MATLAB

Device Driver User Guide

Agilent Technologies

Notices

User Guide Part Number

G5415-90028

July/2007

Contact Information

Agilent Technologies Inc.
Automation Solutions
5301 Stevens Creek Blvd.
Santa Clara, CA 95051
USA

Technical Support: 1.800.979.4811
or +1.408.345.8011
service.automation@agilent.com

Customer Service: 1.866.428.9811
or +1.408.345.8356
orders.automation@agilent.com

European Service: +44 12081443513
euroservice.automation@agilent.com

Documentation feedback:
documentation.automation@agilent.com

Web: http://www.agilent.com

Acknowledgements

Microsoft and Windows are registered trademarks of the Microsoft Corporation in the United States and other countries.

Warranty

The material contained in this document is provided "as is," and is subject to being changed, without notice, in future editions. Further, to the maximum extent permitted by applicable law, Agilent disclaims all warranties, either express or implied, with regard to this manual and any information contained herein, including but not limited to the implied warranties of merchantability and fitness for a particular purpose. Agilent shall not be liable for errors or for incidental or consequential damages in connection with the furnishing, use, or performance of this document or of any information contained herein. Should Agilent and the user have a separate written agreement with warranty terms covering the material in this document that conflict with these terms, the warranty terms in the separate agreement shall control.

Technology Licenses

The hardware and/or software described in this document are furnished under a license and may be used or copied only in accordance with the terms of such license.

Restricted Rights Legend

If software is for use in the performance of a U.S. Government prime contract or subcontract, Software is delivered and licensed as "Commercial computer software" as defined in DFAR 252.227-7014 (June 1995), or as a "commercial item" as defined in FAR 2.101(a) or as "Restricted computer software" as defined in FAR 52.227-19 (June 1987) or any equivalent agency regulation or contract clause. Use, duplication or disclosure of Software is subject to Agilent Technologies' standard commercial license terms, and non-DOD Departments and Agencies of the U.S. Government will receive no greater than Restricted Rights as defined in FAR 52.227-19(c)(1-2) (June 1987). U.S. Government users will receive no greater than Limited Rights as defined in FAR 52.227-14 (June1987) or DFAR 252.227-7015 (b)(2) (November 1995), as applicable in any technical data.

Safety Noticies

⚠ **A WARNING notice denotes a hazard. It calls attention to an operating procedure, practice, or the like that, if not correctly performed or adhered to, could result in personal injury or death. Do not proceed beyond a WARNING notice until the indicated conditions are fully understood and met.**

A **CAUTION** notice denotes a hazard. It calls attention to an operating procedure, practice, or the like that, if not correctly performed or adhered to, could result in damage to the product or loss of important data. Do not proceed beyond a **CAUTION** notice until the indicated conditions are fully understood and met.

Agilent Technologies

Letter to our Customers

Dear Customer,

The Agilent Technologies acquisition of Velocity11 resulted in the following changes:

- Creation of Agilent Technologies Automation Solutions, formerly Velocity11
- Renaming of some Velocity11 products
- New Customer Service and Technical Support contact information
- New website address for product information

Please make a note of the following changes as they impact this user guide.

Velocity11 product name changes

Velocity11 product name	Changes to ...
Access2 Automated Microplate Loader	Automated Centrifuge Loader
Element Automation System	BioCel 900 System
IWorks Device Driver Programming Interface	VWorks DCL Interface
PlatePierce Seal Piercing Station	Microplate Seal Piercer
VCode Barcode Print and Apply Station	Microplate Barcode Labeler
Velocity11 Robot	3-Axis Robot
VHooks Integration Interface	VWorks Hooks Interface
VPrep Pipetting System	Vertical Pipetting Station
VSpin Microplate Centrifuge	Microplate Centrifuge
VStack Labware Stacker	Labware Stacker

New contact information

Documentation feedback: documentation.automation@agilent.com

Technical Support: 1.800.979.4811 or +1.408.345.8011
service.automation@agilent.com

Customer Service: 1.866.428.9811 or +1.408.345.8356
orders.automation@agilent.com

European Service: +44 12081443513
euroservice.automation@agilent.com

Web: http://www.agilent.com

Contents

Introduction 1

This chapter introduces Velocity11 device drivers and provides some basic procedures that are needed to use them.

A Velocity11 device driver is software that plugs into VWorks or BenchWorks software to allow them to control a specific device.

Before reading this guide, you should be familiar with the VWorks or BenchWorks software user interface. Information about using VWorks or BenchWorks software can be found in the *VWorks Version 3 Automation Control User Guide* or *BenchWorks Automation Control User Guide*.

To set up and use Velocity11 device drivers, become familiar with the content in this guide as well as the guides for the devices that use VWorks or BenchWorks software.

This chapter contains the following topics:

Who should read this guide

Job roles

This user guide is for people with the following job roles:

Job role	Responsibilities
Integrator	Someone who writes software and configures hardware controlled by device drivers.
Lab manager, administrator, or technician	Someone who is responsible for: ❑ Installing device drivers ❑ Managing device drivers ❑ Developing the applications that are run using device drivers ❑ Solving the more challenging problems that might arise ❑ Developing training materials and standard operating procedures for operators
Operator	Someone who performs the daily production work using the device driver and solves routine problems. Your organization may choose to create its own procedures for operators including the procedures in this guide.

Related topics

For information about...	See...
Contacting Velocity11	http://www.velocity11.com/contact.html
Accessing online help	"About Velocity11 user guides" on page 3
Device drivers	"About device drivers" on page 7

About Velocity11 user guides

About this topic

This topic describes the different formats of Velocity11 user information and explains how to access the user information.

Formats available

Velocity11 user information is provided to you as:

- ❑ Online help
- ❑ A PDF file
- ❑ A printed book

The information in each format is the same but each format has different benefits.

Where to find user information

Online help

The online help is added to your computer with the Velocity11 lab automation system software installation.

PDF file

The PDF file of the user guide is on the software CD that is supplied with the product.

Velocity11 website

You can search the online help or download the latest version of any PDF file from the Velocity11 website at www.velocity11.com.

Note: All Velocity11 user information can be searched from the website at www.velocity11.com.

Online help

The online help is the best format to use when you are working at the computer and when you want to perform fast or advanced searches for information.

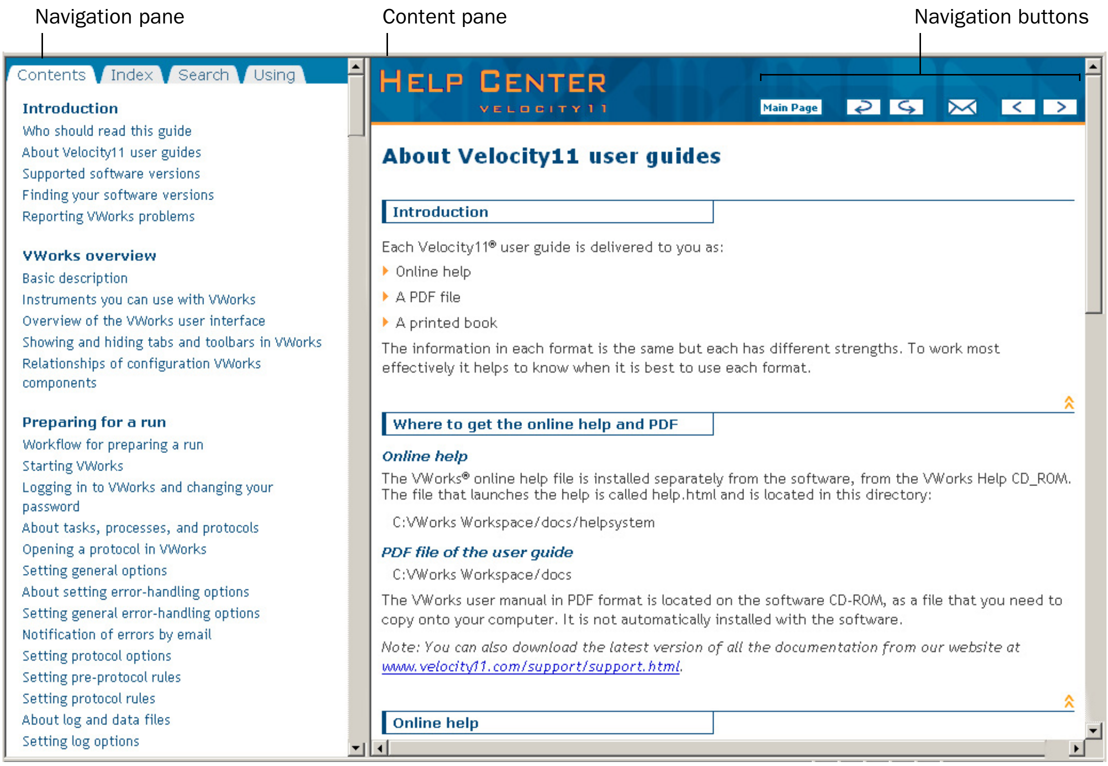
To open the online help:

1. In the Velocity11 lab automation software, press F1. The online help window opens.

Main features

The online help window contains the following:

- ❑ *Navigation pane*. Consists of four tabs. The Contents, Index, and Search tabs provide different ways to locate information. The Using tab contains information about using the help system.
- ❑ *Content pane*. Displays the online help topics.
- ❑ *Navigation buttons*. Enables you to navigate through the pages.The online help includes a navigation pane, content pane, and navigation buttons.



Introduction
Who should read this guide
About Velocity11 user guides
Supported software versions
Finding your software versions
Reporting VWorks problems

VWorks overview
Basic description
Instruments you can use with VWorks
Overview of the VWorks user interface
Showing and hiding tabs and toolbars in VWorks
Relationships of configuration VWorks components

Preparing for a run
Workflow for preparing a run
Starting VWorks
Logging in to VWorks and changing your password
About tasks, processes, and protocols
Opening a protocol in VWorks
Setting general options
About setting error-handling options
Setting general error-handling options
Notification of errors by email
Setting protocol options
Setting pre-protocol rules
Setting protocol rules
About log and data files
Setting log options

About Velocity11 user guides

Introduction

Each Velocity11® user guide is delivered to you as:

- Online help
- A PDF file
- A printed book

The information in each format is the same but each has different strengths. To work most effectively it helps to know when it is best to use each format.

Where to get the online help and PDF

Online help

The VWorks® online help file is installed separately from the software, from the VWorks Help CD_ROM. The file that launches the help is called help.html and is located in this directory:

C:\VWorks Workspace/docs/helpsystem

PDF file of the user guide

C:\VWorks Workspace/docs

The VWorks user manual in PDF format is located on the software CD-ROM, as a file that you need to copy onto your computer. It is not automatically installed with the software.

Note: You can also download the latest version of all the documentation from our website at www.velocity11.com/support/support.html.

Online help

PDF user guides

Computer requirements

To open a user guide in PDF format, you need a PDF viewer. You can download a free PDF viewer from the internet.

Printing and searching

The user guides in PDF format are mainly for printing additional copies. You can perform simple searches in the PDF file, although these searches are much slower than online help searches.

More information

For more information about using PDF documents, see the user documentation for the PDF viewer.

Related topics

For information about...	See...
Who this guide is for	"Who should read this guide" on page 2
What's in this guide	"What this guide covers" on page 5
Device driver plug-ins	"About device drivers" on page 7

What this guide covers

About this topic

This topic presents an overview of what procedures and information are provided in this user guide.

This guide explains how to:

- □ Install the driver for the device
- □ Configure the device in the device manager
- □ Set and use the tasks associated with the device
- □ Use *Device* Diagnostics

Also read

Information about device drivers not covered in this guide and about running VWorks or BenchWorks software can be found in the *VWorks Version 3 Automation Control User Guide* or the *BenchWorks Automation Control User Guide*.

Driver version

To find version information for a driver in VWorks:

1. Start VWorks.
2. Click **Help** and select **About VWorks**.

   The **About VWorks** dialog box lists the version numbers of all the current software for all the devices and plug-ins.

To find version information for a driver in BenchWorks:

1. Start BenchWorks.
2. Click **Help** and select **About BenchWorks**.

   The **About BenchWorks** dialog box lists the version numbers of all the current software for all the devices and plug-ins.

Firmware version

Some devices have firmware installed on them. Because each device is different, the version number may not be the same for all devices.

To find version information for device firmware:

1. Open ***Device*** **Diagnostics** dialog box.
2. Click **About**.

   The **About *Device* Control** message box appears displaying the current version of firmware.

What this guide does not cover

This guide does not cover the following:

- □ The operation of the device
- □ The operation of VWorks or BenchWorks software
- □ Velocity11 devices, such as the PlateLoc Sealer, VCode Microplate Labeler, and VPrep Pipettor when used in stand-alone mode

VWorks or BenchWorks compatibility

If you have purchased a device driver plug-in and are installing it yourself, check with the Velocity11 Technical Support to be sure your version of VWorks or BenchWorks software and the device driver plug-in are using the same version of IWorks software.

BenchWorks versions

Device driver plug-ins used with BenchWorks software may not include some newer features that were specifically added for use with VWorks software and that are described in this manual.

Related topics

For information about...	See...
Who this guide is for	"Who should read this guide" on page 2
User documentation	"About Velocity11 user guides" on page 3
Device driver plug-ins	"About device drivers" on page 7

About devices

About this topic

This topic presents a definition of a Velocity11 device and the device file.

Read this topic if you are unfamiliar with Velocity11 devices and VWorks or BenchWorks software.

Device defined

A device is an item on your lab automation system that has an entry in the device manager. A device can be a robot, an instrument, or a location on the lab automation system that can hold a piece of labware.

Examples of devices:

❑ Velocity11 robot

❑ Human robot

❑ PlateLoc Thermal Plate Sealer

❑ Labcyte Echo550

❑ Platepad

❑ VPrep shelf

❑ Waste

Device file defined

The data entered into the device manager and saved as a device file contains the configuration information for your devices.

Device file location

Device files have the file name format *file name*.dev and are stored in the folder location that you specify when saving the file.

Related topics

For information about...	See...
Device diagnostics	"About diagnostics" on page 11
Device profiles	"About profiles" on page 15
Adding a device to the device manager	"Adding devices" on page 10

About device drivers

About this topic

This topic describes what device drivers are and what they do.

Velocity11 device drivers enable mechanical devices or software programs to work with VWorks or BenchWorks software.

Read this topic if you are:

❑ An administrator in charge of installing device drivers and managing Velocity11 devices

❑ A lab automation system integrator who writes software and configures hardware controlled by VWorks or BenchWorks software

Device driver defined

A Velocity11 device driver enables VWorks or BenchWorks software to control and communicate with the specific type of device. Each type of device that you operate with VWorks or BenchWorks software requires a device driver.

For example, VWorks software uses the:

❑ VPrep Pipettor device driver to communicate with the Velocity11 VPrep Pipettor device

❑ Softmax Reader device driver to communicate with Molecular Devices readers

Plug-in defined

A plug-in is a software program that when added to another program extends it.

Plug-in device drivers

Some device drivers are incorporated directly into the VWorks or BenchWorks software application. Other device drivers are distributed as plug-ins. All the device drivers covered in this guide are the plug-in type.

Advantages of distributing device drivers as plug-ins are:

❑ You only need to install the plug-ins for the devices you use

❑ When new plug-ins become available, they can be easily added. There is no need to re-install the VWorks or BenchWorks software application

IWorks interface

The device driver plug-ins and VWorks or BenchWorks software use IWorks software as a common interface to communicate with each other. Using a common interface allows the creation of a device driver plug-in without the necessity of changing the software.

!! IMPORTANT !! Both VWorks or BenchWorks software and the device driver must be using the same version of IWorks to work properly.

Writing your own device driver

If you are a lab automation system integrator who writes software and configures hardware controlled by VWorks or BenchWorks software, you can write your own driver plug-in for a new device. Contact the Velocity11 Technical Support for information about how to do this.

What functions do the device drivers provide?

Once installed, the following items are enabled:

❑ Tasks associated with the device.

Device-specific tasks appear in the Protocol Tasks list and are available for use in protocol editor processes.

❑ Task parameters associated with the device.

Device-specific task parameters appear in the Protocol Task Parameters toolbar. These determine the conditions with which to execute the tasks of the device.

❑ Diagnostic commands specific to the device.

Device-specific diagnostic commands and options appear in the *Device* Diagnostics dialog box. These commands enable direct control of the device.

Related topics

For information about...	See...
Adding a device to the device manager	“Adding devices” on page 10
Opening diagnostics	“Opening diagnostics” on page 12
Installing a device driver	“Installing device drivers” on page 9
Devices	“About devices” on page 6

Installing device drivers

About this topic

Devices are integrated into VWorks or BenchWorks software using device driver plug-ins. Plug-ins need to be installed before the device can be configured and used.

This topic describes how to install device drivers if they are not already installed on your system. Read this topic if you are an administrator in charge of managing Velocity11 devices.

Procedure

To install device drivers:

1. Insert the device driver installation disc into the CD-ROM of the computer running VWorks or BenchWorks software.
2. Follow the on-screen instructions for installation, selecting the default values when available.
3. When finished, exit VWorks or BenchWorks software.
4. Log off Windows and restart your computer.
5. Start VWorks or BenchWorks software.

For this application...	**The default location for the device driver is...**
VWorks software	C:\VWorks Workspace\bin\plugins
BenchWorks software	C:\Program Files\Velocity11\BenchWorks\plugins

Related topics

For information about...	See...
Device drivers	“About device drivers” on page 7
Opening diagnostics	“Opening diagnostics” on page 12

Adding devices

About this topic

To configure your lab automation system to use a device, you need to add it to a device file in VWorks or BenchWorks software. The VWorks or BenchWorks software device manager uses the information in the device file to communicate and operate the device within the automation system.

This topic describes how to:

□ Create a new device file (if one does not already exist)

□ Add devices

□ Save the device file

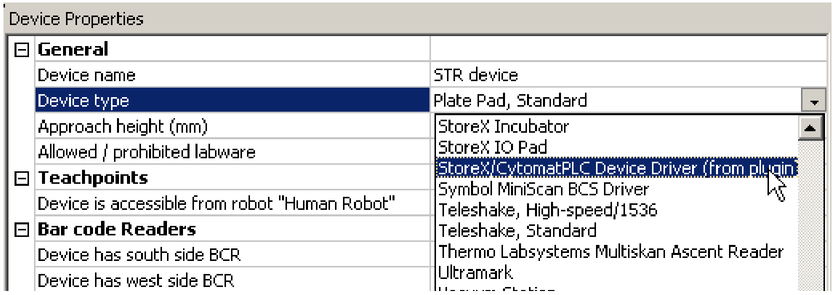
Read this topic if you are an administrator in charge of managing Velocity11 devices.

Procedure

To add devices to a device file:

1. Make sure that the devices are physically networked to the VWorks or BenchWorks software computer and turned on.
2. Start VWorks or BenchWorks software and login as an Administrator.
3. Do one of the following:
   - If you have an existing device file that you want to add to, select **File** > **Device File**, click **Open,** and select your device file.
   - If you are creating a new device file, select **File** > **Device File** and click **New**.
4. Click the **Device Manager** tab.
5. Click **New device** in the **Device List** toolbar and enter a name for the device you are adding.
6. In the device manager, set the **Device type**.

   The default type is **Plate Pad, Standard**.



Device Properties

⊟ **General**	
Device name	STR device
Device type	Plate Pad, Standard
Approach height (mm)	
Allowed / prohibited labware	
⊟ **Teachpoints**	
Device is accessible from robot "Human Robot"	
⊟ **Bar code Readers**	
Device has south side BCR	
Device has west side BCR	

StoreX Incubator
StoreX IO Pad
StoreX/CytomatPLC Device Driver (from plugin
Symbol MiniScan BCS Driver
Teleshake, High-speed/1536
Teleshake, Standard
Thermo Labsystems Multiskan Ascent Reader
Ultramark

7. Repeat step 5 and step 6 for each device.

8. Select **File** > **Device File** > **Save**.

   If you are creating a new device file, you are prompted to enter a name for your device file.

   Alternatively, you can select **File** > **Save All**. This saves the device file and the current protocol file at the same time.

Related topics

For information about...	See...
Device drivers	"About device drivers" on page 7
Setting generic device properties	"Setting the properties for a device" on page 16
Adding a sub-process to a protocol	"Adding and linking Sub Process tasks" on page 19
Opening diagnostics	"Opening diagnostics" on page 12

About diagnostics

About this topic

This topic presents an overview of diagnostics software.

Read this topic if you need to set up or troubleshoot a device running VWorks or BenchWorks software.

Background

Devices can be controlled in real time directly through the VWorks or BenchWorks software Diagnostics using simple commands.

Diagnostics software is used for:

❑ Troubleshooting

❑ Setting teachpoints

❑ Performing manual operations outside a protocol

❑ Creating and editing profiles

For example, if an error occurs during a run that leaves a plate and the robot where they should not be, you can use robot diagnostics to move the plate and return the robot to its home position.

Types of diagnostics software

Devices and robots manufactured by Velocity11 include their own diagnostics software. You can find instructions for using this software in the relevant user guide.

Related topics

For information about...	See...
Opening diagnostics	“Opening diagnostics” on page 12
Adding a device to the device manager	“Adding devices” on page 10
Device drivers	“About device drivers” on page 7
The definition of devices	“About devices” on page 6

Opening diagnostics

About this topic

Every device has diagnostics software to assist you with troubleshooting and setting up the device. This topic describes how to open a device’s diagnostics in VWorks or BenchWorks software.

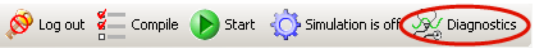
Read this topic if you need to access a device’s diagnostics to perform a device setup task or manually operate a device.

Procedure 1

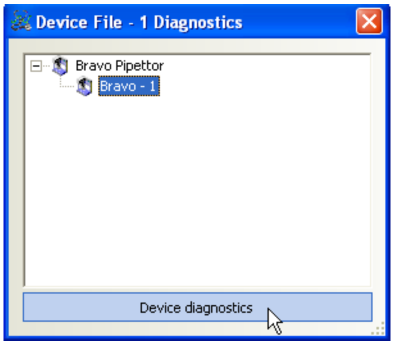
If you are using VWorks4 software

To open Diagnostics:

1. Click **Diagnostics** on the Control toolbar.



2. In the device file’s window, select the device. Expand the general name of the device, if necessary.

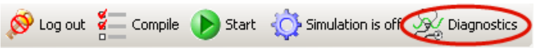


3. Click **Device diagnostics** located at the bottom of the window. The device’s diagnostics dialog box opens.

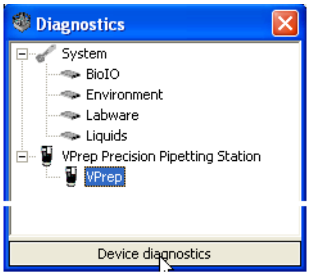
If you are using VWorks3 or BenchWorks software

To open Diagnostics:

1. Click **Diagnostics** on the Control toolbar.



2. In the **Diagnostics** window, select thedevice. Expand the general name of the device, if necessary.



3. Click **Device diagnostics**. The device's diagnostics dialog box opens.

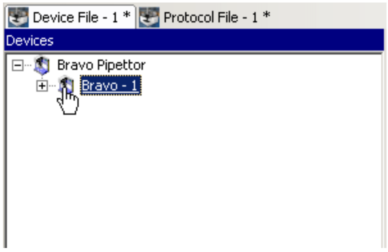
Procedure 2

If you are using VWorks4 software

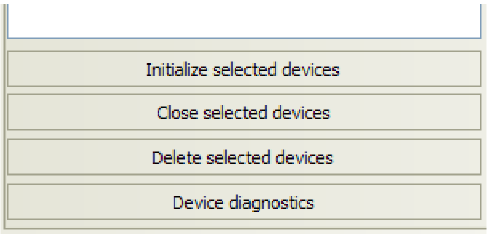
To open Diagnostics:

1. Click the **Device File** tab.
2. Select the device from the **Devices** toolbar.

   Expand the general name of the device, if necessary.



3. Click **Device diagnostics** located at the bottom of the **Devices** toolbar.

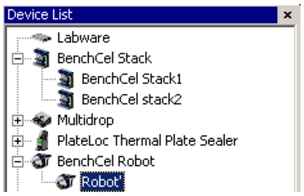


   The device's diagnostics dialog box opens.

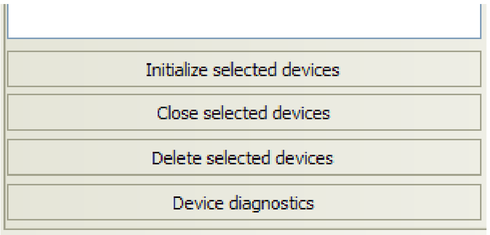
If you are using VWork3 or BenchWorks software

To open Diagnostics:

1. Click the **Device Manager** tab.
2. Select the device from the **Device List** toolbar. Expand the general name of the device, if necessary.



3. Click **Device diagnostics** located at the bottom of the **Device List** toolbar.



The device's diagnostics dialog box opens.

Related topics

For information about...	See...
Diagnostics	"About diagnostics" on page 11
About device drivers	"About device drivers" on page 7
Adding a device to the device manager	"Adding devices" on page 10
Setting generic device properties	"Setting the properties for a device" on page 16

About profiles

About this topic

This topic describes what profiles are and what they do.

Read this topic if you are an administrator in charge of managing Velocity11 devices.

Profiles defined

A profile contains the initialization settings needed for communication between a device and device driver. The data in a profile is used by VWorks or BenchWorks software to identify each device on the network.

A profile can also contain other basic settings that you are unlikely to change once set up.

Because profiles identify device driver devices on the network, each device driver device must have its own profile.

You can create, modify, and delete profiles as needed.

Stored settings

Profiles are stored in the Windows registry.

The settings stored in a device driver profile include:

❑ Whether the device is connected using serial or Ethernet

❑ If the device is connected using Ethernet, the Device ID of the device on the network

❑ If the device is connected using serial, the COM port that the controlling computer uses for communication

❑ Configuration of accessories

Related topics

For information about...	**See...**
Device drivers	“About device drivers” on page 7
Adding a device to the device manager	“Adding devices” on page 10
Opening device diagnostics	“Opening diagnostics” on page 12

Setting the properties for a device

About this topic

The device properties provide VWorks or BenchWorks software with additional information about the device's current configuration, such as which profile to use, and stores the information in the device file. The device file is automatically loaded when you open a protocol.

The device properties need to be set when configuring the device. Typically, these properties only need to be set once. This topic describes how to set the following device properties:

- ❑ General
- ❑ Teachpoint
- ❑ Barcode
- ❑ Location (for devices with multiple teachpoints)
- ❑ *Device* Properties

Read this topic if you are an administrator in charge of managing Velocity11 devices.

Before you start

Make sure that you have installed the device driver plug-in and have added the device to the device manager.

See "Related information" for procedures on how to do these tasks.

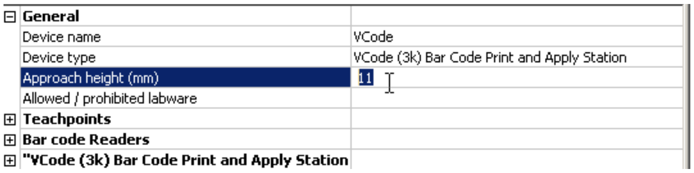
Setting general properties

To set the general properties for a device:

1. Click the **Device Manager** tab.
2. Select the device from the **Device List** toolbar. (Expand the device name, if necessary.)

   Note: For devices with **Locations**, see "Setting location properties" on page 17. If no Locations, continue with step 3.
3. In the **General** group, set the following:

   a. **Approach height**. This is the height to raise the robot gripper above the teachpoint when the robot moves the plate horizontally towards or away from it.



⊟ **General**	
Device name	VCode
Device type	VCode (3k) Bar Code Print and Apply Station
Approach height (mm)	11
Allowed / prohibited labware	
⊞ **Teachpoints**	
⊞ **Bar code Readers**	
⊞ **"VCode (3k) Bar Code Print and Apply Station**	

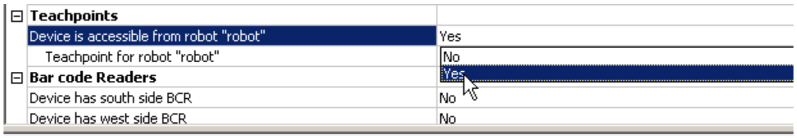
   b. **Allowed/prohibited labware**. Click the adjacent field to open the dialog box. Move the labware classes by selecting them and clicking one of the arrow buttons.
4. In the ***Device* Properties**, select the desired profile if it is not already selected.
5. Select **File** > **Device File** > **Save** to save the changes to the device file.

Setting teachpoints

Teachpoints are the coordinates in space that a robot travels to in order to interact with a device. Only the devices that are accessible by robots are able to have teachpoints.

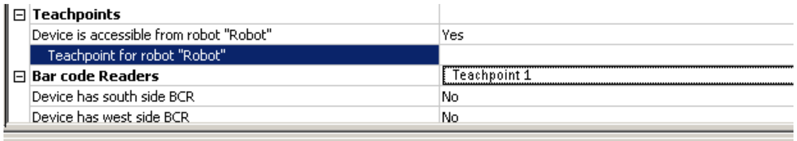
To set the teachpoint properties:

1. Open the **Device Properties** page.
2. In the **Teachpoints** property group, set the following:
   a. **Device is accessible from robot *robot's name***. Choose **Yes** or **No**.



⊟ **Teachpoints**	
Device is accessible from robot "robot"	Yes
Teachpoint for robot "robot"	No
⊟ **Bar code Readers**	Yes
Device has south side BCR	No
Device has west side BCR	No

   b. **Teachpoint for robot *robot's name***. Choose a file.



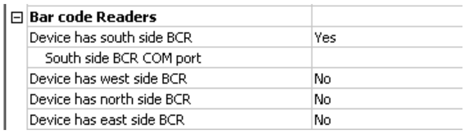
⊟ **Teachpoints**	
Device is accessible from robot "Robot"	Yes
Teachpoint for robot "Robot"	
⊟ **Bar code Readers**	Teachpoint 1
Device has south side BCR	No
Device has west side BCR	No

Setting barcode location

If your device has a barcode reader, indicate where the reader is located.

To set the barcode readers property:

1. In the **Barcode Readers** property group, set the side that has the barcode to **Yes**.



⊟ **Bar code Readers**	
Device has south side BCR	Yes
South side BCR COM port	
Device has west side BCR	No
Device has north side BCR	No
Device has east side BCR	No

2. Enter the number of the COM port to which the device is connected.

Setting location properties

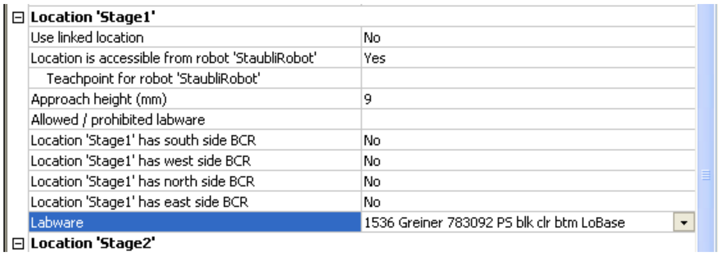
Note: The options available under Location groups might differ for software and hardware device drivers. Software devices do not have robot-accessible labware positions.

For hardware devices that have more than one robot-accessible labware position, the approach height, allowable/prohibited labware, teachpoint, and barcode properties are located under Location groups.

To set the Location properties:

1. *Hardware device drivers only.* Set the **Use linked location**. Follow the procedure in "Setting the Use linked location" on page 18.
2. *Hardware device drivers only* Set the **Teachpoints**. Follow the procedure in "Setting teachpoints" on page 17.

3. *Some software device drivers only.* Set the **Approach height** and **Allowed/prohibited labware**. Follow the procedure in “Setting general properties” on page 16.
4. Set the **Barcode Readers** location. Follow the procedure in “Setting barcode location” on page 17.
5. Assign the **Labware** used by the location by selecting the correct labware type from the list.



⊟ **Location 'Stage1'**	
Use linked location	No
Location is accessible from robot 'StaubliRobot'	Yes
Teachpoint for robot 'StaubliRobot'	
Approach height (mm)	9
Allowed / prohibited labware	
Location 'Stage1' has south side BCR	No
Location 'Stage1' has west side BCR	No
Location 'Stage1' has north side BCR	No
Location 'Stage1' has east side BCR	No
Labware	1536 Greiner 783092 PS blk clr btm LoBase
⊟ **Location 'Stage2'**	

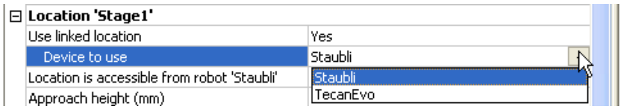
6. In the ***Device* Properties**, select the desired profile if it is not already selected.
7. Select **File** > **Device File** > **Save** to save the changes to the device file.

Setting the Use linked location

Currently, this feature is enabled for the special situations in which there is a storage device such as a PlateHub Carousel, StoreX, or Cytomat and a robot, such as the Velocity11 Translator robot that is shuttling plates between systems.

To use this feature, select yes and then select the device location to which you want to link. This tells the software that the current device location is the same physical location as the device selected from the Device to use list.

Note: Selecting this option when it is not enabled will have no effect on the system.



⊟ **Location 'Stage1'**	
Use linked location	Yes
Device to use	Staubli
Location is accessible from robot 'Staubli'	Staubli TecanEvo
Approach height (mm)	

Related topics

For information about...	See...
Device drivers	“About device drivers” on page 7
Installing a device driver plug-in	“Installing device drivers” on page 9
Profiles	“About profiles” on page 15
Adding a device to the device manager	“Adding devices” on page 10
Opening diagnostics	“Opening diagnostics” on page 12

Adding and linking Sub Process tasks

About this topic

This topic describes how to add a sub-process to a protocol and configure it. Read this topic if you are an administrator or technician and are responsible for creating protocols in VWorks or BenchWorks software.

Before you read this

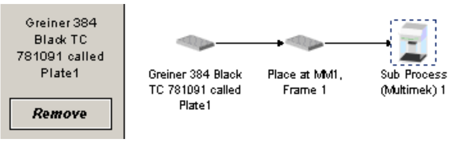
Before you read this topic, become familiar with the topics in the *VWorks Version 3 Automation Control User Guide* or *BenchWorks Automation Control User Guide* describing what a protocol is and how it is created.

Sub Process task defined

Sub Process tasks indicate the existence of a subroutine within a protocol. Sub-processes typically contain a series of liquid handling tasks used by devices such as the VPrep Pipettor or Multimek dispenser.

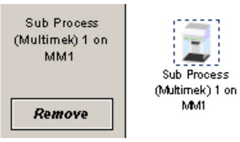
Adding a Sub Process task

The first step in creating a pipette process is to add a Sub Process task to the protocol editor. Drag the Sub Process icon into the process.


Greiner 384 Black TC 781091 called Plate1
Remove
Greiner 384 Black TC 781091 called Plate1
Place at MM1, Frame 1
Sub Process (Multimek) 1

Setting Sub Process task parameters

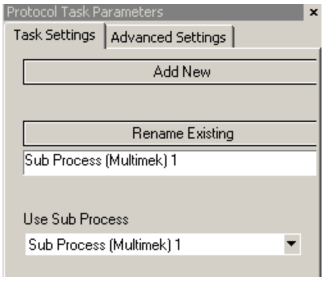
When you add the Sub Process task, a new sub-process is started in the pipette process editor. This process is identified by its sub-process link icon.


Sub Process (Multimek) 1 on MM1
Remove
Sub Process (Multimek) 1 on MM1

Because you can have more than one sub-process in a protocol, you must link the Sub Process task to the correct sub-process.

To link the Sub Process task to the correct sub-process:

1. In the **Protocol Editor**, add a Sub Process task to the protocol and then select it in the protocol sequence.
2. In the **Protocol Task Parameters** toolbar, select the sub-process that you want to use for this pipetting task from the **Use Sub Process** list.



Sub Process (Multimek) 1

Use Sub Process: Sub Process (Multimek) 1

3. If there is only one sub-process and you need to create a second one, click **Add New**.

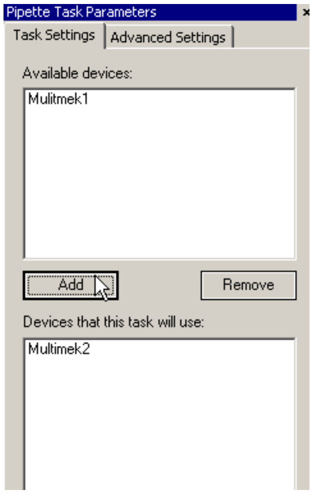
Associating the sub-process to a device

Because you can have more than one device that uses sub-processes on a lab automation system, you must link each sub-process link icon with one or more devices that you want the sub-process to be able to use. You do this by setting the parameter for the sub-process link icon.

To link a Sub Process task to a device:

1. In the **Pipette Process Editor**, select the **Sub Process** link icon.

2. In the **Available devices** list of the **Pipette Task Parameters** toolbar, select one or more pipettors to link to and click **Add**.



Available devices: Mulitmek1

Devices that this task will use: Multimek2

The selected pipettors move to the lower box and become available for use.

Related topics

For information about...	See...
Device drivers	"About device drivers" on page 7
Setting common device properties	"Setting the properties for a device" on page 16
Adding a device to the device manager	"Adding devices" on page 10
Creating protocols	❑ *VWorks Version 3 Automation Control User Guide* ❑ *BenchWorks Automation Control User Guide*

Using JavaScript to set task parameters

About this topic

JavaScript programs (scripts) can be used to change the parameters of a protocol task immediately before it is scheduled. This extends the capability of VWorks or BenchWorks software because the parameters can be changed dynamically during a run, based on the following:

- ❑ Information passed from an external source, such as a database
- ❑ The number of times the protocol has cycled
- ❑ Feedback on changing conditions during the run

This topic describes the use of JavaScript to set task parameters in a protocol.

Read this topic if you are an administrator or technician responsible for creating VWorks or BenchWorks software protocols and want to add functionality to a task using JavaScript.

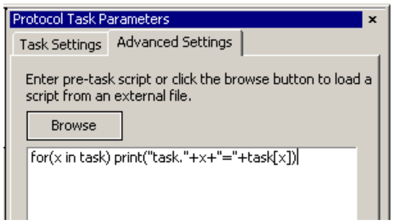
Where scripts are written

Scripts can be written in two ways:

- ❑ Directly into the box in the Advanced Settings tab of the Task Parameters toolbar
- ❑ As an external file that is located by clicking Browse in the Advanced Settings tab and navigating to its location on the hard drive

Note: You can also call an external file by embedding the "`open()`" function in the box.

The following screenshot displays a short script that prints the parameters of a task to the log toolbar, just before the task runs. In this case, the script is written directly in the Advanced Settings box.



Enter pre-task script or click the browse button to load a script from an external file.

```
for(x in task) print("task."+x+"="+task[x])
```

For more information about using JavaScript, refer to the *VWorks Version 3 Automation Control User Guide* or the *BenchWorks Automation Control User Guide*.

Related topics

For information about...	See...
Using JavaScript in protocols	□ *VWorks Version 3 Automation Control User Guide* □ *BenchWorks Automation Control User Guide*
Adding tasks to protocols	□ *VWorks Version 3 Automation Control User Guide* □ *BenchWorks Automation Control User Guide*

About reader output files

About this topic

Plug-in device drivers that are written for plate readers have a common way of naming their output files. This topic explains the concepts related to output file naming. By reading this topic, you will learn how to prevent data in the reader output files from being overwritten by newer data.

Read this topic if you are an operator who wants to make changes to the task parameters for one of these readers:

□ VR4000

□ Analyst GT

□ Fusion

□ Viewlux

□ Tecan readers

Plug-in default output file

When you first install a reader device driver plug-in, all data recorded during a protocol or by a manual read using diagnostics software is written to a single file stored in the C: drive.

The exact name of the file is specific to the device. For example, the RVSI VR4000 device driver creates a file with the name vialreaderresults.txt.

This file can only store data for one read, which means that the set of data for each read overwrites the last set in the file. To avoid this problem you must set up an output file naming convention.

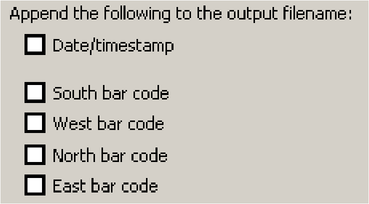
Profile default output file name

Some device drivers allow more than one device of that type to be used in the lab automation system. In this case, each device must have its own profile. Even if you have only one device, you can still set up multiple profiles for it, with each storing different settings.

In these cases, you probably want each profile to have a separate default output filename to prevent the data from runs using one profile overwriting those of another.

Filename suffixes

To prevent the data from one read overwriting the data from another, you need to append a variable suffix to the file name. You can append a date/time stamp and one or more bar codes on the rack or plate.



Append the following to the output filename:

□ Date/timestamp

□ South bar code

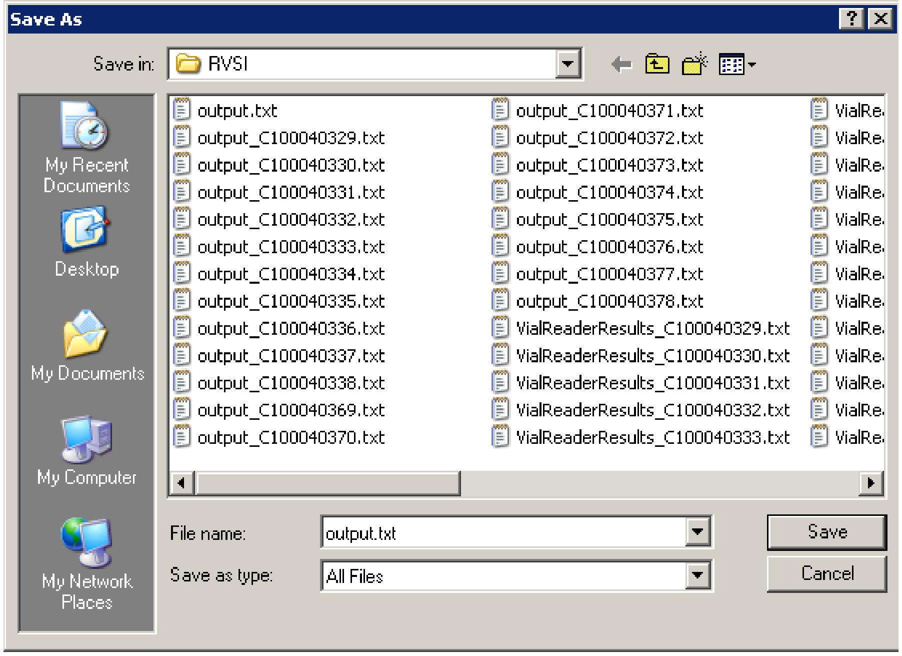
□ West bar code

□ North bar code

□ East bar code

Example

The example output file folder below shows that a profile default file name of output.txt was created at one time. At another time, a suffix was appended in the profile for the device driver, which added a barcode identifier to the file name (for example output_C100040329.txt).



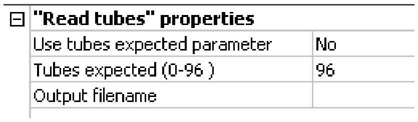
Save in: RVSI

File name: output.txt

Save as type: All Files

Overriding output file names with tasks

You can override the default output file name that is set in the profile using the Output filename property of the Read task parameters.



"Read tubes" properties	
Use tubes expected parameter	No
Tubes expected (0-96)	96
Output filename	

This allows you to use different output file names for every task.

The suffix used for the file name that you set in the task parameters is taken from the suffix specified in the device diagnostics profile. So if you select date/time stamp in the profile, the date/time stamp will also be appended during a run in which you have specified a different file name.

Related topics

For more information about...	See...
Opening diagnostics	"Opening diagnostics" on page 12
Profiles	"About profiles" on page 15

About device initialization

About this topic

When working in device diagnostics software, you are often required to initialize the device. This topic explains why device initialization is necessary.

Opening communications

Initializing a device opens communications with it. For example, if the device is connected with a serial cable, the COM port is opened, and if the device is connected with an Ethernet cable, the TCP/IP socket is connected.

Homing motors

Initializing a device homes motors that do not track their position along their line of travel. Homing a motor moves it until it triggers an event, called a home flag. This tells the motor its location.

The motors on some devices automatically move to their home positions when the device is turned on. The motors on other devices must be initialized to be homed.

Setting profile parameters

Initializing a device applies relevant parameters set in the device's profile.

Setting state and memory variables

Most devices store variables in software or firmware. Initializing a device sets these variables to their initial values.

Related topics

For information about...	See...
Using Diagnostics	❑ "About diagnostics" on page 11 ❑ "Opening diagnostics" on page 12
Workflow for configuring devices	"Adding devices" on page 10

MATLAB 2

Mathworks MATLAB is software for performing scientific and engineering calculations. MATLAB can be opened and run from VWorks or BenchWorks using the MATLAB device driver.

Note: The MATLAB driver is a software driver and not a hardware device driver. However, it is configured using the device manager in same way as Velocity11 device drivers. For this reason, it is treated as a device.

This chapter contains the following topics:

Workflow for configuring the MATLAB device driver

About this topic

This topic presents the workflow for configuring the MATLAB device driver.

Read this topic if you are an administrator responsible for setting up devices in VWorks.

Before you start

Driver installation instructions

Before you can configure the MATLAB device driver you must have installed it on the same computer as VWorks.

About multiple devices

You should only install one MATLAB device.

Note: Installing more than one device will give you an error message and you will need to restart VWorks.

Workflow

Step	Topic
1	"Adding devices" on page 10
2	"Creating a MATLAB profile" on page 29
3	"Setting the properties for a device" on page 16

Related topics

For information about...	See...
Device drivers	"About device drivers" on page 7
Device driver installation	"Installing device drivers" on page 9
Using_MATLAB tasks	"Setting MATLAB task parameters" on page 30
Working with MATLAB profiles	"Managing MATLAB profiles" on page 32

Creating a MATLAB profile

About this topic

This topic describes how to create a profile for the MATLAB device driver.

Read this topic if you are an administrator responsible for setting up devices in VWorks.

Before you start

Before you create a profile, you must:

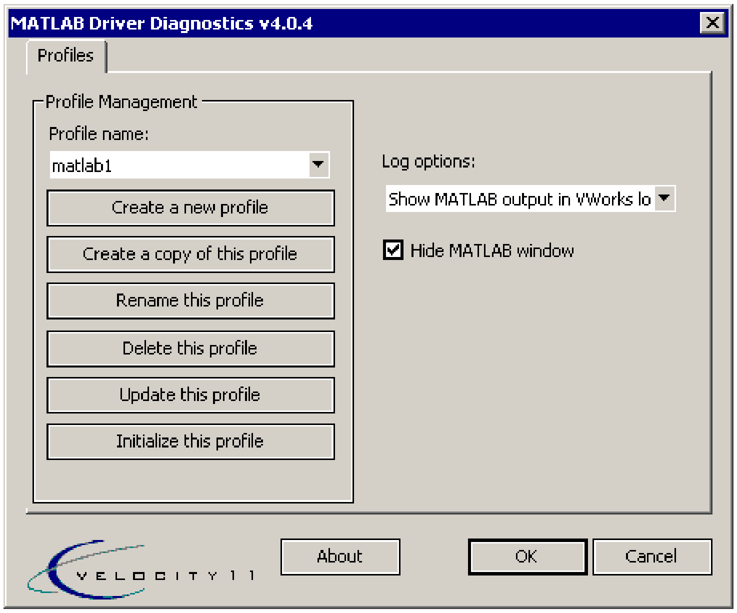
❑ Install the MATLAB device driver

❑ Add the MATLAB to the device manager

Procedure

To create a MATLAB profile:

1. Open **MATLAB Diagnostics**.



Profile name: matlab1

Log options: Show MATLAB output in VWorks lo

☑ Hide MATLAB window

2. In the **Profile Management** area, click **Create a new profile**.
3. Enter a name and click **OK**.
4. To choose an output format, select a **Log options** preference:
   - ◆ Show MATLAB output in the VWorks log
   - ◆ Don’t show output
   - ◆ Show the first 128 characters of output
5. If you want to make the MATLAB window invisible, click **Hide MATLAB window**.

   Note: The MATLAB software can start and run in the background.
6. Click **Update this profile** to save your changes.

Related topics

For information about...	See...
The workflow to which this topic belongs	"Workflow for configuring the MATLAB device driver" on page 28
Profiles	"About profiles" on page 15
The next step	"Setting the properties for a device" on page 16
Using MATLAB tasks	"Setting MATLAB task parameters" on page 30

Setting MATLAB task parameters

About this topic

This topic describes how to set the MATLAB device driver task parameters. Read this topic if you are: Read this topic if you are:

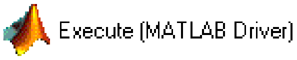
- ❑ An administrator or technician responsible for creating protocols
- ❑ An operator who wants to make changes to the MATLAB task parameters in a protocol

  Note: Operators cannot save changes to protocols.

About the MATLAB task

When MATLAB is added to the device manager, the **Execute (MATLAB Driver)** task associated with the MATLAB driver becomes available in the Pre-Protocol Editor, Protocol Editor, Pipette Process Editor, and Post-Protocol Editor.

The task is represented by the following icon:



Execute (MATLAB Driver)

About MATLAB commands and scripts

You can run simple MATLAB commands or more complex scripts through VWorks or BenchWorks.

Commands

A command is one or more mathematical expressions separated by semi-colons. For example:

```
x=2;x=x+1
```

Scripts

MATLAB scripts are stored in text files and given the file name extension .m. If the result of a script or command is the display of a graph, a MATLAB window will open to display it.

Note: To learn more about writing MATLAB commands and scripts, refer to the MATLAB help or user documents.

Setting Execute (MATLAB Driver) task parameters

It is important to understand that the **Execute (MATLAB Driver)** task can involve both MATLAB scripts and JavaScript scripts. The Command to execute property is where you define a MATLAB script.

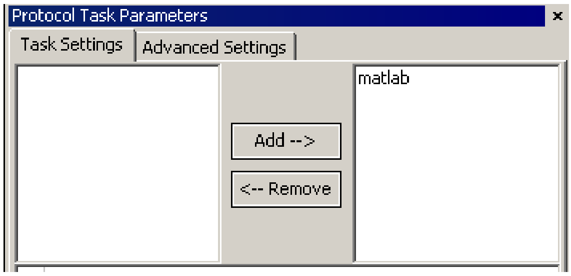
To set the Execute (MATLAB Driver) task parameters:

1. Add the **Execute (MATLAB Driver)** task to the protocol process.

   Note: This task is available in the Pre-Protocol Editor, Protocol Editor, Pipette Process Editor, and Post-Protocol Editor.

2. In the **Protocol Task Parameters** toolbar, click the **Task Settings** tab.

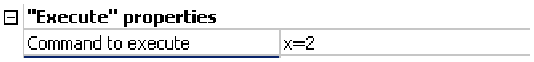
3. Move the device (profile) that you created in the device manager to the right-hand box using the **Add** button.

   Note: This links the task with the device that you created. Only one copy of MATLAB is loaded no matter how many MATLAB devices are defined in the device manager.

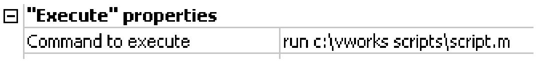
4. In the **Command to execute** box, type a command or a script:

   ◆ Type the expression that you want MATLAB to compute.

⊟ **"Execute" properties**	
Command to execute	x=2

   ◆ Type `run` *`<path to script\file name>`*

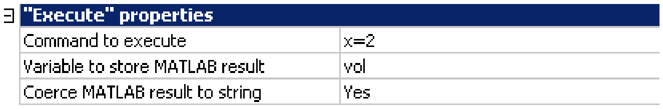
   

⊟ **"Execute" properties**	
Command to execute	run c:\vworks scripts\script.m

5. If you plan to use the result of a MATLAB computation as a variable in a JavaScript script:

   a. Enter either a MATLAB script, or a path to a MATLAB script file in the **Command to execute** box.

   b. In the **Variable to store MATLAB result** box, enter the name of the JavaScript variable that will be assigned the value returned by the MATLAB script.

⊟ **"Execute" properties**	
Command to execute	x=2
Variable to store MATLAB result	vol
Coerce MATLAB result to string	Yes

   Note: In this example, the value of "2" is stored in the "vol" global variable. This value is used for any JavaScript scripts containing the "vol" variable that are subsequently executed.

c. Click **Yes** if you want to **Coerce MATLAB result to string**.

d. Add another **Execute (MATLAB Driver)** task to the process *after* this task.

e. Click the **Advanced Settings** tab of the **Protocol Task Parameters** toolbar.

f. Enter the JavaScript or browse to the JavaScript file that uses the variable.

6. To execute the command, run the protocol in the same way that you would run a protocol that moves labware.

Related topics

For information about...	See...
Creating scripts	"Using JavaScript to set task parameters" on page 21
Other ways to create JavaScript scripts	*VWorks Version 3 Automation Control User Guide*
Creating protocols	*VWorks Version 3 Automation Control User Guide*
Starting a protocol run	*VWorks Version 3 Automation Control User Guide*

Managing MATLAB profiles

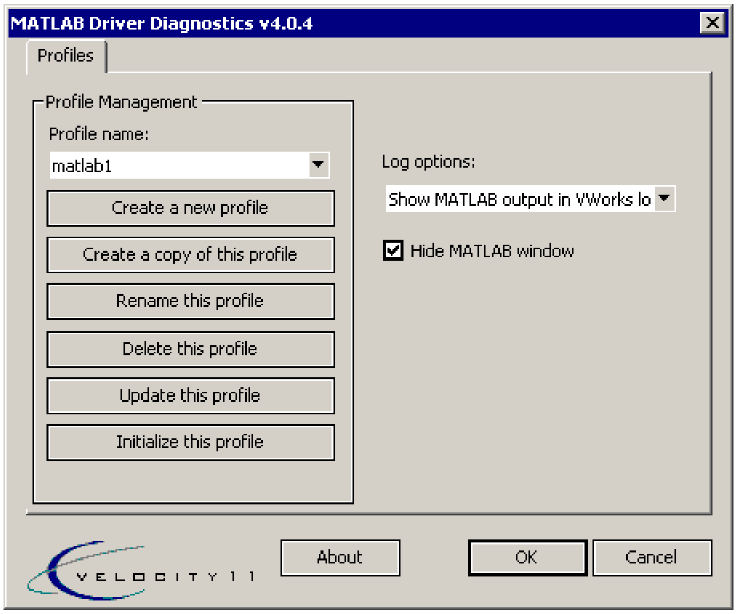
About this topic

This topic describes how administrators and technicians can manage MATLAB profiles.

Managing profiles

To manage MATLAB profiles:

1. Open **MATLAB Diagnostics**.



Profile name: matlab1

Log options: Show MATLAB output in VWorks lo

☑ Hide MATLAB window

2. Select a profile from the **Profile name** list.
3. Perform the management tasks:

   Management tasks include the following:

   - Updating the profile.
   - Copying a profile.
   - Renaming a profile.
   - Deleting a profile.
   - Saving changes.

     Note: Click **Update this profile** to save edits.

Related topics

For more information about…	See…
Profiles	"About profiles" on page 15
Creating a MATLAB profile	"Creating a MATLAB profile" on page 29
Device properties	"Setting the properties for a device" on page 16

Agilent Technologies

User Guide

G5415-90028