User Guide

Cavro Fusion

Evaluation Software for Tecan Cavro Components

April 2014

20738568-G

This page has been left intentionally blank.

User Guide

Cavro Fusion

Evaluation Software for Tecan Cavro Components

Tecan Systems, Inc.
2450 Zanker Road
San Jose, CA 95131 USA
T 1 408 953 3100, Toll Free 1 800 231 0711
F 1 408 953 3101
E-mail: tecansystemsinfo@tecan.com
Web Site: www.tecansystems.com

April 2014

20738568-G

Part Number 20738568-G

Copyright and Trademark Information

Product Warranty Information

Tecan Systems warrants that robots manufactured and sold by Tecan Systems will be free from defects in materials and workmanship for a period of twenty-four (24) months from the date of shipment to customer. Tecan Systems' liability for the breach of the foregoing warranty is limited to the repair or replacement of the products found to be other than warranted. Such products will be accepted for return only if the customer returns them to Tecan Systems' factory or repair depot within thirty (30) days from the time of discovery of the alleged defect, and prior to return, fills out a Certificate of Decontamination (document P/N 20730171), obtains a return authorization number from Tecan Systems, provides Tecan Systems with the serial number of each robot to be returned, and prepays freight charges to the factory or a designated Tecan Systems repair depot. No warranty is expressed or implied for:

- Breakage
- Maltreatment
- Unauthorized service
- Units not returned in original or adequate packaging
- Units which are "life-cycled"
- Syringes
- Syringe seals
- Tubing and tubing connections
- Cavro valves
- Cavro probes

The foregoing warranties and limitations are customer's exclusive remedies and are in lieu of all other warranties, express or implied, including without limitation any warranty of merchantability or fitness for a particular purpose.

Product Documentation Warranty Information

The information contained in this document is subject to change without notice. Tecan Systems makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose.

Tecan Systems shall not be held liable for errors contained in this document or for incidental or consequential damages in connection with the furnishing, performance, or use of this material.

Table of Contents

This page has been left intentionally blank.

1 Getting Started

1.1 About This Guide

1.1.1 Introduction

This user guide is designed to describe the installation and operation of the Cavro® Fusion software. In this reference, the assumption is made that the reader is technically proficient in electrical, mechanical, and software engineering.

All questions about Cavro Fusion and this reference should be directed to a Tecan customer support representative.

1.1.2 Warnings, Cautions, and Notes

There are three informational notices used in this reference. These notices are designed to highlight important information or to warn the user of a potentially dangerous situation.

WARNING! *Indicates a possibility of severe personal injury, loss of life, or equipment damage if instructions are not followed.*

Caution! *Indicates a possibility of equipment damage if instructions are not followed.*

Note: *Gives helpful information about Cavro Fusion.*

1.1.3 Abbreviations, Symbols, and Measurements

Both the SI (System International), and English Standard abbreviations/ measurements are used. However, the “English Standard” measurements have been converted from SI.

1.2 Introduction to Cavro Fusion

Cavro Fusion is an evaluation software tool to provide customers a simple way to demonstrate, evaluate, and troubleshoot Tecan Cavro liquid handling components using any Windows-based PC. Any other use is considered improper and may result in damage to the products and/or unreliable test results.

1.2.1 Demo Mode

Cavro Fusion provides a graphical user interface (GUI) that allows you to manipulate robotic arm modules, diluters, and pumps by dragging and dropping the moveable parts of the module in three dimensions. The GUI is used as a tool to demonstrate the capabilities of a device.

1.2.2 Script Mode

Cavro Fusion also provides simple scripting capabilities. You can write scripts to operate any connected modules, and use the scripts to automate your tasks and operate the module without user intervention.

Cavro Fusion reads scripts from text files, so you can write a script in any text editor and then use it in Cavro Fusion to operate modules.

1.2.3 Supported Modules

This version of Cavro Fusion supports the following modules:

- Cavro® Centris, XLP, and XC syringe pumps (diluters)
- XP, XL, XE, and XMP syringe pumps (limited support)
- Smart Valves
- Smart Peristaltic pumps
- Linear Option modules
- Input/Output boards
- RSP, one and two arms
- MSP, one and two arms
- Cavro® Omni Robot, single or dual arms
- Cavro® ADP

1.2.4 Supported Omni Versions

Table 1-1 shows supported Omni versions and their corresponding CIB firmware versions, revisions, and axis control program or TML script version.

Table 1-1 Supported Omni Versions

Omni Version	CIB Revision	CIB Version	TML version and axis type
V2.0	30046329 Rev-A	V2.1.3	(0102) (0202) (0302)
V2.2	30046329 Rev-C	V2.4.1	(0103) (0203) (0303) (0801) (0901)
V2.2.2	30046329 Rev-C	V2.4.1	(0103) (0203) (0303) (0801) (0901)
V3.0	30071193 Rev-A	V3.0.1	(0104) (0204) (0304) (0802) (0902)
V3.1	30089476 Rev-A	V3.0.2	(0104) (0204) (0304) (0802) (0902)
V4.0	30087595 Rev-A	V4.0.1	(0105) (0205) (0305) (0803) (0903)

Note: *This version of Fusion supports multiple versions of Omni. Please refer to the* Omni Operator's Manual *for version-specific features.*

1.3 System Requirements and Prerequisites

Before using the Cavro Fusion software, you must set up your robotic and/or pump modules according to the instructions included with your hardware.

WARNING! *The Cavro Fusion software supports serial (RS-232) communication. If your PC does not have a serial port, you can use a USB-to-serial converter. The Cavro Fusion software has only been tested with converters that are compatible with the FTDI USB to Serial converter driver supplied with the software. Using another type of converter or another version of the driver may result in communication problems.*

- *Exception: The Cavro Fusion software communicates with the Cavro® Omni Robot in Command Processor mode and any attached modules, using the TCP/IP communication protocol.*
- *Exception: The Cavro Fusion software communicates with the Cavro® Omni Robot in Embedded mode and any attached modules, using the TCP/IP or RS232 communication protocols.*
- *Exception: This version of the Cavro Fusion software requires the Cavro® Centris Pump CAN capabilities to be disabled when used with the Cavro® Omni Robot in Command Processor mode. The RS-485 protocol should be used for communication between the Omni in Command Processor mode and the Centris pump when using this version of Cavro Fusion.*

The Cavro Fusion software must be installed on a PC running Windows that supports .NET Framework.

Your monitor DPI should be 96 (or the default setting).

The Cavro Fusion software supports only the US standard keyboard (101 keys).

WARNING! *Read the safety notes and communication instructions in your device manuals before operating your devices.*

Operate your devices with caution. Risks to users, property and the environment can arise when devices are used carelessly or improperly.

2 Installing and Starting Cavro Fusion

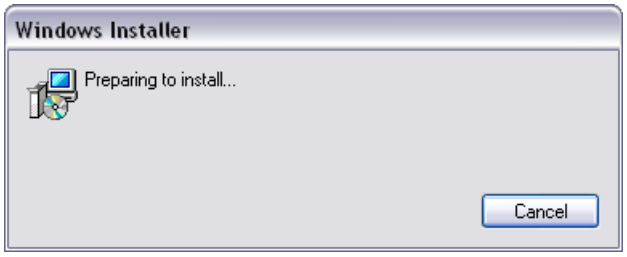
The following sections describe how to install Cavro Fusion and the drivers needed for Cavro Fusion to operate your modules, and how to start up Cavro Fusion for the first time after installation.

This chapter shows the installation sequence for a host computer that has the .NET Framework installed. If you are installing Cavro Fusion on a host computer that does not have the .NET Framework installed, you will be prompted to install the program before installing Cavro Fusion.

2.1 Installing Cavro Fusion

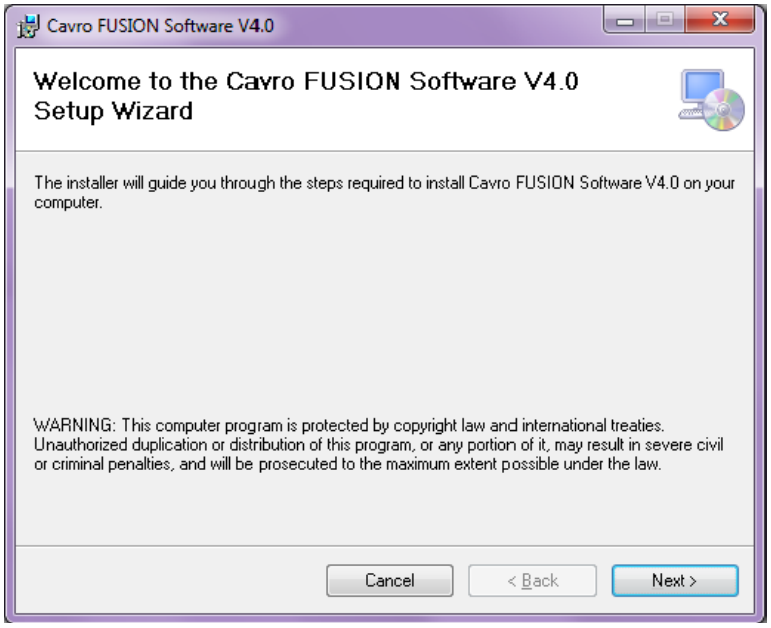
When you insert the Cavro Fusion installation CD into your PC, the Cavro Fusion installer starts automatically (Figure 2-1). If installation does not begin automatically, locate your CD drive using Windows Explorer and double-click on Setup.exe.

Figure 2-1 *Cavro Fusion Installation Page 1*



When the installer is ready to proceed, the window will close. The Welcome page appears (Figure 2-2).

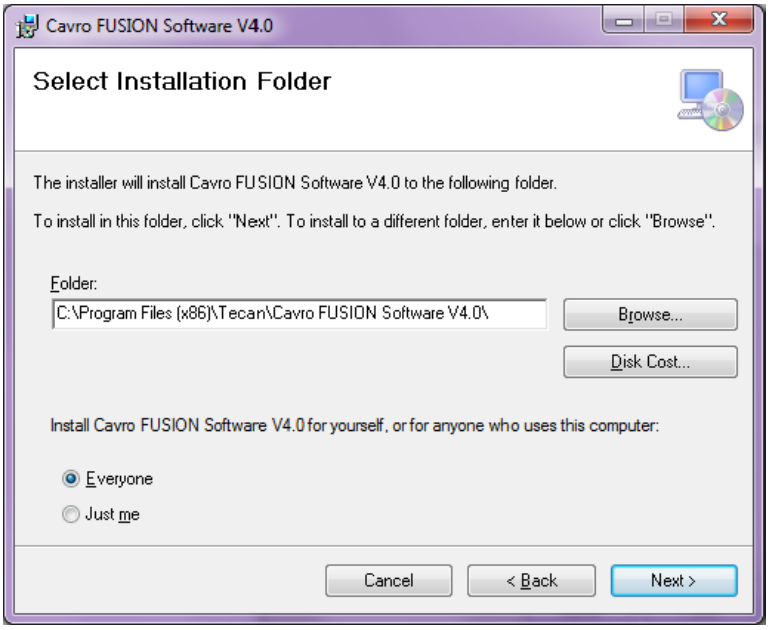
Figure 2-2 *Cavro Fusion Installation Page 2*



Note: *From this point forward, you can end the installation by clicking* ***Cancel****.*

Click **Next >** to proceed to the Select Installation Folder page (Figure 2-3).

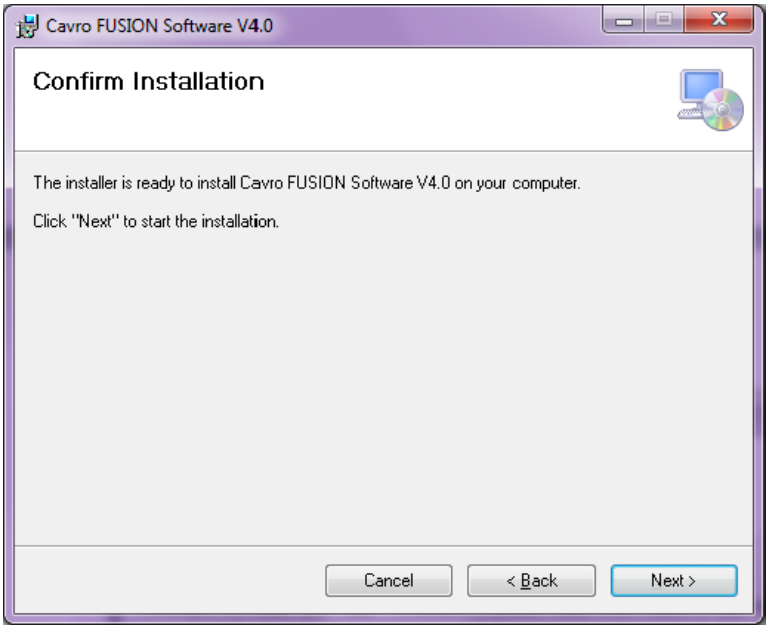
Figure 2-3 *Cavro Fusion Installation Page 3*



To change the location where Cavro Fusion will be installed, click the **Browse** button and use Windows Explorer to select the correct location.

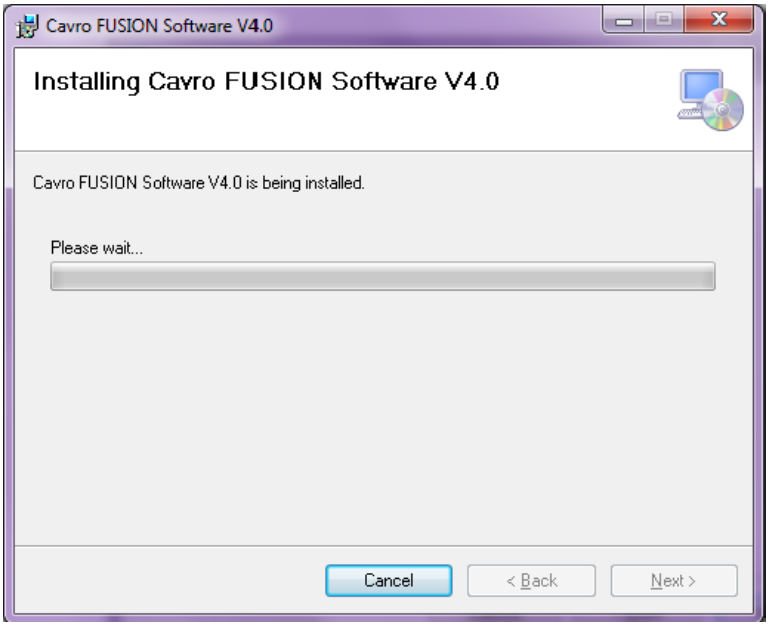
When you are satisfied with the location where Cavro Fusion will be installed, click **Next >** to proceed to the Confirm Location page (Figure 2-4).

Figure 2-4 *Cavro Fusion Installation Page 4*



Click **Next >** to install the program. The progress of the installation is shown in the Installing page (Figure 2-5).

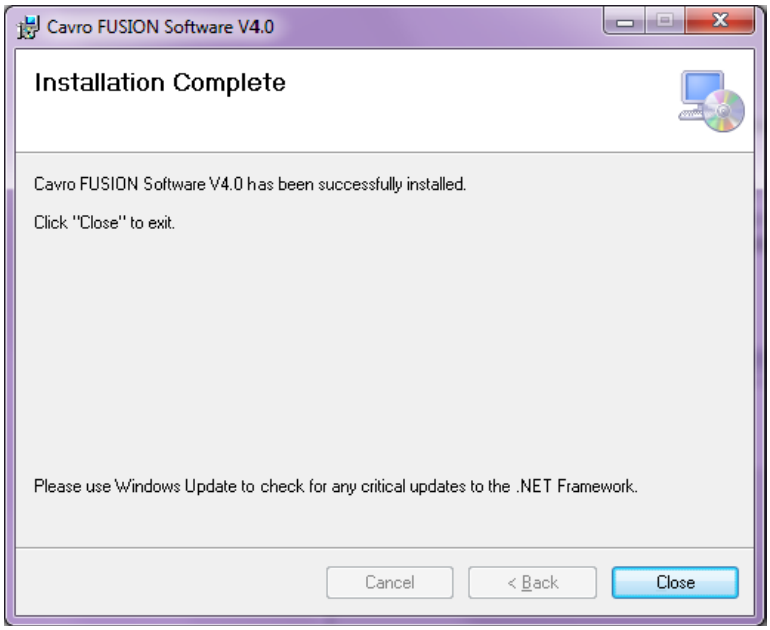
Figure 2-5 *Cavro Fusion Installation Page 5*



Upon finishing installation of the software, the Installation Complete page appears (Figure 2-6).

The Cavro Fusion installation will copy the FTDI CDM driver to the application's FTDI Driver subdirectory. The user needs to manually install the driver if the system does not already have the driver installed.

Figure 2-6 *Cavro Fusion Installation Page 7*



Click **Close** to exit the installer.

This page has been left intentionally blank.

3 Starting Cavro Fusion

Start Cavro Fusion from the shortcut on the Windows desktop (Figure 3-1).

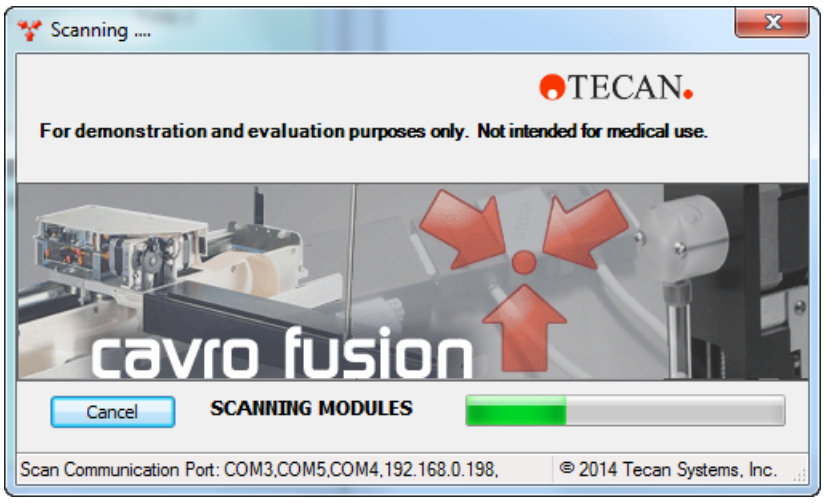
Figure 3-1 *Cavro Fusion Shortcut*

The Cavro Fusion splash page appears (Figure 3-2), and Cavro Fusion starts (by default) a quick scan for:

- RSPs / MSPs and pumps with addresses 0 to 3 through all available serial ports
- Omni and attached pumps with addresses 0 to 3 through TCP/IP

Note that Omni is operated in Command Processor mode by default; you must change settings to control Omni through its Embedded mode (see Section 3.3, Omni Embedded Mode Settings).

Figure 3-2 *Cavro Fusion Splash Page*

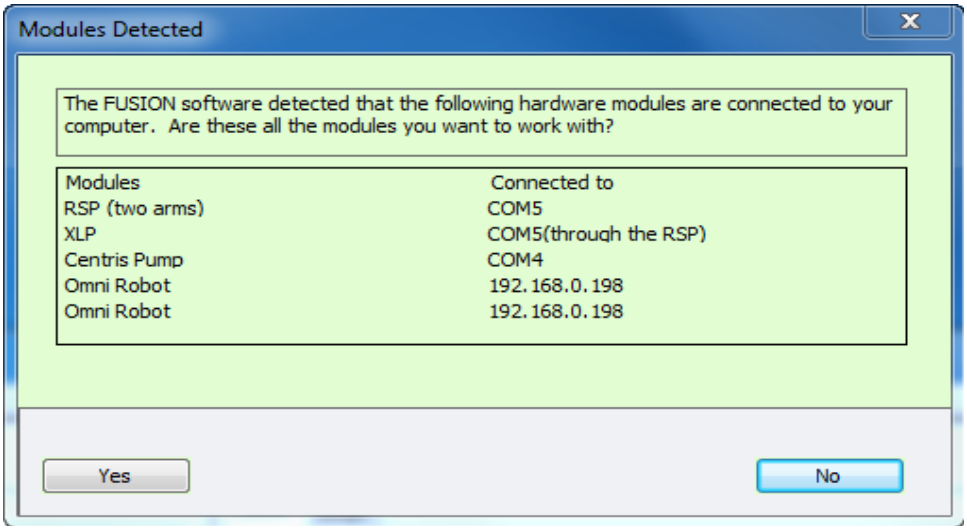


Note: *You can stop the initial scan from running automatically; see "Scanning Options" on page 3-4 for more information.*

At any time during the scan, you can click the **Cancel** button to stop the scan; however, stopping the scan will not stop the Cavro Fusion startup.

After the scan is complete, the splash page closes, the Cavro Fusion window opens, and the Modules Detected window appears (Figure 3-3).

Figure 3-3 *Modules Detected Window*



If the scan ran to completion, then the Modules Detected window shows the results of the scan. If you cancel the startup scan before it has completed, the Modules Detected pop-up window shows an empty device list, and the Detected Module window will be empty.

If the list of modules in the window is accurate, click **Yes** to close the Modules Detected window and display the Cavro Fusion window. If it is not, click **No** to start the diagnostic procedure. The diagnostic procedure is described in "The Scan Did Not Detect All Connected Devices" on page A-1.

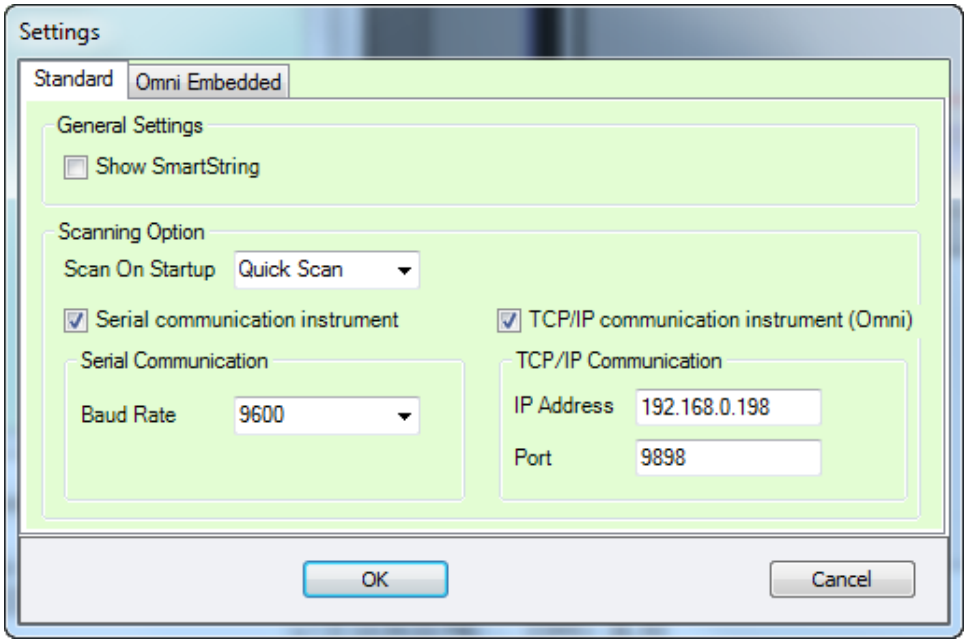
Note: *You can turn off the startup scan (see "Scanning Options" on page 3-4), in which case the results of the most recent scan (saved when you exit Cavro Fusion) are displayed, if those devices are still connected.*

The Cavro Fusion window is described in "Understanding the Cavro Fusion Window" on page 4-1.

3.1 Changing Cavro Fusion Settings

You can change the Cavro Fusion settings in the Settings window (Figure 3-4), which you open by selecting **Settings** from the Edit menu.

Figure 3-4 *Settings Window Standard Tab*



At any time, you can save your changes and close the Settings window by clicking **OK**, or close the Settings window without saving your changes by clicking **Cancel**.

The Settings window has two tabs. The Standard tab is for Omni in its Command Processor mode, and RSPs/MSPs and diluters through serial communication. The Omni Embedded tab is for Omni in its Embedded mode. For information about Omni modes, see Section 3.2, Choosing Command Processor Mode or Embedded Mode for Omni.

The Standard settings tab allows you to turn SmartStrings on and off and customize Scanning Options as shown in Figure 3-4. These settings are described in the following sections.

3.1.1 SmartString Settings

By default, Cavro Fusion software is configured to use SmartString functionality for scripting (**Show SmartString** is checked). Uncheck this box to turn off SmartString functionality.

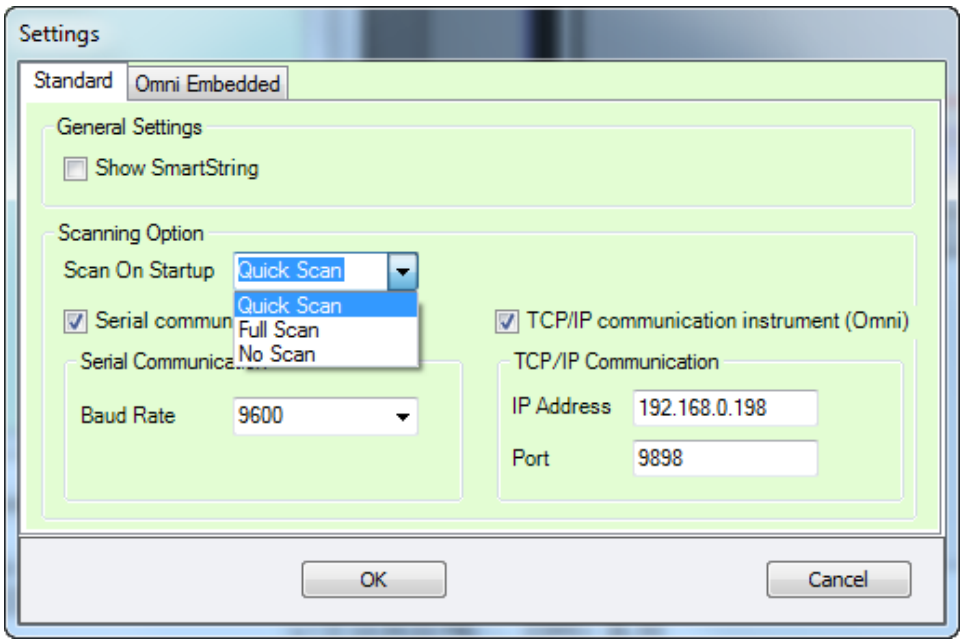
SmartString provides you with a pop-up menu of common scripting commands when you type a colon (:) or greater-than sign (>) in the Script tab. See Section 6.8, Using SmartString, for more information.

3.1.2 Scanning Options

Scan Types

Three scanning options are available. Users can select from the pull-down menu showing these options, next to Scan On Startup:

Figure 3-5 *Scan Type Options*



- A **Quick Scan** scans the first four pumps (addresses 0 through 3) directly connected to the serial port or attached to Omni. Fusion sends a ?23 query, which returns the device part number and revision string.
- A **Full Scan** scans all 15 devices (addresses 0 through 14) directly connected to the serial port or attached to Omni. Fusion sends both a ?23 query and a & query, since some older pumps do not recognize the ?23 query.
- You might choose **No Scan** if a scan has recently been performed and no changes have been made. The software will compare the information saved from the last scan with the currently connected modules. Any modules matching the saved scan information will be redisplayed.

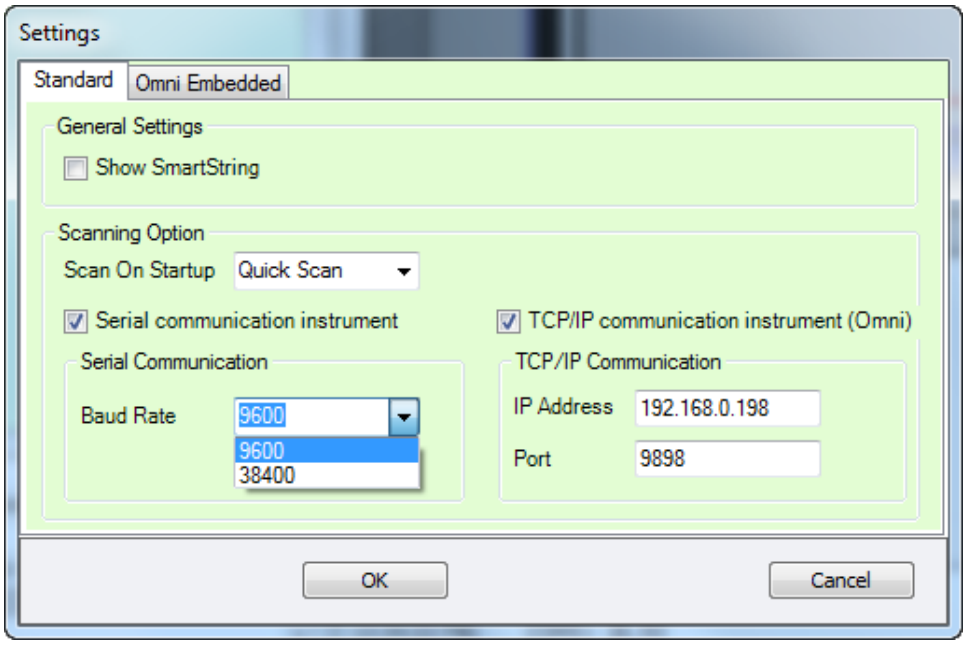
Scan Protocols

In addition to choosing the type of scan to run, you can also choose to scan only serial ports, only TCP/IP ports, or both, using the scan protocol selections in the Scanning Options section. The new settings take effect the next time Cavro Fusion starts.

Using Serial Communication

To scan using serial communication, check **Serial communication instrument** and select the baud rate from the pull-down menu.

Figure 3-6 *Serial Scan Baud Rate Options*



Modules that use the RSP Communication Server always communicate at a baud rate of 9600.

Using TCP/IP Communication

To scan using TCP/IP communication, check **TCP/IP communication instrument**. The IP address and port should be pre-populated with your system's default values, but can be manually edited if necessary.

Cavro Fusion uses the IP address and port number used to communicate with a Cavro Omni Robot and any modules connected to the Omni.

3.2 Choosing Command Processor Mode or Embedded Mode for Omni

Starting with Omni Version 4.0, an Embedded command set is provided as an alternative to the Command Processor command set. You can now control the Omni using either the Omni Command Processor mode or the Omni Embedded mode.

The Embedded commands integrate the Command Processor functionality into the electronics within the Cavro® Omni Robot's Communication Interface Board (CIB). This allows customers to who do not use an external host PC in their system architecture to programmatically control an Omni Robot through a host controller of their own design.

In Command Processor mode, the Omni Robot allows a software developer to incorporate the operation of the Omni Robot with an existing software system on a Windows platform. Commands are transmitted via easy-to-understand scripts transmitted using the Transmission Control Protocol/Internet Protocol (TCP/IP).

To help you choose between operating in Command Processor mode and Embedded mode, consider the following points:

- For a dual arm configuration, programming the Omni with the Command Processor provides built-in collision avoidance features to help avoid accidental collision of the arms in an initialized system. Currently, the Omni operated in Embedded Mode does not provide a collision avoidance feature.
- In Embedded mode, command strings are transmitted using either the TCP/IP or RS-232 communications interface.
- RS-485 devices are supported by the Command Processor only.

Command Processor mode fully supports automatic detection of properties for Omni 4.0. Embedded mode can only be operated through the Fusion script.

By default, Cavro Fusion is configured to control Omni through its Command Processor mode. To control Omni through its Embedded mode, you must change settings as described in Section 3.3, Omni Embedded Mode Settings.

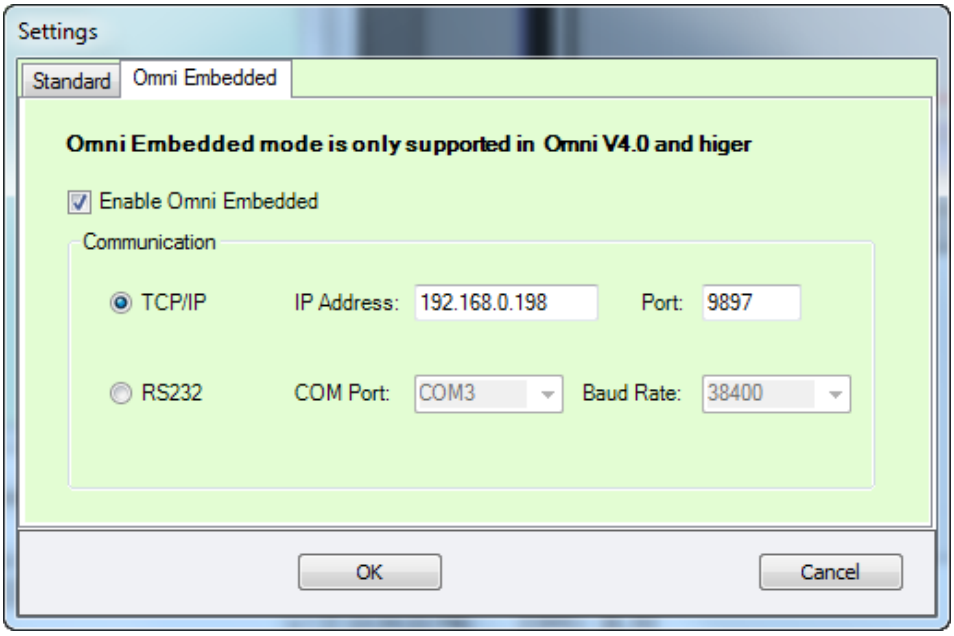
3.3 Omni Embedded Mode Settings

You can change the Cavro Fusion to operate Omni in Embedded Mode in the Settings window (Figure 3-7), which you open by selecting **Settings** from the Edit menu, and checking the Enable Omni Embedded check box.

Using TCP/IP Communication

To use TCP/IP communication, select **TCP/IP**. The IP address and port should be pre-populated with your system's default values, but can be manually edited if necessary.

Figure 3-7 *Settings Window Omni Embedded Tab*



To use RS232 communication, select **RS232**. The baud rate default value is 38400. You must choose the correct COM port from the list for RS232 communication.

At any time, you can save your changes and close the Settings window by clicking **OK**, or close the Settings window without saving your changes by clicking **Cancel**. Changing Omni mode or communication channel takes effect when you power cycle the instrument and restart the Cavro Fusion program.

See Chapter 4,Chapter 5, and Chapter 6 for information about Cavro Fusion for all devices other than embedded Omni.

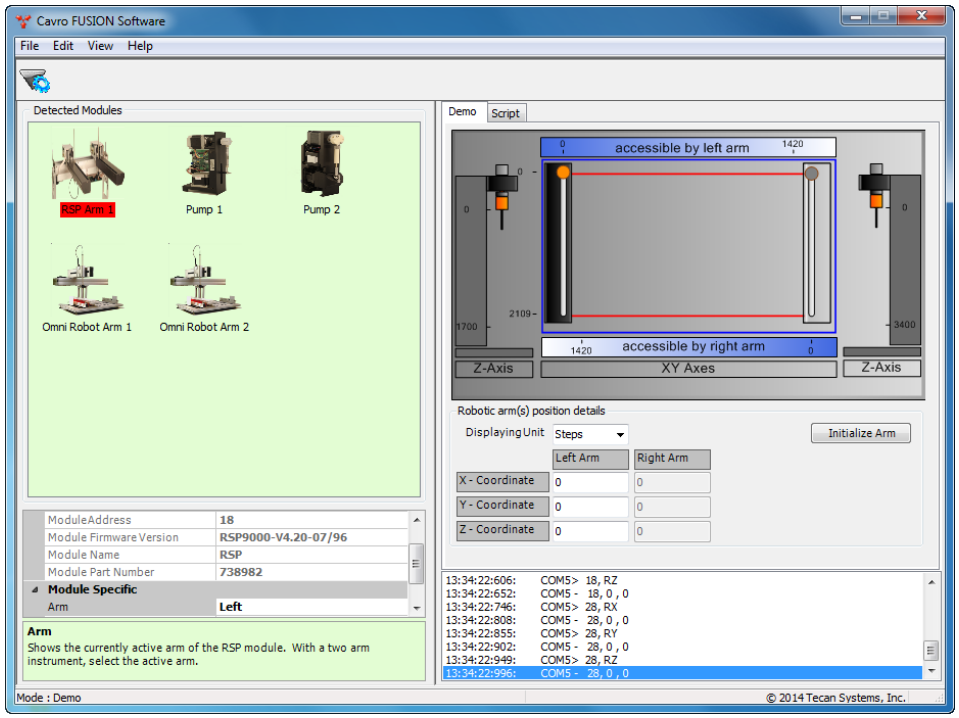
See Chapter 7, “Using the Omni Embedded Script Window” for information about using Cavro Fusion for embedded Omni.

This page has been left intentionally blank.

4 Understanding the Cavro Fusion Window

The Cavro Fusion window (Figure 4-1) provides your interface with all modules connected to your computer.

Figure 4-1 *Cavro Fusion Window*



The Cavro Fusion window includes four panes:

- The Detected Modules pane, at the upper left of the window, displays all of the modules connected to your computer.
- The Properties pane, at the lower left, displays information about the currently-selected module.
- The Module pane, at the upper right, contains two tabs:
  - The Demo tab shows a graphical representation of the selected module and lets you perform individual commands through the GUI.
  - The Script tab allows you to write, test, and run scripts to automate your module's activities.
- The Log pane, at the lower right, displays all of the commands that are sent from your computer to the module, and all replies from the module.

The Cavro Fusion window also includes three menus and a row of buttons. The buttons displayed vary depending on the tab displayed in the Module pane.

4.1 Using the Detected Modules Pane

The Detected Modules pane displays icons representing all of the modules connected to your computer, and the model number of the module.

Select a device by double-clicking the icon or the name in the Detected Modules pane. The icon name is editable; to change the name, click on the name and enter the preferred name.

When you select a device, the Properties pane will automatically update to show the values associated with the selected device, and the Demo tab of the Modules pane will show the GUI associated with the selected device.

You can also rearrange the way that the modules are displayed by dragging and dropping them in the Detected Modules pane. Doing this will not affect the modules themselves in any way.

For example, if you are using three pumps, the icons in the Detected Modules pane might not appear in the same order that the pumps are actually arranged. For clarity in using the pumps, you can drag and drop them so the icons are arranged in the same order as the physical pumps.

If you are connected to a dual-arm Cavro Omni Robot, each arm of the robot will appear as a separate module in the Detected Modules pane. If you are connected to a RSP or MSP robotic module, it appears as a single unit; use the Properties pane to select a different arm.

4.2 Using the Properties Pane

The Properties pane displays information about the currently-selected module. The properties of each device are read automatically during a scan.

The properties that are defined by the module (and therefore not editable by the user) are displayed in gray, and the properties that can be edited are displayed in black. The following are the properties that are editable by the user:

- The active arm of a two-arm RSP or MSP robotic module
- The volume of the syringe connected to a pump module, in microliters (µl)
- The valve type of a Smart Valve or pump module
- The maximum range of a Linear Option (LO) module, in steps

4.2.1 Selecting the Arm of a RSP or MSP Robotic Module

You can send commands to only one arm of a two-arm RSP or MSP robotic module at a time. To select the desired arm, set the value next to **Arm** to either **Left** or **Right**.

4.2.2 Specifying the Volume of a Syringe

Cavro Fusion does not automatically read the volume of the syringe attached to a pump module. The default syringe volume value is 1000 µl.

To specify a different syringe volume, select the value from the list next to **Syringe Size**. The volume values in the Demo tab will update automatically to reflect this change.

4.2.3 Specifying the Valve Type of a Smart Valve or Pump Module

When you detect a Smart Valve or pump module, Cavro Fusion automatically reads the valve type. However, if you want to change the valve type, you can do so by selecting a new valve type next to **Valve Type**. Cavro Fusion will prompt you to confirm the change; after you confirm the change, the Demo tab will update automatically.

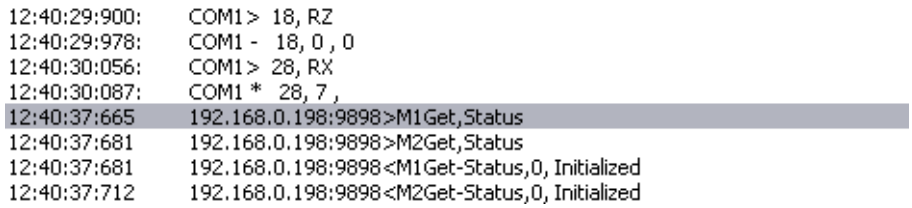
4.2.4 Specifying the Range of a Linear Option Module

The range of a Linear Option module specifies the maximum position (in steps) that the linear slide will display in the Demo tab. By dragging the slide, you can move the attached stepper motor clockwise and counter-clockwise between 0 and the specified range. For the maximum possible range, refer to the Linear Option module documentation.

4.3 Reading the Log Pane

The Log pane, which is in the lower right corner, displays all of the commands that are sent from your computer to the module, and all replies from the module.

Figure 4-2 *Log Pane*



The Log pane displays commands being sent to the module as follows:

hh:mm:ss:sss: PORT > ad, nn

where:

- *hh:mm:ss:sss* is the time that the command was issued by Cavro Fusion or that the response was received from the device.
- *PORT* is the serial port where the device is attached, or the IP address and port of the Cavro Omni Robot.
- > indicates that the rest of the line is a command sent by Cavro Fusion.
- *ad* is the address of the device.
- *nn* is the command issued to the device.

The Log pane displays responses being sent from the module as follows:

hh:mm:ss:sss: PORT result ad, er, rs

where:

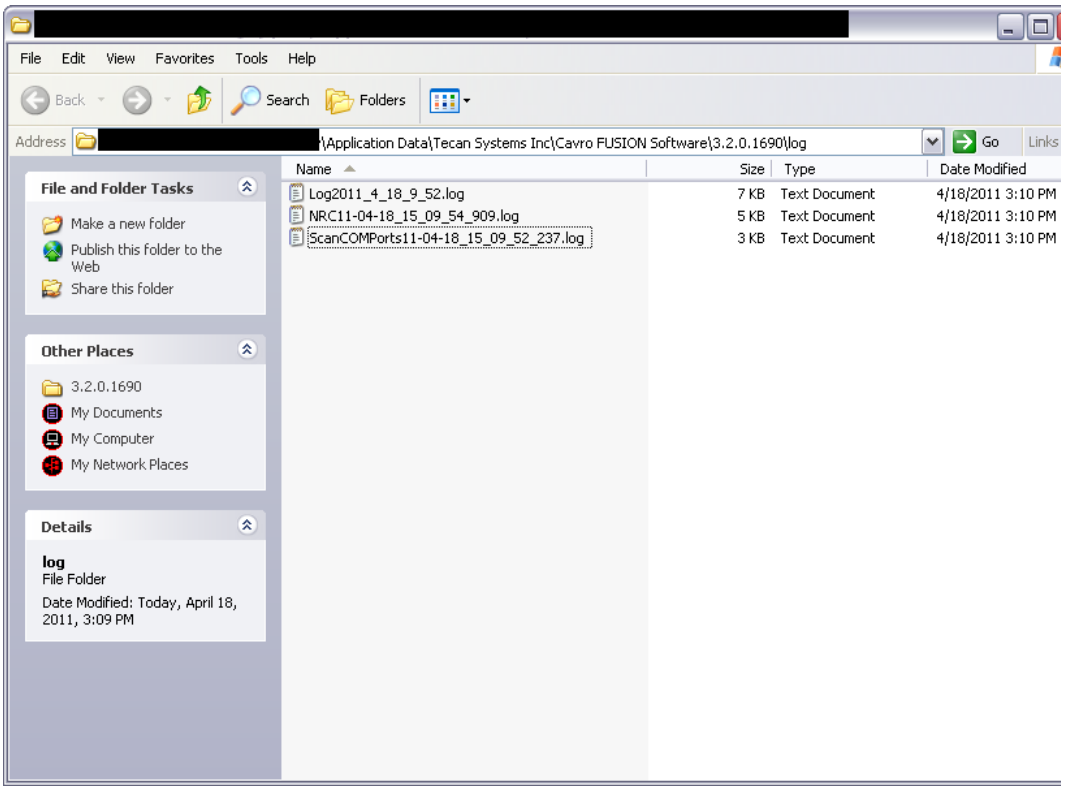
- *hh:mm:ss:sss* is the time that the command was issued by Cavro Fusion or that the response was received from the device.
- *PORT* is the serial port where the device is attached, or the IP address and port of the Cavro Omni Robot.
- *result* is one of the following:
  - \- indicates that the device command was successful, and the rest of the line is a response received from the device.
  - * indicates that the device command generated an error.
  - < indicates a response from the Cavro Omni Robot.
- *ad* is the address of the device, or the device ID of the component if the module is an Omni Robot.
- *er* is the result code generated by the module.
- *rs* is the reply string, if any.

Cavro Fusion automatically writes the contents of the Log pane to a text file that is saved to a standard location in the user's data directory, as shown in Figure 4-3. Storing log files in the user's directory prevents users from seeing any log files other than their own.

To view your saved log files:

1. Click **View** at the top of the Cavro Fusion window and select **Log File**.
2. Choose a log file and open it in a text editor.

Figure 4-3 *Location of log files*



4.4 Manually Scanning for Modules

In addition to the scan that Cavro Fusion automatically performs when you start the software, you can scan for connected modules at any time by clicking the Scan button in the Cavro Fusion window (Figure 4-4).

Figure 4-4 *Scan Button*

During the scan, Cavro Fusion will automatically detect the available ports and search them for supported modules.

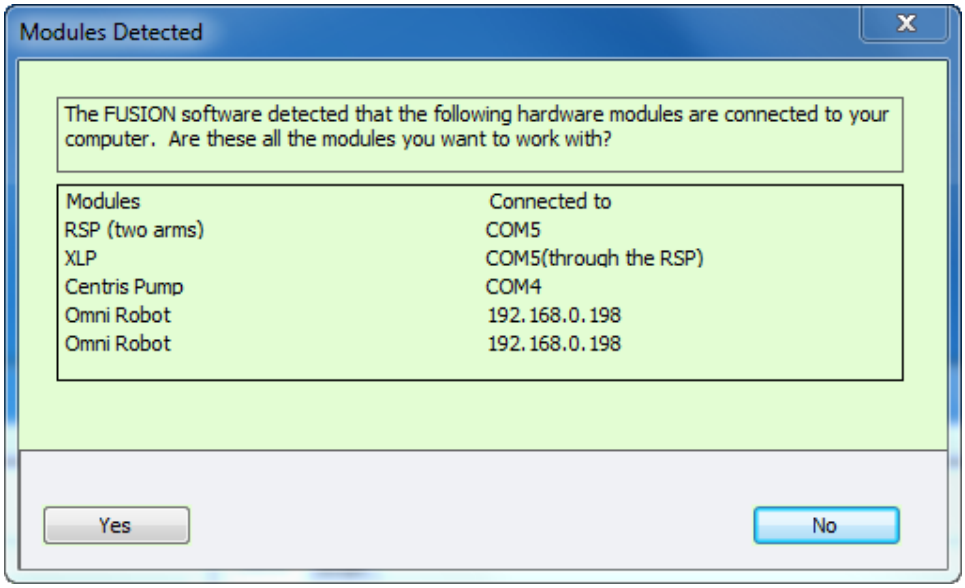
Note: *Two types of scans can be performed.* ***Quick scans*** *scan up to four devices;* ***full scans*** *scan all attached devices, up to a maximum of 15. Users can select either type of scan at any time. See* *"Changing Cavro Fusion Settings" on page 3-2* *for more information.*

If Cavro Fusion finds a diluter or pump connected to a serial port, it automatically scans addresses 0 through 3. If it finds an RSP module connected to a serial port, it automatically scans addresses 0 through 7 for connected modules.

If Cavro Fusion finds a Cavro Omni Robot connected to the given IP address and port (according to the settings described in “Scanning Options” on page 3-4), it automatically scans addresses 0 through 3 for connected modules, if it is performing a quick scan. For full scans, the scan will include addresses 0 through E.

The results of the scan are displayed in the Modules Detected window (Figure 4-5).

Figure 4-5 *Modules Detected Window*



If the list of modules in the window is accurate, click Yes to close the Modules Detected window and display the Cavro Fusion window. If it is not, click No to start the diagnostic procedure. The diagnostic procedure is described in “The Scan Did Not Detect All Connected Devices” on page A-1.

If Cavro Fusion cannot identify a device that is at one of the scanned ports and addresses, it will refer to the device as an Unrecognized Device. The Demo tab is not available for Unknown Modules, but the Script tab is available.

Unknown Modules appear as question marks in the Detected Modules pane.

The results of the most recent successful scan are automatically saved when you exit the application. If you turn off the “Scan on Startup” feature, Cavro Fusion will check the saved results and display them if the same modules are still connected.

If you cancel a manual scan, the previous scan results are retained.

4.5 Changing the Cavro Fusion Settings

You can change the Cavro Fusion software settings in the Settings window, which you open by selecting Settings from the Edit menu.

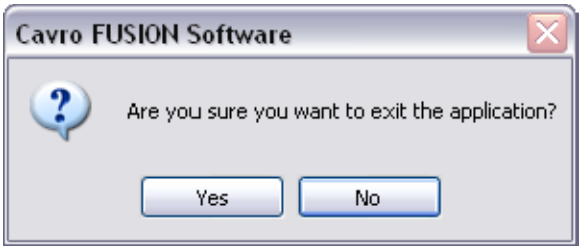
The Cavro Fusion software settings are described in “Changing Cavro Fusion Settings” on page 3-2 and “Omni Embedded Mode Settings” on page 3-6.

4.6 Exiting the Software

To exit Cavro Fusion, select Exit from the File menu.

A dialog box appears and asks if you are sure that you want to exit (Figure 4-6).

Figure 4-6 *Exit Dialog Box*



Click **Yes** to close the dialog box and the Cavro Fusion window. Click **No** to close the dialog box and return to the Cavro Fusion window.

This page has been left intentionally blank.

5 Using the Demo Tab

The Demo tab shows a graphical representation of the selected module and lets you perform individual commands through the GUI. The GUI is used as a tool to demonstrate the capabilities of a device.

View the Demo tab at any time by clicking the Demo tab above the Module pane.

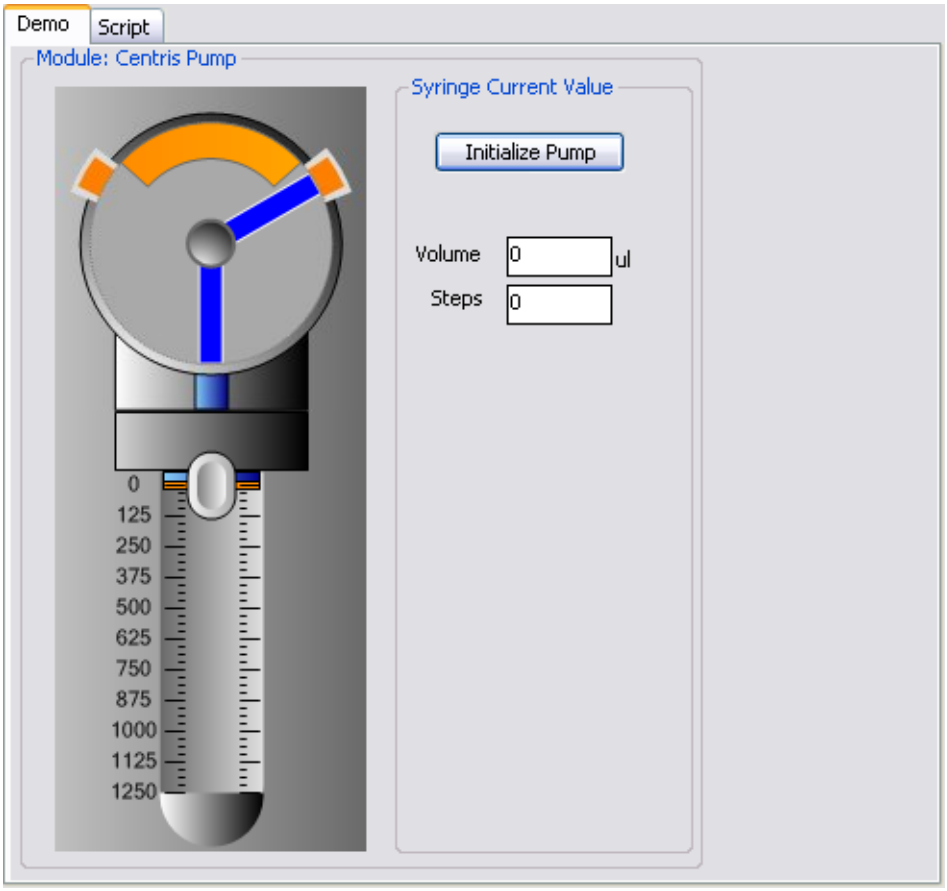
The following sections describe the capabilities of the Demo tab to manipulate the following supported modules:

- pump modules
- robotic modules, including the RSP, MSP, gripper, and single or dual arm Cavro Omni Robot (Command Processor mode)
- Smart Valve
- Smart Peristaltic (SP) pump
- Linear Option (LO) module
- Input/Output (I/O) board module
- ADP

5.1 Manipulating a Pump Module

Figure 5-1 shows the Demo tab as it appears when a pump or diluter module is selected in the Detected Modules pane (in this case, a Cavro Centris Pump).

Figure 5-1 *Demo Tab Displaying a Pump Module*



The graphic in the left-hand portion of the Demo tab represents the pump module. The top half shows the pump valve and represents all of the available ports on the valve; the bottom half shows the pump syringe.

The syringe volume displayed is controlled in the Properties pane; the default value is 1000 µl. See “Specifying the Volume of a Syringe” on page 4-3 for instructions on setting the syringe volume.

5.1.1 Initializing the Pump

Initialize the pump by clicking the **Initialize Pump** button.

The valve returns to its initial position, the syringe pump returns to the 0 position, and the pump is initialized.

5.1.2 Changing the Valve Position

You can change the path taken by fluid through the valve by selecting a different valve port. The orange areas of the graphic represent the available valve ports, and the blue rectangles represent the path that the fluid will take.

On a standard three-port valve pump, change the location from which the pump will draw fluid by clicking the valve port that represents the desired path.

You can also set the pump to bypass the syringe by clicking the orange arc at the top of the pump; both blue rectangles representing the liquid path will point to the orange rectangles, representing the valve ports. Reset the path to go through the syringe by clicking any of the valve ports.

Note: *Some pumps do not have a bypass position.*

5.1.3 Moving the Syringe Plunger With the GUI

You can move the syringe plunger in the GUI by clicking on the gray oval in the bottom portion of the graphic and dragging it to the desired location.

When you move the plunger, the values in the text fields to the right of the graphic update dynamically to display the current position.

5.1.4 Moving the Syringe Plunger With the Text Fields

The two text fields in the right-hand portion of the Demo tab display the current position of the syringe plunger, shown in volume (in µl) and in motor steps. The top of the syringe represents 0 µl and 0 steps.

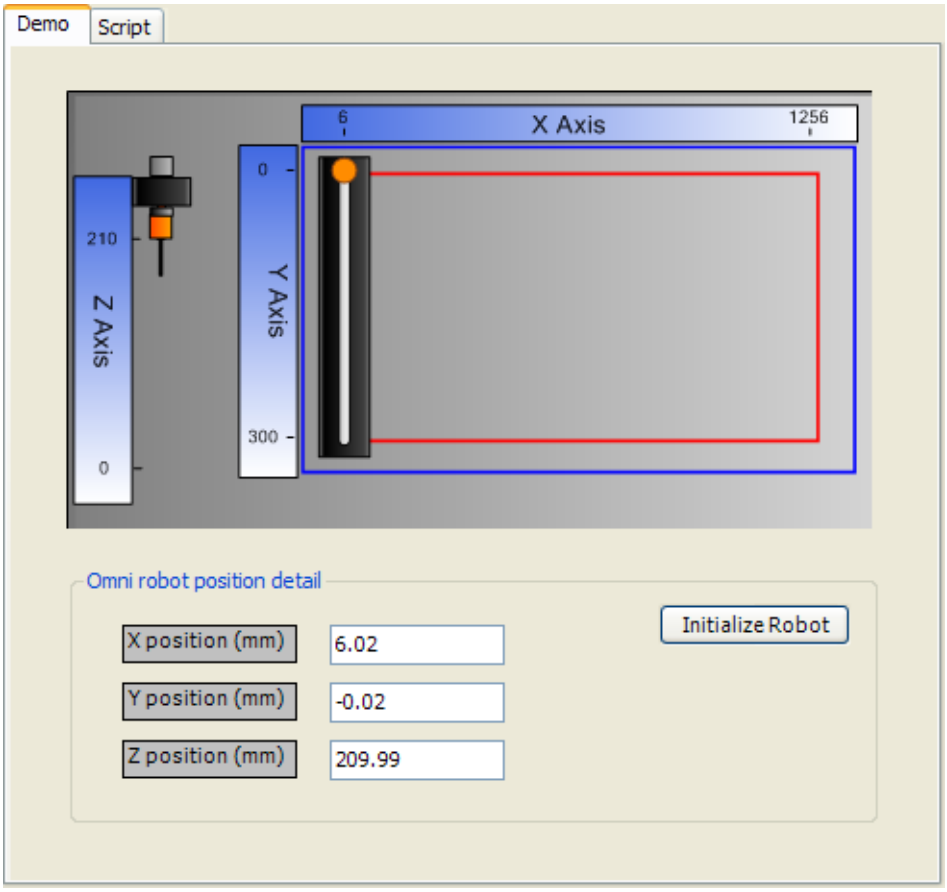
Type the desired position of the plunger in one and only one of the text fields and press Enter. The plunger will move to the specified position, and the value in the unused text field automatically updates to reflect the new position.

If you type positions in both text fields and press Enter, the plunger will move to the position specified in the text field where the cursor was active when you pressed Enter.

5.2 Manipulating a Cavro Omni Robot

Figure 5-2 shows the Demo tab as it appears when the left arm module of a Cavro Omni Robot is selected in the Detected Modules pane.

Figure 5-2 *Demo Tab Displaying a Cavro Omni Robot left arm module*



The graphic at the left of the diagram represents a side view of the Cavro Omni Robot, and can be used to move the Z axis of the Omni by dragging the orange rectangle, representing the tip position, up and down. The central part of the diagram represents a top view of the Omni Robot, and is used to move the X and Y axes, by dragging the orange circle representing the tip position.

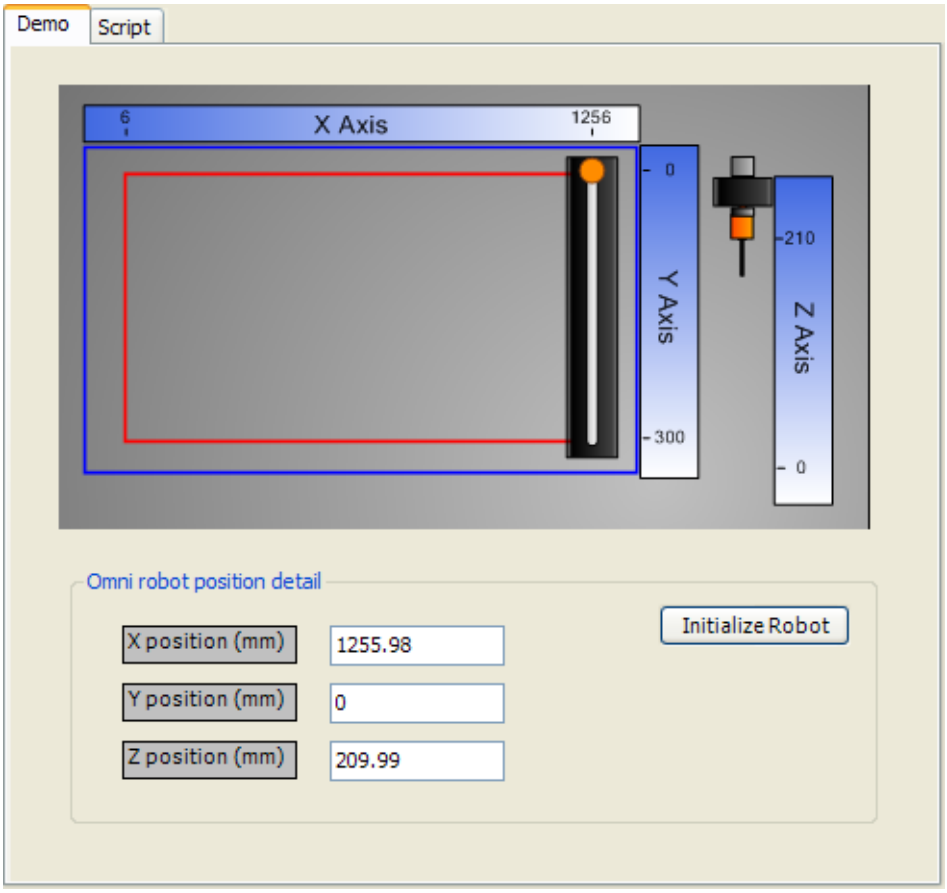
The orange rectangle and orange circle in each graphic represent the current position of the probe or tip attached to the Z axis.

You can move the tip to any point in all three axes. The top left corner of the center graphic represents the home position (Range Low) on both X and Y axes for the left arm. The top of the graphic on the left represents the maximum position (Range High) for the Z axis.

The area below the GUI includes text fields that display the current X, Y, and Z coordinates of the selected module. You can also use these fields to move each axis to a specific point.

If the right arm module of a dual-arm Omni Robot is selected, the Z axis is shown at the right side of the diagram, as shown in Figure 5-3.

Figure 5-3 *Demo Tab Displaying a Cavro Omni Robot right arm module*



In this case, the top right corner of the center graphic represents the initial (home) position of the X axis, which is the Range High for that axis (the actual value will depend on the length of the axis of your Omni Robot). The top right corner also represents the home position (Range Low) of the Y axis of the right arm. The top of the graphic on the right represents the maximum position (Range High) for the Z axis.

5.2.1 Initializing an Arm

The Omni Robot is initialized by the scan performed at start up. If multiple arms are present, they are all initialized. All arms and axes are also initialized when you perform a manual scan by clicking the **Scan** button in the toolbar at the top of the Cavro Fusion window.

You can re-initialize the selected arm at any time.

To re-initialize a Cavro Omni Robot arm module, do the following:

1. In the Detected Modules pane, select the appropriate arm module.

   The Demo tab displays the selected module, and the Properties pane displays the properties of the selected module.
2. In the Demo tab, click the **Initialize Arm** button.

   This initializes the three axes of the selected arm module.

5.2.2 Moving an Arm with the GUI

You can move an arm in any axis by clicking the orange (active) area and dragging it to the desired location.

When the arm has finished moving, the values in the text fields will update to display the current position.

Note: *In a dual-arm Omni Robot, the collision avoidance feature will prevent one arm from executing a motion that will collide with the other arm. If you try to drag an arm to a position that would cause a collision, an error message is displayed, and the move is not executed.*

5.2.3 Moving an Axis with the Text Fields

You can move the axes of an arm to specific positions by using the text fields below the graphic. Simply type the desired location in the text box and press Enter.

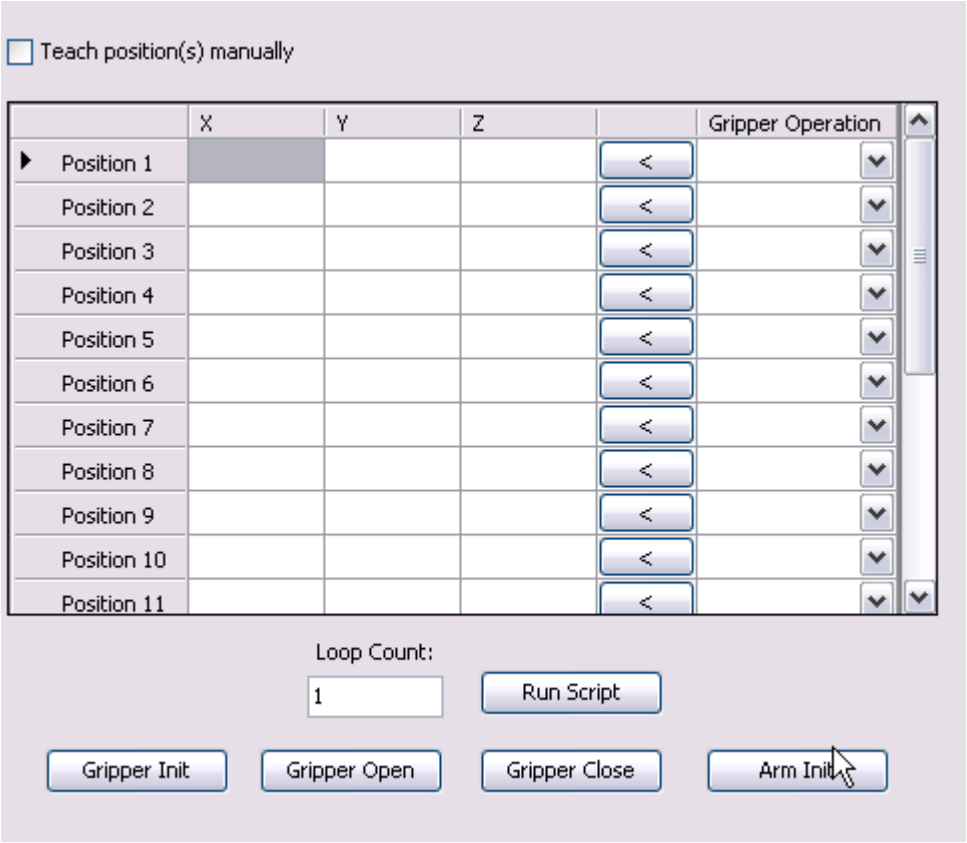
You can only move one axis at a time using the text fields; the move command will be executed only for the axis specified by the text field where the cursor was active when you pressed Enter.

Note: *In a dual-arm Omni Robot, if you enter a value into a text field that would cause a collision with the other arm, an error message is displayed, and the move is not executed.*

5.3 Manipulating a Cavro Omni Gripper

Figure 5-4 shows the Demo tab as it appears when an Omni arm with a gripper is selected in the Detected Modules pane. The Demo tab allows you to manually teach the Omni a sequence of up to 20 individual movements and then run them as a script.

Figure 5-4 *Demo Tab Displaying a Cavro Omni Robot gripper module*



5.3.1 Example of Teaching a Movement Sequence

This example illustrates techniques that are helpful in making the process of teaching a sequence of movements as safe and efficient as possible. We recommend that you follow exactly the steps outlined here the first time you teach a movement sequence, so that you can practice these techniques. After completing this example, you will be prepared to teach the Omni your own custom movements.

Follow these steps to pick up and move a plate:

1. Select the icon of an arm with a gripper on the Detected Modules pane. A warning appears that reminds you to move the gripper to a safe position so that during the initialization you do not run into labware with the gripper.
2. Make sure the gripper is in a closed position.
3. Check the **Teach position(s) manually** box. A message appears to warn you that the gripper might drop.

4 Click **OK** to dismiss the message box.

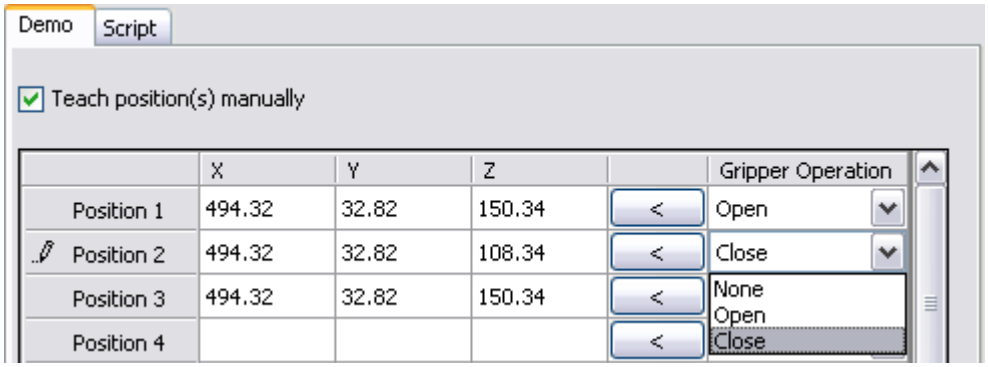
5 Manually move the gripper to the position on the work surface where the plate is located. Rest the gripper on top of the plate, and align the gripper so that the plate is located at the center of the gripper fingers when the fingers are open.

6 Click the **<** buttons in the Position 1, 2, and 3 rows. This inserts the current X, Y, and Z coordinates of the gripper in the three positions. Add 20-50 mm to the Z coordinates for Positions 1 and 3. Subtract 2-3 mm from the Z coordinate of Position 2.

Note: *The numbers to add and subtract from the Z coordinates in this example are suggestions only. You may need to use numbers outside the suggested ranges if there is not sufficient clearance in your work environment to accommodate our suggestions.*

7 Click the **V** button for the Position 1 row and select **Open** from the list of Gripper Operations. This instructs the Omni to open the gripper fingers.

Figure 5-5 *Gripper operations*



8 Click the **V** button for the Position 2 row and select **Close** from the list of Gripper Operations. This instructs the Omni to close the gripper fingers, gripping the plate.

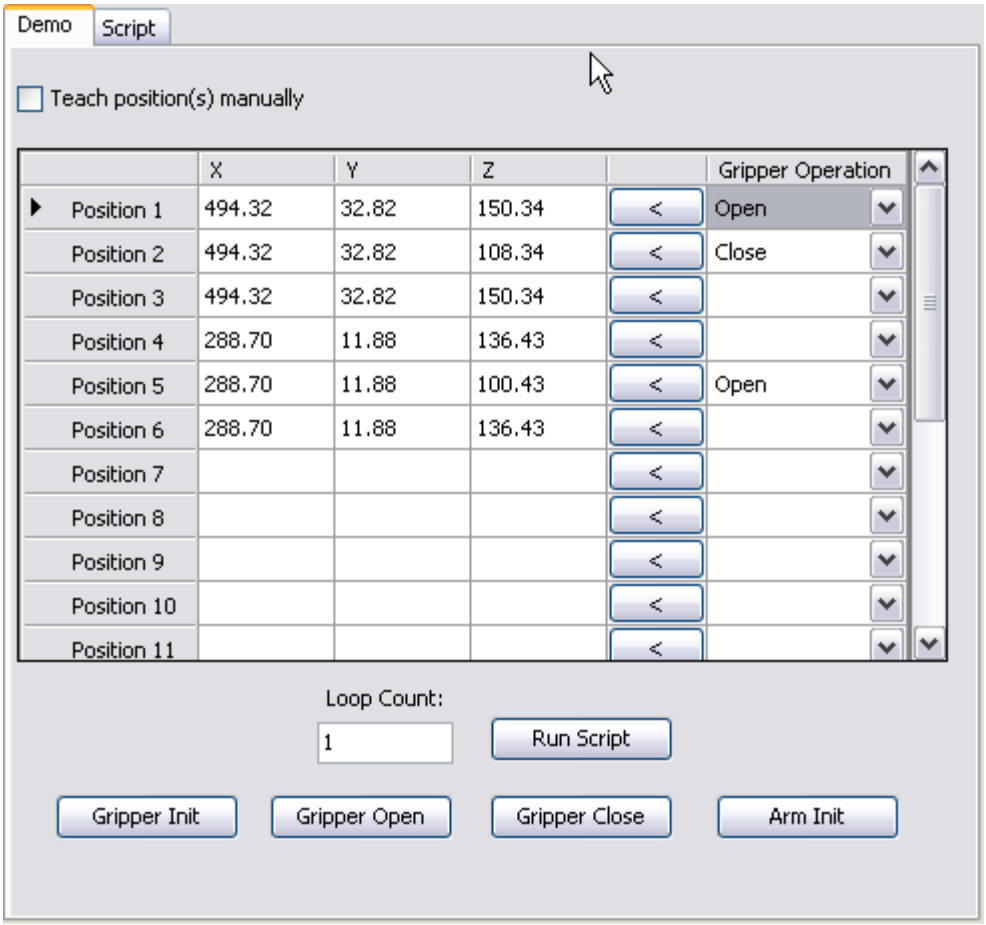
9 Move the gripper to the position where you want to release the plate.

10 Click the **<** buttons for the Position 4, 5, and 6 rows. Add 20-50 mm to the Z coordinates for Positions 4 and 6. Add 2-3 mm to the Z coordinate for Position 5.

11 Click the **V** button for the Position 5 row and select **Open** from the list of Gripper Operations.

12 Uncheck the **Teach position(s) manually** box.

Figure 5-6 *Run script to test*



13 Click **Arm Init** to return the arm to the home position.

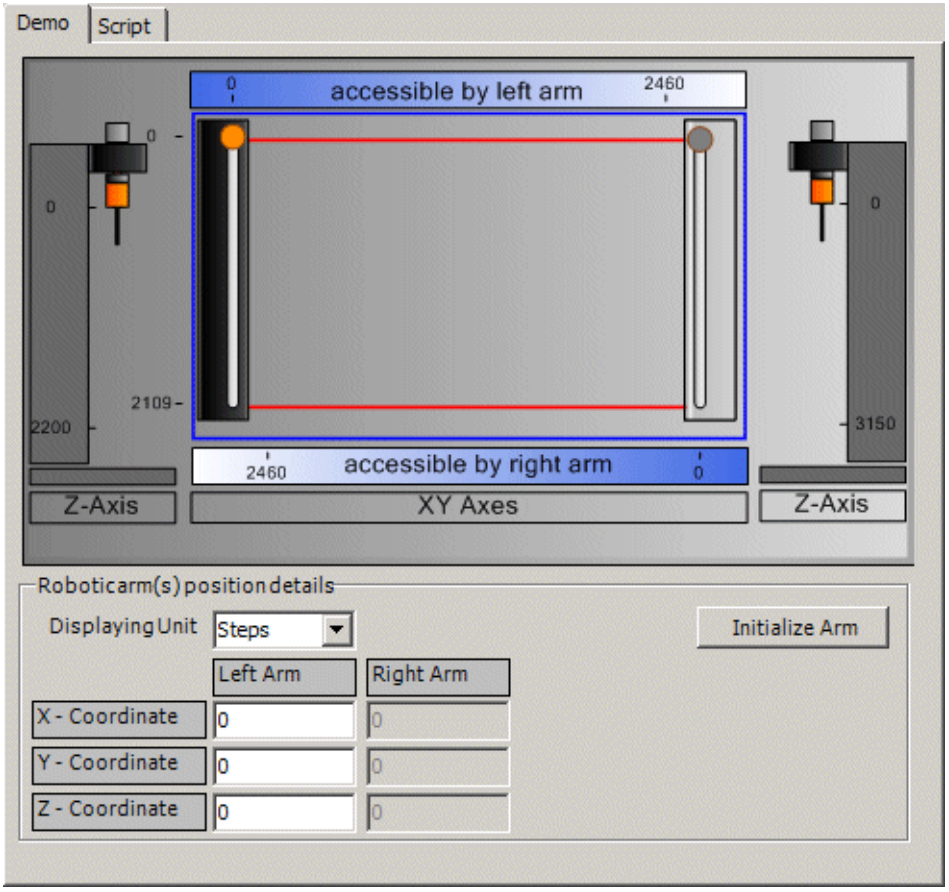
14 Click **Run Script** to test.

Note: *You can select one or more lines and click* ***Run Script*** *to test only those lines. Click a line to select it; use CTRL-click to select multiple lines. If no lines are selected, all lines run.*

5.4 Manipulating a Robotic Module (RSP or MSP)

Figure 5-7 shows the Demo tab as it appears when a robotic module (in this case, an RSP with two arms) is selected in the Detected Modules pane.

Figure 5-7 *Demo Tab Displaying a Robotic Module*



The graphic in the top portion of the tab represents the robotic module; the center displays the top view of the module, and the graphics on the left and right display a side view of the left and right arms, respectively. If the module has only one arm, then only one graphic appears, on the correct side of the tab.

The orange rectangles in each graphic represent the current position of the arm(s).

You can move the arm to any point in all three axes. The center graphic is used to move either arm in the X axis and Y axis. The top left corner of the graphic represents the 0 position on both axes for the left arm, and the top right corner represents the 0 position on both axes for the right arm. The graphics on the left and right are used to move the left or right arm in the Z axis, respectively; the top of the graphic represents the 0 position.

The area below the GUI includes text fields that you can use to move the arms to a specific point.

5.4.1 Initializing an Arm

Initialize an arm on a robotic module by performing the following tasks:

1. In the Detected Modules pane, select the appropriate module.

   The Demo tab displays the selected module, and the Properties pane displays the properties of the selected module.
2. In the Demo tab, click the **Initialize Arm** button.

   Both arms, regardless of which arm is selected in the Properties pane, return to their initial position and are initialized.

5.4.2 Changing the Displaying Units

The text fields below the graphic display the current position of each arm in either steps or millimeters.

Change the units displayed in the text boxes by choosing the desired units in the Displaying Unit drop-down menu.

5.4.3 Moving an Arm with the GUI

You can move an arm in any axis by clicking the orange (active) area and dragging it to the desired location. If you move the arm in both the x-axis and y-axis with the same movement, the arm will move in both directions simultaneously.

When you move the arm, the values in the text fields to the right of the graphic update dynamically to display the current position.

5.4.4 Moving an Arm with the Text Fields

You can move an arm to a specific position by using the text fields below the graphic. Simply type the desired location in each text box and press Enter.

If you move the arm in the X axis and Y axis with the same command, the arm will move in both directions simultaneously. If you move the arm in the Z axis and one of the other axes with the same command, the arm will move on the Z axis first, and then in the other axes.

Note: *Remember to check the units in the Displaying Unit pull-down menu, to ensure that you are not entering values that will cause a collision or move the arm outside its maximum range.*

5.5 Manipulating a Smart Valve

Figure 5-8 shows the Demo tab as it appears when a Smart Valve module is selected in the Detected Modules pane (in this case, a four-port module).

Figure 5-8 *Demo Tab Displaying a Four-Port Smart Valve*

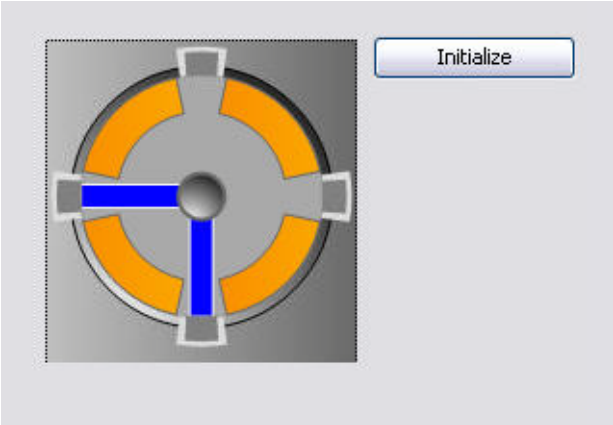
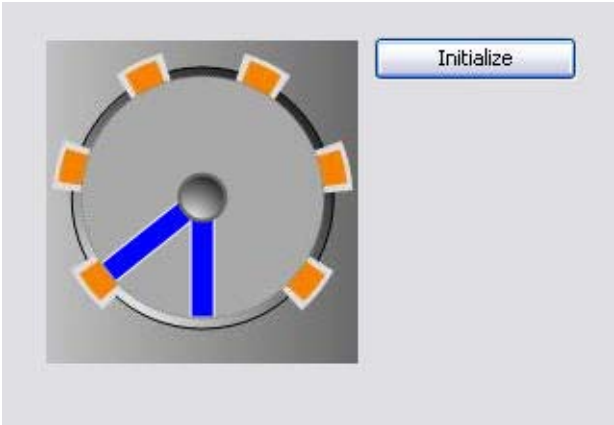


Figure 5-9 shows a distribution Smart Valve module, with a different number of ports (in this case, six ports).

Figure 5-9 *Demo Tab Displaying a Six-Port Distribution Smart Valve*



5.5.1 Initializing the Smart Valve

Initialize the valve by clicking the **Initialize** button.

The valve returns to its initial position, and the valve is initialized.

5.5.2 Changing the Valve Position

You can change the path taken by fluid through the valve by selecting a different valve port. The blue rectangles represent the path that the fluid will take.

The boxes around the rim of the module represent the available valve ports.

The port to dispense fluid always appears at the bottom of the module. If a box appears at this position, such as in Figure 5-8, then the module includes a bypass position (that is, the fluid path can be moved away from this port, and you can move both blue rectangles in the Demo tab). If no box appears at this position, such as in Figure 5-9, then the module does not include a bypass position (that is, the fluid path cannot be moved away from this port, and you can only move one blue rectangle in the Demo tab).

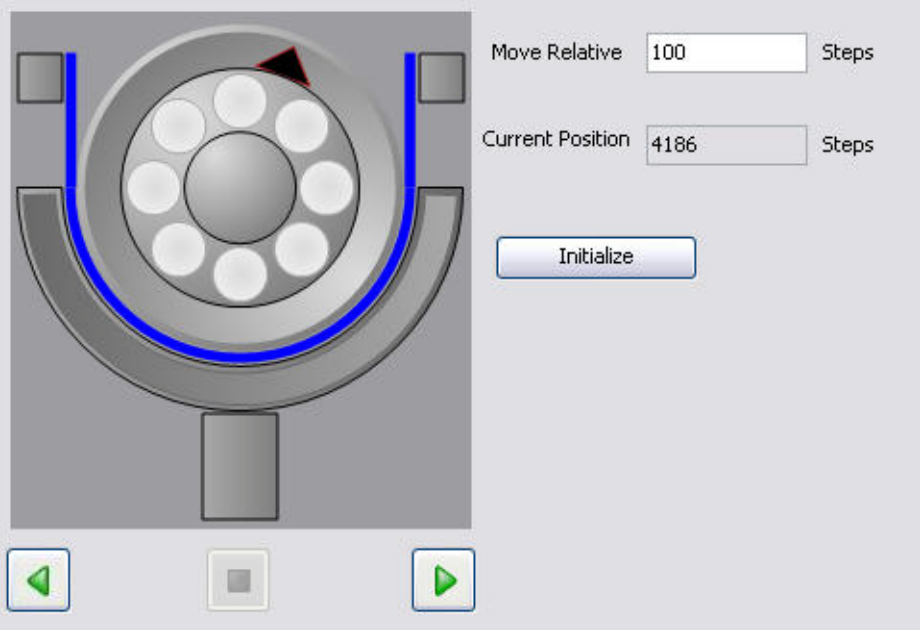
Change the fluid path in a Smart Valve with a bypass position by clicking one of the orange arcs in the Demo tab. this sets the fluid path to the two ports on either side of the arc.

Change the fluid path in a Smart Valve without a bypass position by clicking the orange box at the desired valve position.

5.6 Manipulating a Smart Peristaltic (SP) Pump

Figure 5-10 shows the Demo tab as it appears when a Smart Peristaltic (SP) Pump module is selected in the Detected Modules pane.

Figure 5-10 *Demo Tab Displaying an SP Pump*



5.6.1 Initializing the Pump

Initialize the module by clicking the Initialize button.

The pump returns to its initial position, the Move Relative and Current Position fields are set to 0, and the pump is initialized.

5.6.2 Changing the Pump Position With the Text Field

You can move the pump a specific number of steps in one direction with the Move Relative text field. A positive number moves the pump the given number of steps clockwise, and a negative number moves the pump counterclockwise.

Type the number of steps to move in the Move Relative text field, and press Enter.

When the move is completed, the black triangle moves to show the current position of the pump, and the Current Position field updates to show the current position of the pump relative to the 0 position.

Note: *The Move Relative text field always moves the pump relative to its current position, while the Current Position field always shows the position of the pump relative to the 0 position.*

5.6.3 Running the Pump Continuously

You can set the SP Pump to run continuously in the clockwise direction by clicking the right-pointing arrow button, and counterclockwise by clicking the left-pointing arrow button.

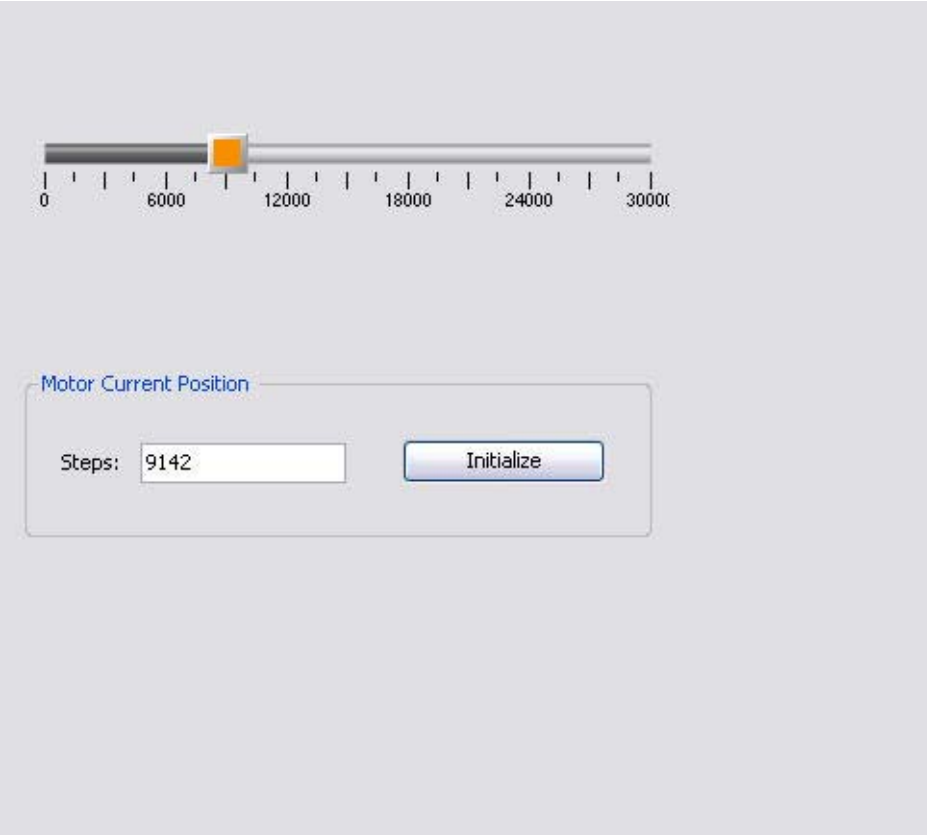
While the pump is running, the black triangle on the rim of the display and Current Position field do not continuously update.

Stop the continuously-moving pump by clicking the red Stop button. After the pump stops, the Current Position field updates to show the current absolute position.

5.7 Manipulating a Linear Option (LO) Module

Figure 5-11 shows the Demo tab as it appears when a Linear Option (LO) module used to control a stepper motor is selected in the Detected Modules pane.

Figure 5-11 *Demo Tab Displaying an LO Module*



5.7.1 Initializing the Module

Initialize the module by clicking the **Initialize** button.

The motor returns to its initial position, the Steps field is set to 0, the orange slider moves to the leftmost position, and the pump is initialized.

5.7.2 Moving the Motor

Move the motor in one of two ways:

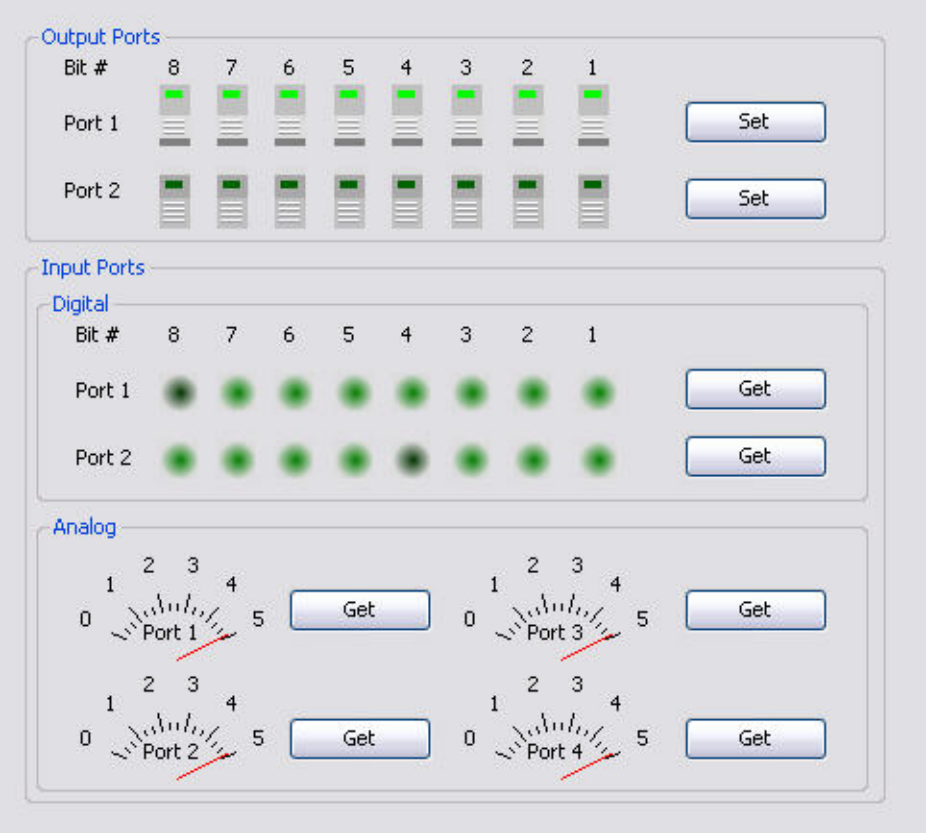
- Enter the absolute value of the desired position in the Steps field and press Enter.
- Click the orange rectangle and slide it to the desired position.

In either case, the motor moves to the specified position, and the Steps field and the GUI update to reflect the current position.

5.8 Manipulating an Input/Output (IO) Board

Figure 5-12 shows the Demo tab as it appears when an Input/Output (IO) board module is selected in the Detected Modules pane.

Figure 5-12 *Demo Tab Displaying an IO Board Module*



Note that the output bits of port 1 (JP4) are “Active Low,” which means that if you set a logical 1 for a particular bit, the corresponding output level will be 0. For example, if you set output port 1 to 255 (all eights bits set to 1), all of the output levels will be low (0). Please note that output port 2 (JP10) has the reverse logic; its bits are “Active High,” which means that if the bit is set to 1, the corresponding output level will be high.

For the digital input ports (JP3 and JP9), the dark green LED reflects a bit that IS NOT set (low level) and a light green LED reflects a bit that IS set (high level).

5.8.1 Setting Output Port Bits

Set the value of an output port bit by clicking the switch representing the bit. For Port 1, the color will toggle between dark green (representing a bit that is set) and light green (representing a bit that is not set). For Port 2, the color will toggle between light green (representing a bit that is set) and dark green (representing a bit that is not set).

When you have set the desired values for all eight bits on an output port, click the **Set** button for the port.

5.8.2 Getting Values of Input Port Bits

Get the current value of the input port bits by clicking the **Get** button next to the desired input port.

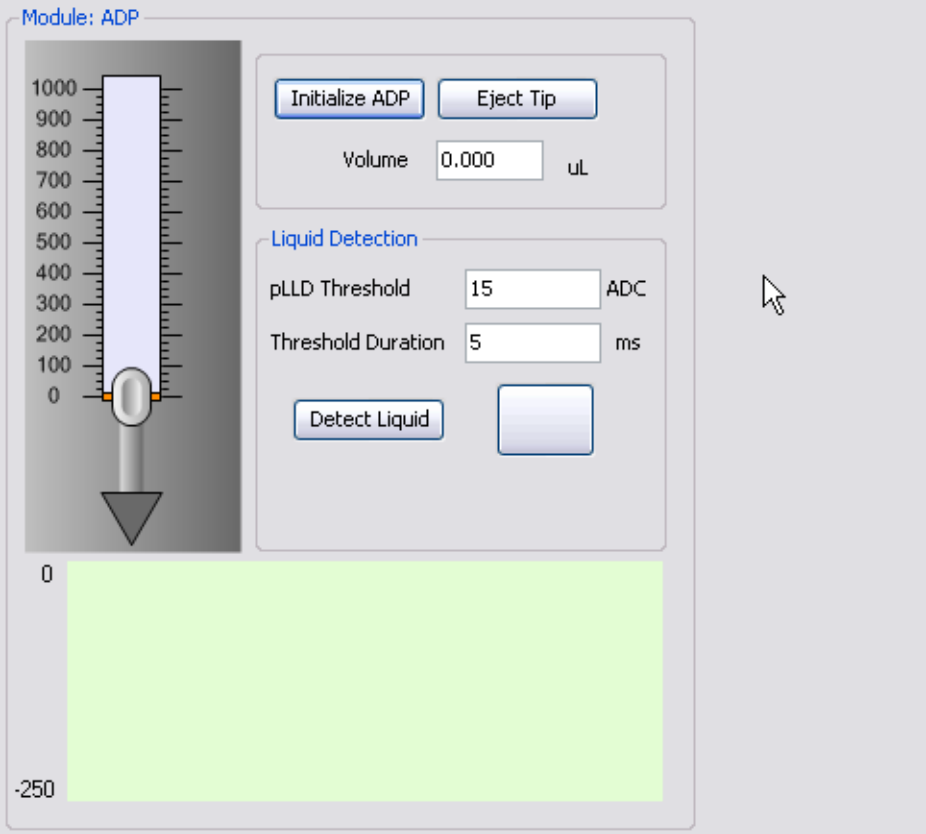
The GUI will update to reflect the current values.

5.9 Manipulating an ADP Module

Figure 5-13 shows the Demo tab as it appears when an ADP module is selected in the Detected Modules pane.

Note: *An ADP module can be manipulated using the Demo tab only if it connects directly through serial port. An ADP module accessed through Omni via CAN bus can be operated through a Fusion script (see Chapter 6, "Using the Script Tab").*

Figure 5-13 *Demo Tab Displaying an ADP Module*



The graphic in the left-hand portion of the Demo tab represents the ADP module probe assembly. The plunger can be moved up and down using the oval button in the center of the plunger. The triangle at the bottom of the plunger represents the disposable tip.

ADP properties displayed in the Properties pane cannot be edited.

5.9.1 Initializing the ADP

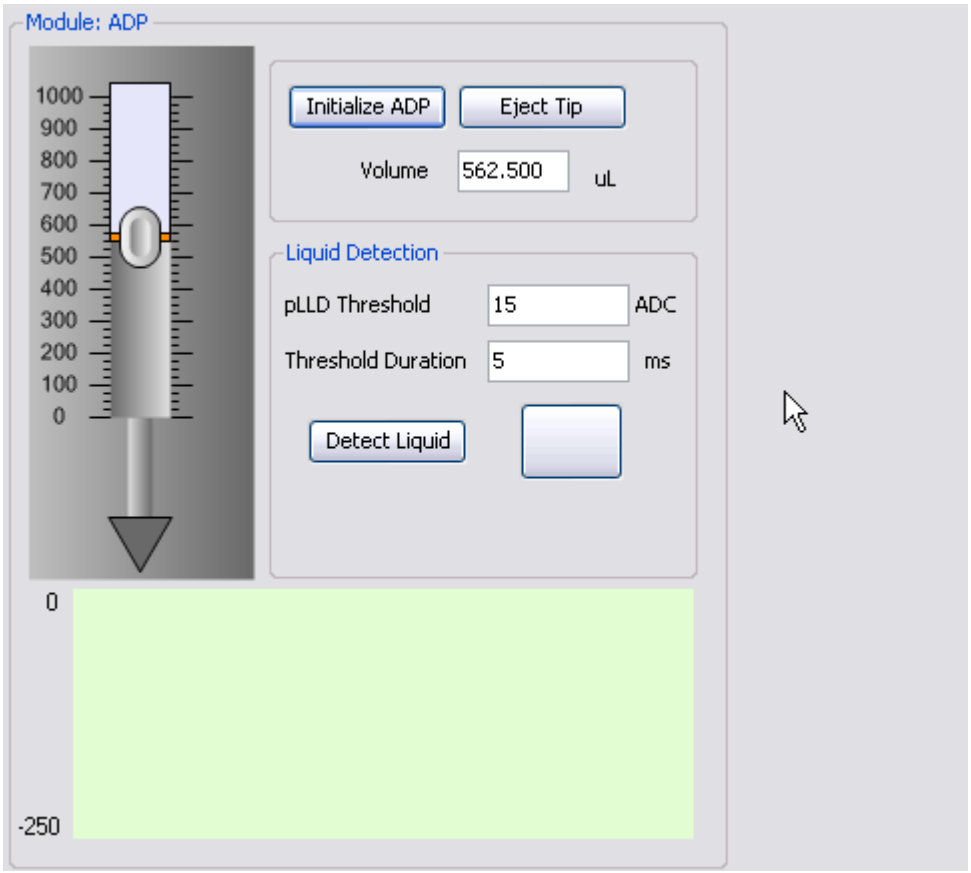
Initialize the ADP by clicking the Initialize ADP button. Fusion will send the initialize command <WR> to the ADP. The plunger will move to its 0 position; when it finishes, the ADP GUI will refresh to the 0 position and the text box for volume will display 0, as shown in Figure 5-13.

5.9.2 Changing Plunger Volume With the GUI

You can raise or lower the probe assembly by dragging the button in the center of the plunger graphic. The orange line represents the current volume level.

When you release the button in the graphic, the volume shown in the text box will update to the currently selected plunger volume. See Figure 5-14.

Figure 5-14 *ADP Plunger Volume Change*



5.9.3 Changing Plunger Volume With the Text

The plunger volume can be changed by entering a specific volume in the text box. The graphic will update to simulate this volume level, represented by the orange line.

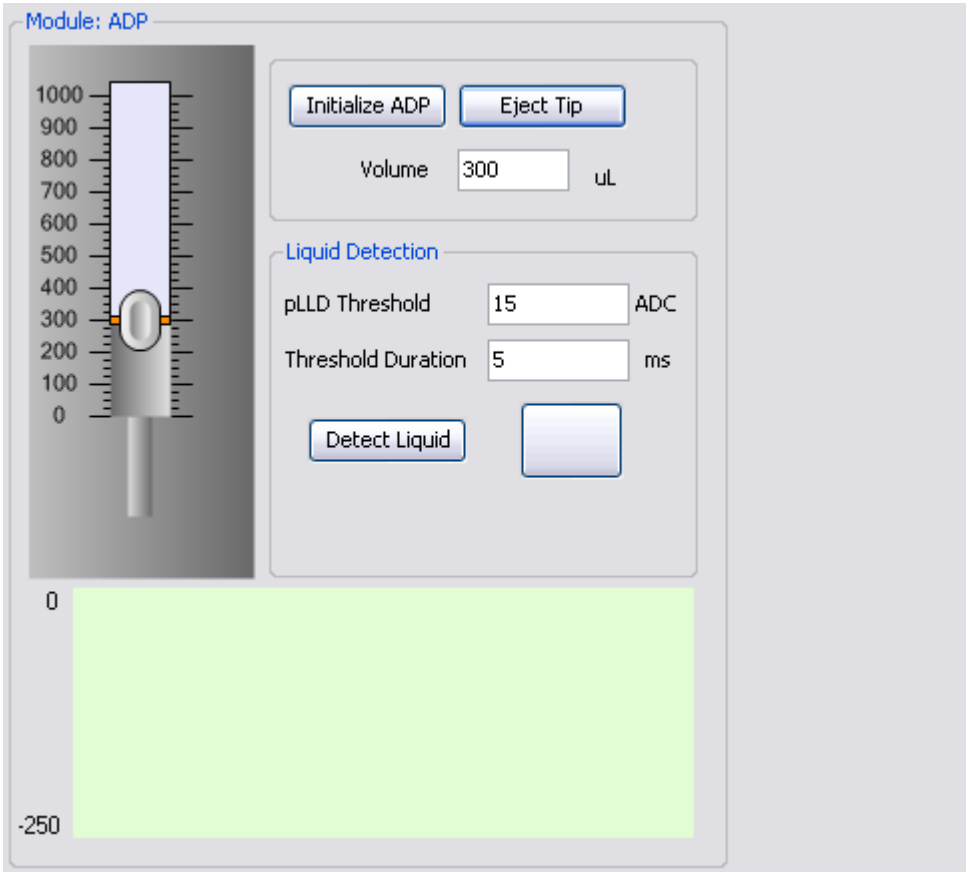
5.9.4 Handling Disposable Tips

To eject a disposable tip, click the Eject Tip button. This is equivalent to issuing an Eject Tip command.

If present, the tip will pop off the end of the probe assembly and the graphic will update to show no tip present.

If no tip is present, clicking Eject Tip will display Tip Lost error message.

Figure 5-15 *ADP with No DiTi Present*



To attach a new disposable tip, physically mount the new DiTi on the ADP probe assembly. The graphic will update to show the tip attached.

5.9.5 Pressure Liquid Level Detection (pLLD)

The Liquid Detection section of the ADP Module graphic displays two values:

- pLLD Threshold
- Threshold Duration

Liquid level is detected by performing an aspirate/dispense action while lowering the probe assembly until the presence of liquid is detected through changes in air pressure.

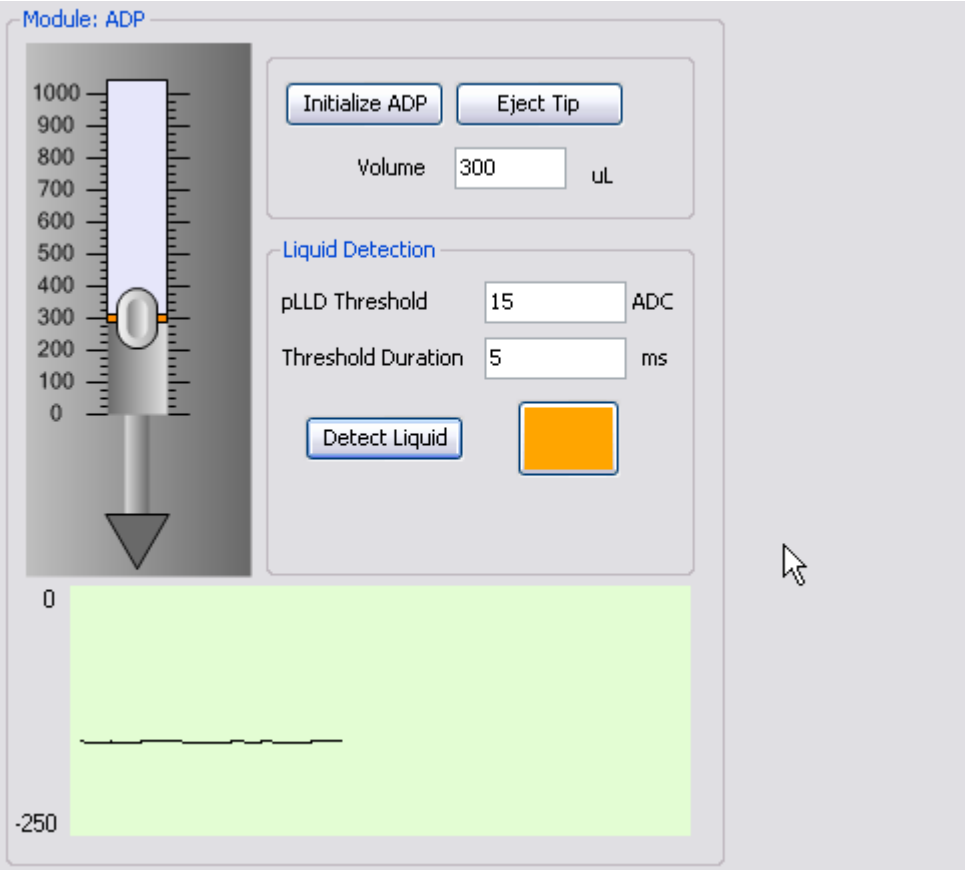
The pLLD threshold is the point at which the pressure differential exceeds a predetermined level, identifying a difference in pressure sufficient to signal the presence of liquid. The default pLLD threshold value is 15 ADC.

The threshold duration is the length of time, in milliseconds, that the pressure remains above the threshold. The default threshold duration value is 5 ms.

To detect liquid level using pLLD through the Cavro Fusion software, click the Detect Liquid button.

The status box next to the Detect Liquid button will turn orange to indicate that liquid detection is in process.

Figure 5-16 *Liquid Level Detection Begins*



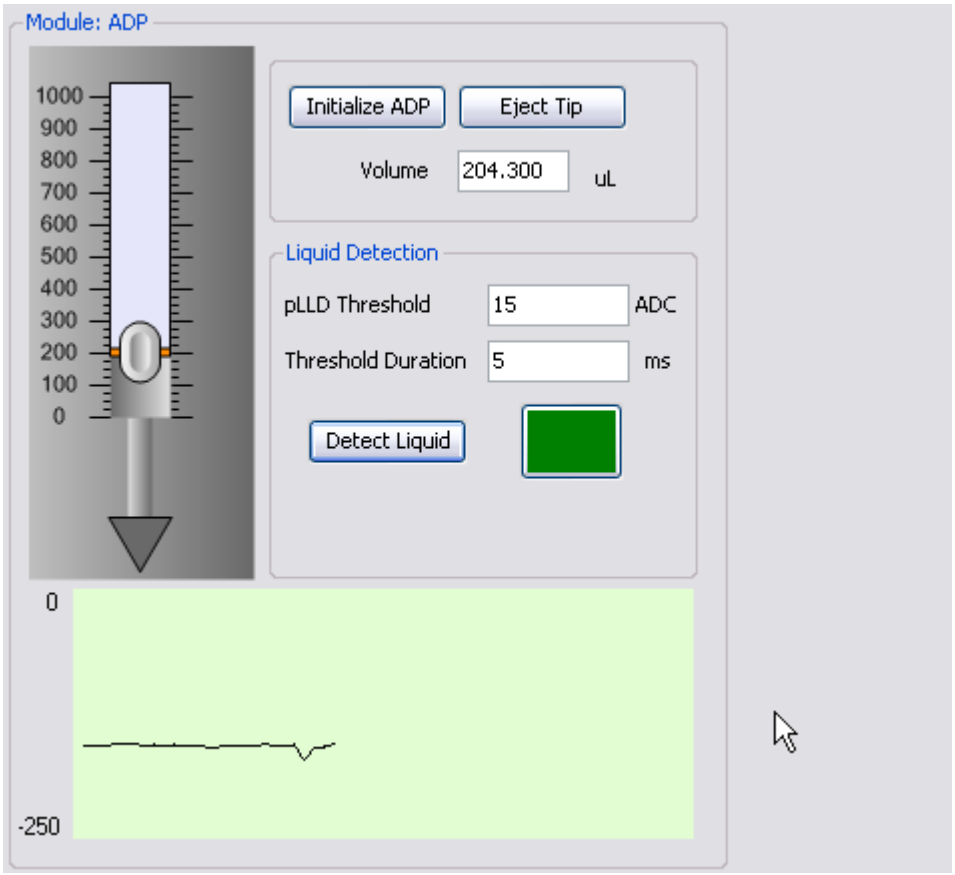
The green panel below the ADP module graphic displays pressure data during the liquid detection process.

When liquid level is detected, the following things happen simultaneously (see Figure 5-17):

- The pressure data line shows a significant dip
- System instantly stops monitoring liquid level detection
- Aspirate/dispense action instantly ceases
- Status box turns green
- Device returns to “ready” state

If no liquid is detected, or the pressure sensor fails, the status box turns red.

Figure 5-17 *Liquid Level Detected Using Fusion*

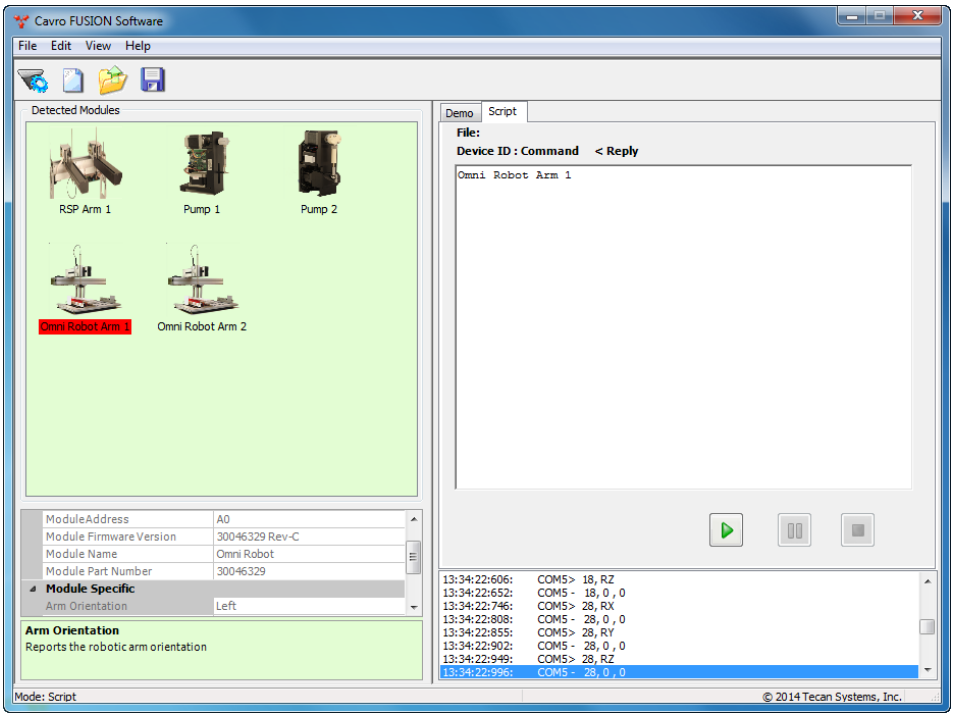


6 Using the Script Tab

The Script tab (Figure 6-1) provides the interface to write scripts to operate all connected modules other than Embedded Omni.

Note: *You cannot use the Script tab to operate Omni in Embedded mode. To use Omni in Embedded mode, see "Omni Embedded Mode Settings" on page 3-6 and Chapter 7, "Using the Omni Embedded Script Window".*

Figure 6-1 *Script Tab*



View the Script tab at any time by clicking the Script tab next to the Demo tab.

The appearance and use of the Script tab is the same whether a pump module or robotic module is selected in the Detected Modules pane.

Note that when the Script tab is visible, three new buttons are available below the menu bar: the New button, Open button, and Save button. The use of these buttons is documented later in this chapter.

Note: *The commands used in a script to operate a module are dictated by the type of module that will execute the commands, not by the Cavro Fusion software. Keep the documentation for your hardware modules close at hand for reference on the commands accepted by the modules.*

6.1 Starting a New Script

To start a new script from a blank file in the Script tab, click the New button (Figure 6-2).

Figure 6-2 *New Button*

If the script has not been saved, Cavro Fusion prompts you to save it before a new script is opened.

You can also start a new script using the New command on the File menu, or by typing Ctrl-N.

6.2 Opening a Script

Open a script by clicking the Open button (Figure 6-3).

Figure 6-3 *Open Button*

If the script has not been saved, Cavro Fusion prompts you to save it before a new script is opened.

The Windows Open window appears. Open the script as you would any file on a Windows computer.

The procedure for opening a script is identical for scripts written in Cavro Fusion and scripts written with a text editor.

You can also open a script using the Open command on the File menu, or by typing Ctrl-O.

6.3 Saving a Script

Save a script as a .txt file at any time by clicking the Save button (Figure 6-4).

Figure 6-4 *Save Button*

The Windows Save As window appears. Save the script as you would any file on a Windows computer.

You can also save your script using the Save command on the File menu, or by typing Ctrl-S.

6.4 Entering and Reading Script Commands

Add commands to a script simply by typing the command and pressing Enter.

Script commands must conform to the following format:

DeviceID:com [param]

or

DeviceID>com [param]

where:

- *DeviceID* is the module name (as shown in the Detected Modules pane) that represents the port of the device, or the IP address and port of the robot.
- *:* indicates that the command will be sent to the device after the previous command is finished (that is, “Send and Wait”).
- > indicates that the command will be sent to the device immediately, without waiting for the previous command to finish (that is, “No Wait”).

Note: *Either : or > must be entered, but not both.*

- *com* is the command.
  In the case of the Omni Robot, the command must include the DeviceID of the component (arm or axis) to which the command should be directed (for example A0CavroInit or M1CavroInit).
- *param* is the list of parameters required by the command (if required).

For example:

RSP Arm 1:PI

Omni Robot Arm 1:M1Get, Pos

Pump1:ZR

WARNING! *If a command is sent in "No Wait" mode, then there will be no answer from the module and there may be no error information returned, depending on the module and command used.*

If the currently-selected module is a diluter or a pump, then the software automatically appends an "R" to the end of the command; this addition tells the module to execute the command, and is not needed for robotic modules.

The DeviceID name of the device that is currently selected in the Detected Modules pane appear automatically at the beginning of the bottom line of the script. Selecting a new module results in the DeviceID name automatically changing to the value for the newly-selected module.

6.5 Using Variables

You can use variables in your script to store values, perform limited calculations, and act as loop counters. Variables can be used in commands such as Assign, Add and Loop as well as in low level device commands to replace numerical constants.

Cavro Fusion scripting supports ten predefined variables: (V0), (V1), (V2), (V3), (V4), (V5), (V6), (V7), (V8), and (V9). The variable name includes the surrounding parentheses: (V1) is a valid variable in a script command; V1 will produce a syntax error.

(V0) is a reserved variable, used to store the reply from instrument after executing a command by the script. For example, after execution of the script line

Omni Robot Arm 1:M1Get, Pos

(V0) will contain current position value for axis M1

6.5.1 Using the Assign Command

You can use an Assign command to assign values to a variable. You can assign an integer or floating point value to a variable:

Assign (V1) 5.0 assigns the value 5.0 to variable (V1).

You can also assign the value of one variable to another:

Assign (V2) (V1) assigns the value stored in variable (V1) to variable (V2).

Note: *A space is required between the keyword Assign, the variable, and the value or second variable: Assign<space>(V1)<space>5.0. The variable name must include the parentheses.*

6.5.2 Using the Add Command

The Fusion scripting language supports addition of two variables, or addition of a variable and a numeric constant (integer or floating point value).

Add (V1) 5.0 Adds the value 5.0 to the value stored in (V1). The result is stored in (V1).

Add (V1) (V2) Adds the value stored in (V2) to the value stored in (V1), and the result is stored in (V1)

Note: *A space is required between the Add keyword, the variable name, and the value or second variable: Add<space>(V1)<space>(V2).*

6.5.3 Using Variables in a Loop Command

You can use a variable in the LOOP command to represent how many times the loop should be executed. The value stored in the variable must be an integer.

See the end of Section 6.6, Using Loops for an example of a loop using a variable.

6.5.4 Using Variables in Device Commands

You can use variables in low level device commands to numerical constants. For example:

Assign (V1) 105.23

Assign (V2) 150.45

Assign (V3) 50.5

Omni Robot Arm 1:A0MoveAbs,(V1),(V2),(V3)

When sending the command to the robot, each variable will be replaced by the numerical value it represents. The keyword and the following parameters must be separated by spaces.

6.6 Using Loops

You can use loops in your script to perform a command or set of commands a given number of times. The syntax of a loop is:

LOOP loop_name number

...

ENDLOOP loop_name

where:

- *loop_name* is the name of the loop. The loop name must be unique within the script.
- *number* is the number of times to execute the loop.

Note: *The value of loop_name in the ENDLOOP command must match the value of loop_name in the closest preceding LOOP command; if the values do not match, an error is raised and the script will not execute.*

For example:

LOOP Rows 12

...

ENDLOOP Rows

will execute the contents of the loop named Rows twelve times.

Multiple loops can be nested. For example:

LOOP Rows 12

. . .

LOOP Columns 8

. . .

ENDLOOP Columns

ENDLOOP Rows

As discussed previously, variables can be used as the loop counter. For example:

Assign (V5) 8

Loop TestLoop (V5)

. . .

ENDLOOP TestLoop

will execute the contents of loop TestLoop eight times.

6.7 Using Comments

As with programs written in any other programming language, you are encouraged to comment your Cavro Fusion scripts, to ensure that anyone can understand the script and update it in the future.

Indicate comments in your script with two slashes (//). The Cavro Fusion script engine will ignore any text that falls between the two slashes and the end of the line.

6.8 Using SmartString

SmartString functionality allows you to insert common script commands with a mouse click.

SmartString functionality is available only if SmartString is on. See “SmartString Settings” on page 3-3 for instructions on turning SmartString on and off.

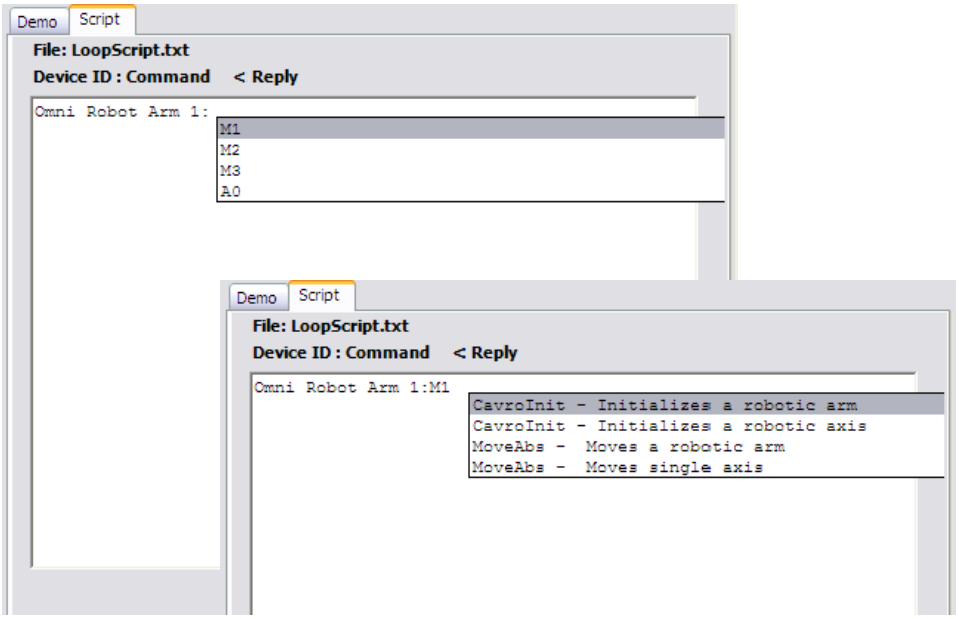
When you type the : or > character after the port and address, the SmartString menu appears and lists the most common commands used by the device.

For the Cavro Omni Robot, the first SmartString menu lists the devices (arm and axes) available, and then provides a list of common commands.

Note: *The SmartString menu does not list all of the commands accepted by the device; it lists only the most commonly-used commands.*

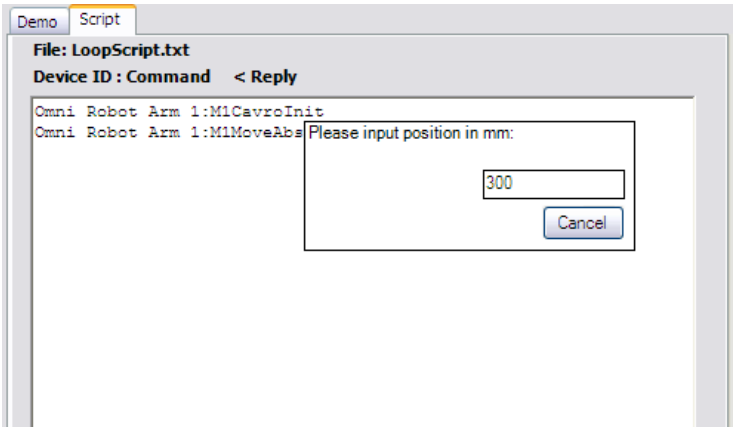
Select a command by clicking it in the SmartString menu. Figure 6-5 shows the progression of SmartString menus for a Cavro Omni Robot.

Figure 6-5 *SmartString Menus for a Cavro Omni Robot*



If the command requires one or more parameters, a window appears and requests the required parameter (Figure 6-6).

Figure 6-6 *SmartString Parameter Window*



Enter the parameter in the text field and hit Enter.

If any further parameters are needed, a new window will appear; continue adding parameters and hitting Return until all needed parameters are entered.

When all parameters are entered (or if no parameters are needed), the command is inserted in the Script tab.

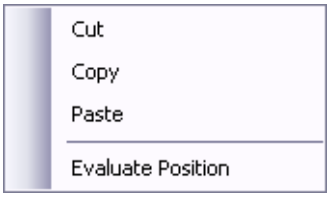
6.9 Performing a Position Evaluation on a Robot

To discover the coordinates of an arm position on an RSP or Cavro Omni Robot:

1 In the Script tab, move the cursor to a line with the device ID of the desired robot.

2 Right-click to display a pop-up menu.

Figure 6-7 *Pop-up Menu*



3 In the pop-up menu, select Evaluate Position.

The arm is unlocked and a message appears, prompting the user to manually move the arm to the desired position.

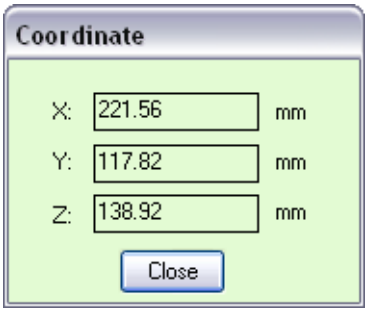
Figure 6-8 *Evaluate Position Message*

4 Move the arm to the desired location.

5 Click the OK button.

The Coordinate window appears and displays the X, Y, and Z coordinates of the current position.

Figure 6-9 *Coordinate Window*



6 Click the Close button to end the evaluation and lock the arm.

WARNING! *After ending the evaluation, the arm could move unexpectedly. Take care to remove your head or hands from the path of moving parts before clicking the Close button; a risk of injury, blunt force trauma, or piercing exists.*

6.10 Controlling an RSP Arm with Device 19 or Device 29

If a one- or two-arm RSP is detected on a particular serial port, the Cavro Fusion scripting interface can communicate directly with the left or right arm (bypassing the Central Control Unit) as device 19 or device 29, respectively.

The device ID RSP Arm 1 is individually controlled with device 19 script commands addressed as “MoloX 1”. Similarly, the device ID RSP Arm 2 has device 29 identified as address “MoloX 2”.

For more information about manipulating device 19 and device 29, see your RSP documentation.

6.11 Running a Script

Run an entire script by ensuring that no commands are selected and clicking the Run button (Figure 6-10).

Figure 6-10 *Run Button*

Run a portion of a script by selecting a continuous string of commands and clicking the Run button.

Run a single line of a script at any time by double-clicking the desired command.

As the script runs, the command that is currently being executed is highlighted in the Script tab.

If one of the commands results in a message being returned from a device, then the message appears in the Script tab in the following format:

DeviceID[:][>]com [param] **<answer**

where:

- < indicates that the following text is a message from the device.
- *answer* is the message.

If one of the commands raises an error, then the script stops and an error message appears on the screen in the following format:

DeviceID[:][>]com [param] ***error**

where:

- * indicates that the command raised an error.
- *error* is the error description.

6.12 Pausing a Script

6.12.1 Using the Pause Button

Pause a script while it is running by clicking the Pause button (Figure 6-11).

Figure 6-11 *Pause Button*

Restart a command at the point where it left off by clicking the Run button again.

Note: *A script cannot pause in the middle of a command; if you click the Pause button while a command is in progress, the script will pause after the current command has finished.*

6.12.2 Using the Pause Command

Pauses can be inserted in the script with the Pause command. The syntax for the Pause command is:

<Pause> <number of milliseconds>

For example, the following command pauses script execution for 100 milliseconds and then resumes with the next command line:

Pause 100

6.13 Stopping a Script

Stop a script by clicking the Stop button (Figure 6-12).

Figure 6-12 *Stop Button*

Note: *A script cannot stop in the middle of a command; if you click the Stop button while a command is in progress, the script will stop after the current command has finished.*

6.14 Example: Writing and Executing a Script

The sample script that appears in this section uses the following modules and device IDs:

- Cavro Centris pump - Pump 1
- Cavro Omni Robot, Arm 1 - Omni Robot Arm 1

Perform the following steps to write and execute a script that will automatically initialize all three attached modules:

1 In the Cavro Fusion window, click the Script tab.

The Script tab appears, with a new script opened and the device ID of the currently-selected module inserted on the first line.

2 Double-click the Cavro Centris pump in the Detected Modules pane.

The address on the first line of the script changes to Pump 1.

3 Enter a colon after the device ID.

The SmartString menu appears and displays commands for the Cavro Centris pump.

4 Select Z - Initialize Plunger.

The command is inserted after the colon.

5 Press the Enter key.

An R is inserted at the end of the command, and the device ID Pump 1 appears on the next line.

6 Double-click the Omni Robot Arm 1 in the Detected Modules pane.

The device ID on the current line of the script changes to Omni Robot Arm 1, which is the device ID of the left arm.

7 Enter a colon after the device ID.

The SmartString menu appears and displays commands for the Omni.

8 Select A0, which is the address of the arm.

A new SmartString menu appears and displays the commands for the Omni arm.

9 Select CavroInit - Initializes a robotic arm.

The command is inserted after the colon.

10 Press the Enter key.

The device ID Omni Robot Arm 1 appears on the next line.

11 Click the Run button.

The Cavro Centris pump and the Omni return to their initial positions and are initialized.

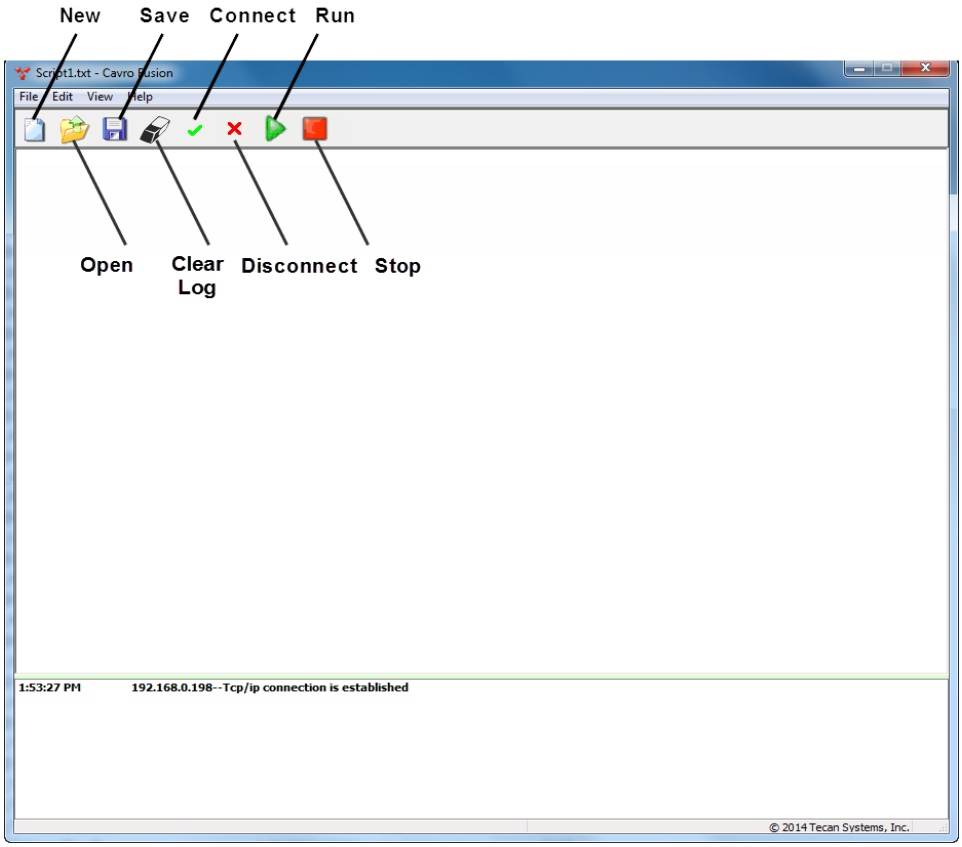
This page has been left intentionally blank.

7 Using the Omni Embedded Script Window

The Omni Embedded Script window provides the interface to write scripts to operate embedded Omni. See Chapter 6, “Using the Script Tab” for information about writing scripts for RSP / MSP, pump and Omni in Command Processor mode.

Note: *By default Cavro Fusion operates Omni in Command Processor Mode. To use Omni Embedded mode, see “Omni Embedded Mode Settings” on page 3-6 and then return to this chapter for information about writing scripts.*

Figure 7-1 *OE Script Window*



- New: clears content in the editor
- Open: loads previously saved script into the editor
- Save: saves the script in text format
- Clear Log: clears content in the log window

- Connect: establishes communication with the embedded Omni through the selected communication channel
- Disconnect: closes communication with the Omni
- Run: starts to execute the script or selected lines
- Stop: stops the current script execution

Note: *The commands used in a script to operate a module are dictated by the type of module that will execute the commands, not by the Cavro Fusion software. Keep the documentation for your hardware modules close at hand for reference on the commands accepted by the modules.*

7.1 Starting a New Script

To start a new script from a blank file in the Script window, click the New button (Figure 7-2).

Figure 7-2 *New Button*

If the script has not been saved, Cavro Fusion prompts you to save it before a new script is opened.

You can also start a new script using the New command on the File menu, or by typing Ctrl-N.

7.2 Opening a Script

Open a script by clicking the Open button (Figure 7-3).

Figure 7-3 *Open Button*

If the script has not been saved, Cavro Fusion prompts you to save it before a new script is opened.

The Windows Open window appears. Open the script as you would any file on a Windows computer.

The procedure for opening a script is identical for scripts written in Cavro Fusion and scripts written with a text editor.

You can also open a script using the Open command on the File menu, or by typing Ctrl-O.

7.3 Saving a Script

Save a script as a .txt file at any time by clicking the Save button (Figure 7-4).

Figure 7-4 *Save Button*

The Windows Save As window appears. Save the script as you would any file on a Windows computer.

You can also save your script using the Save command on the File menu, or by typing Ctrl-S.

7.4 Entering and Reading Script Commands

Add commands to a script simply by typing the execution mode indicator (: or >) followed by an OE command and pressing Enter.

Script commands must conform to the following format:

[execution mode indicator] [OE command]

where:

- *Execution mode indicator* can be : or >
  - *:* indicates that the command will be sent to the devices after the previous command is finished (that is, “Send and Wait”).
  - > indicates that the command will be sent to the devices immediately, without waiting for the previous command to finish (that is, “No Wait”).

Note: *Either : or > must be entered, but not both.*

- *OE command* see the Omni Operator’s Manual for OE command syntax.

For example:

- To initialize Omni X, Y, Z axes and ADP addressed at 0 in sequence:

  :1/1Z

  :1/2Z

  :1/3Z

  :1/0WR

- To initialize Omni X, Y, Z axes and ADP addressed at 0 simultaneously:
  >1/1Z
  >2/2Z
  >3/3Z
  :4/0WR

7.5 Using Variables

You can use variables in your script to store values, perform limited calculations, and act as loop counters. Variables can be used in commands such as Assign, Add and Loop as well as in low level device commands to replace numerical constants.

Cavro Fusion scripting supports ten predefined variables: (V0), (V1), (V2), (V3), (V4), (V5), (V6), (V7), (V8), and (V9). The variable name includes the surrounding parentheses: (V1) is a valid variable in a script command; V1 will produce a syntax error.

(V0) is a reserved variable, used to store the reply from instrument after executing a command by the script. For example, after execution of the script line

:1/1?0

(V0) will contain current position value for axis M1

7.5.1 Using the Assign Command

You can use an Assign command to assign values to a variable. You can assign an integer or floating point value to a variable:

Assign (V1) 5.0 assigns the value 5.0 to variable (V1).

You can also assign the value of one variable to another:

Assign (V2) (V1) assigns the value stored in variable (V1) to variable (V2).

Note: *A space is required between the keyword Assign, the variable, and the value or second variable: Assign<space>(V1)<space>5.0. The variable name must include the parentheses.*

7.5.2 Using the Add Command

The Fusion scripting language supports addition of two variables, or addition of a variable and a numeric constant (integer or floating point value).

Add (V1) 5.0 Adds the value 5.0 to the value stored in (V1). The result is stored in (V1).

Add (V1) (V2) Adds the value stored in (V2) to the value stored in (V1), and the result is stored in (V1)

Note: *A space is required between the Add keyword, the variable name, and the value or second variable: Add<space>(V1)<space>(V2).*

7.5.3 Using Variables in a Loop Command

You can use a variable in the LOOP command to represent how many times the loop should be executed. The value stored in the variable must be an integer.

See the end of Section 7.6, Using Loops for an example of a loop using a variable.

7.5.4 Using Variables in Device Commands

You can use variables in low level device commands to numerical constants. For example to move the left arm to position 105,150,50:

Assign (V1) 200

Assign (V2) 105

Assign (V3) 150

Assign (V4) 50

:1/3A(V1)(/1A(V2)/2A(V3))/3A(V4)

When sending the command to the robot, each variable will be replaced by the numerical value it represents. In the example above, first axis 3 moves to 200 (V1); then axes 1 and 2 move simultaneously to 105 (V2) and 150 (V3) respectively; finally axis 3 moves to 50 (V4). For detailed information about embedded command execution sequence, see the *Omni Operator's Manual* Version 4.0 or later.

7.6 Using Loops

You can use loops in your script to perform a command or set of commands a given number of times. The syntax of a loop is:

LOOP loop_name number

...

ENDLOOP loop_name

where:

- *loop_name* is the name of the loop. The loop name must be unique within the script.
- *number* is the number of times to execute the loop.

Note: *The value of loop_name in the ENDLOOP command must match the value of loop_name in the closest preceding LOOP command; if the values do not match, an error is raised and the script will not execute.*

For example:

LOOP Rows 12

...

ENDLOOP Rows

will execute the contents of the loop named Rows twelve times.

Multiple loops can be nested. For example:

LOOP Rows 12

. . .

LOOP Columns 8

. . .

ENDLOOP Columns

ENDLOOP Rows

As discussed previously, variables can be used as the loop counter. For example:

Assign (V5) 8

Loop TestLoop (V5)

. . .

ENDLOOP TestLoop

will execute the contents of loop TestLoop eight times.

7.7 Using Comments

As with programs written in any other programming language, you are encouraged to comment your Cavro Fusion scripts, to ensure that anyone can understand the script and update it in the future.

Indicate comments in your script with two slashes (//). The Cavro Fusion script engine will ignore any text that falls between the two slashes and the end of the line.

7.8 Using the Pause Command

Pauses can be inserted in the script with the Pause command. The syntax for the Pause command is:

<Pause> <number of milliseconds>

For example, the following command pauses script execution for 100 milliseconds and then resumes with the next command line:

Pause 100

7.9 Running a Script

Run an entire script by ensuring that no commands are selected and clicking the Run button (Figure 7-5).

Figure 7-5 *Run Button*

Run a portion of a script by selecting a continuous string of commands and clicking the Run button.

Run a single line of a script at any time by double-clicking the desired command.

As the script runs, the command that is currently being executed is highlighted in the Script tab and the script window is disabled. The script window is enabled when the command finishes executing.

If one of the commands results in answer(s) being returned from device(s), then the answer(s) appear in the Script tab in the following format:

[execution mode indicator][OE command]<(Device ID, reply string)
(Device ID, reply string)....

where:

- < indicates that the following text contains answer(s) from the device(s).
- (Device ID,reply string) indicates the answer from that device.

For example:

:1/1?0/2?0/3?0 <(1, 51.000)(2, -0.040)(3, 88.825)

If one of the commands raises an error, then the script stops and an error message appears on the screen in the following format, asking whether to continue or terminate execution as shown in Figure 7-6.

[execution mode indicator][OE command] *error

where:

- * indicates that the command raised an error.
- *error* is the error description.

For example:

:1(/1Z/2Z/3Z) *Device ID: 1 Error Code: 0x0067

Figure 7-6 *Omni Embedded error window*

7.10 Stopping a Script

Stop a script by clicking the Stop button (Figure 7-7).

Figure 7-7 *Stop Button*

Note: *Clicking the Stop button sends a global termination O/T command to the instrument and stops all device operations immediately.*

7.11 Example: Writing and Executing a Script

The following script demonstrates how to program Omni in Embedded mode to transfer a reagent from a source to the wells of micro plate. It shows liquid handling operations such as:

- picking up and dropping a disposable tip
- detecting liquid using the ADP's pLLD mechanism
- aspirating and dispensing a reagent with the ADP

The example shows how to use variables and loops to iterate the operations.

The hardware configuration for this example is as follows:

- Left arm Omni with a Universal Z with ADP, with these addresses:
  - Axis X: 1
  - Axis Y: 2
  - Axis Z: 3
  - ADP: 0
- Z has an assumed travel pos of 200

Example

1 Initialize the left arm and ADP. Make sure no tip is mounted on ADP:

:1((/3Z)/1Z/2Z)/0WR

2 Assign the X position of the 1st tip of the DiTi tip rack to 200:

Assign (V1) 200

3 Assign the X position of the 1st well of the 96-well micro plate to 100:

Assign (V2) 100

4 Begin a loop named “Reagent” that executes three times:

LOOP Reagent 3

5 Assume the Y position of the first row of DiTi tip rack is 150; move the arm to the first tip at the first iteration; then move the arm to the second tip at the second iteration, and so on:

:1((/3A200)/1A(V1)/2A150)

6 Pick up tip, starting from position 100, descending as much as 50 mm from 100, and stopping at or before 75 mm below 100:

:1/3P100,50,75

7 Raise the arm to position Z:200 and move the arm to a reagent rack positioned at X:350, Y:75:

:1((/3A200)/1A350/2A75)

8 Set the ADP to pLLD mode. Liquid detection must use single detection mode:

:1/0U70R

9 Detect liquid with submerge distance set to 2 mm using single detection mode; start from position 180 and descend as far as 0:

:1(/0B1R/3u21,2u25,1B180,0)

10 Aspirate 200 µl:

:1/0P200,1R

11 Assume the Y position of the first row of the 96-well micro plate is 100. Raise the arm to position Z:200; then move the arm to the first well at the first iteration; then move the arm to the second well at the second iteration and so on; lower the arm to position Z:130:

:1(((/3A200)/1A(V2)/2A100)/3A130)

12 Dispense 200 µl:

:1/0D200,1R

13 Raise arm to position Z:200; then move arm to a waste container positioned at X:50, Y:50; eject tip:

:1((/3A200)/1A50/2A50)/0ER

14 Increment X position to the next tip:

Add (V1) 8

15 Increment X position to the next well:

Add (V2) 8

16 End the Reagent loop:

ENDLOOP Reagent

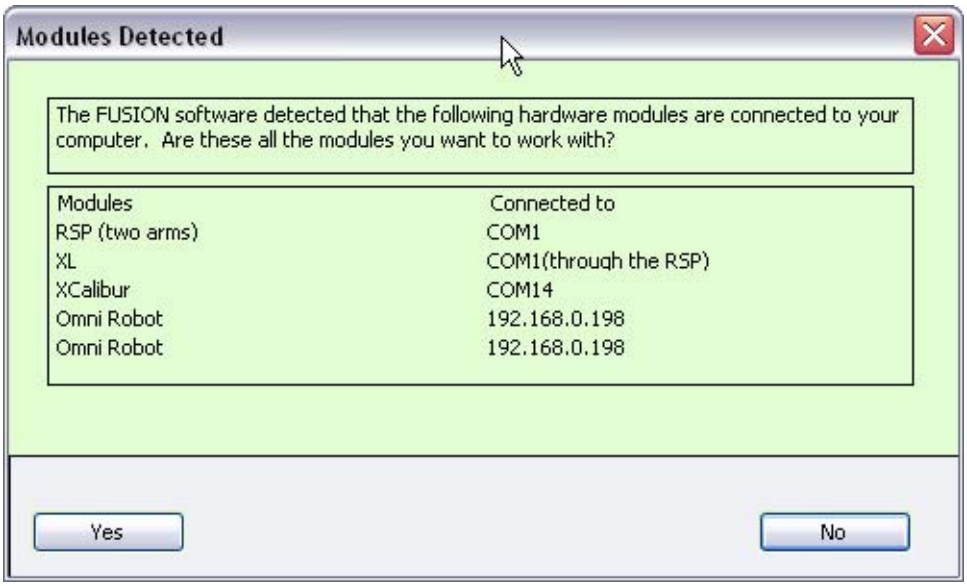
A Troubleshooting

The following sections contain the solutions for some common Cavro Fusion issues. If you have an issue similar to one described below, please try the solution provided before contacting Tecan for help.

A.1 The Scan Did Not Detect All Connected Devices

At the conclusion of a scan for modules, the results of the scan are shown in the Detected Modules window .

Figure A-1 *Detected Modules window (from chapter 3)*

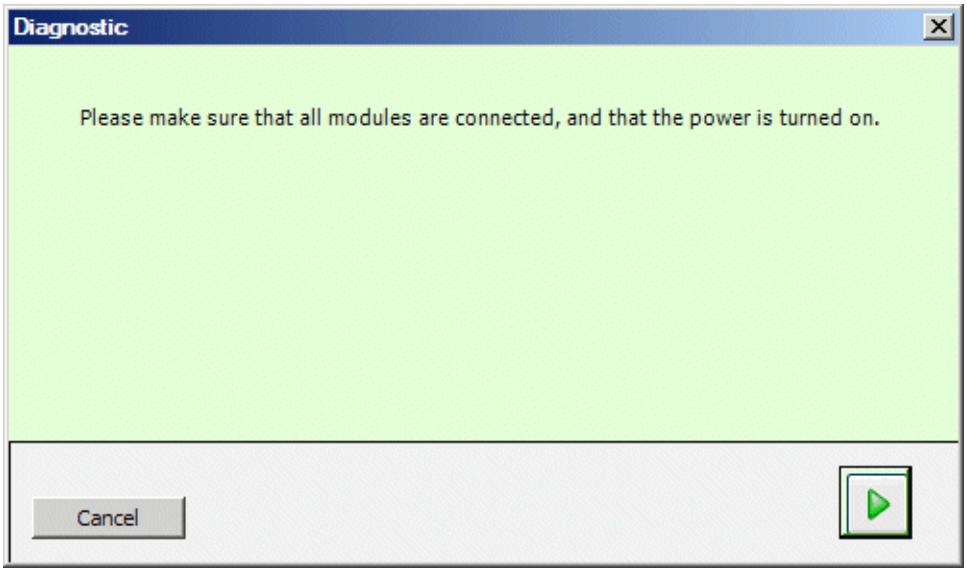


If the list of detected modules does not include all connected devices, then perform the following tasks to correct some common errors:

1 In the Detected Modules window, click No.

The first page of the diagnostic process appears (Figure A-2).

Figure A-2 *Diagnostic Page 1*



Note: *You can exit the diagnostic and return to the Cavro Fusion window at any time by clicking Cancel.*

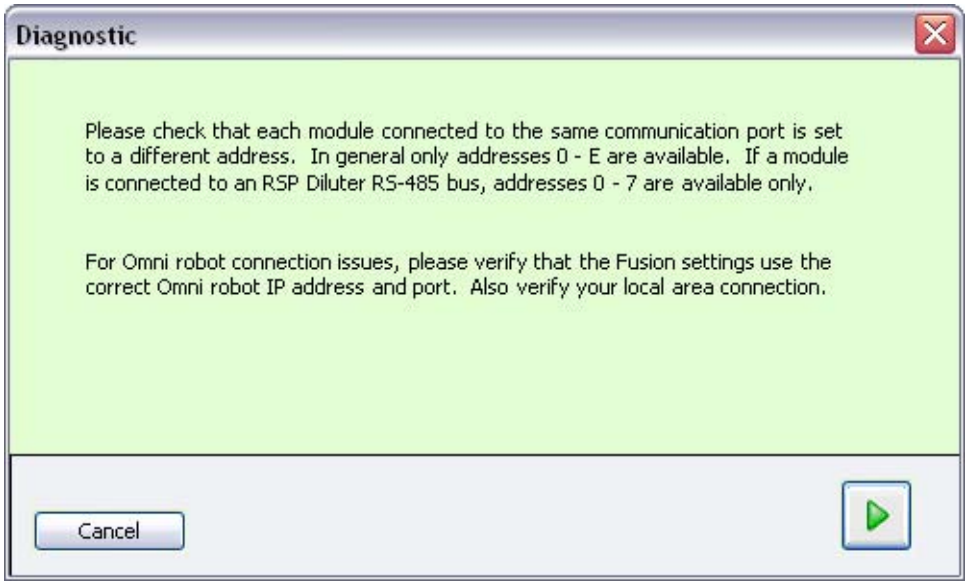
2 Check that all of the modules are properly connected and turned on.

3 If you are using a Cavro Omni robot, make sure that its Category 5 Patch cable is properly connected to the PC and the Omni.

4 Click the Next button , at the bottom right.

5 The second page of the diagnostic process appears (Figure A-3).

Figure A-3 *Diagnostic Page 2*



6 Check that all devices connected to the same serial port are set to unique addresses.

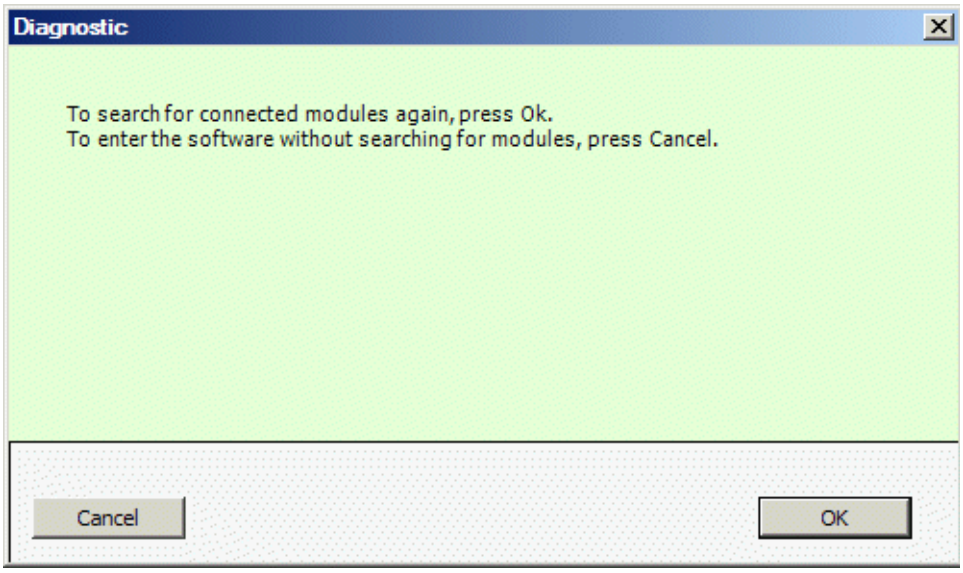
7 Check that all devices connected to the same Cavro Omni Robot are set to unique addresses.

8 Verify that the Cavro Omni Robot IP address and port are properly set in the Fusion Settings window, as described in “Scanning Options” on page 3-4.

9 Click the Next button.

The third page of the diagnostic process appears (Figure A-4).

Figure A-4 *Diagnostic Page 3*



10 Click the OK button to scan for modules again.

A.2 The Module is Not Responding to Commands

If the module is not responding to commands from either the Demo tab or the Script tab, the module might not be initialized.

Initialize the module by selecting the module in the Detected Modules window and clicking the **Initialize Pump** or **Initialize Arm** button in the Demo tab.

A.3 The Module is Not Responding to a Script

If a module that was previously responding to commands (either from a script or from the Demo tab) is not responding to commands from a script, the module could need to be initialized explicitly in the script. To correct this, insert the command to initialize the module into your script before the first command used to manipulate the module.

B Technical Service

For information or questions regarding operating the Cavro Fusion software, please contact Tecan Systems Technical Service using one of the methods listed below.

By phone	408-953-3100 or 800-231-0711
By fax	408-953-3101
By e-mail	helpdesk-sy@tecan.com

Technical support is available 7:00 a.m. to 5:30 p.m. PST, Monday through Friday.

Our mailing address is:

Tecan Systems, Inc.
2450 Zanker Road
San Jose, CA 95131
USA

When calling for technical service, please have the following information ready

- Description of the problem
- Software version
- PC configuration
- Types, part numbers, and serial numbers of connected modules

This page has been left intentionally blank.

Index