controlled velocity is achieved by adjusting the driver gain to minimize the difference between the required position and the actual position at regular time intervals. Position sensors can be of four types:

- **NSE-5310 magnetic sensor.** Measures position along a magnet that has 2 mm pole pairs.
- **AS5510 magnetic sensor.** Provides up to 1024 divisions between the maximum North & South pole field strengths.
- **Digital (incremental) encoders.** The position resolution is determined by the resolution of the encoder.
- **Analog position sensors.** The position resolution is determined by the resolution of the A/D converter, the resolution of the position sensor and the resolution of the motor.

Both open- and closed-loop commands are entered using the software.

Setting up the software

Follow the procedures in this section in the order shown to prepare for motor evaluation using the software. Then refer to Chapter 2 to begin moving the motor.

Note: You should install the motor as described in the product manuals before you set up the software.

Installing the software

Complete this section if:

Your computer should have at least 200 MB of disk space available. You will need 2.2 GB total if you do not already have Microsoft .NET V4.0 installed.

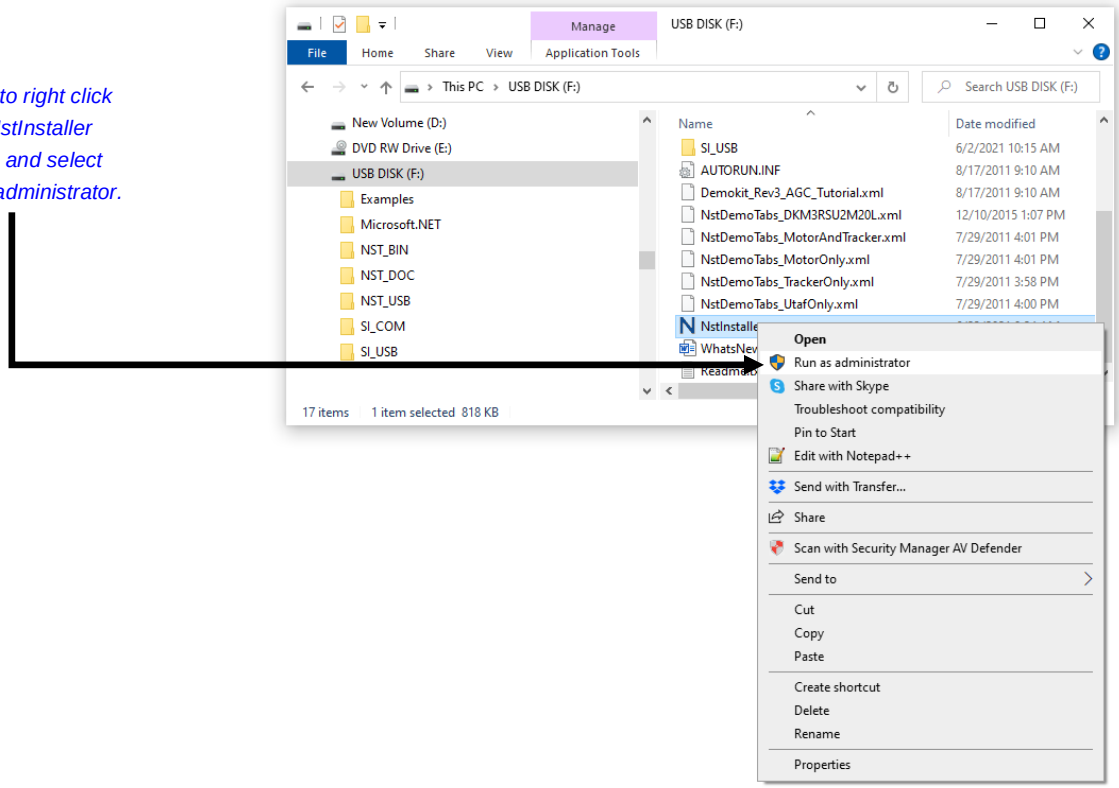
- You are using the software to evaluate the motor or encoder *OR*
- You are using the handset to evaluate the motor but want monitor status and/or perform frequency calibrations. (If this isn't the case, skip ahead to Chapter 2.)

To install the software:

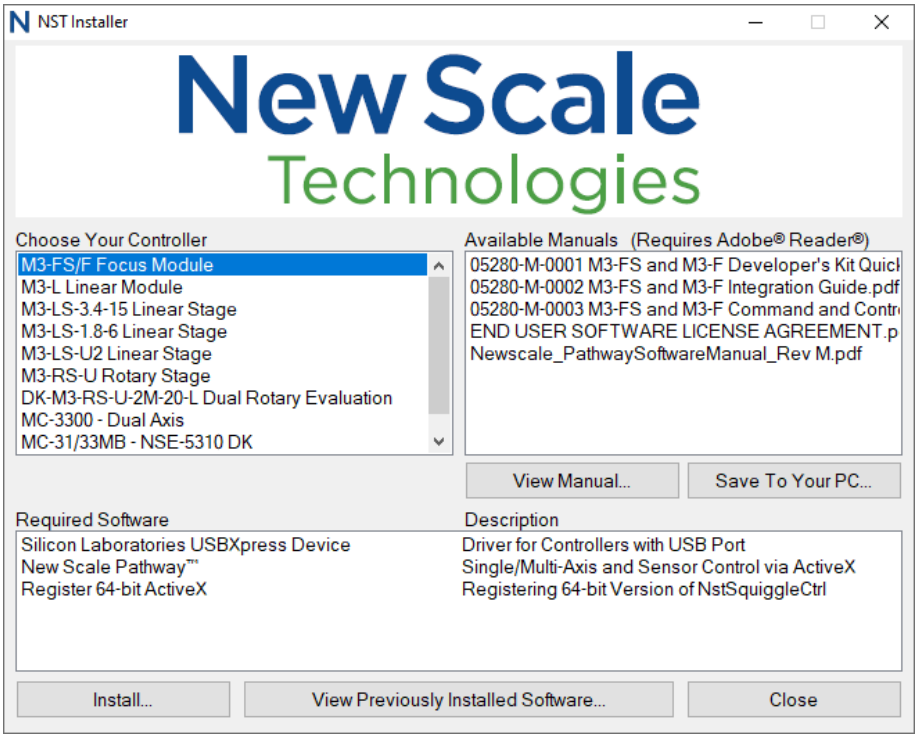
1. If necessary, close any previously-installed software from New Scale.
2. Insert the New Scale Pathway Software Flash Drive into a USB port on your PC. You will have to display the contents of the USB

flash drive, select NstInstaller.exe and start it by right clicking on it and selecting **Run as administrator**.

Be sure to right click on the NstInstaller program and select Run as administrator.



The NstInstaller user interface will be displayed as shown below.



3. Verify that the controller you purchased (e.g. **M3-LS-3.4-15 Linear Stage)** is selected in the *Choose Your Controller* section. *If you don't see the controller you selected, that's OK, the same software is installed regardless of the selection.* The point of the selection is to display the associated documentation on the right.

   Note: You can view and save product manuals using the *Available Manuals* section of this window. After installation, you can also access manuals from the flash drive (in the NST_DOC directory).

4. Click the **Install…** button located at the bottom of the window.
5. The software will detect the software needed on your computer. You may be notified that various programs will be installed and/or uninstalled. Proceed through any prompts.
6. Prior to loading Pathway, the installer will check to see if you have Microsoft .NET Framework 4.0 or later installed. If not, you will be asked it install it before continuing. The link below can be used

https://www.microsoft.com/en-us/download/details.aspx?id=17851

8. Continue responding to prompts as installation proceeds until you see the *Setup Complete* window.

   When you connect the controller's USB interface to the computer later in this process, Windows may attempt to reinstall the USB driver. Allow Windows to locate it automatically.

9. Complete installation as prompted. After installation is complete, an icon for launching the New Scale Pathway software will appear on your desktop.

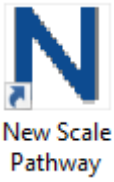
10. You may remove the flash drive if desired (follow the Window's protocol for dismounting a flash drive).

Note: To uninstall the software, use **Add or Remove Programs** on the Windows Control Panel.

Starting the software

To start the software:

- Double-click the **New Scale Pathway** icon on the desktop. The icon looks like this:



The Main window may open to the *Motor Control* tab.

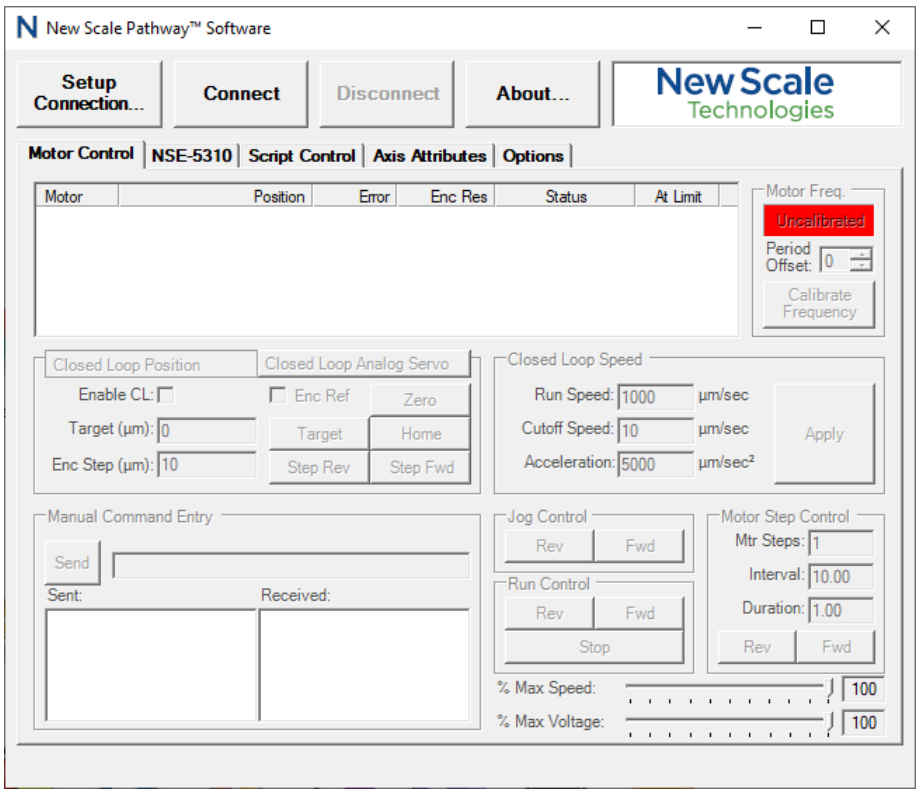


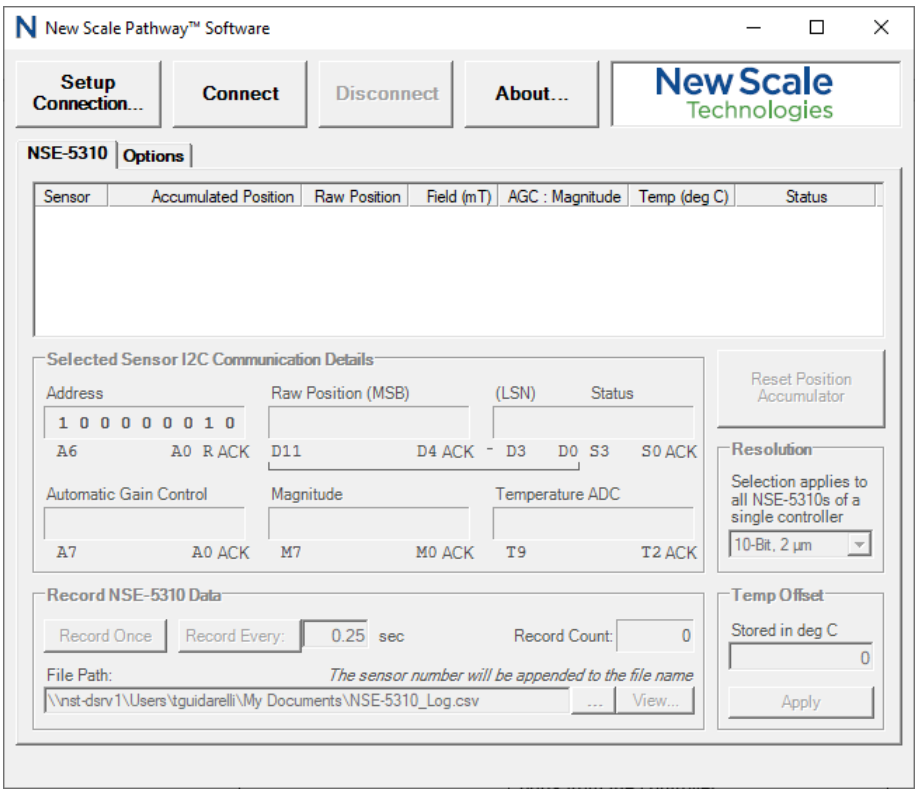
Figure 1-1. *The Main window, Manual Control tab*

Note: After you have used the **Connect** button to connect to a motor as described next, the axes you are connected to will appear in the above window.

In this guide, "axis" is the same as "motor."

The software can be used to control multiple motors, often in a two-axis (x-y) or three-axis (x-y-z) configuration.

However, if you purchased the NSE-5310 evaluation kit (without a motor), the main window may appear as follows.



Selecting the controller port(s)

Before the controller can communicate with a motor, you must identify the port characteristics used in the connection. You use the *Setup Connection* window to connect a port to the controller.

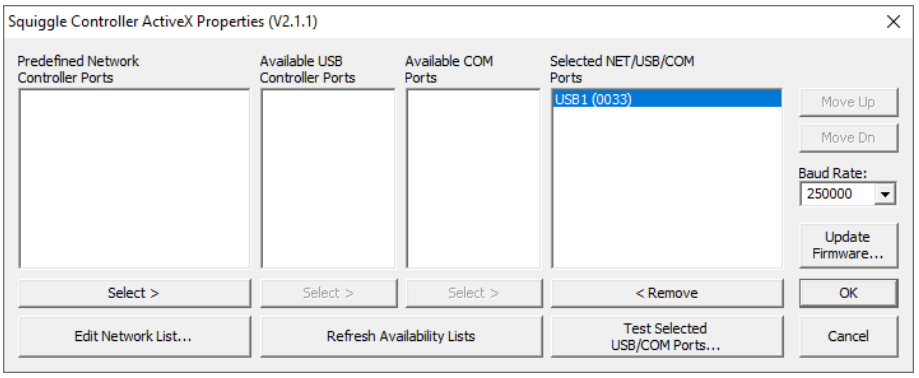


Figure 1-2. *The Setup Connection window*

Table 1-1. *Setup Connection window components*

Item	Description
Predefined Network Controller Ports	Even if M3-PoE-3:1 multiplexer units are connected to the network, this list will be initially blank. Click the **Edit Network List…** button to access a tool to locate the units (discover their IP assignments). Any units that you select for this list will be remembered.
Edit Network List…	Displays a dialog that you can use to discover what M3-PoE-3:1 units are available on the Network.
Selected USB/COM Ports	Lists ports that are selected for use with the controller. For selected USB ports, the controller serial number is shown in parentheses. For selected COM ports, the baud rate is shown in parentheses.
Select >	Connects the selected USB and COM ports to the controller.
< Remove	Disconnects the selected USB and COM ports from the controller.
Move Up, Move Dn	Changes the order of the axis (motor) connections listed on the Main window (available only if multiple ports are selected).
Baud Rate	Identifies the baud rate of the selected USB/COM port. (The MC-1000 controller uses a COM port, but other New Scale controllers use a USB port.)
Update Firmware	Updates the controller firmware. See Chapter 4.
Refresh Availability Lists	Refreshes the list of available USB and COM ports.
Test Selected USB/COM Ports	Attempts to connect to the selected ports.
OK	Saves any port selection changes and closes the window.
Cancel	Removes any port selection changes and closes the window.

Port selection:

1. On the Main window (Figure 1-1), click **Setup Connection**. The *Setup Connection* window opens (Figure 1-2).

Note: If the **Setup Connection** button is dimmed, click **Disconnect** in the Main window.

You can also double-click a port to select and move it.

2. Select the port you want to connect to the controller in the *Available USB Based Controllers* section.
3. Click **Select >**. The port name will move to the *Selected USB/COM Ports* section.

If multiple ports are selected, you can change the motor (axis) display order in the Main *window by clicking* ***Move Up*** *and* ***Move Dn****.*

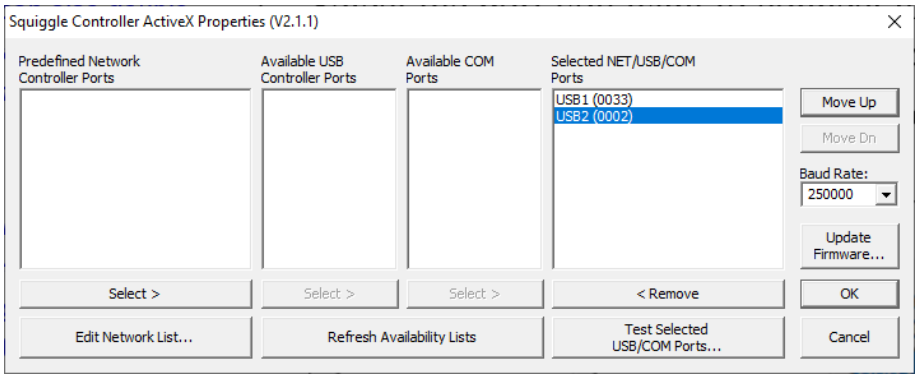


Figure 1-3. *The Setup Connection with Multiple Ports Selected*

4. (Optional) To verify that the controller can communicate with the selected port(s), select the port(s) and click **Test Selected USB/COM Ports**. A dialog box opens, indicating if the port was found or if an error occurred. Click **OK** to close the dialog box.
5. Click **Close** to close the *Setup Connection* window. The Main window returns.

Note: The list of selected ports is saved in the NstClosedLoopDemo2.xml file, located in your **My Documents** directory.

Establishing the PC to controller connection

After you make port selections, you can connect to the selected controller(s).

To establish the connection:

1. In the Main window Figure 1-1), click **Connect**. The area beneath the tabs shows the motors (axes) currently connected.

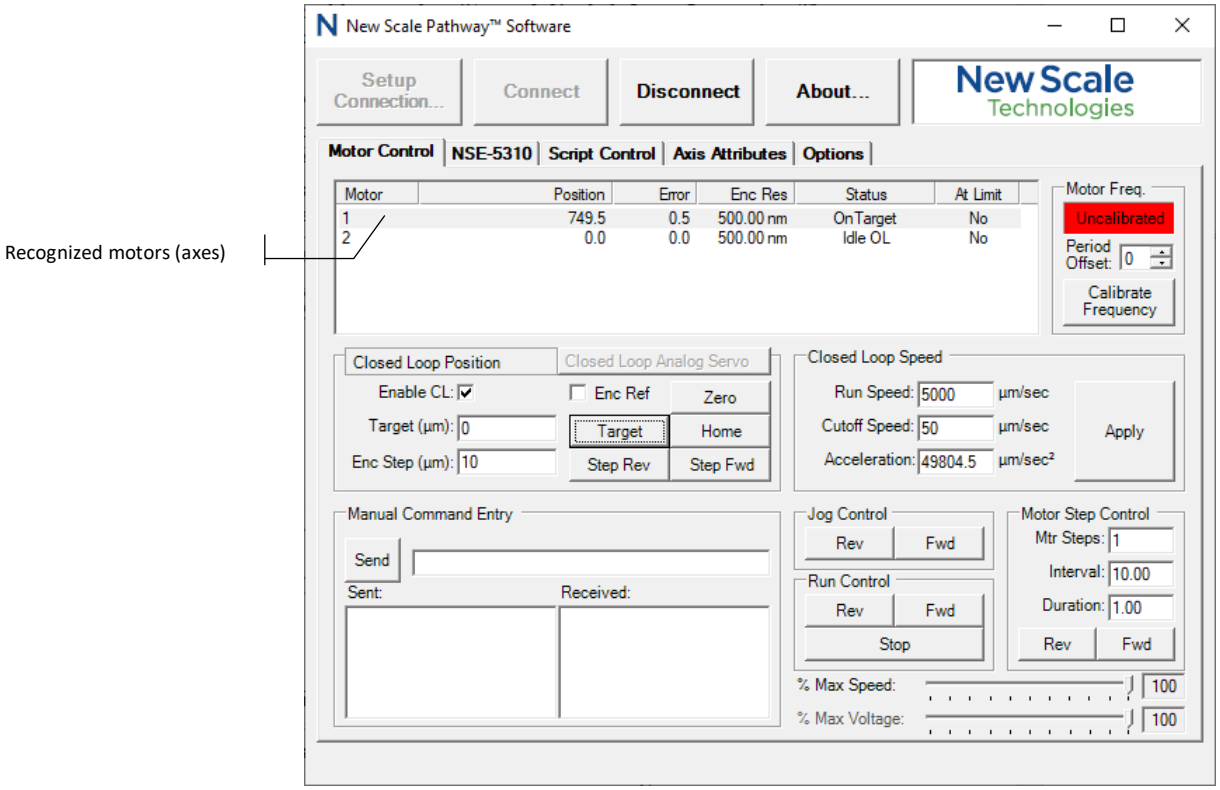
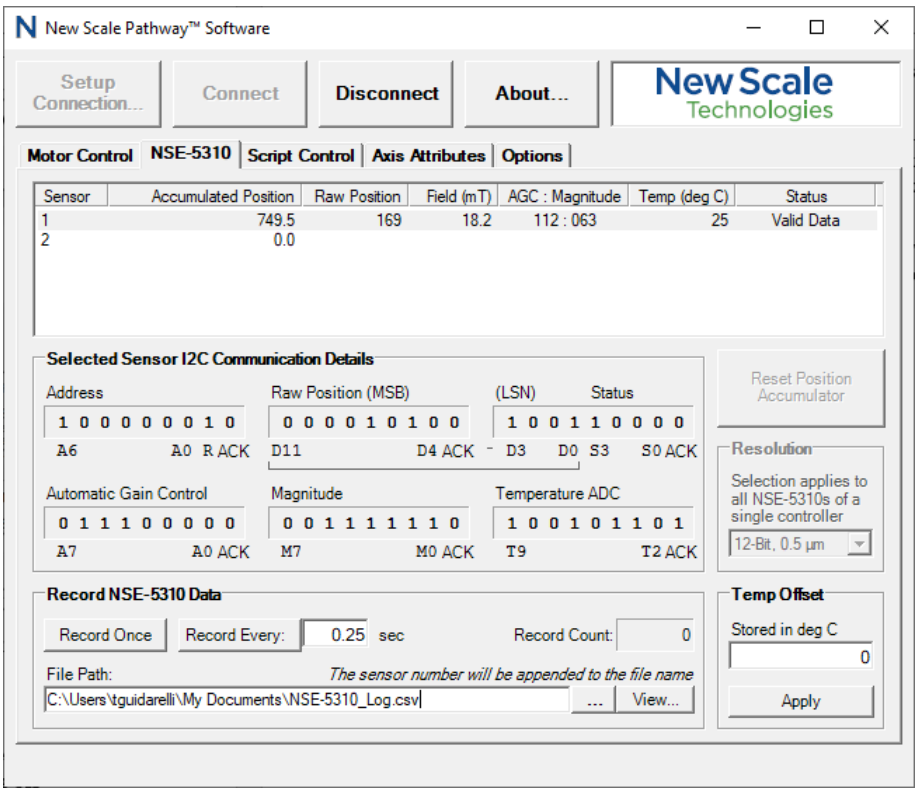


Figure 1-4. The Main window, Motor Control tab

2. Select the desired motor (also known as Axis).
3. Begin using the motor as described in Chapter 2.

Here again, if you are using the NSE-5310 evaluation kit, the screen will have the following appearance after clicking **Connect**.



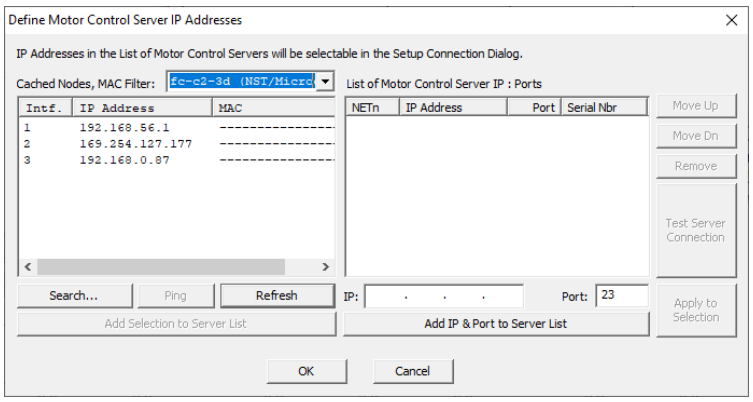
Disconnecting a motor from the controller

When you are done using the motor, be sure to click **Disconnect** or close the program before unplugging the USB cable or turning off power.

You must also click **Disconnect** before setting up new port connections.

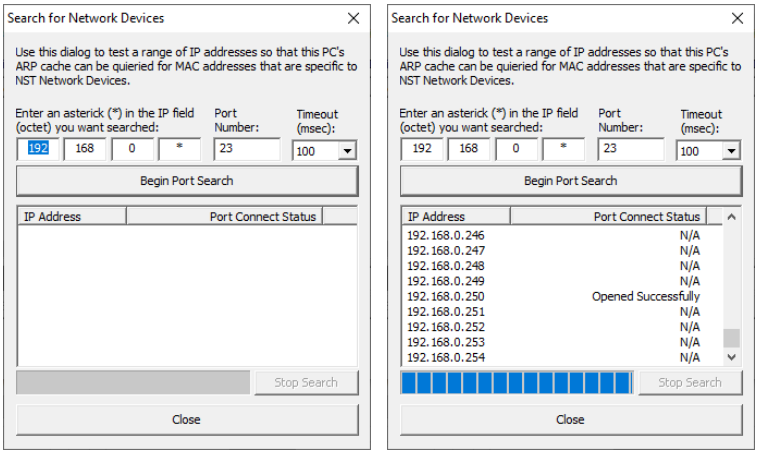
Connecting to one or more M3-PoE-3:1 Units

The IP address of each M3-PoE-3:1 unit (a device similar in purpose to the M3-USB-3:1 except it connects to the network via an IEEE-802at compliant switch instead of to a USB port) is assigned by a local DHCP server. To find the IP addresses, click on the **Edit Network List…** button. The following dialog will appear.

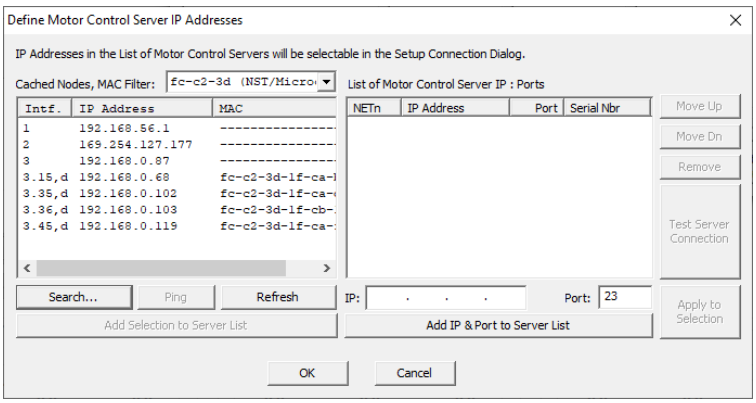


The list box on the left enumerates all the network nodes that your PC knows about (from the PC's ARP cache). By default, this list is filtered according to the common MAC address prefix of the M3-PoE-3:1units so it may be initially empty if the PC hasn't detected any M3-PoE-3:1 units you may have already connected to the network.

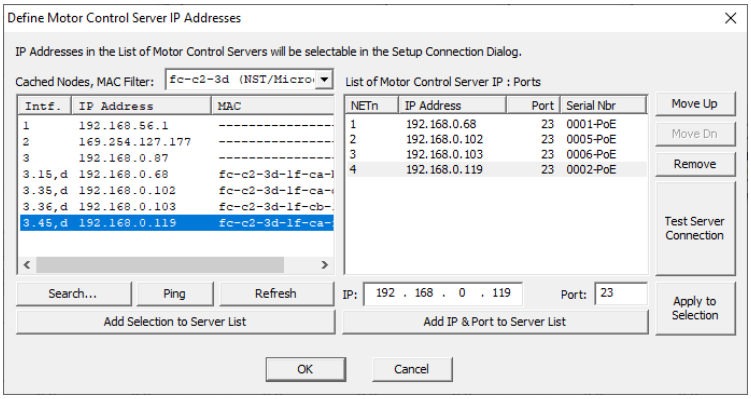
To initiate a search, click the **Search…** button. The following dialog will appear…



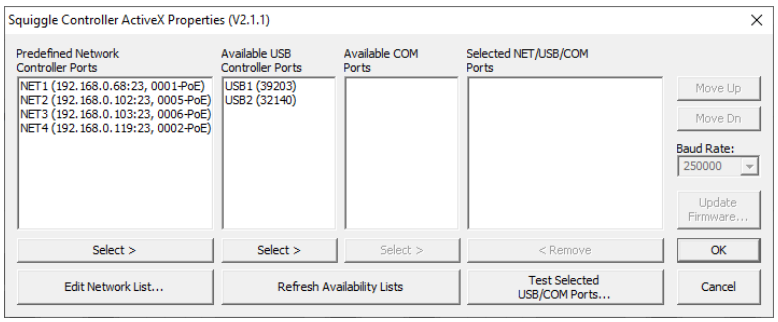
With this dialog you can place a wildcard ('*') in the IP address octet you want searched (only one of the four IP address octets can have a wild card). The M3-PoE-3:1 units will all be listening to port 23. Click **Begin Port Search** to start the process. The s/w will attempt to open port 23 at each of the available 254 IP addresses and list the result of each attempt. This will take ~26 seconds (given a 100 msec timeout/port). When it's finished click **Close**.



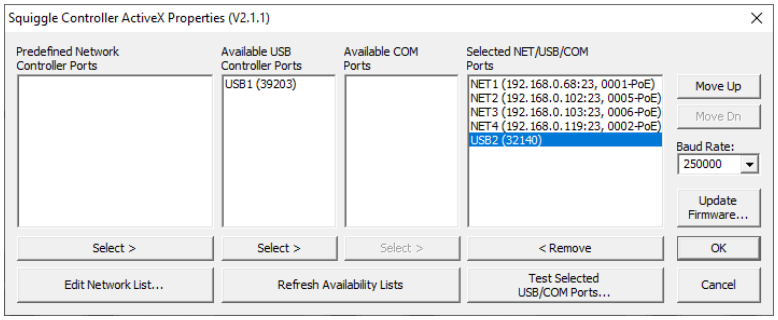
If any M3-PoE-3:1 units were detected, they will be listed on the left. You can add each node you want to use to the list on the right by either double clicking on the item in the list or by selecting it and clicking the **Add Selection to the Server List** button.



You may change the order each unit appears in the list by selecting the unit you want to reposition and click the **Move Up** or **Move Dn** buttons. The serial number that appears in the right most column is that which was assigned at the factory. Each M3-PoE-3:1 has a sticker with that same serial number as well as its MAC address. When you click **OK**, the contents of the list on the right will be available for selection.



If you also have M3-USB-3:1 units, both type may be selected at once.



Finally, to store the port selections, click **OK**.

Note: If a DHCP server is not available in the network where the M3-PoE-3:1 units will be connected, contact NST for a version of the M3-PoE-3:1 that can support a static IP address.

Network Congestion and Communication Timeouts

The Pathway s/w periodically queries each stage for its current position. It's possible, due to network traffic, that a position query of a stage that's connected to an M3-PoE-3:1 module may be delayed. This is unavoidable when the network is shared with others. The operator may occasionally see a message at the base of the main window that there was a timeout with one of the network ports or a dialog may be displayed. The software should recover on its own (apart from the operator dismissing a dialog). If it does not recover, you should check to make sure it's still being powered by the 802.3at (PoE+) switch (i.e., the LEDs on the M3-PoE-3:1 RJ45 jack should be on though one will blink with network traffic).

2: Using a Motor/Sensor via the New Scale Pathway Software

This chapter describes how to use the SQUIGGLE motor after you have:

- Installed the motor as described in the *Quick Guide*
- Installed the Pathway software as described in Chapter 1
- Selected controller ports and connected to the controller described in Chapter 1

Note: In general, always click Disconnect using the New Scale Pathway software *before* you disconnect any cables.

Moving the motor with the Pathway manual controls (open-loop mode)

To move the motor with manual controls, you use the *Motor Control* tab of the Main window.

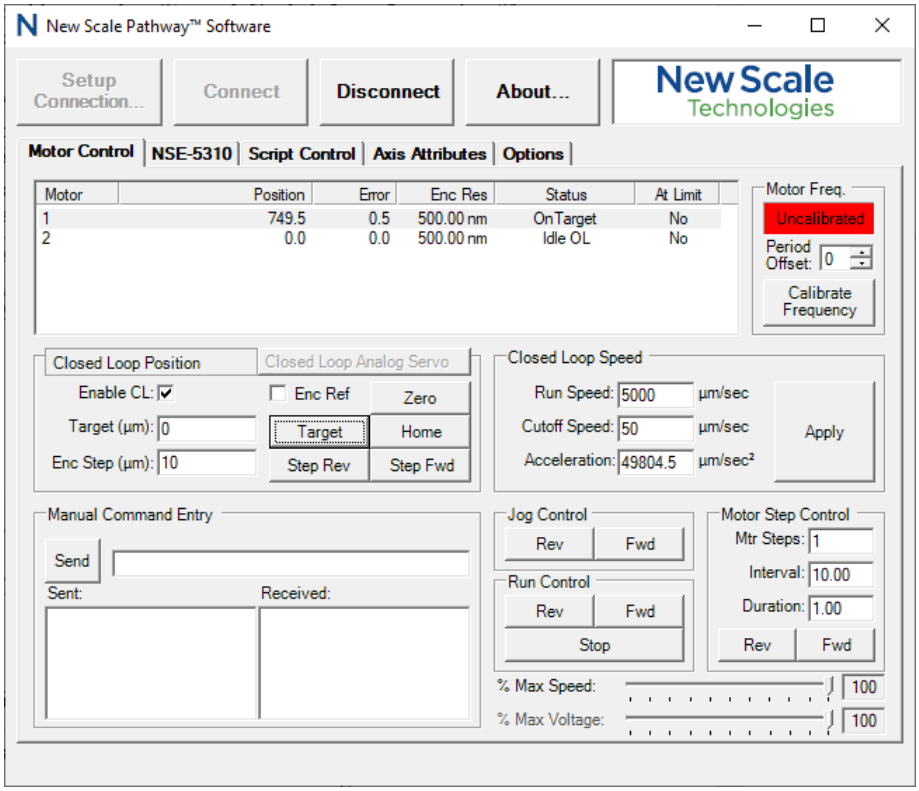


Figure 2-1. *The Main window, Motor Control tab*

Table 2-1. *Motor Control tab components*

Item	Description
Motor list box	Used to select the motor that will receive movement commands. This area also displays information about motor movement and position, including: • Position (µm): Current encoder position for each closed-loop axis (motor), updated every 0.25 seconds. • Err (µm): The distance between where the motor is supposed to be and where it is actually located. • Enc Res (nm): The encoder resolution.

Item	Description
	• Status: Motor status (running or idle). If running, motor direction is indicated. • At Limit: Indicates which travel limit, if any, has been reached (FwdLimit, RevLimit or No).
Motor Freq. area	Visible when the selected axis uses an NSD-2101 motor driver. The colored label indicates the freq. calibration status: Uncalibrated if a motor calibration has not been performed, Calibrated otherwise. To perform a calibration, click the **Calibrate** button below the label. The **Period Offset** is typically 0 unless using a UTAF motor. A calibrated motor provides the maximum speed & push force.
Multiplexer area (typically not shown).	Enabled when motors are Multiplexed (i.e. share the same drive electronics) such as with the SQ-2306 controller. This area is used to select which motor will be driven.
Closed Loop Position / Closed Loop Analog Servo section	Identifies position parameters of the selected motor (axis) in closed-loop mode. Used only if position sensor available and configured. The MC-3300 controller may alternatively be controlled with an analog servo signal from which the target position is derived. The mode (Position vs. Servo) is determined by which of the two buttons are depressed.
Closed Loop Speed section	Identifies speed parameters of the selected motor (axis) in closed-loop mode. Used only if position sensor available and configured. This section is disabled if in Servo mode (see above).
Manual Command Entry section	Allows you to send the native ASCII commands directly to the controller. See Appendix A for details.

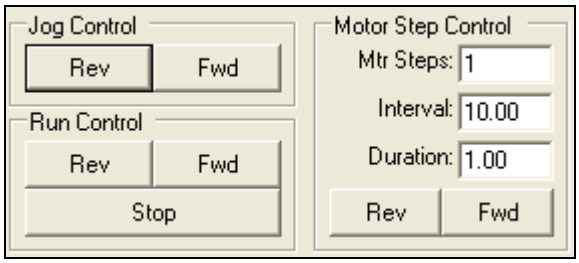
Jog Control section	Starts free-run motor motion in open-loop mode.
Run Control section	Starts free-run motor motion in open-loop mode
Motor Step Control section	Moves the motor using drive signal bursts in open-loop mode
% Max Speed	In open-loop mode, defines either the motor drive signal burst duty cycle or signal amplitude (or PWM) based on the mode selection on the **Axis Attributes** tab.
% Max Voltage	If the controller supports a separate voltage setting, this control will be enabled.
Status line	Displays motor (axis) information, controller name, or communications errors.

Using open-loop controls

The LED on some controllers flash while the motor is moving.

To move the motor in open-loop mode, use the controls in the *Jog Control*, *Run Control* and *Motor Step Control* sections of the Manual Control tab.

Caution! In non M3-x systems, if you run the motor continuously in one direction in open-loop mode, the screw may continue to move until it runs out of the motor body. This should be avoided. If it happens, re-insert the screw and turn clockwise.



To use the Jog Control:

- Click **Rev** or **Fwd** to move the motor in reverse and forward directions. The motor runs for as long as the button is depressed.

The Forward or Reverse directions depend on how the motor is mounted.

To use the Run Control:

- Click **Rev** or **Fwd** to move the motor in reverse and forward directions. The motor runs until **Stop** is depressed.

To use the Motor Step Control:

1. In the *Mtr Steps* box, enter the number of "steps" to move the motor. A motor step is a drive signal burst (up to 65535).

   Note: The SQUIGGLE is not a stepper motor, but rather travels a continuous distance for the duration of the burst. Most people find it convenient to think of this as a "step."

2. In the *Interval* box, enter the microseconds between each burst (up to 104 milliseconds).
3. In the *Duration* box, enter the duration of each burst (in milliseconds). This number must be smaller than the *Interval* and it may be less than 1 msec.

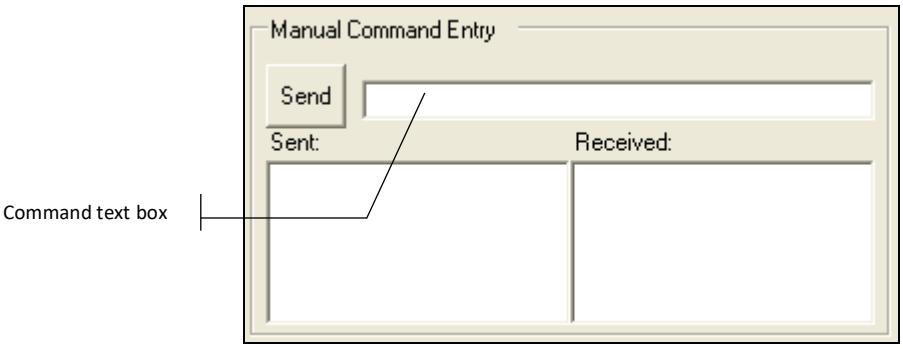
   Note: If using a controller with an NSD-1202 or 2101 driver the maximum duration 10 msec for an SQL series motor.

4. Click **Rev** or **Fwd** to move the motor in reverse and forward directions as defined by the motor step parameters.

Moving the motor with ASCII commands

The LED on some controller flash while the motor is moving.

All ASCII (text) commands, controller responses and sample commands are described in Appendix A. Use the *Manual Command Entry* section of the Manual Control tab to enter ASCII commands.



To use ASCII commands:

1. Enter an ASCII command in the command text box.
2. Click **Send** to send it to the controller. The *Sent* section lists the command sent; the *Received* section lists the controller response.

Viewing the NSE-5310 sensor details

If an NSE-5310 magnetic sensor is connected to the controller and the controller has been configured to use the NSE-5310 as the input sensor, you can view the details of what information is returned by the NSE-5310 sensor from the NSE-5310 tab.

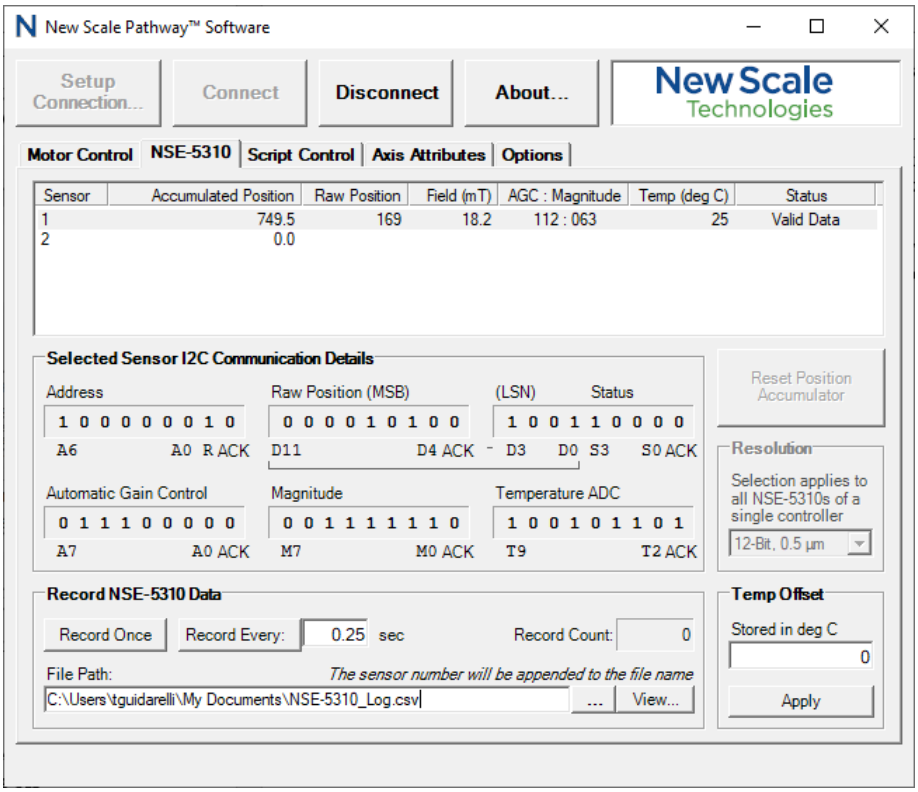


Figure 2-2. *The NSE-5310 Tab*

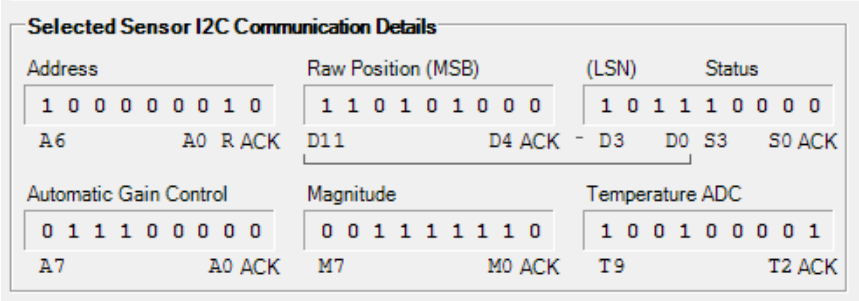
Table 2-2. *Tracker Tab Components*

Item	Description
Sensor list box	Used to select the sensor whose details will be shown in the group boxes below. This area also displays a summary of what the sensor is reporting: • Accumulated Position (µm): Since the NSE-5310 reports position within the current magnetic pole-pair, this value is the sum of the current pole position and all the magnetic pole-pairs that have been traversed relative to a home position. It is maintained by the controller. • Raw Position: The sensor position within the current pole-pair. This value wraps around when a pole-pair boundary is crossed. The controller maintains an accumulated position. • Field (mT): Is an estimate of the field strength (in milli-tesla) based reported amplifier gain and magnitude values. • ACG : Magnitude: AGC is the automatic gain control. It get's larger as the field gets weaker. Magnitude is the relative strength of the field at the current AGC value. • Temp (deg C): The temperature as reported by the NSE-5310. This will be no better than +/- 10 deg C. • Status: The reported tracker status
Selected Sensor I2C Communication Details	Displays what the NSE-5310 sensor is returning over the I^2C channel.
Reset Position Accumulator	Allows you to set the home pole-pair. *This is disabled on M3-x modules*.
Resolution	The NSE-5310 can be set to 10-bit (2 um), 11-bit (1 um) or 12-bit (0.5 um) resolution. *This is disabled on M3-x modules*
Record NSE-5310 Data	Saves position information from each sensor into a file at each invocation or at regular intervals.

Temp Offset	Can be used to apply an temperature offset correction to the NSE-5310's reported temperature value.

I²C Communication Details

The NSE-5310 datasheet describes in detail how to directly communicate with it the I²C channel. This section of the tab is intended to provide a live illustration of what each position query transaction looks like.



Each bit of each byte of data is shown in the boxes – from the address byte that is sent from the host to the sensor to the data bytes that are sent from the sensor to the host. The byte order is left to right and top to bottom (e.g. Address first, Temperature last).

The ACK bit indicates an acknowledgement (0 = OK). In the Address field, it's the sensor doing the acknowledging. In the case of the remaining bytes, it's the host doing the acknowledging. The last byte, however, a 1 is used to indicate the transmission is complete.

In this case, the host is the microprocessor in the controller.

Reset Position Accumulator

For position repeatability from one power cycle to another, it is assumed that a certain magnetic pole-pair on the magnetic strip will be used as the reference position. *Button disabled for M3-x modules.*

Once the strip is moved just in front of that pair, this button may be used to clear the position accumulator so that the next pair starts with an accumulated position of zero. But you are free to click it anywhere along the length of the magnet.

Note: For improved accuracy, reset the position accumulator 1.5 mm or greater from the end of the magnet to avoid magnetic end effects.

Resolution

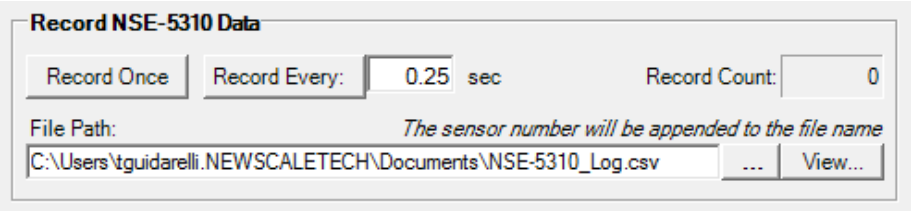
The NSE-5310 always returns 12 data bits (or 0 to 4095 decimal) which correspond to where the sensor is located within the current pole-pair. Given that each pole-pair is 2 mm in length, the effective resolution is 2 mm/4096 or 0.488 um (the controller rounds this to the nearest 0.5 um).

However, noise in the system may limit the stability of this signal. Therefore you have the option from this section to set the reported resolution to be 10-bit, 11 bit or 12-bit.

At 10-bit, the position resolution is 2 um. At 11-bit, the resolution is 1 um. *Drop list disabled for M3-x modules.*

Record NSE-5310 Data

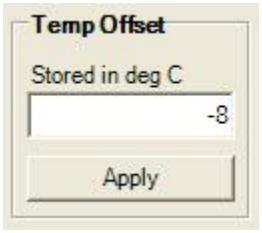
During your evaluation of the NSE-5310, you may wish to record the reported position as you are moving either the sensor or the magnetic strip. This section allows you to specify a file where each value returned by the sensor may be recorded. This also includes the field strength which is not reported directly but is calculated based on the AGC and Magnitude values.



You may record one position at a time for each sensor (via Record Once) or you may have the sensor positions recorded at regular intervals (e.g. ¼ second as shown above). Note that a separate file will be created for each sensor. The sensor number will be appended to the file name. Next to the file name is a file browse button and a file viewer button. **View…** will launch Microsoft® Excel (if available) or Notepad to display the file associated with the currently selected sensor in the above list box.

Temp Offset

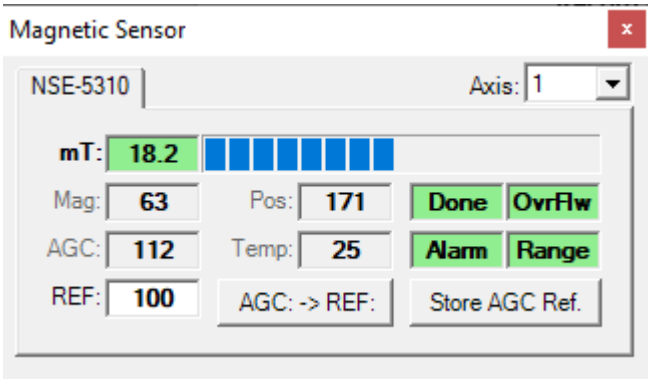
The NSE-5310 has a build in temperature sensor. However, it is not intended to be highly accurate (expect about +/- 10 deg C). But in this section, you can record a temperature offset in the controller that will be applied to the reported temperature.



Pole Position & Field Strength Monitor

A dialog window is available to allow you to monitor the pole position and the field strength of magnet while viewing other tabs (such as the Motor control tab).

To display this dialog, press ***<ALT>M***. The following dialog window will appear.



You do not have to close with window to continue to operate the Pathway software main screen. This way you may move the motor and view the field strength.

The field labeled REF: (reference) is used to set at what AGC value, the controller should consider the start or end of magnet.

The field strength declines at each end of the magnet. But the AGC (automatic gain control) value, which should be used as the reference, may vary from one unit to another based on the strength of the magnet itself and the gap between the magnet and the sensor.

Please refer to the design guide: *Establishing an Absolute Zero Reference with the Tracker NSE-5310 Position Sensor* for more details about how to establish an absolute zero position reference.

Moving the motor with scripts

Scripts are handy when you want to repeat motor movements for automated motor demonstrations.

To move the motor using scripts, use the Script Control tab, which displays the window shown in Figure 2-3.

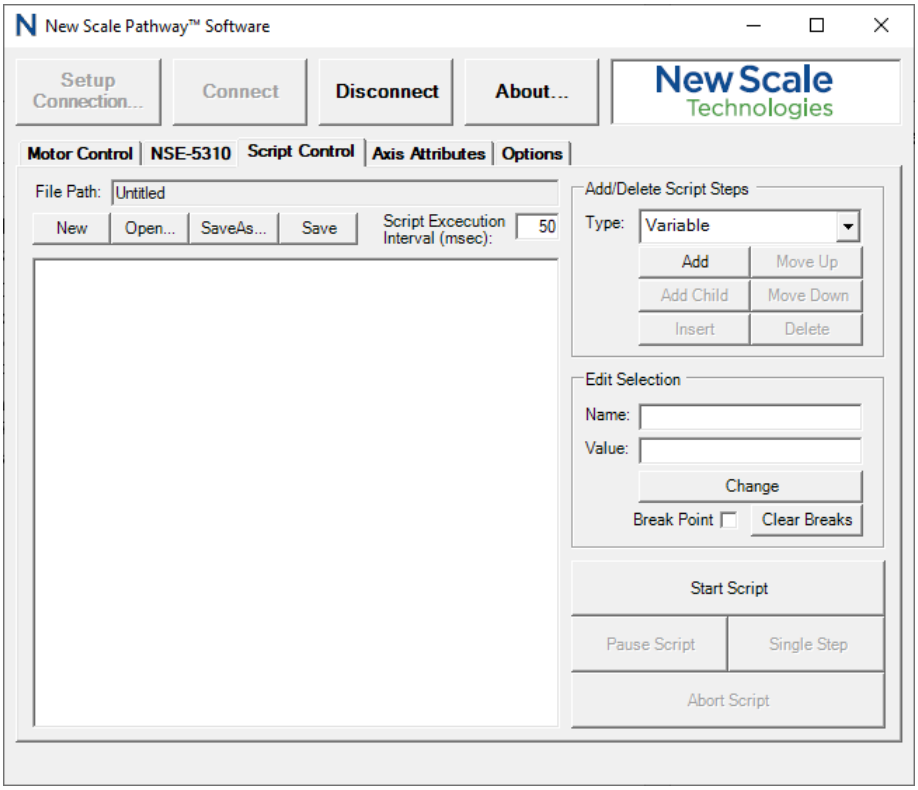
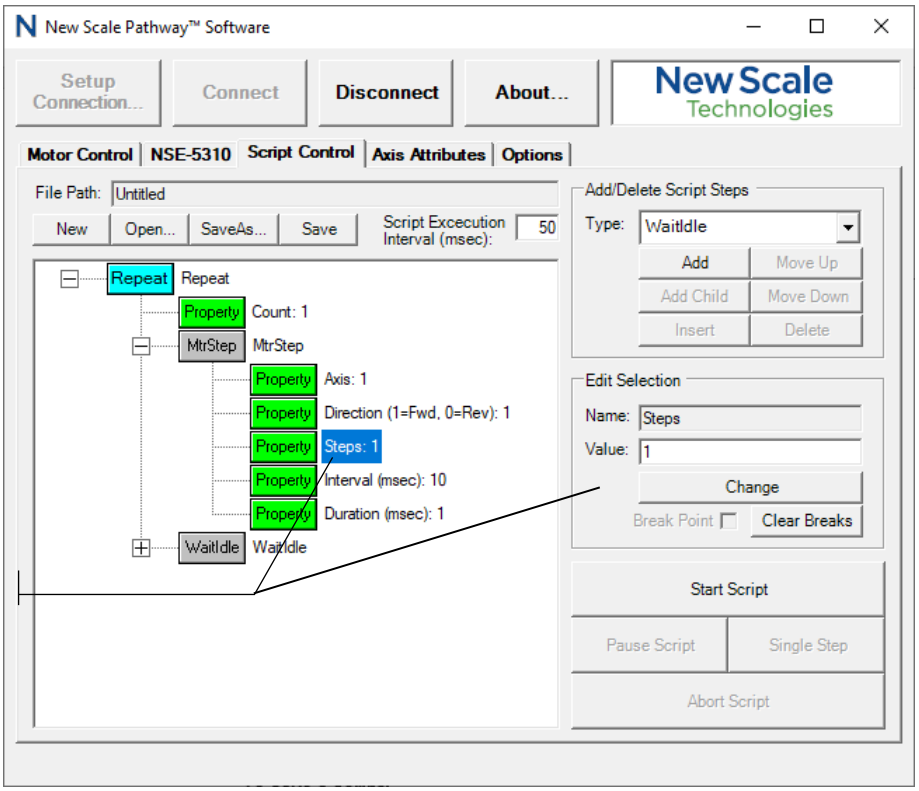


Figure 2-3. *The Script Control tab*

Creating and editing a script

To create and edit a script:

1. From the *Type* box, select the step you want to include in the script (see Table 2-3).
2. Click **Add**. A script tree appears in the box to the left, using default values for each step property (shown in green).



Note: Default step properties are displayed in the script tree and in the *Edit Selection* section.

3. **To edit step properties**, click on the property name and change its value in the *Edit Selection* area.

   For example, to change the Axis:1 property above, you would click on its accompanying Property box and change the *Value* listed in the *Edit Selection* area.

4. **To edit step/command names**, change them in the *Edit Selection* area. The modified name appears next to the default name in the script tree.
5. Click **Change**. The new property value is shown in the script tree.
6. Continue to add new commands as needed.
7. **To arrange the script**, use these commands:
   - **Add Child**: When clicked, the command selected in the *Type* box is added as child of the repeat command. This button is

available when the selected tree node is either a repeat step or inside a repeat step.

- **Insert**: Inserts the step selected in the *Type* box before the step selected in the script tree.
- **Move Up**: Moves the step selected in the script tree before the previous step.
- **Move Down**: Moves the step selected in the script tree after the next step.
- **Delete**: Deletes the step select in the script tree. You must confirm a step deletion.

8. **To pause the script at a specific step**, select the step and click **Break Point**. Break points are indicated in red in the script tree.

   To clear all pauses in the script, click **Clear Breaks**.

Saving a script

Scripts are stored on your computer as .xml files.

To save a script:

- Click **Save** at the top of the Script Control tab. The standard Windows *Save* or *Save As* window opens.

To save a script with a new name:

- Click **Save As** at the top of the Script Control tab. The standard Windows *Save As* window opens.

Executing a script

To start a script:

- With the desired script open, click **Start Script** in the Script Control tab.

To pause a script:

*Scripts can also be paused using **Break Point** when you create them.*

- With the script running, click **Pause Script** in the Script Control tab.

To execute a single step of a script:

- Click **Single Step** in the Script Control tab.

To abort a script:

- With the script running, click **Abort Script** in the Script Control tab.

Opening a script

To open a script:

- Click **Open** at the top of the Script Control tab. The standard Windows *Open* window opens.

Table 2-3. *Summary of script steps*

Step	Description
Variable	Each variable must have a unique name and can hold a single numeric value. Variables may be specified in the Property field of other steps in place of numeric constants. Variables are given an initial value but that value may be changed via the Calculation, Ask, Mask and Query Position steps.
Calculation	Updates the value of a variable using some combination of other constants and/or variables.
Ask	Displays a dialog through which an operator may enter the value of a variable. Upper/Lower limit values for the operator entry may be specified and/or a list of possible choices (comma delimited).
Mask	Used to set (OR), clear (AND) or toggle (XOR) one or more bits in a variable. The bits to be effected are specified with a mask value. This might be used to isolate a single bit (e.g. OnTarget) so that it can be used in a conditional execution step.
Repeat	Executes the child step(s) contained in the Repeat step a specified number of times. The Repeat step can contain other Repeat steps.
While	Executes the child step(s) contained in the While step until a specified condition is false or a Break step is encountered.
IF	Executes child step(s) once if the specified condition is true. An Else step may be placed after an IF step. If the specified condition is false and an Else step is present, the child step(s) of the Else step will be executed.
Else	Executes child step(s) once if the preceding IF

Step	Description
	statement's specified condition is false. If an Else step is encountered without a preceding IF then the script will abort.
Break	If encountered within a Repeat or While loop the break step will terminate the loop. The next step after the loop step would be executed. If encountered outside a loop then the script will terminate. The Break step would normally be used within an IF or Else step which was itself in a loop.
WaitTime	Pauses a script for a specified amount of time.
Run User Program	Allows you to launch an external application. You can either wait for it to complete or continue to the next script command.
Display Tab	Displays the selected tab defined as a numeric index. For example, 0 is the Manual Control tab, 2 is the Display Only tab, etc.
CL-Mode	Enables or Disables closed loop control mode for the specified axis. Closed loop mode requires that a position sensor be integrated with the controller. While in closed loop mode you can specify a motion velocity and acceleration as well as target positions. The MtrStep command, however, still operates in the same manner even in close loop mode.
CL-Speed	Sets the closed-loop velocity and acceleration (in um) of the motor.
Target	If in closed loop mode, allows you to specify an absolute target for a given axis. After starting the motion the script execution advances to the next step. This allows you to begin moving one or more other axes in parallel. However, if you want to wait until motion on a particular axis completes, use the WaitIdle step. Alternatively you may use WaitTime.
EncStep	If in closed loop mode, allows you to specify distance to move from the last target position of a given axis in either the forward or reverse direction. Note that the distance is added to the last target not the current position.

Step	Description
	Like the Target command, this command does not wait for motion to complete.
Stall Detect	Enables or disabled stall detection while in closed loop control mode.
Zero Ref. Enable	If the controller supports a reference mark, this step may be used to enable it. That is, if enabled, the controller will stop motion and zero the encoder count (i.e. the position). This step is not recommended for M3-x modules.
Log Position	Used to log the one or more position values of a specfied axis to a file. If more than one value is to be recorded, a time interval between recordings (in seconds) may be specifed. Five columns are generated for each record. They are… • Number, the sample number* • Elapsed, time since the log was started* • Position, the position of the motor • DeltaPos, the change in position since the last query.* • Error, the distance from the position to the last target (if stationary) or set-point (if moving). *These are only meaningful if more than one position value is being generated by the same Log Position step.
Query Position	Used to query position, error and status and place the values into the specified variables which may be used later in calculation and or conditional execution steps.
OL_Speed	Sets the open-loop speed of the motor.
MtrStep	Moves the motor a specified number of motor steps (signal bursts) in the specified direction. The interval between bursts and the duration of each burst (in msec) are also specified. Note that for ASIC based drivers (e.g. NSD-2101, NSD-1202) the maximum duration of an individual burst is 10 msec. However, a continuous burst is achieved by setting the interval equal to the duration.

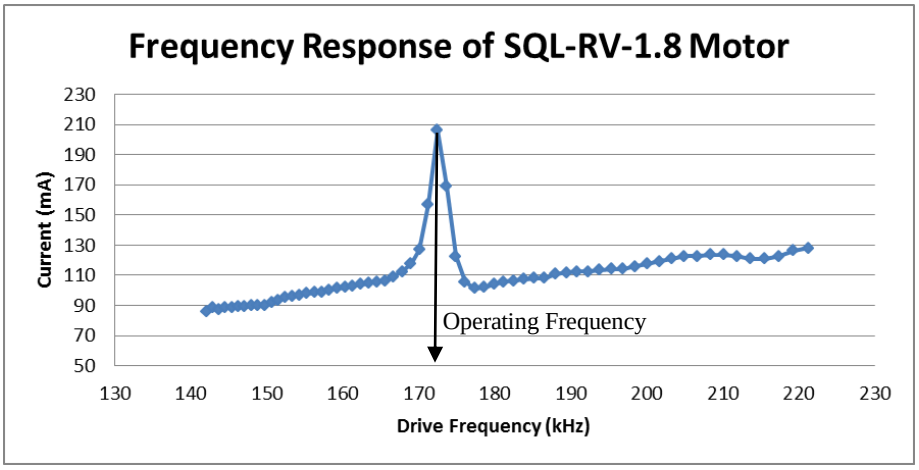
Step	Description
FreeRun	Runs the motor of a specified axis in the specified direction for a specified duration or until stopped.
Stop	Halts motor movement of the specified axis.
WaitIdle	Pauses the script until the motor of the specified axis stops moving.
Custom	Sends the specified ASCII command to the motor. See Appendix A for a list of available commands.

Confirming Motor Operating Frequency and Q

Controllers such as the MC-3300 and MC-1100 have an on-board current sensor that can be used to confirm the best operating frequency of a motor based on the current consumed over a range of frequencies. This can also be done with an M3-F, M3-L or M3-LS controller as long as they have F/W V4.2.6a or later and are connected through an M3 USB Interface that was shipped after 1/31/2012.

Frequency Response of Direct Drive Motors

The MC-3300-RV and some of the M3-x controllers use a square wave driver chip (NSD-2101). The frequency response of an SQL-RV-1.8 squiggle motor connected to this driver consists of a gradually increasing background current level (due to the motor capacitance & drive frequency) and a peak in current at the operating frequency.



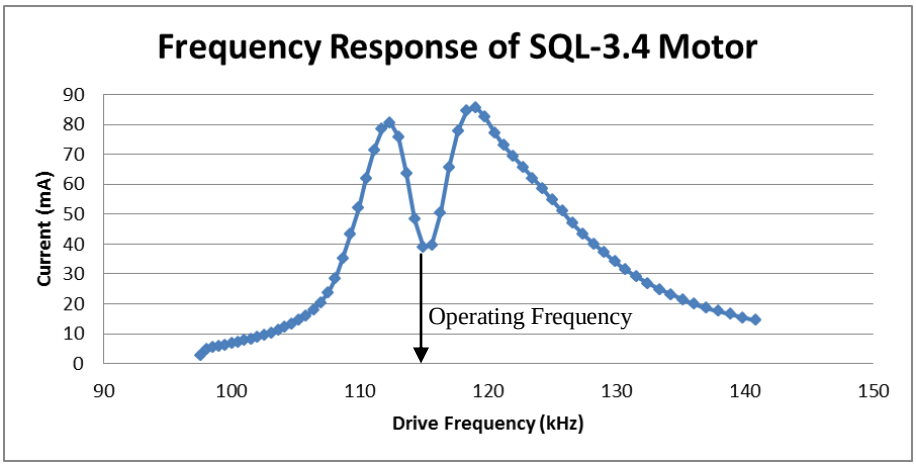
The peak in current consumption is due to the higher dissipation of the motor at its operating frequency.

Frequency Response of Tank Circuit Driven Motors

The MC-1100 controller feeds a PWM signal into a tank (LC) circuit which boosts the voltage on its way to the SQL-3.4 motor. The tank circuit's electrical resonance is tuned to match the nominal mechanical resonant frequency of the SQL-3.4 motor.

If the motor were only a capacitor, the frequency response would be similar to a Gaussian curve where the peak occurred at the electrical

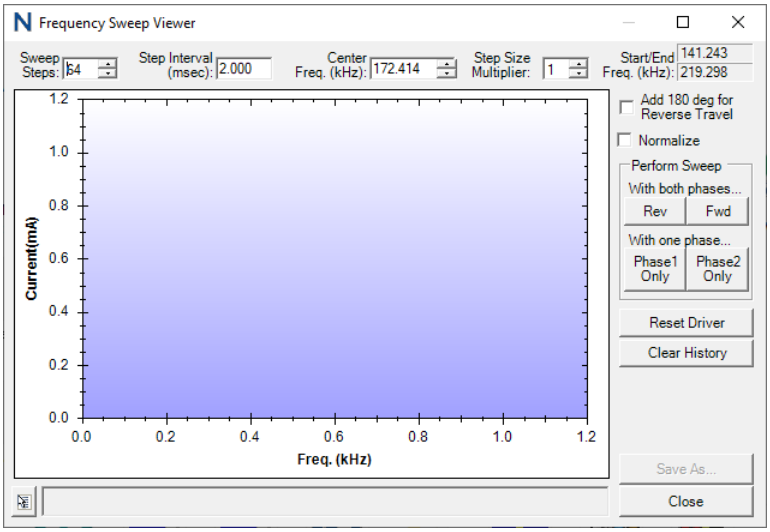
resonant frequency of the tank circuit. However, since the motor consumes the most energy at about that same frequency there is a valley near the center of the curve.



Therefore the best operating frequency of the motor is at the lowest point in that valley.

Using the Pathway to Generate a Frequency Response Curve

While connected to a controller via the Pathway software (and while the Pathway window has input focus) the operator may press <Alt>S to display the **Frequency Sweep Viewer** dialog. This is a modeless dialog which means that the operator can still use the main Pathway screen.



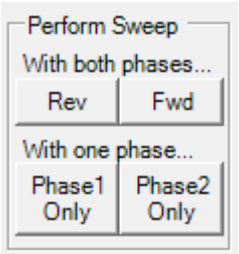
The controls at the top of the dialog are used to set the range over which the frequency sweep will be performed. Most controllers are able to collect up to 64 data points per sweep.

The **Step Interval** (i.e. the duration of the drive signal at a given frequency) can be changed but a minimum of 2 msec is recommended. The **Center Freq.** will default to the currently programmed operation frequency of the motor. In the above example, a sweep would be started 32 steps (1/2 of 64) below the center frequency.

The size of each step (in kHz) depends on the resolution of the master clock. The NSD-2101 has a 25 MHz clock so the change in pulse period for each step is 0.04 usec (or ~1.2 kHz in the range of 173 kHz). The **Step Size Multiplier** can be used to cause the sweep to cover a wider range of frequencies but at the cost of resolution. Finally the **Start/End Freq.** boxes on the right display the frequency range to be covered based on the settings to the left and the master clock frequency reported by the controller.

The sweep is always performed on the currently selected Axis in the Pathway main window.

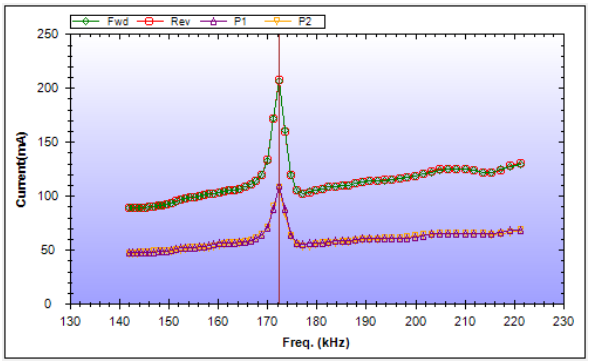
The sweep may be initiated using the buttons on the right side of the viewer. Clicking the **Rev** or **Fwd** button will trigger the sweep using both drive phases, the difference being which phase leads which determines the direction of motion. If supported by the controller, sweeps may be performed with only one or the other phase (see **Phase1 Only** and **Phase2 Only** buttons).



After each sweep completes, the results will be read and displayed by the Pathway S/W. Pathway will locate the point of peak current and place a cursor there so that the report at the bottom of the viewer will show frequency and current consumed at that point.

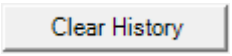
Though the example given here was generated with an MC-3300-RV, results for an MC-1100 and SQL-3.4 motor can be generated in the same manner though the shape of the peak will be very different.

The sweep buttons may be pressed multiple times and the results of each will be added to the plot. In the example to the right the **Fwd**, **Rev**, **Phase1 Only** and **Phase2 Only** buttons



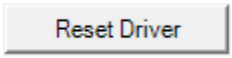
were pressed. In this example there is good alignment between both phases.

Results will continue to accumulate as the sweep buttons are pressed.



Click the **Clear History** button once to remove all but the most recent results for each type of sweep. Clicking a **Clear History** a 2nd time will remove all results.

Since the NSD-2101 driver has the ability to perform a motor frequency calibration internally (which adjusts the trimming on its master oscillator) the **Reset Driver** can be used to clear that internal calibration. This button is disabled for all other types of controllers.



The **Add 180 Degrees for Reverse** check box would not be used for an SQL motor and should remain unchecked.

All displayed results can be saved to a CSV file via the **Save As…** button. The results are written in a format which can be imported and plotted by Microsoft® Excel.

Conditions that Affect Sweep Results

The operator does need to be aware of the available travel in the direction of the sweep motion as the motor may be forced to stop before the frequency sweep can complete. There can be some motion even when sweeping a single phase.

Ambient temperature, motor duty cycle, screw loading, motor body mounting constraints and damage to the motor all have an effect on the mechanical resonant frequency of the motor.

Mechanical resonance varies inversely with temperature. In the case of the SQL-RV-1.8 motor it changes by -40.7 Hz per deg C. Motor temperature will increase with use so a motor being operated at a high duty cycle will be warmer than an idle motor.

A high load or a highly constrained motor mount will increase the mechanical resonance and lower the motor Q, or mechanical quality factor (and thus the push force). This is reflected in the sweep results by a peak that is lower in amplitude and shifted up in frequency. A side load on the screw or non-symmetrical mounting constraints may appear as differences between the sweep results of each phase individually.

If the motor is damaged such that the electrical connection to one or both piezo plates of a phase is lost, it will be reflected in a reduced

amount of current consumed. Sweeping phases individually will reveal which phase has been damaged.

A flat line response can indicate a problem with the driver or a complete electrical disconnect from the motor.

If the overall height (in current consumed) of the results plot is lower than expected, check the supply voltage to the controller since the lower the supply the lower the current consumption.

If using the MC-3300 controller, keep in mind that although it is able to respond to the host while connected to a power supply that is below 2.3V (as it has a boost regulator to supply the processor), the driver ASIC runs directly off the supply and will not operate below 2.3V (if equipped with the NSD-2101; below 2.8V if equipped with the NSD-1202).

3: Setting Motor (Axis) Attributes

This chapter describes how to view and change the attributes associated with a motor (axis), including:

- Motor EEPROM parameters
- Open-loop parameters
- Closed-loop parameters
- Motor drive settings (frequency, or resonance)

Each of these options is described later in this section.

To view and change motor (axis) attributes, use the Axis Attributes tab in the software as shown in Figure 3-1.

Note: Previous versions of this software required that an EEPROM password be entered prior to changing some of these parameters. This requirement has been removed (i.e. Pathway sends the password as part of the Connect sequence).

However, if the controller is reset without disconnecting it will return an invalid command message if an attempt is made to change an EEPROM parameter. If this occurs, just click Disconnect then Connect.

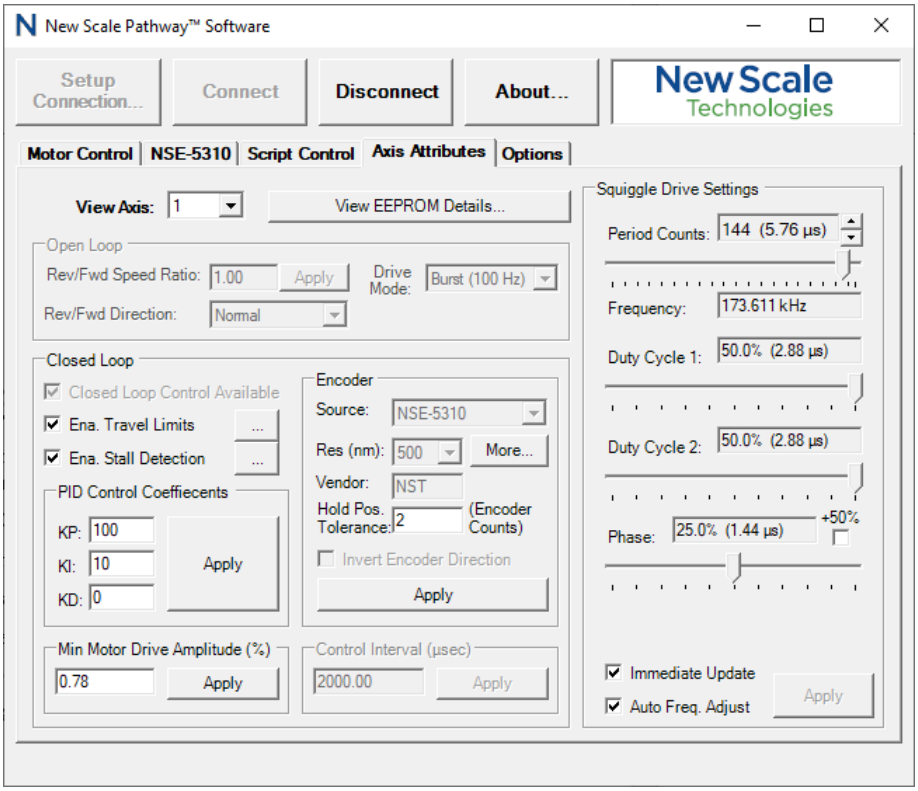


Figure 3-1. *The Main window, Axis Attributes tab*

Viewing and editing motor EEPROM parameters

Many parameters, such as motor frequency, are stored in electrically erasable programmable read-only memory (EEPROM).

On SQ Series controllers (such as the SQ-2300), the EEPROM is located in the motor connector (it goes with the motor). In MC Series controllers (such as the MC-3300), the EEPROM is within the processor itself (it goes with the controller). In either case, you can view these parameters and change them.

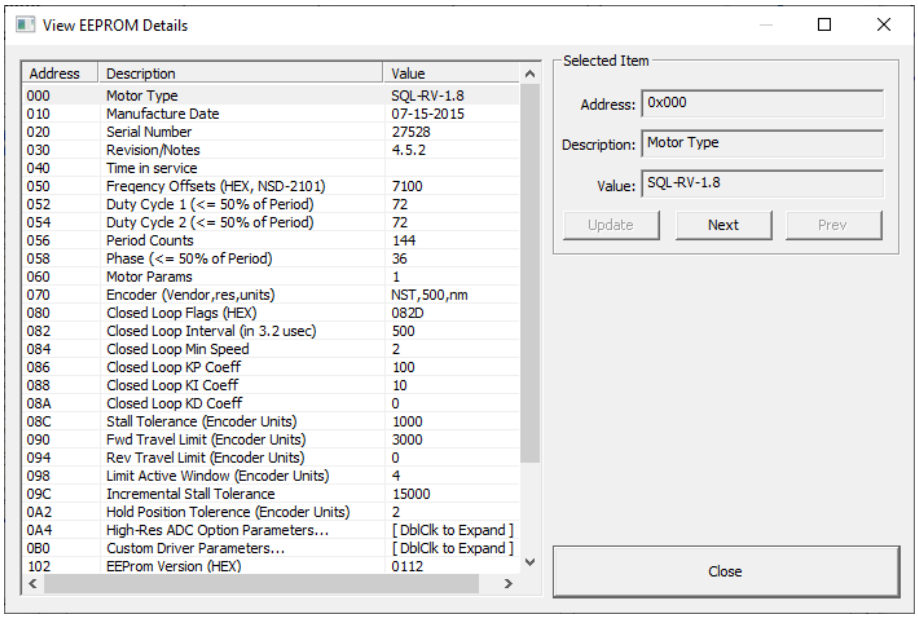
In most cases, you will not need to change these parameters. *On M3-x Modules some of these parameter settings are not enabled since they are configured at the factory only.*

To view and edit motor EEPROM parameters:

The motor selected in the Manual Control tab will be selected by default.

1. From the drop-down box in **View Axis**, select the motor whose parameters you want to view or edit.

   The *View EEPROM Details* dialog box opens, showing all of the EEPROM parameters for the selected motor.



2. Select the parameter you want to edit using the **Next** and **Prev** buttons in the *Selected Item* section.
3. Change the value in the *Value* box.
4. Click **Update**. The value changes and the next parameter is displayed in the *Selected Item* section.
5. Click **Close** when finished.

Urgent Note: You should record the factory supplied EEPROM settings prior to making any changes.

Viewing and editing open-loop parameters

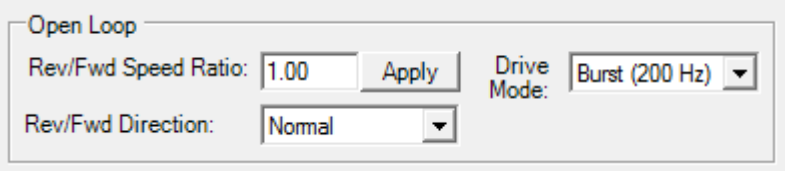
You can control the relative motor speed (for forward vs. reverse), select the drive mode used and the switch the default motor directions. These parameters are stored in the motor's EEPROM.

To view and edit open-loop parameters:

The motor selected in the Manual Control tab will be selected by default.

1. From the drop-down box in **View Axis**, select the motor whose parameters you want to view or edit.

   The open-loop parameters are displayed in the following section:



2. **Rev/Fwd Speed Ratio**. This parameter is used to equalize the motor speed when the load is different between forward and reverse directions. Then click **Apply**.
3. **Drive Mode**. Defines whether the motor drive signal is intermittent (in bursts) or continuous (but adjusting voltage or pulse width) to regulate speed:
   - **Burst mode** (100/200 Hz) is a better choice for open-loop applications except when audible noise is an issue. This mode has a more linear speed range and may overcome stiction more easily than Amplitude mode.

     Note: The MC-3100/3300 controllers run at 200 Hz while in Burst Mode.

   - **Amplitude or PWM mode** is a better choice for closed-loop applications. It is quieter and provides for smoother motion. Amplitude mode is a continuous drive signal where, depending on the controller type, either the voltage or the duty cycle of each motor pulse is adjusted according to the specified speed.
4. **Rev/Fwd Direction**. If the motor does not move in the direction you consider forward, when clicking the Jog Fwd button, you may invert the default forward direction by selecting **Inverted** from the drop list.

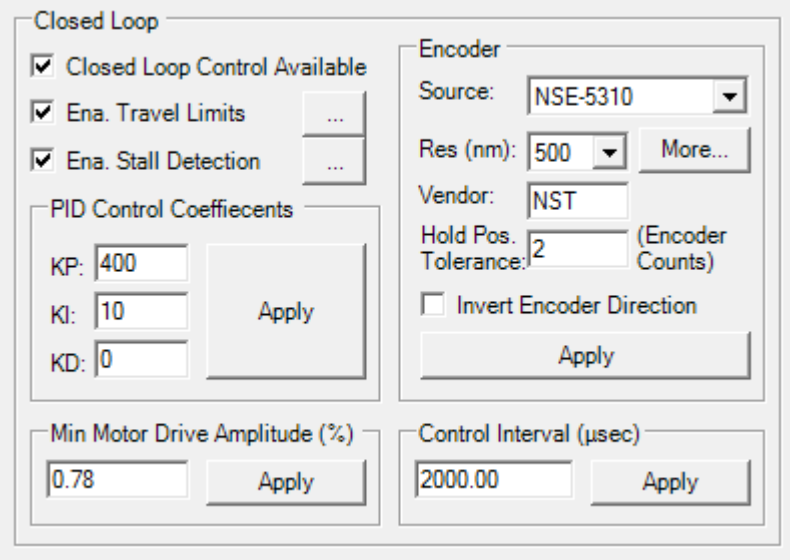
Viewing and editing closed-loop parameters

To view and edit closed-loop parameters:

The motor selected in the Manual Control tab will be selected by default.

1. From the drop-down box in **View Axis**, select the motor whose parameters you want to view or edit.

   The closed-loop parameters are displayed in the following section:



2. Refer to Table 3-1 to change the closed-loop parameters.
3. Click **Apply** as needed to save the changes in the motor's EEPROM.

Table 3-1. *Closed-Loop Parameters*

Parameter	Description
Closed Loop Control Available	When selected, indicates that closed-loop hardware is available. This parameter is distinct from the closed-loop enabled flag in Appendix A. The closed-loop enabled flag is ignored unless this option is selected. At power-up, this parameter is read and if it is enabled, the closed-loop enabled flag can be set to true. When enabled, even if the closed-loop enabled flag is set to false, the encoder position will still be read and made available via the position query command. This parameter corresponds to bit 0 of EEPROM address 0x080.
Enable Travel Limits	See "Enabling travel limits" next.
Enable Stall Detection	See "Enabling stall detection" on page 52.
PID Loop Coefficients	The PID loop is a common mechanism for closed-loop positioning. The following three coefficients are applied to the position error in order to determine the motor drive gain: • **KP** (Proportional): Used when driving to an absolute/relative target or when free running the motor.

Parameter	Description
	• **KI** (Incremental): Used when holding the last targeted/halted motor position. • **KD** (Differential): Typically used only in custom applications where there is a large load (i.e. high inertia). Guidelines: • Larger PID coefficients result in a more responsive, less stable motor. • Smaller PID coefficients result in a less responsive, more stable motor. • The larger the load on the motor, the larger the PID coefficients need to be.
Encoder	• **Source**: Identifies the type of position sensor - digital (quadrature), analog or the NSE-5310 magnetic sensor. • **Res (nm)**: Resolution of the encoder/sensor being used with the motor. This value is used by the host PC so that it can convert between encoder count and μm units. • **Vendor**: Optional identity of the encoder vendor (not required). • **More…:** If an analog sensor is being used, this dialog allows you to specify actual vs. reported resolution. For example, using a reference sensor, you determine that you get 1.534 um per ADC count, you can set the actual resolution to that value and the reported resolution to 2 um. The controller will perform the conversion for you. If a NSE-5310 sensor is being used, a dialog will be displayed that lets you specify the actual pole-pair length of the magnet you are using. The standard pole-pair length is 2000 um. But if you have one that is slightly longer (e.g. 2007) you will accumulate 7 um of error for each pole pair traversed. However, by specifying the exact length, this accumulated error can be removed. • **Invert Encoder Direction:** When checked, indicates that the encoder value will be inverted to match the forward direction of the motor. This may be necessary to ensure that position value increases when moving in the forward direction. Alternatively, the motor direction may be inverted. This parameter corresponds to bit 1 of EEPROM address 0x080.
Min Motor Drive Amplitude (%)	The smallest drive gain required to overcome the stiction of the load on the motor screw. Make this value as small as possible (typically <1%) to minimize overshoot when holding position.

Parameter	Description
Control Interval	The time interval between each sampling of the encoder count, the calculation of the position error, and the application of the appropriate motor drive gain. This value is typically not changed.

Enabling travel limits

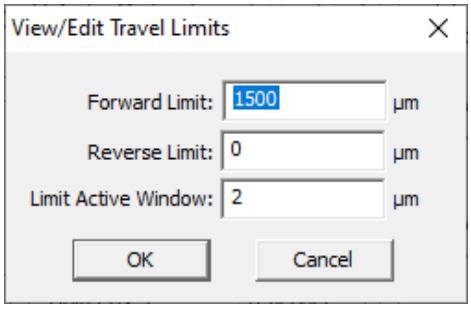
This section describes how to enable travel limits in closed-loop mode. With travel limits, you can limit motor travel using an absolute forward and reverse position (with respect to the encoders' reference or index mark).

When using an incremental encoder, such as a glass scale, the reported position is relative to wherever the stage happened to be when the controller was turned on. For travel limits to work the encoder's reference mark must be found by turning on the **Enc Ref** check box on the **Manual Control** tab and moving the stage until the reference mark is found). Even if travel limits are enabled, they will be ignored until the reference mark is detected.

However, if the position sensor is absolute, as with a potentiometer or hall sensor, it is not necessary to locate a reference mark since the reported position is consistent from one power cycle to the next. To enable travel limits:

1. Select the **Enable Travel Limits** check box.
2. Click  to the right of the check box.

   The *View/Edit Travel Limits* dialog box opens.



3. Set the following values:
   - **Forward Limit**: The absolute forward position.

– **Reverse Limit**: The absolute reverse position. **Limit Active Window**: The distance in front of each limit position over which the limit flag will stay active after it is tripped.

4. Click **OK**.

Enabling stall detection

This section describes how to enable stall detection in closed-loop mode.

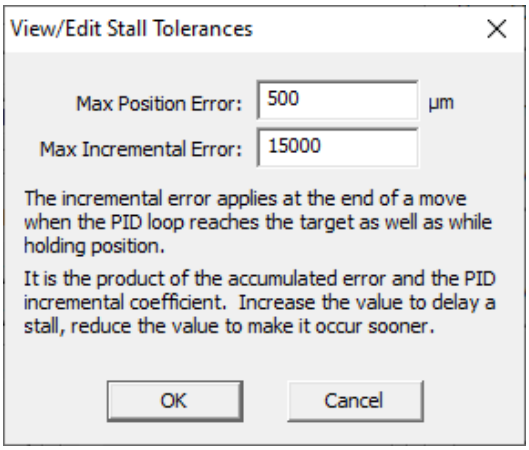
Position error is the distance between where the motor is supposed to be located and where it is actually located.

If stall detection is enabled, the controller continuously monitors the position error while the motor is being driven to an absolute/relative position or free running. If the position error (the distance between where the motor actually is and where it is supposed to be) reaches or exceeds the maximum position error, the motor will be halted and the stalled status flag will be set.

To enable stall detection:

1. Select the **Enable Stall Detection** check box.
2. Click  to the right of the check box.

   The *View/Edit Stall Tolerances* dialog box opens.



3. Set the following values:
   – **Max Position Error**: The maximum distance between where the motor actually is and where it is supposed to be.
   – **Max Incremental Error**: The cumulative error.

Note: The Max Incremental Error prevents the motor from pushing when it's not making any progress even if the distance to the target may be within the maximum position error.

4. Click **OK**.

Viewing and editing motor drive settings

By changing the motor drive settings, you're tuning the motor to its mount. Drive settings are stored on the motor's EEPROM chip.

Input parameters for integrated circuits capable of generating square waves typically include **period** and **duty cycle**. Other parameters may include **phase** and **amplitude** as show in Figure 3-2.

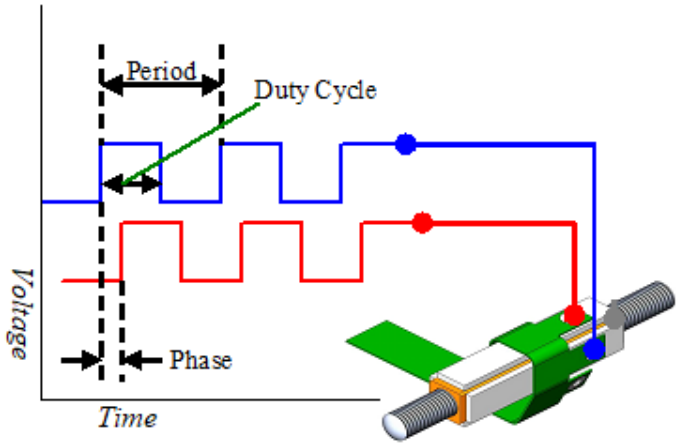


Figure 3-2. *Voltage versus Time effect on motor parameters*

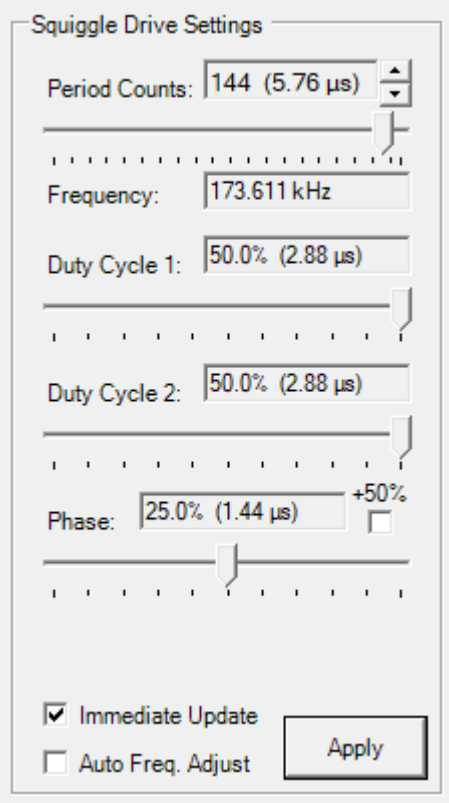
To view and edit motor drive settings:

The motor selected in the Manual Control tab will be selected by default.

You can drag the slider bar or click on the slider bar line to move incrementally.

1. From the drop-down box in **View Axis**, select the motor whose drive settings you want to view or edit.

   The motor drive settings are displayed in the following section:



2. Refer to Table 3-2 to change drive settings.
3. Select the *Immediate Update* check box to have the frequency value used immediately by the controller as you are changing it.
4. Click **Apply** to save the changes in the motor's EEPROM.

Table 3-2. *SQUIGGLE Drive Settings*

Parameter	Description
Period Counts Actual Frequency	Period Counts are the time between the start of each cycle (the inverse of frequency).* • To decrease the Period Count and increase the Actual Frequency, move the slider to the right. • To increase the Period Count and decrease the Actual Frequency, move the slider to the left. • **Nominal Value for SQL-RV-1.8 Motor**: Inverse of ~173.6 KHz or ~5.76 µsec

Parameter	Description
Duty Cycle 1	Time that the drive signal is high in each cycle (Phase 1). For SQL-RV-1.8 motor, this value should be 50% best performance.
Duty Cycle 2	Time that the drive signal is high in each cycle (Phase 2).
Phase	Separation in time between the start of a cycle in each drive signal. For best performance, this value is typically 25%.
+50%	This checkbox is used to force a 180 degree phase shift when changing direction. This is intended for UTAF (Ultra Thin Auto Focus) motors.
Auto Freq. Adjust	If enabled, the controller will, while in open loop mode, periodically perform a frequency calibration on the motor to maintain optimal performance. In closed loop mode, it's up to the operator to request a calibration. For controllers equipped with the NSD-2101 driver, this can be done via the Calibrate button on the Motor Control tab.

* The period value must match or be near the period of the mechanical resonance of the motor.

4: Using Additional Options

This chapter describes a few additional options you may find helpful in evaluating the motor.

Selecting Visible Tabs

To show or hide available tabs in the Pathway software, use the Options tab shown in Figure 4-1. Changes made to the tab selections are saved in the user's My Documents directory and are restored at the next invocation of the software.

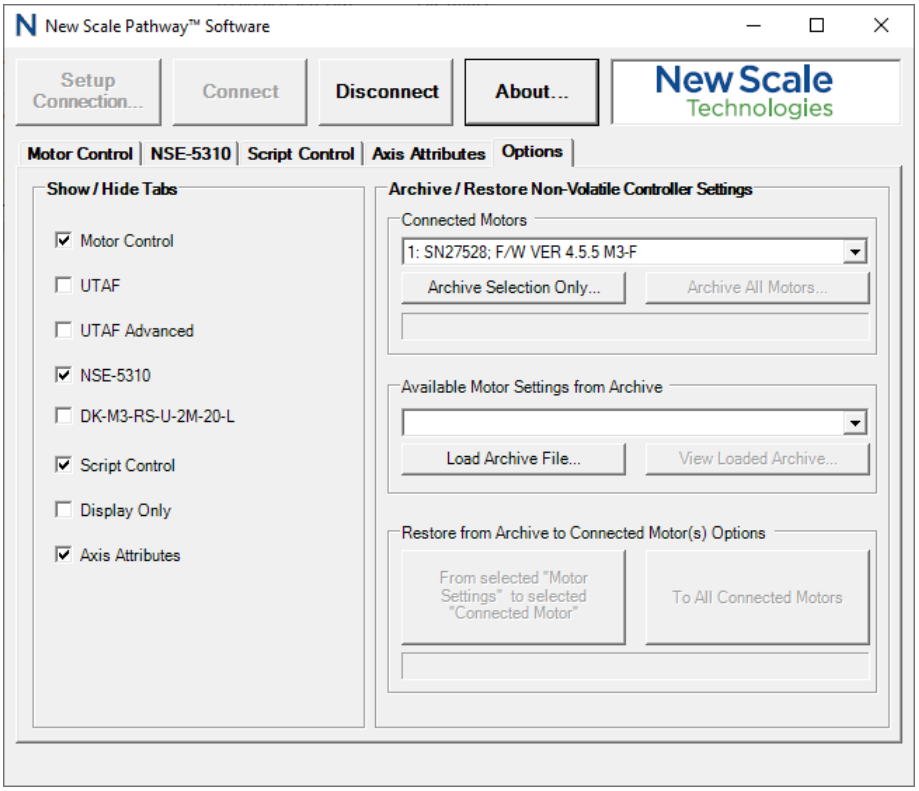


Figure 4-1. *The display Options tab*

This tab also allows the EEPROM data within the controller to be archived to a PC file and later restored in case an operator accidently changes parameters to values which lead to unacceptable performance (e.g. such as the motor drive frequency).

Displaying motor (axis) position

To display the current position of the first three motors in large format, use the Display Only tab shown in Figure 4-1. This option is useful when you want to demonstrate motor movement to a group.

Colors in the *Position* column indicate motor (axis) activity as follows:

- **Blue** indicates that the motor is idle.
- **Green** indicates that the motor is moving.
- **Red** indicates that the motor has stalled.
- **Black** indicates that the motor is not in use.

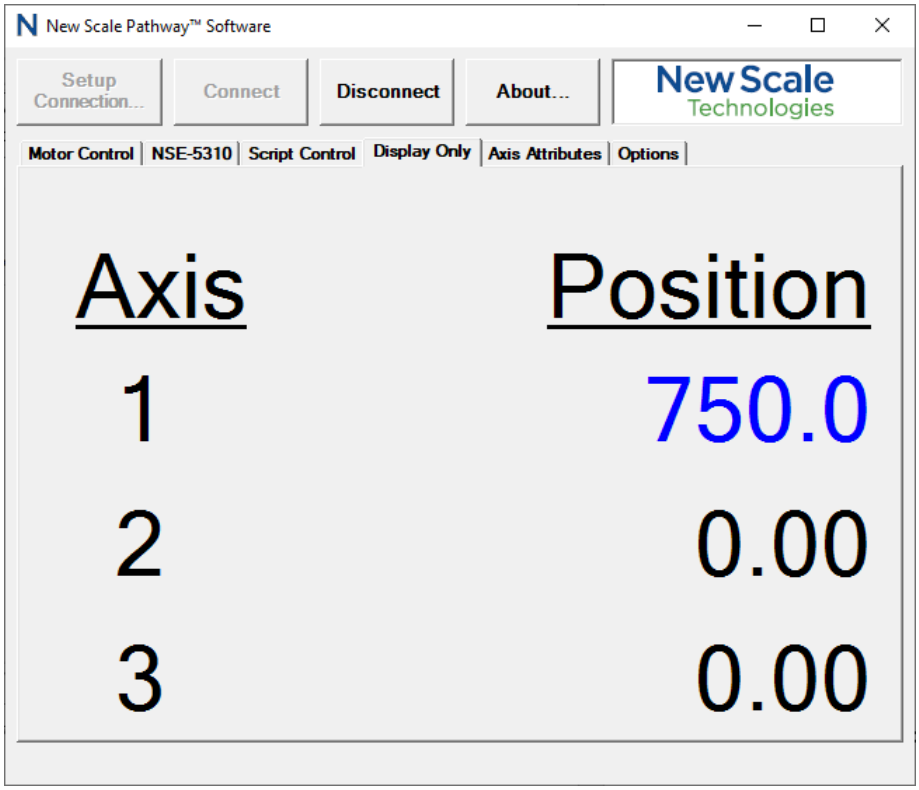


Figure 4-1. *The Display Only tab*

Generating a Motion Performance Log Plot

A performance log viewer has been added to the Pathway s/w to allow the operator to see the effect of a change to PID parameters or assess the friction/load on the motor over the travel range.

Controllers that support closed loop motion execute a routine at regular intervals that fetches the encoder position and decides how hard to drive the motor and in what direction.

.

*This is called the **Closed Loop Interval** and it can be anywhere from 0.5 to 2 msec.*

When the feature is enabled, data is logged by the controller at every closed loop interval (typically every 2 msec) or every *N* intervals (where *N* is 2 – 10). The data logged consists of the encoder position at that interval, the position error (difference between encoder position and where the motor is supposed to be located at that moment) as well as the motor gain that was applied at that interval.

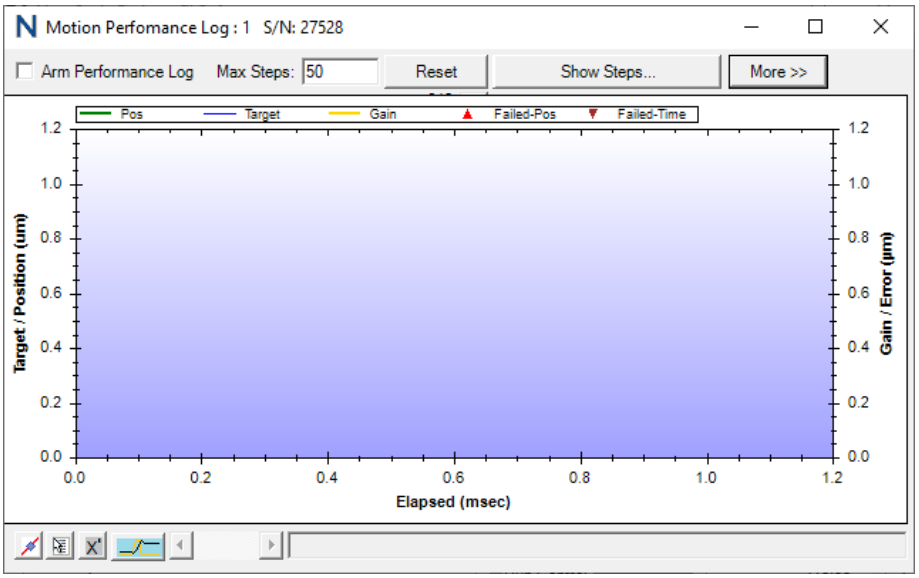
This feature requires that the controller have a position sensor integrated. The following controllers are able to provide this performance log data.

- MC-3300, F/W V1.9.8 or later.
- M3-F, M3-L or M3-LS, F/W V4.2.6a or later. Also requires an MC USB Interface that was shipped 1/31/2012 or later. V4.4.3 or later is required for the extended interval feature.
- MC-1100, F/W V2.2.4 or later

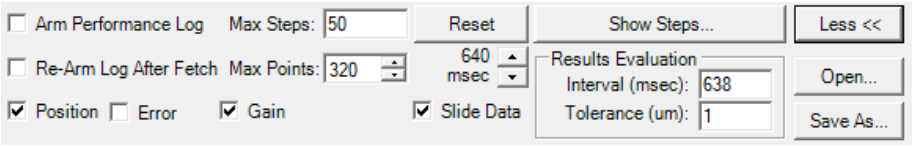
Since available memory varies with each controller, the amount of data that can be collected at one time is controller dependent.

Displaying the Performance Log Viewer

To display the performance log viewer, press <Alt>P while the Pathway main screen has input focus.



You may click the **More >>>** button to reveal more setup controls. Each control is described in the table following the image below.



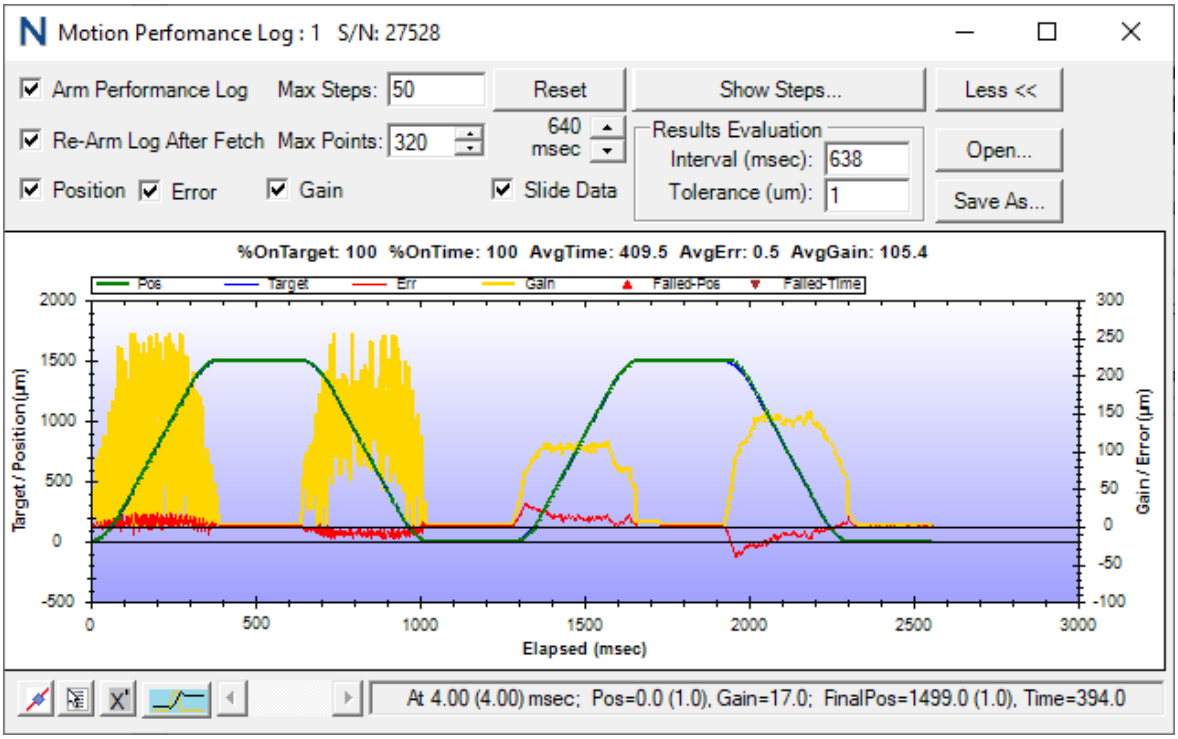
Control	Description
Arm Performance Log	Checking this box tells the controller to start logging data at the start of the next move. It may start logging immediately if position maintenance is enabled and there is some external vibration. Once the controller's log array is full the viewer will read the data out and plot it.
Re-Arm Log After Fetch	If checked, the viewer will automatically rearm the performance log after the data is read from the last move.
Max Steps: 50	In this case a ***Step*** refers all the data points in a single performance log acquisition. **Max Steps** is the maximum number of steps that will be graphed at one time.
Max Points: 320	This is the maximum number of data points to read out of the controller for a single Step. The controller has a limited amount of space so **Max Points** cannot exceed the limit set by the controller (for the MC-3300 the maximum is 200, for the M3-F/L it's 320).
Position Error Gain	These check boxes enable the data associated with the name to be plotted. If **Position** is checked both the position and the set-point are plotted. If **Error** is check, the difference between the position and the set-point is plotted. If **Gain** is checked, the computed motor gain (0-255, where 255 is the maximum motor gain) being applied at that interval is plotted.
Slide Data	If checked and the number of displayed steps has reached the maximum (as defined by Max Steps) then after the next step completes the first step will

	be removed and the last step will be appended to the end of the plot. Otherwise the data will be reset and the last step will be the only step visible.
Reset	If clicked, clears all visible steps as well as the Arm check box.
640 msec	This is time over which the data logging will be active. Note that It will vary from one type of controller to another (due to differences in the log array size and the closed loop interval) . If the Up/Down arrow is enabled, the time may be increased up to 10x. This is done in the controller by skipping closed loop intervals (i.e. not logging data at every interval). This means that the greater time range comes at the expense of resolution.
Show Steps...	This button causes a dialog to be displayed which enumerates each step showing the target, error (based on specified tolerance, see below), duration to reach that tolerance, average & max gain applied over that step.
Interval (msec): 20 Tolerance (um): 4	As each step is loaded by the viewer it can be evaluated according to the criteria in these edit boxes (i.e. did the motor reach the current target within the specified time interval and within the specified tolerance). The default interval (20 msec) is quite short for most SQL motor applications so if it is to be used, the interval will have to be increased.
Open... Save As...	The data currently visible in the performance viewer can be saved to a CSV file via the **Save As…** button. Files previously saved via **Save As…** can be read and plotted via the **Open…** button.

Example Performance Log

The next figure illustrates the performance log of an M3-F with two PID coefficient settings. The forward/reverse cycle on the left was taken with a KP value of 2000. The forward/Reverse cycle on the right was taken with a KP value of 200.

The point is to make clear what effect too high a KP coefficient has on the calculated motor gain and thus the constant over/undershooting of the set-point as to motor moves toward its target.

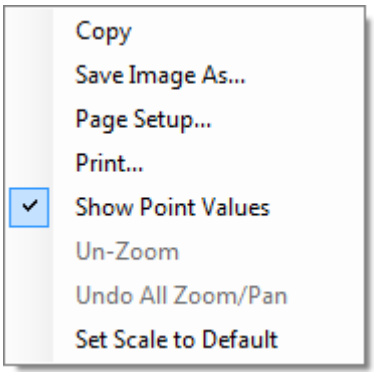


Other Features of the Performance Viewer

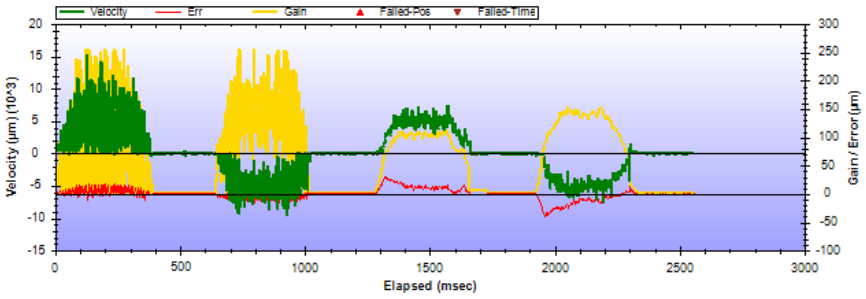
- Zoom In/Out. You may zoom into any area of the plot by pressing the left mouse button over the plot and dragging it across the area of interest. Releasing the mouse expands the selected area. You may zoom in multiple levels. Right clicking the mouse un-zooms to the previous level.

- Show Points: Depressing the  button (lower left-hand corner) enables the drawing of a symbol for each data point. This may be helpful if zoomed in such that only a few dozen or so points are visible. However, the plot becomes more difficult to read if enabled while all the data is visible.

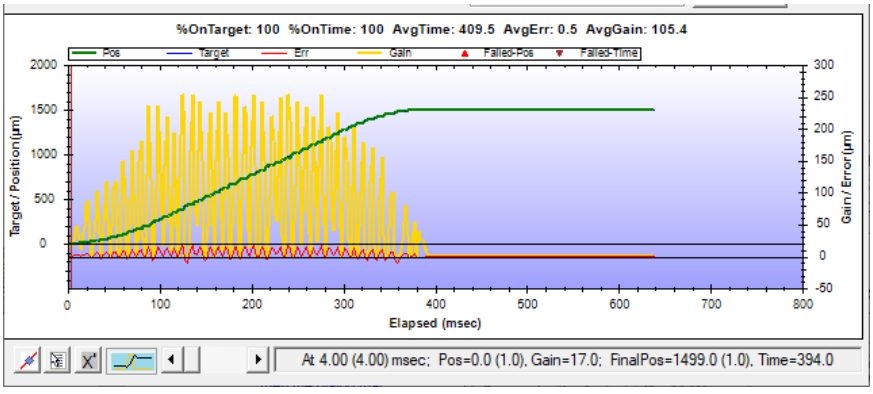
- Enable Popup Menu: Depressing the  button changes the behavior of the right mouse button. Instead of just un-zooming it will display a popup menu that allows you to save the image to a file, print the image, copy to the clip board, etc.



- Enable Instantaneous Speed: Depressing the  button changes the view to plot speed vs time instead of position vs time. Doing this with the data shown in the previous graph displays the following. Click again to go back to a position vs. time plot.



- Plot One Step at a Time: Depressing the  button changes the view to plot a single step. Then, using the scroll bar to the right, you may select which step to plot.



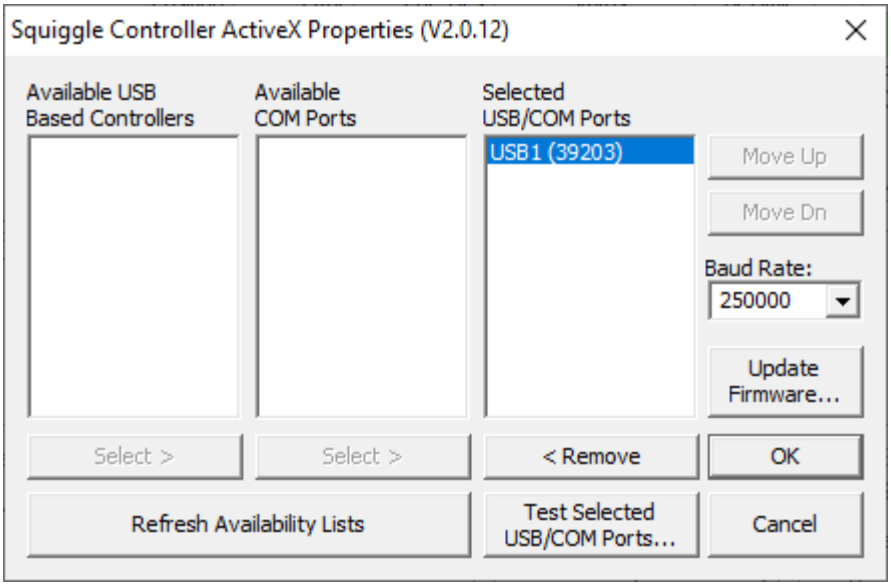
- Display Cursor: Double clicking the left mouse button over any point on the graph will position a cursor at that spot. The data at the base of the graph then displays details about that point.

Updating Firmware

This procedure describes how to update controller firmware.

To update firmware:

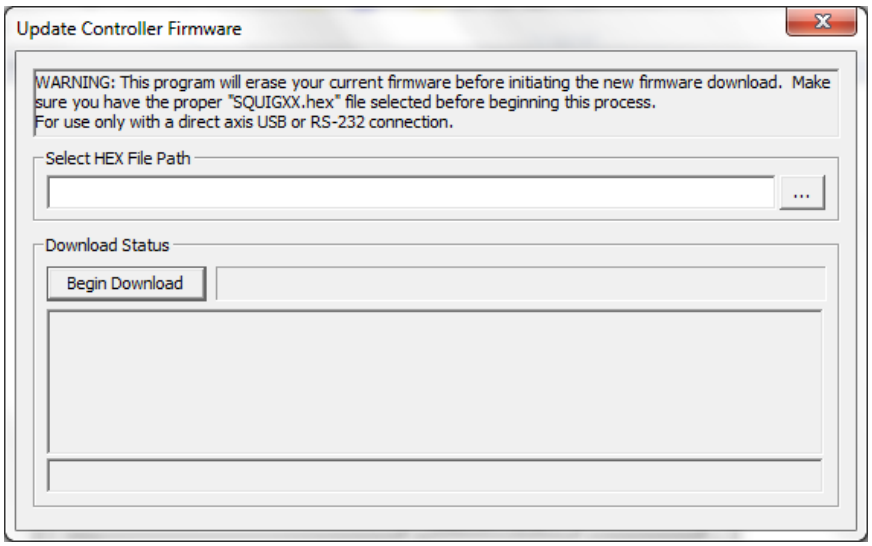
1. If necessary, click **Disconnect** in the Main window.
2. Click **Setup Connection**. The *Setup Connection* window opens.



3. Select the port connected to the controller that will be updated in the *Selected USB/COM Ports* section.
4. Click **Update Firmware**. The following dialog box opens.

If the selected USB port is an M3 USB Interface (shipped after 1/31/2012) then the dialog that appears after clicking Update Firmware… allows you to choose which device to update (i.e. the M3-x, M3 USB Interface or M3 Multiplexer [if attached]).

See next section for details.

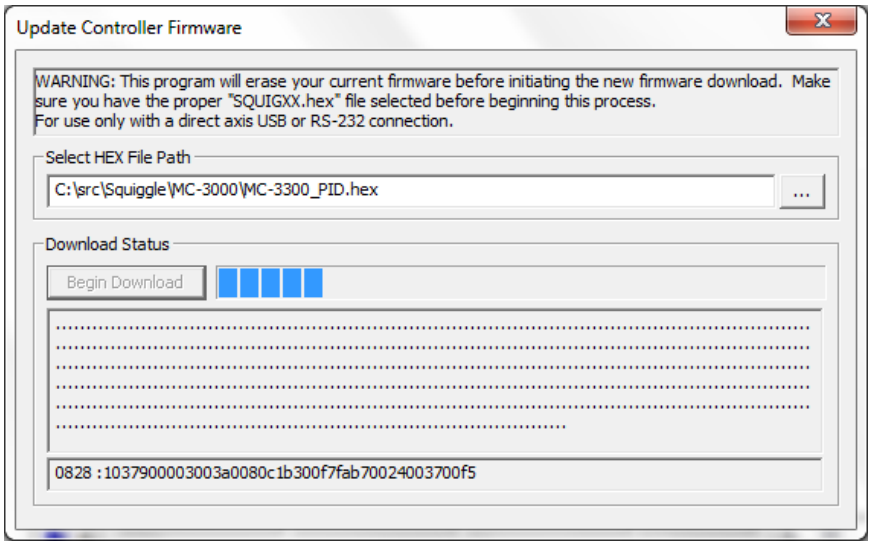


5. In the *Select HEX File Path* text box, type the path and file name of the HEX file that will be used to update the firmware.

Tip: Click the ... button to browse to the desired HEX file location.

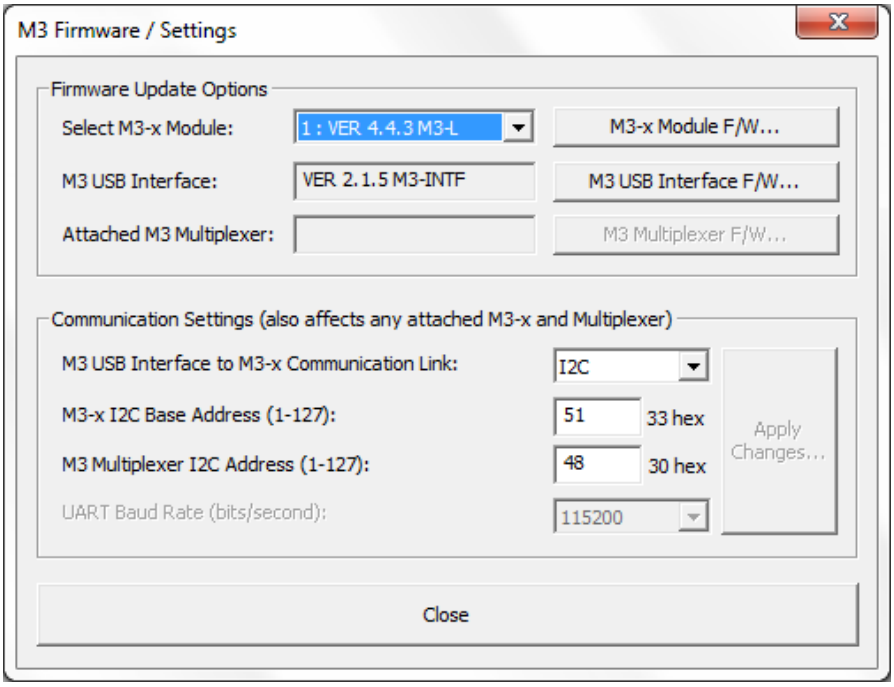
6. Click **Begin Download** to begin the file download. This begins the download process. The progress bar to the right of the Begin Download button shows the % complete while the periods in the box below reflect each file record that was sent to the controller.

WARNING: DO NOT DISCONNECT or SHUTDOWN CONTROLLER DURING DOWNLOAD!
*Any interruption, once the **Begin Download** button has been clicked, will result in the controller having to be reprogrammed at the factory.*



Difference if the Selected USB device is an M3 USB Interface

Instead of showing the **Update Controller Firmware** dialog immediately an intermediate dialog will be displayed which allows you to choose which device to update.



In most cases you would only be updating the F/W on the M3-F or M3-L attached to the M3 USB Interface. In that case you would then just click the **M3-x Module F/W…** button to bring the **Update Controller Firmware** dialog into view.

If an M3 Multiplexer is connected to the M3 USB Interface there could be up to eight M3-x devices to choose from in the **Select M3-x Module:** drop list.

The M3-x devices have always been able to communicate over either I^2C or SPI, however, the M3 USB Interfaces leave the factory configured for one or the other. Since 1/31/2012 (V2.1.5), however, the M3 USB Interfaces can be switched between I^2C or SPI. If using I^2C, the address of the M3-x can also be changed. These changes can be made using the **Communication Settings** group box (changing the I^2C address of an M3-x device F/W V4.2.6a or later).

Support for a UART connection between the M3 USB Interface and the M3-x was added later. If the M3 USB Interface has F/W V2.2.2 or later and the M3-x device has F/W V4.4.3 or later then the UART mode may be selected and a baud rate may be assigned.

Reporting Problems with Pathway S/W and/or Controller(s)

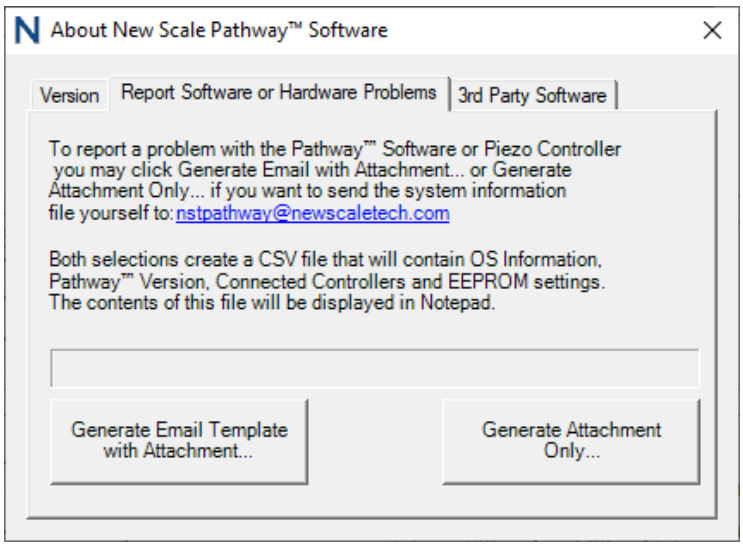
Pathway S/W version 2.6.0 or later includes a tool, via the About dialog box, that can be used to provide New Scale Technologies (NST) with current information about the Pathway software you are using as well as the configuration of the controllers connected to your PC.

The intent is to gather this information into a single text file that can be emailed to NST. Having this information is very helpful in determining the nature of the problem.

To access this tool, click the large **About…** button at the top of the Pathway window. This displays a dialog shows the current version of the software.

The second tab, **Report Software or Hardware Problems**, contains two buttons.

Please have your controllers powered and connected to the PC when clicking either of the Generate… buttons.



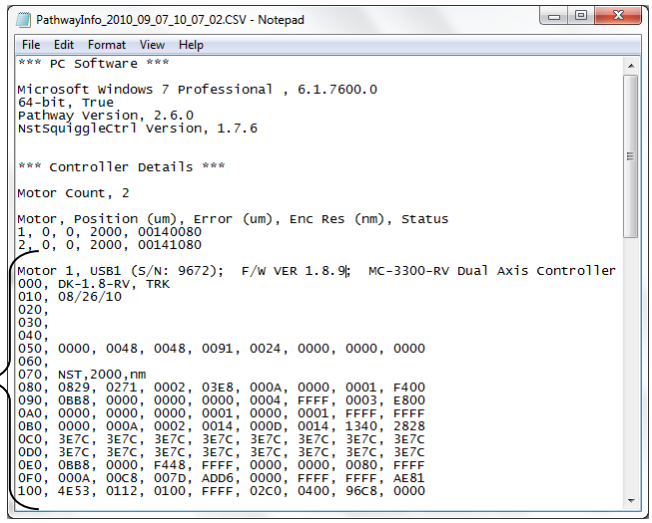
Either of the above buttons will create a CSV file in your MyDocuments directory with the name:

PathwayInfo*_yyyy_mm_dd_HH_MM_SS*.csv.

This is a text file that will contain operating system information, Pathway and NstSquiggleCtrl (our programming library) version as well as details about each controller connected to the PC. After generating the file it will be displayed to you via the Notepad editor.

This file contains no personal information!

These numbers represent the data stored in the controller's EEPROM.



If your PC has local email capability (e.g. Microsoft Outlook) you may use the **Generate Email Template with Attachment**... button to both create the text file and an email template that you may modify before clicking Send.

If you don't have email available on the PC where Pathway is running or it's not the type can be launched automatically from another program you can still copy the text file to where email is available and add it as an attachment to an email addressed to nstpathway@newscaletech.com

Appendix A: ASCII Command Reference

Overview

This appendix lists the commands used in the *Manual Command Entry* section of the **Manual Control** tab. See "Moving the motor with ASCII commands" in Chapter 2. These are the generally available commands for NST controllers. Each specific type of controller has its own command reference manual which takes precedence (though the format is always consistent with what is described herin).

Command conventions

- All commands must be preceded and ended with **<** and **>** characters (for example, **<01>**).
- All commands are terminated with a carriage return character.
- Brackets indicate optional values and are not entered with the commands. For example, the command **<04 D [TTTT]>** could be entered: **<04 1>** or **<04 1 1234>** where **1** represents **D** and **1234** represents **TTTT**.
- Commands use 7-bit ASCII characters only.
- All hexadecimal (hex) numbers sent or received use only capital letters (e.g., 0xF4 = F4).

Note: Command **<01>** must be used to establish computer control before entering other motion control commands in this appendix.

Using the command integrity prefix (optional)

The command format described in remainder of this appendix does not verify command integrity. To do so, use the 7-byte command prefix described in this section.

The command prefix includes a checksum, command count and command length, and allows the controller to verify that the command has not been corrupted or already processed. The controller's reply includes a 7-byte prefix so that the host PC can evaluate the reply and re-transmit if necessary.

Note: The command prefix is used automatically when the NstSquiggleCtrl ActiveX control is talking to any NST Squiggle Motor Controller.

Enabling and disabling the command prefix

Use the commands that follow to enable or disable prefix processing. The **$** character represents the **ESC** character (1B hex, 27 decimal).

Send to controller	Receive from controller	Usage
`$[0`	No response	Disables prefix processing
`$[1`	No response	Enables prefix processing **Note**: *Prefix processing is enabled implicitly if a standard command is received that includes a prefix*.
`$[2`	Previous command reply	Re-transmits the last reply

When the command prefix is enabled, the controller returns a **NAK** character (15 hex, 21 decimal) under these circumstances:

- An end of command carriage return character (0D hex, 13 decimal) is received *AND*
- The command length or checksum in the prefix does not match *OR*
- A prefix was not sent

Otherwise, the command will be processed, and the reply returned with the appropriate prefix.

Command prefix syntax

The command prefix uses the following syntax.

Command syntax	Usage
`$CSCCCL[command]/CR`	• **$** is the **ESC** character • **CS** is a 2-byte checksum in hex. The checksum includes the command count, the command length (hex characters) and the command itself but does not include the carriage return terminator. • **CC** is a 2-byte command count in hex. This should be incremented for each command sent (ActiveX control does this automatically). **A CC value of 00 (zero) is not supported!** • **CL** is a 2-byte command length in hex (the number of characters in the command itself). • **/CR**is the carriage return character.

Example

To…	Send to controller	Receive from controller
Send the 54th command to the controller, requesting a move to target where the target position is 250 µm from the encoder zero position (assuming a 20 nm encoder).	`$7A360D<08 000030D4>/CR`	`$AF3604<08>/CR` Notice that the command number (36 hex, 54 decimal) is echoed in the reply prefix.

01: Establishing computer control

This command establishes computer control. In the MC-1x00 and SQ-2x00 controllers it also locks out the analog velocity servo input).

Send to controller	Receive from controller	Usage
`<01>`	`<01 A VER X.X.X>`	• If **A = 1**, a motor is present. • If **A = 0**, no motor is present.

Send to controller	Receive from controller	Usage
		• **VER X.X.X** is the controller firmware version.

02: Releasing computer control

This command releases computer control and allows the analog servo control mode of the MC-1x00 or SQ-2x00 controllers to be used.

Send to controller	Receive from controller
<02>	<02>

03: Halting the motor

This command halts motor motion regardless of where the movement command was issued.

Send to controller	Receive from controller
<03>	<03>

04: Running the motor

This command runs the motor. The motor will continue to move until a command <**03>** is received or until an optional time value elapses.

Note: Use this command only after establishing computer control with command <**01>**.

Send to controller	Receive from controller	Usage
<04 D [TTTT]>	<04>	• If **D = 1**, motor runs forward. • If **D = 0**, motor runs in reverse. • **TTTT** is an optional run duration, entered in 0.1-second units.*

* Hex format.

Examples

To…	Send to controller	Receive from controller
Run the motor forward	<04 1>	<04>
Run the motor in reverse for two seconds	<04 0 0014>	<04>

05: Moving the motor in timed steps

*Use command <**20**> on page 85 to select closed-loop mode.*

This command sends one or more bursts of resonant pulses to the motor at 100Hz (every 10msec) or at the period indicated by the **PPPP** parameter.

Note: For controllers that support amplitude mode, command <**09>** on page 79 still applies even while in closed-loop mode.

Send to controller	Receive from controller	Usage
<05 [D [SSSS PPPP TTTT]>	<05>	• If **D = 1**, motor runs forward. • If **D = 0**, motor runs in reverse. • **SSSS** is an optional number of steps to take.* • **PPPP** is an optional step interval*. The step interval is the time between the start of each step. • **TTTT** is an optional step duration*. The step duration is the time that the resonant pulses are active, which must be less than the step interval.

* 4-digit hex format. Use command <52> to query the time units. For the M3-L and M3-F/FS the units are 4 µsec, For the M3-LS-1.8 it's 1.6 µsec. For the MC-3300 it's 3.2 µsec though for those with f/w versions > 2.0.0, it may be lower.

Example

To…	Send to controller	Receive from controller
Run the motor 100 steps forward with an interval of 20 msec and a duration of 10 msec per step	<05 1 0064 186A 0C35>	<05>

To…	Send to controller	Receive from controller
(assumes interval of 3.2 µsec)		

06: Moving the motor using closed-loop steps

*Use command <**40**> on page 86 to set the closed-loop speed.*

This command adds or subtracts the specified step size (in encoder counts) to the current target position, and then moves the motor to the new target at the previously defined speed.

Note: This command will return an illegal command error (24) if issued while the axis (motor) is in open-loop mode. See command <**20>** on page 85 to select closed-loop mode.

Send to controller	Receive from controller	Usage
<06 D [SSSSSSSS]>	<06>	• If **D = 1**, motor runs forward. • If **D = 0**, motor runs in reverse. • Set **D** to **N** to set the step size without moving the actuator. • **SSSSSSSS** is the number of encoder counts to advance/backup from the current target position.* If not specified, the previously stored step size will be used.

* Hex format.

Example

To…	Send to controller	Receive from controller
Run the motor forward 100 encoder counts from the target position	<06 1 00000064>	<06>

07: Clearing the encoder count

This command clears the encoder count register, thereby setting the current position as the zero position.

Send to controller	Receive from controller

<07>	<07>

08: Moving to target

This command:

- Sets a target position.
- Moves the motor to the target position at the speed defined by command <**40**> (see page 85).

Note: This command will return an illegal command error **<24>** if issued while the axis (motor) is in open-loop mode. See command <**20>** on page 85 to select closed-loop mode.

Send to controller	Receive from controller	Usage
<08 TTTTTTTT>	<08>	• **TTTTTTTT** is the target position in encoder counts.*

* Hex format.

Example

To…	Send to controller	Receive from controller
Sets a new target and moves to position -1000. (assuming 0.5 um per encoder count)	<08 FFFFF830>	<08>

09: Setting the open-loop speed

*Use command <**20**> on page 85 to select open-loop mode.*

This command sets the open-loop speed of the motor.

Note: In closed-loop mode, this command applies only to the motor step command **<05>** on page 77. In closed-loop mode, motor drive gain is normally based on the requested velocity, but the open-loop speed still applies for controllers that support amplitude mode when using the motor step, which is strictly an open-loop command.

Send to controller	Receive from controller	Usage
<09 EE>	<09>	• **EE** is the speed command of the value 1 to 256 (0 x 01 to 0 x FF) where the speed is the ratio of (EE / 256) x 100%.* **Tip**: Use command <**51>**on page 92 to

Send to controller	Receive from controller	Usage
		automatically set the reverse speed to a fraction or multiple of the forward speed. This is useful to maintain similar forward or reverse speeds when there is a significant load difference between directions.

* Hex format.

Example

To…	Send to controller	Receive from controller
Set the speed to (0 x 80 / 0 x FF) x 100% == (128 / 256) x 100% = 50%	<09 80>	<09>

10: Viewing closed-loop status and position

This command is used to view the motor status and position.

Send to controller	Receive from controller	Usage
<10>	<10 SSSSSS PPPPPPPP EEEEEEEE>	• **SSSSSS** is the motor status.* See "Motor status values," next. • **PPPPPPPP** is the absolute position in encoder counts.** • **EEEEEEEE** is the position error in encoder counts.** This value is the distance between where the motor is located and where it is supposed to be located.

* 6-digit hex format, 24-bit integer.

** 8-digit hex format, 32-bit integer.

Motor status values

Bit	Description	Values
0	Reserved	N/A
1	Motor direction	0 = Reverse

Bit	Description	Values
		1 = Forward
2	Running	1 = Motor is running
3	Motor interlock	1 = Motor is disconnected or, in the case of an M3-x, the driver chip is not responding.
4	Numbered Burst mode	1 = Command 05 in progress which uses a command specified number of motor bursts.
5	Timed run	1 = Timed free run in progress
6	Multiplexed axis	1 = Multiplexed axis (e.g., SQ-2306, 2206)
7	Controller status	1 = Under computer control (analog servo control, if supported, is not available)
8	Reserved	N/A
9	Forward limit	1 = Forward travel limit reached
10	Reverse limit	1 = Reverse travel limit reached
11	Motor burst or amplitude mode	0 = Amplitude mode (always used in closed-loop mode. 1 = Burst mode (200 Hz)
12 - 15	Reserved	N/A
16	Encoder count error	1 = An error was detected in the encoder quadrature signal. Cleared by sending command <**07>** on page 78.
17	Zero reference enabled	1 = Encoder zero reference mark detection is enabled (see command <**42>** on page 88)
18	Motor on target	1 = Encoder position error is within the target tolerance.
19	Motor moving toward target	1 = Motor is moving toward a target position; appears after command <**08>** or move step command <06>. Once the target is reached, bit 19 is set to zero.
20	Maintenance mode enabled	1 = Controller will actively hold the last target position **Note**: If bits 20 and 21 are set 1 and bit 18 is set to 0, the controller is in the

Bit	Description	Values
		process of moving back toward the last targeted position.
21	Closed loop enabled	1 = Motion commands use the encoder for feedback
22	Motor accelerating	1 = The motor is accelerating to the desired velocity (set at the start of closed-loop motion) 0 = Required motor speed is reached, motor is decelerating, or motor is stopped
23	Stalled	1 = The position error exceeds the stall detection threshold while the motor is moving

11: Writing a hex byte to the motor EEPROM

This command writes a hex byte to the motor EEPROM. You must have permissions enabled to use this command.

Caution: Using this command can damage the motor or affect its performance.

Send to controller	Receive from controller	Usage
<11 XXX YY>	<11>	• **XXX** is the hex address in EEPROM. • **YY** is the hex number.

12: Reading a hex byte from the motor EEPROM

This command reads a hex byte from the motor EEPROM.

Send to controller	Receive from controller	Usage
<12 XXX>	<12 YY>	• **XXX** is the hex address in the controller EEPROM. • **YY** is the hex number returned.

13: Writing ASCII to the motor EEPROM

This command writes an ASCII string to the motor EEPROM. You must have permissions enabled to use this command.

Caution: Using this command can damage the motor or affect its performance.

Send to controller	Receive from controller	Usage
<13 XXX Y…Y>	<13>	• **XXX** is the hex address in the controller EEPROM. • **Y…Y** is the ASCII string (up to 16 characters).

14: Reading ASCII from the motor EEPROM

This command reads an ASCII string from the motor EEPROM.

Send to controller	Receive from controller	Usage
<14 XXX ZZ>	<14 Y…Y>	• **XXX** is the hex address in the controller EEPROM. • **ZZ** is the hex number, from 0 to 15, that coincides with the number of ASCII characters returned. • **Y…Y** is the ASCII string returned (up to 16 characters).

15: Updating firmware

This command erases the controller's internal flash memory to prepare for a new firmware download (hex source code). ***It is recommended that you use the Demo software to control proper download timing.***

Caution: If this procedure is interrupted, a partial or corrupt firmware installation may result. The latest firmware can **only** be obtained from New Scale Technologies.

Send to controller	Receive from controller	Usage
<15>	See Usage	After controller flash memory is erased, the following response will be sent: **Send new .HEX file now**. Use the Demo software or a terminal emulation program with ASCII text file download control to send each line of the hex file with at least a 10 msec delay after each line. The controller will return a period (.) after it processes each line.

18: Selecting or toggling the multiplexer controller axis (motor)

This command selects or toggles the selected axis (motor) for the multiplexer controller (e.g., SQ-2306). However, it may also be used for axis selection in the MC-3100.

Note: The front panel axis indicators on the SQ-2306 show the axis (motor) selected.

Send to controller	Receive from controller	Usage
<18 X>	<18 X>	• If **X = I**, will increment to the next available axis (motor). • If **X = D**, will decrement to the next available axis (motor). • If **X = R**, the controller will report the currently selected axis (motor) without changing it. • If **X = 1, 2, 3, 4, 5 or 6**, the controller will try to select the axis (motor) number indicated. If that axis does not contain a motor, the controller will revert back to the previously selected

Send to controller	Receive from controller	Usage
		axis.

19: Reading motor flags from the controller

This command reports internal flags used by the controller to monitor motor conditions.

Send to controller	Receive from controller	Usage
<19>	<19 XXXX>	**XXXX** is the status flags returned. See "Motor status values" on page 80 (first 16 bits only).

* 16-bit hex format.

20: Viewing and selecting the open- or closed-loop drive mode

This command sets the drive mode for the current axis (motor).

Send to controller	Receive from controller	Usage
<20 X>	<20 X>	• If **X = 0**, open-loop drive mode is selected. • If **X = 1**, closed-loop drive mode is selected. • If **X = R**, the controller will report the current drive mode. **Notes**: • The open-loop motion commands (<**03>**, <**04>** and <**05>** on pages 76 to 77) will work in closed-loop mode. • Command <**10>** on page 80 will continue to work in open-loop mode.

23: Illegal command format

If the controller returns **<23>**, the command was improperly formatted. Verify that the proper begin and end terminating characters are present.

24: Illegal command

If the controller returns **<24>**, an illegal command was performed. This occurs if you attempt a run command before establishing computer control or if a user is not authorized to access a feature.

40: Setting the closed-loop speed

This command sets the closed-loop speed of the current axis (motor).

Note: This command will return an illegal command error **<24>** if issued while the axis is in open-loop mode. See command <**20>** on page 85 to select closed-loop mode.

Send to controller	Receive from controller	Usage
<40 SSSSSS CCCCCC AAAAAA IIII>	<40>	• **SSSSSS** is the speed of the axis (motor) in encoder counts per interval.* • **CCCCCC** is the cutoff (minimum) speed of the axis (motor) in encoder counts per interval.* • **AAAAAA** is the acceleration/deceleration of the axis (motor) in encoder counts per interval.* • **IIII** is the interval duration in closed-loop interval units.**

* 6-digit hex format. The last two hex digits are a fractional encoder count.

** 4-digi hex format.

Example

To…	Send to controller	Receive from controller
Set the speed to 1000 µm/sec with a cutoff speed of 10 µm/sec and an acceleration of 4000 µm/sec, assuming a 20 nm encoder and a 500 µsec closed-loop interval. **Explanation**: 001900 hex translates to 6400/256 or 25 encoder counts per interval. Since the duration of each interval was set to 1, that's 50000 encoder counts/sec. A 20 nm encoder translates to 1000 µm/sec.	<40 001900 000040 00000D 0001>	<40>

41: Setting position error thresholds and stall detection

This command is used to enable stalled motor detection and set position thresholds for the current axis (motor).

Send to controller	Receive from controller	Usage
<41 X EEEEEE IIIIII>	<41 X EEEEEE IIIIII>	• If **X = 0**, stalled motor will not be detected. • If **X = 1**, stalled motor will be detected. • If **X = R**, the controller will report the current stall mode. • **EEEEEE** is an optional stall position error threshold in encoder counts.* This value sets the maximum allowable difference between where the motor is currently located and where it is supposed to be located. • **IIIIII** is an optional stall incremental error threshold.* This value is used at the end of a move and during position hold and is the sum of encoder errors and the incremental coefficient. **Tip**: Use this command when the motor is close to the required target (within position error threshold) but is unable to reach the target.

* 16-digit hex format.

Example

To…	Send to controller	Receive from controller
Enable stall detection with a position error threshold of 100 encoder counts	<41 1 000064>	<41 1 000064 0003E8>

42: Detecting the zero-reference mark

This command is used to enable detection of the zero-reference mark. This state remains enabled until the reference mark is detected or power to the controller is cycled.

When the reference mark is detected, the encoder count is zeroed and the motor is stopped.

Note: This command will return an illegal command error <**24**> if issued while the axis (motor) is in open-loop mode. See command <**20>** on page 85 to select closed-loop mode.

Send to controller	Receive from controller	Usage
<42 X>	<42>	• If **X = 0**, the zero reference mark will not be detected. • If **X = 1 or is not specified**, the zero reference mark will be detected.

43: Viewing and setting closed-loop PID coefficients in motor EEPROM

This command is used to view and set the closed-loop PID coefficients in the motor's EEPROM.

Closed-loop PID coefficients are used in closed-loop mode to adjust the motor drive signal gain based on how far the motor is from where it is supposed to be.

Send to controller	Receive from controller	Usage
<43 PPPP IIII DDDD>	<43 PPPP IIII DDDD>	To view the current PID coefficients, use this command without any parameters. The following parameters are optional: • **PPPP** = P coefficient.* • **IIII** = I coefficient.* • **DDDD** = D coefficient (typically unused).

* 4-digit hex value (16-bit).

Example

To…	Send to controller	Receive from controller

To…	Send to controller	Receive from controller
View the PID coefficients The coefficients in this example have the following decimal values: P = 32, I = 16, D = 1.	<43>	<43 0020 0010 0001>

44: Viewing and setting closed-loop encoder details

This command is used to view and set closed-loop control encoder details. These details are used by the host PC when it converts distance values into encoder units.

Send to controller	Receive from controller	Usage
<44 Vendor,Res,Units>	<44 Vendor,Res,Units>	To view the current details, use this command without any parameters. The maximum length of the Vendor,Res,Units string is 15 characters. • **Vendor** is the name of the encoder vendor. This value can be but the comma-delimiter is required. • **Res** is the decimal encoder resolution. • **Units** is the encoder resolution units (up to 15 characters).

Example

To…	Send to controller	Receive from controller
View encoder setup parameters	<44>	<44 MicroE,20,nm>
Set encoder resolution to 50 nm (0.05 um) using a MicroE encoder.	<44 MicroE,50,nm>	<44 MicroE,50,nm>

45: Viewing and setting the minimum closed-loop drive amplitude

This command is used to view and set the minimum relative motor drive gain in closed-loop mode.

Piezo motors often have some stiction that must be overcome before the shaft will move at all. The PID loop will start with the specified relative drive (e.g., amplitude or burst duration) for small position errors.

Send to controller	Receive from controller	Usage
<45 XX>	<45 XX>	• **XX** is the minimum gain.*

* 00 to FF (0 to 255 decimal).

Example

To…	Send to controller	Receive from controller
Set the minimum gain to 10 (decimal) or 3.92% (10/255).	<45 0A>	<45 0A>

46: Viewing and setting forward and reverse travel limits

*To activate travel limits, use command **<47>**, next.*

This command is used to view and set motor travel limits for motors or stages (typically a spring-loaded slide pushed by the motor) equipped with an encoder. Using travel limits eliminates the need for limit switches.

Once the motor/stage reaches or exceeds the travel limit, it will be stopped and the appropriate limit flag will be set. For incremental (as opposed to absolute) encoders, the reference or index mark must first be found before the limits will be used.

Send to controller	Receive from controller	Usage
<46 FFFFFFFF RRRRRRRR WWWW>	<46 FFFFFFFF RRRRRRRR WWWW>	• **FFFFFFFF** is optional the forward limit, in encoder units*. • **RRRRRRRR** is the optional reverse limit, in encoder units.*. • **WWWW** is the optional distance (in encoder units) before each travel limit position at which the limit flag will be

		remain asserted once it has been tripped.** This value appears on the Limit Active window.

* Signed 32-bit hex format.

** Unsigned 16-bit hex format.

Example

To…	Send to controller	Receive from controller
Set the forward and reverse travel limits to 6500 m and -6500 µm, assuming a 20 nm encoder. This also sets the Limit Active window to 1 µm.	<46 0004F588 FFFB0A78 0032>	<46 0004F588 FFFB0A78 0032>

47: Viewing, activating and deactivating forward and reverse travel limits

*To define travel limits, use command <**46**>, above.*

This command is used to view, activate and deactivate motor travel limits.

Note: Activated travel limits will only be used if both closed-loop control (command **<20>** on page 85) is selected and the encoder reference mark has been found (use command <**42>** to enable reference mark detection). However, if an analog position sensor is being used, no reference mark is required.

Send to controller	Receive from controller	Usage
<47 X>	<47 X>	• If **X = 0**, travel limits are not activated. • If **X = 1**, travel limits are activated.

Example

To…	Send to controller	Receive from controller
Query the current travel limits enabled flag.	<47>	<47 1>

50: Viewing and activating motor burst or amplitude mode

This command is used to view current settings and activate amplitude or burst mode for the motor of the current axis (motor).

- **Burst mode** has a more linear speed range and may overcome stiction more easily than amplitude mode.
- **Amplitude mode** is quieter and provides for smoother motion. Amplitude mode is a continuous drive signal where, depending on the controller type, either the voltage or the duty cycle of each motor pulse is adjusted according to the specified speed.

Note: The MC-3100 does not support amplitude mode.

This mode is stored on the motor's EEPROM so that it will be restored at power-up.

Send to controller	Receive from controller	Usage
<50 X>	<50 X>	• If **X = 0**, burst mode is selected (100 Hz). • If **X = 1**, amplitude mode is selected. • If **X = R**, the current mode is returned.

51: Viewing and setting the forward and reverse speed ratio

*Use command **<09>** on page 79 to set the actual forward speed.*

This command is used to view and set the forward and reverse speed ratio for the current axis (motor).

Reverse speed is calculated by multiplying the forward speed by the speed ratio. For example, if the speed ratio is 1.0, the reverse speed will be the same as the forward speed. If the ratio is 0.5, the reverse speed will be half of the forward speed.

Send to controller	Receive from controller	Usage
<51 XXXX>	<51 XXXX>	• **XXXX** is the speed ratio.* If no value is specified, the current mode is returned. **Note**: To create the integer fraction, multiply the desired ratio by 256. Then convert the result to a 4-digit hex value. See "Integer fraction examples" next.

* Hex 8.8 integer fraction format.

Integer fraction examples

To create this decimal ratio…	Use this speed ratio value
0.25	0040
0.333	0055
0.50	0080
0.75	00C0
1.00	0100
1.25	0140

52: Viewing time interval units

This command is used to view controller time interval units, which are applied to timed step command values.

Send to controller	Receive from controller	Usage
<52>	<52 XXX UUUU>	• **XXX** is the interval value. • **UUUU** is the interval units (e.g., a reply of <52 3.2 µsec> means that each controller timer interval is in 3.2 µsec units, so a period of 2 msec would be 625).

54: Viewing and setting the port baud rate

This command is used to view and set the port baud rate used by the controller when the controller is reset. It is not recommended that this be used on the MC-3100 for host communication, as the baud rate is assumed by the host software to be 250000.

Note: With the SQ-2301M handset, set the baud rate of the RS485 port to 19200.

Send to controller	Receive from controller	Usage
<54 P XX>	<54 P XX>	• If **P = 1**, the RS232/USBport is selected. • If **P = 2**, the RS485 port is selected. • **XX** is the baud rate of the selected port (optional). Use these values:

		– 00 = 19200 baud – 01 = 38400 baud – 02 = 57600 baud – 03 = 115200 baud – 04 = 250000 baud

Examples

To…	Send to controller	Receive from controller
View the baud rate to be used by the RS-232 on reset. The returned value of 00 indicates 19200 baud.	<54 1>	<54 1 00>
Set the baud rate of the RS-232 port to 115200 (applied at next power-up).	<54 1 03>	<54 1 03>

87: Viewing and Starting Motor Freq. Calibration

For controllers equipped with the NSD-2101 piezo driver chip, this command is used to view and start a frequency calibration of the motor using the 2101's internal calibration feature. A calibrated motor provides the maximum speed and push force.

As stated in the NSD-2101 datasheet there are two types of frequency calibrations available: Sweep and Incremental.

The sweep calibration steps through a wide range of frequencies centered on centered on the frequency loaded into the period register (e.g. ~172 kHz or a period value of 145). The frequency that generates the highest performance is selected. The incremental starts at the current frequency and uses a hill climbing method (stepping the frequency in the direction of increasing performance). The sweep is a coarse calibration. The incremental is a fine calibration.

It is recommended that a sweep be performed first, followed by an incremental.

Note: See the NSD-2101 datasheet for more information about the frequency calibration types (sweep, incremental) and the use of the offset register.

Send to controller	Receive from controller	Usage
<87[D OO FFFF]>	<87 D OO FFFF>	• **D** is direction and calibration type. It is recommended that calibrations be performed against the load (typically forward). ○ 0 = Sweep in Reverse direction. ○ 1 = Sweep in Forward direction. ○ 2 = Incremental in Reverse direction. ○ 3 = Incremental in Forward direction. ○ 4 = Sweep/Incremental in Reverse direction. ○ 5 = Sweep/Incremental in Forward direction. • **OO** is the offset register value (in hex). A value of 70 is recommended for the SQL-RV-1.8 motor. A value of 75 is recommended for a UTAF since the best operating frequency for the UTAF is 5 steps below the resonant frequency of phase 1. • **FFFF** are the calibration flags. • Bit 0 reserved • Bit 1 Phase 1 only (suppress phase 2) • Bit 2 Phase 2 only (suppress phase 1) • Bits 3-15 reserved It is recommended that both phases be used for an SQL-RV-1.8 motor. The UTAF motor requires that phase 2 be suppressed.

The offset register is not fully described here. Only the recommended values are stated. For more information about the offset register, please refer to the NSD-2101 datasheet.

Examples

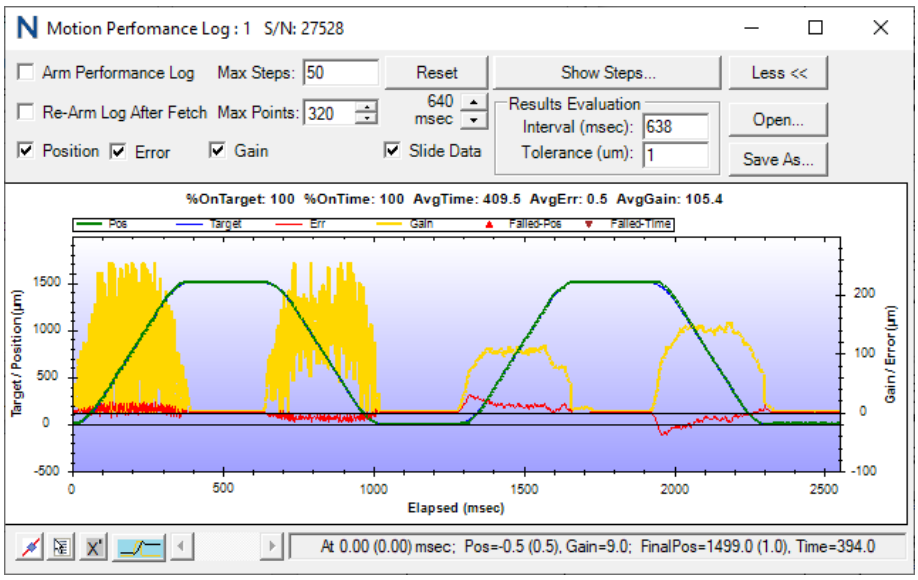
To…	Send to controller	Receive from controller
Query frequency calibration status (in this example, just after a power cycle)	<87>	<87 0 00 0000>

To…	Send to controller	Receive from controller
The offset register being zero indicates no calibration has been performed.		
To calibrate an SQL-RV-1.8 motor No freq. offset. No phases are suppressed	<87 5 70 0000>	<87 5 70 0000>
To calibrate a UTAF motor Freq. offset of 5 steps. Phase 2 is suppressed (calibrates using Phase 1 only).	<87 5 75 0002>	<87 5 75 0002>
Query frequency calibration status (in this example, just after a full calibration of an SQL-RV-1.8 motor).	<87>	<87 5 70 0000>

A9, AA: Viewing and Starting a Performance Log

To make it easier to assess the step response of the motor a position log feature has been added to some of NST's motor controllers (e.g. M3-F, M3-L [V4.2.6a or later]; MC-3300 [V1.9.8 or later]; MC-1100 [V2.2.4 or later]).

From the Pathway Software (beta version 2.7.3x9 or later) this log feature of those motor controllers can be configured and the results may be displayed. The following figure illustrates two cycles of full travel motion on an M3-F. In the first cycle, the KP coefficient was 400. In the second cycle KP was 150. In the latter case there is less variation in the motor gain though the following error increases. If there are spots of high friction and/or load, they would be clearly visible in this plot.



Plot of Multiple Position Log Queries

The following table lists the commands used to query status, arm/disarm and query results from the position log.

Send to controller	Receive from controller	Usage
<A9>	<A9 X CCCC AAAA s>	*To Query Log Status* X 0=Unarmed, 1=Armed (and will start recording at start of next move) or armed and logging in progress (see CCCC) CCCC 4 digit hexadecimal number of data points collected so far (0000 on power-up) AAAA 4 digit hexadecimal number of possible data points (i.e. the size of the array) s 1 digit number (0-9) indicating the number of closed loop intervals to skip (i.e. to not record a position value) between each recorded value. This is used to extend the time over which the log is active. The parameter is shown in blue because it was not in the initial f/w implementations. If it does not appear in the response to <A9> it is not supported by the controller. See command <A9 S s> below to set the skip value. If armed, it will switch to disarmed when the maximum possible data points are collected. In the case of the M3-F that will be 640 msec after collection starts because its

		array size is 320 (0140 hex) and each closed loop interval is 2 msec. In future versions the maximum number of data points may increase so it is important to use the value (represented by AAAA) returned by the controller.
<A9 1>	<A9 1 CCCC AAAA s>	*To Arm Position Log* Once armed, the controller will start recording position, target and motor gain data at the start of the next move.
<A9 0>	<A9 1 CCCC AAAA s>	*To Disarm Position Log*
<A9 S s>	<A9 1 CCCC AAAA s>	*To Set Skip Value* s is the required 1 digit skip value (0-9). The default value is zero which means, for the M3-F, the log duration is 640 msec (320 x 2msec/CL interval). The max value of 9 would make the log duration 6400 msec (320 x 10 x 2 msec) since only 1 point out of every 10 is being logged.
<AA 0 OOOO>	<AA 0 OOOO DDDD DDDD … DDDD>	*To Query Position Data* OOOO is the 4 hex digit array offset. Since only 16 points are returned at a time, the offset would be incremented by 16 (10 hex) until all data points have been returned. DDDD is the 4 hex digit position value at that time interval (in encoder counts).
<AA 1 OOOO>	<AA 1 OOOO GG GG GG GG … GG>	*To Query Motor Gain Data* OOOO same as above. GG is the 2 hex digit motor gain value generated at that time interval (00 = off, FF = max gain).
<AA 2 OOOO>	<AA 2 OOOO EEEE EEEE … EEEE>	*To Query Position Error Data* OOOO same as above. EEEE is the 4 hex digit position error value calculated that time interval (where Error = SetPoint – Position).