BenchCel Microplate Handling Workstation X-Series

BenchWorks version24.x

User Guide

Agilent Technologies

Notices

User Guide Part Number

G5400-90002

January 2009 rebranded edition including, June 2008 Labware Rack guide, January 2006 BenchCel Workstation X-series user guide

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 (0)1763853638
euroservice.automation@agilent.com

Documentation feedback:
documentation.automation@agilent.com

Web:
www.agilent.com/lifesciences/automation

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 Device Driver 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 (0)1763853638
euroservice.automation@agilent.com

Web: www.agilent.com/lifesciences/automation

Labware Rack Handling Guide

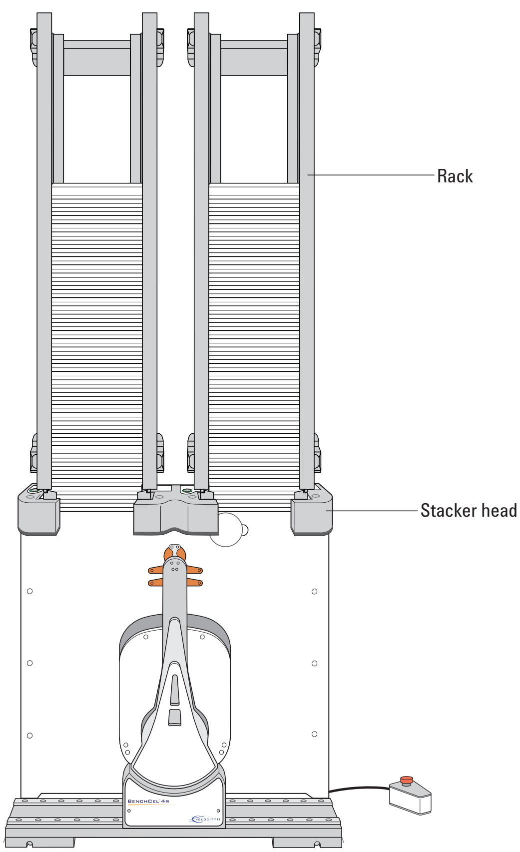
This guide explains how to handle the labware racks safely. The topics are:

- About the labware racks
- Carrying the racks
- Lifting the racks
- Loading labware into the racks
- Accessing product user documentation
- Contacting Velocity11

About the labware racks

The BenchCel® Microplate Handling Workstation and the VStack® Labware Stacker use labware racks to store the stacks of labware (microplates, tipboxes, and tube racks) that are processed during a protocol run.

Figure BenchCel Workstation with front-load labware racks



For a description of the BenchCel device or VStack device, see the *BenchCel User Guide* or *VStack User Guide*.

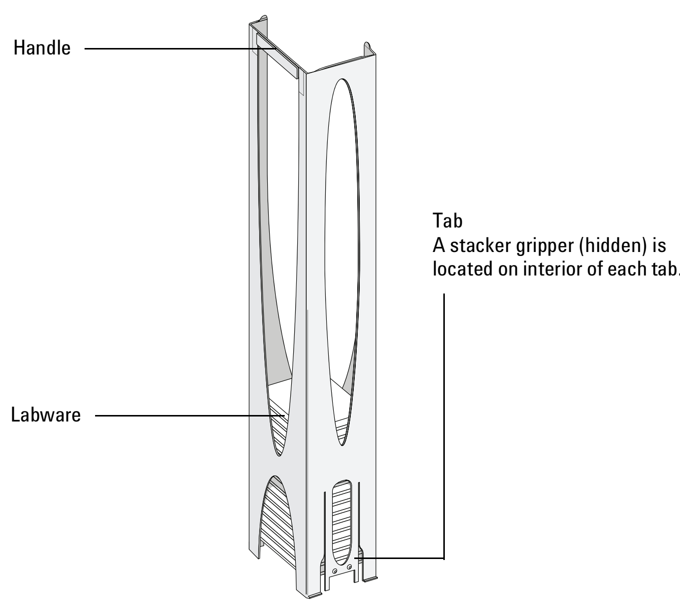
The labware racks are available in three models: standard rack, top-load rack, and front-load rack.

Figure Three models of labware racks

All the rack models have the following basic parts:

- *Carrying handle*. The standard rack has a polished top bar that can be used as a carrying handle. The top-load and front-load racks have fold-down carrying handles.
- *Tabs*. A pair of vertical tabs are located at the bottom sides of the rack. The tabs insert into slots on the device when you mount the rack.
- *Stacker grippers*. A gripper is located on the interior bottom of each tab. The pair of grippers hold a microplate during the labware loading, unloading, downstacking, and upstacking processes. A clamp in the device opens and closes the grippers.

Figure Standard rack containing labware



22825.00.00.00_RackHandlingGuide_June08

An Agilent Technologies Company

Carrying the racks

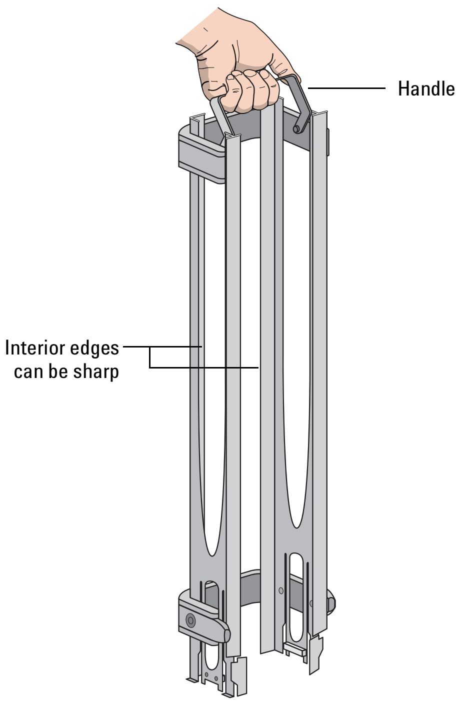
WARNING **Do not hold a rack by the interior edges. The interior edges can have sharp surfaces that can cause cuts if handled improperly.**

CAUTION A rack that is fully loaded with labware can be heavy. Grasp the rack handle firmly to prevent the rack from slipping or tilting.

To carry a rack:

Firmly grasp the rack by the handle.

Figure Carrying a front-load rack



Lifting the racks

Make sure to use the proper lifting technique when mounting a rack on a device or removing a rack from a device.

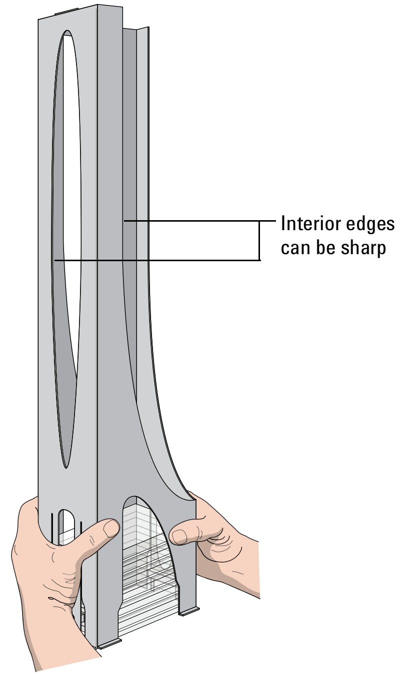
WARNING **Avoid touching the interior edges of a rack when lifting the rack. The interior edges can have sharp surfaces.**

IMPORTANT See your *BenchCel User Guide* or *VStack User Guide* for the procedure to mount a rack on the device or to release a rack from the device.

To lift a rack:

Use both hands to grasp the rack securely around the four corners near the base, as the following figure shows.

Figure Lifting a standard rack



Loading labware into the racks

Before loading the labware into a rack:

- Position the rack so that the opening is facing you.
- Determine how the microplates should be oriented in the rack.

  For example, if the BenchCel orientation-sensing feature is enabled, make sure the A1 wells are oriented in the rack as specified.

Loading labware in standard and top-load racks

IMPORTANT See your *BenchCel User Guide* or *VStack User Guide* for the details on how to release a rack for removal or to prepare for loading a mounted rack.

To load labware into a standard or top-load rack:

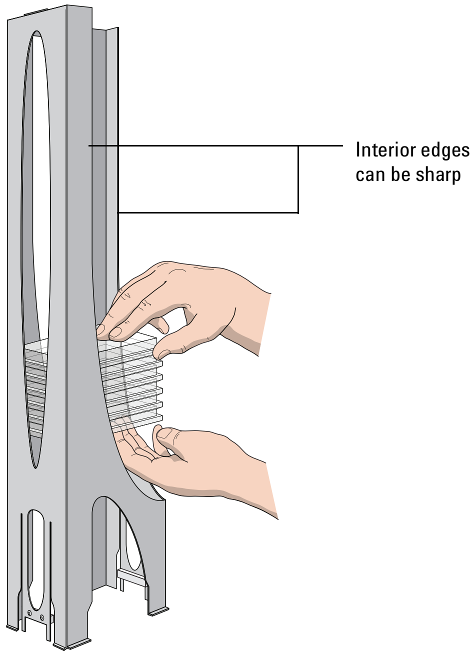
1 If possible, remove the rack from the device, and place the rack on a flat, level surface.

2 Using both hands, carefully slide a small stack of labware down through the top of the rack.

  You can use one hand to support underneath the labware stack, while the other hand holds the top of the labware to keep it level. See the following figure.

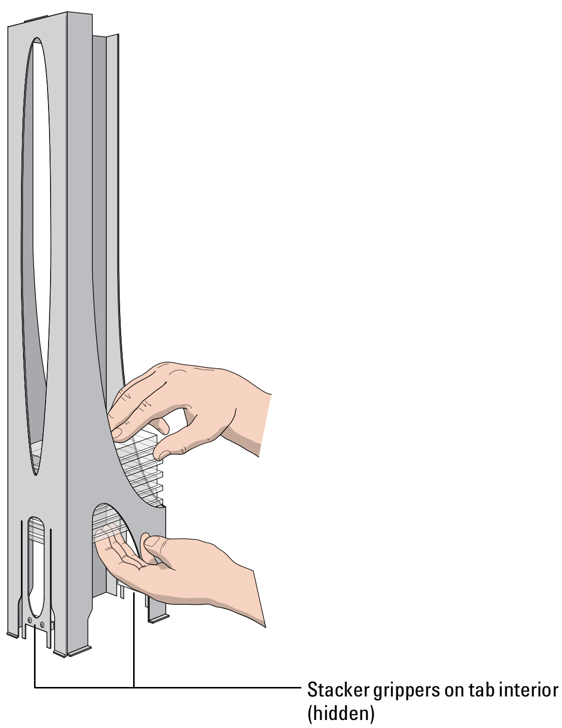
WARNING **Use care to avoid sliding your hand on the interior edges in the rack. The edges can have sharp surfaces.**

Figure Loading a standard rack



3 (Standard racks only) When you reach the bottom of the open slot, transfer your hand positions so that you continue supporting the labware through the bottom slot.

Figure Supporting labware through the bottom slot



4 Ensure that the bottom labware in the stack rests on the rack stacker grippers.

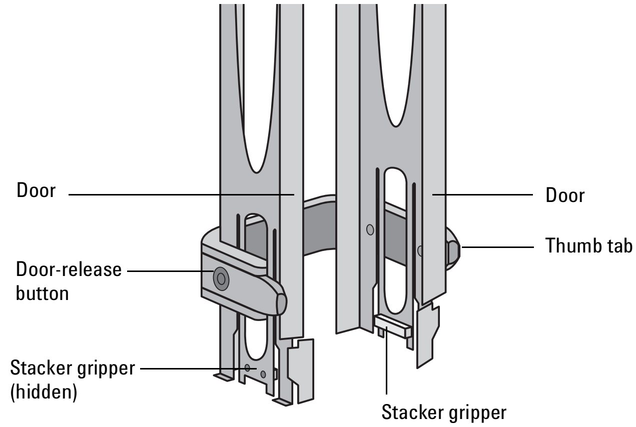
To unload labware from a standard or top-load rack:

With your hands positioned as shown in the previous figures, carefully slide the labware in small stacks up and out of the top of the rack.

Loading labware in a front-load rack

The doors on the front-load rack provide easy access for loading labware into the front of a rack that is mounted on a device.

Figure Door mechanism on the front-load rack

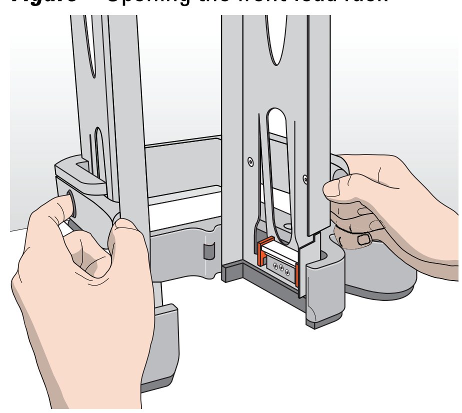


To load labware in a front-load rack:

IMPORTANT Before you attempt to load the labware in a mounted rack, ensure the device is ready for loading. For example, the clamps in the BenchCel stacker head must be closed (extended). See your device user guide for details.

1 On each side of the rack, slide the Door-release (black) buttons forward, while pushing outward on the thumb tabs. The rack doors open.

Figure Opening the front-load rack



2 Place the labware directly through the open rack doors so that the bottom labware rests on the rack stacker grippers. Ensure the labware is level and securely in the rack.

3 To close the doors, press the thumb tabs inward until the doors snap shut.

To unload labware from a front-load rack:

IMPORTANT Before you attempt to unload the labware from a mounted rack, ensure the device is ready for unloading. For details, see your device user guide.

1 To open the rack doors, slide the black Door-release buttons forward on each side of the rack, while pressing outward on the thumb tabs.

2 Carefully, lift the labware out through the front of the rack.

Accessing product user documentation

Velocity11 product user documentation is available in the following formats:

- Online help available within the software
- PDF files on the software CD
- Printed books

You can also search the online help or download the latest version of any PDF file from the Velocity11 website at:

http://www.velocity11.com/support/knowledge_base

Contacting Velocity11

- Technical Support: 1.800.979.4811 or +1.650.846.6611
- Customer Service: 1.866.428.9811 or +1.650.846.6601
- Email: info@velocity11.com
- Web: http://www.velocity11.com

BenchCel User Guide

January 2006

BenchWorks version 24.x

16967.00.00.00_BenchCelUG_Jan06

Velocity11
3565 Haven Avenue
Menlo Park CA 94025
USA

Phone: 650-846-6600
Fax: 650-846-6620
Service: 1-800-979-4811
Sales: 1-866-428-9811
Email: info@velocity11.com
http://www.velocity11.com

Contents

Chapter 1. Introduction 1
Who should read this guide 2
About Velocity11 user guides 3
What this guide covers 5
What's new in this version 7
Finding your software versions 8
Reporting BenchCel problems 9
Safety information 10
BenchCel hazard warnings 11

Chapter 2. BenchCel overview 13
BenchCel description 14
BenchCel configurations 15
BenchCel features 16
BenchCel accessories and integration options 17
Laboratory setup requirements 19
Utility and environmental requirements 20
The computer and networking 21
Connection panel description 22
Stacker-gripper control description 24
Connecting the controlling PC and BenchCel 24
Connecting the BenchCel to compressed air 26
About BenchWorks 27
Relationships of BenchWorks components 30

Chapter 3. Preparing for a run 31
Workflow for preparing for a run 32
Turning on the BenchCel 33
Logging into the operating system 34
Starting BenchWorks 34
Logging in to BenchWorks 36
About tasks, processes, and protocols 37
Opening a protocol in BenchWorks 40
Setting general options 41
About setting error options 43
Setting general error-handling options 44

Setting up email error notification 45
Setting protocol error-handling options 47
Setting pre-protocol rules 48
Setting protocol rules 49
About log and data files 51
Setting log options 54
Importing a log file to Excel 57
Understanding the protocol 59
Printing a protocol 60
Preparing plates, instruments, and accessories 61
Setting up a VSpin counterweight plate 63
Installing a rack 64

Chapter 4. Performing a run 67
About performing a run 68
Performing pre-run checks 69
Starting a run from BenchWorks 70
Starting a run from a command line 73
Pausing or stopping a run 74
Monitoring a run 76
Working with the Log toolbar 77
Cleaning up after a run 80
Logging out of BenchWorks 81
Shutting down the BenchCel 81

Chapter 5. Creating a protocol: basics 83
About protocol files 84
Workflow for creating a protocol 85
Setting up a pre-protocol or post-protocol process 87
Setting up a plate instance 88
Setting the number of simultaneous plates 91
Adding and deleting tasks and pipette tasks 92
Compiling and saving protocols 93
About setting task parameters 95
About setting pipette task parameters 96
Simulating a run 97

Chapter 6. Setting task parameters 99
Setting AliQuot task parameters 100
About setting Apply Label task parameters 101
Setting Apply Label task parameters 102
About combining bar code modifiers 106
Setting Delid/Relid task parameters 108
Setting Downstack and Upstack task parameters 109

Setting Incubation task parameters 112
Setting Multidrop task parameters 114
Setting Nanodrop task parameters 116
Setting Pierce task parameters 117
Setting Place Labware task parameters 118
Setting Place Plate task parameters 120
Setting QFill2 task parameters 121
Setting Restack task parameters 122
Setting Seal task parameters 124
Setting User Message task parameters 125
Setting Waitfor task and Signal task parameters 127
Setting WellMate task parameters 128
Setting VSpin with Access2 task parameters 130
Specifying task order across processes 132

Chapter 7. Setting pipette task parameters 135
Configuring a pipette process: example 136
Adding and configuring a Pipette Process task 140
Configuring a VPrep shelf as a device 142
Setting Aspirate pipette task parameters 144
Setting Change Instance pipette task parameters 146
Setting Change Tips pipette task parameters 148
Setting Dispense pipette task parameters 150
Setting Dry Tips pipette task parameters 153
Setting Loop pipette task parameters 153
Setting Mix pipette task parameters 155
Setting Pump Reagent pipette task parameters 157
Setting Wash Tips pipette task parameters 159

Chapter 8. Creating a protocol: advanced topics 163
Setting up the LabwareSelector plug-in 164
About the FileReader plug-in 164
About the FileReader file format 166
Using the FileReader plug-in in a protocol 168
Using JavaScript in BenchWorks 171
The JavaScript task object and properties 178
Understanding the configuration settings in a pipette task 192
Working with device files 195
About bar code reading and tracking 197
Using bar code input files 198
Using bar code data files 200

Chapter 9. Administrator procedures 203
About user accounts and privileges 204
Adding and deleting a user account 205
Setting up email 206
Moving or sending a registry file 208
Obtaining information about the BenchCel network cards 210

Chapter 10. Defining labware 213
About defining labware in BenchWorks 214
About the labware editor 215
Labware editor overview 216
About the defining labware process 218
Opening the labware editor 219
Adding a labware entry 221
Deleting a labware entry 222
Renaming a labware entry 223
Copying a labware entry 224
Defining general properties 225
Defining plate properties 226
Defining BenchCel properties 229
Defining stacker properties 230
Inserting an image 232
Defining labware classes 233
About the Labware Parameters group box. 236
Opening the Labware Parameters group box 237
Changing labware parameters 238

Chapter 11. Setting liquid-handling definitions 243
About the liquid library editor 244
Opening the liquid library editor 246
Creating a liquid class 248

Chapter 12. Using BenchCel Diagnostics 251
About diagnostics 252
Opening BenchCel Diagnostics 253
About the Controls page 254
Homing the robot 257
Moving the robot manually 258
Jogging the robot 259
Changing the robot speed 261
Using Move-To-Position commands 263
About stacker controls 264
Loading and releasing stacks 265
Extending and retracting stacker clamps 267

About teachpoints 269
Adding a teachpoint. 270
Editing a teachpoint 272
Deleting a teachpoint 273
Managing teachpoint files 274
Moving between teachpoints 275
About general settings. 277
Changing general settings 278
Making global general settings changes 281
About profiles 282
Creating a BenchCel profile 283
Managing profiles 284
Using Multidrop diagnostics 285
Using Nanodrop Diagnostics 288
Using QFill2 Diagnostics 291
Using WellMate Diagnostics 293

Chapter 13. Maintenance and troubleshooting 299
Routine maintenance 300
About error handling 301
Resolving device initialization errors 302
Compilation warnings and errors 303
Resolving common problems and error messages 304
Checking the arm alignment. 309
Resolving labware problems. 310
Resetting the orientation sensors for a plate 313
Checking the stacker sensors 314
Resolving teachpoint problems. 315
Resolving non-VCode bar code reader errors 317
Resolving VCode bar code reader errors 319
Adjusting the stacker gripper pressure 320
Sending a bug report 321

Chapter 14. BenchCel ActiveX controls 323
Setting up properties. 324
Setting up methods. 325

Index 339

1 Introduction

This chapter introduces the *BenchCel User Guide*.

To operate the BenchCel, become familiar with the procedures in this guide as well as the guides for the devices installed on your BenchCel.

Who should read this guide

Job roles

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

Job role	Responsibilities
Integrator	Someone who configures software and hardware to allow integration of the BenchCel into a larger lab automation system.
Installer	Someone who unpacks, puts together, and tests the BenchCel before it is used.
Lab manager, administrator, or technician	Someone who is responsible for: ❑ Managing the BenchCel ❑ Developing the applications that are run on it ❑ Solving the more challenging problems that may arise ❑ Developing training materials and standard operating procedures for Operators
Operator	Someone who performs the daily production work on the BenchCel and solves routine problems. Your organization may choose to create its own procedures for operators based on the information in this guide.

Related information

For information about...	See...
Using Velocity11 user guides	“About Velocity11 user guides” on page 3
Finding firmware version	“About general settings” on page 277
What this guide covers	“What this guide covers” on page 5
Starting BenchWorks	“Starting BenchWorks” on page 34

About Velocity11 user guides

About this topic

This topic describes the different formats of Velocity11 documentation.

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 format has different benefits.

Where to find the user guides

Online help

The online help is automatically added to your computer with the software installation.

PDF

The *BenchCel User Guide* and *Device Driver User Guide* in PDF format are located on the software CD-ROM, as a files named BenchCelUserGuide_Jan2006.pdf and DeviceDriverUserGuide_Dec2005.pdf. You will need to copy the files onto your computer. They are not automatically installed with the software.

Velocity11 website

You can download the latest version of any PDF file from our website at www.velocity11.com/techdocs/docdownloadpage.html.

All of Velocity11 user documentation can be searched from the website at www.velocity11.com/support/support.html.

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. It is also the only format in color.

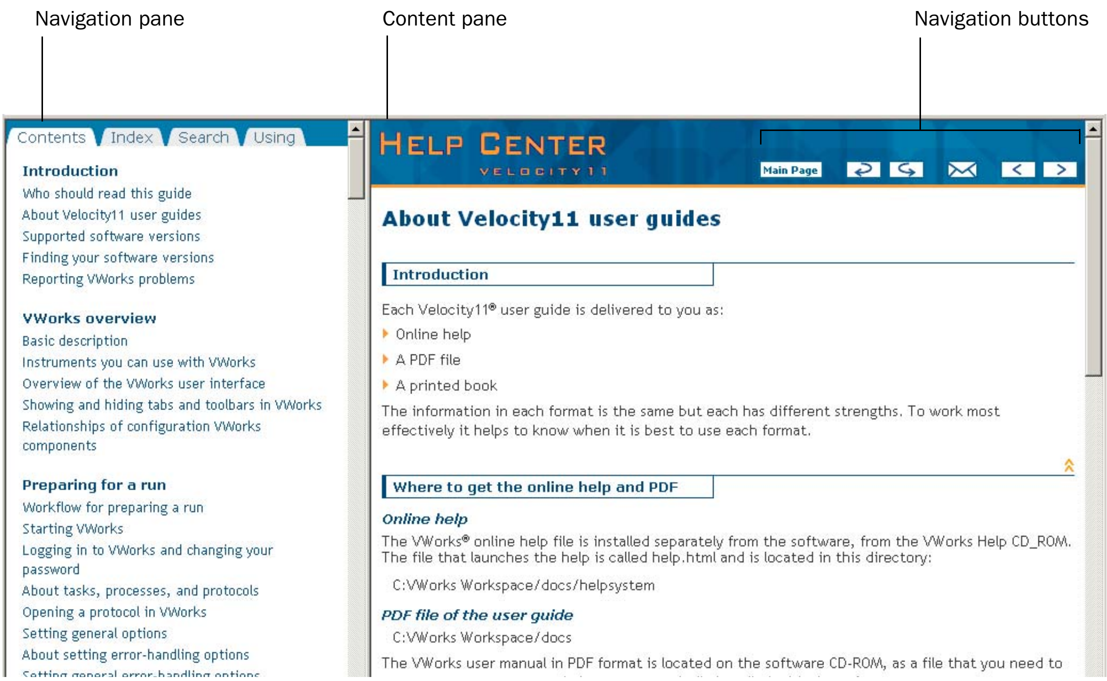
To open the online help:

1. If you are currently using BenchWorks, press F1.
2. If you are not currently using BenchWorks, open help.html in: C:\Program files\Velocity11\BenchWorks\HelpSystem\benchworks_ug.

   Place a shortcut to this file on your desktop for future use.

Main features

The online help includes a navigation pane, content pane, and navigation buttons.



The content pane displays the online help topics.

Navigation buttons in the content pane allow you to navigate through the pages.

PDF user guides

Computer requirements

To open a user guide in PDF format, you need an Acrobat viewer. You can either use the viewer that is built into Adobe Acrobat, or you can download the free Adobe Reader application from http://www.adobe.com/support/downloads/main.html.

Printing and searching

We provide user guides in PDF format mainly for printing additional copies. You can use them for simple searches from the Find button, although these searches are much slower than online help searches:

More information

For more information about using PDF documents, see the Adobe Acrobat PDF help system that can be accessed from your Acrobat viewer.

Related information

For information about...	See...
Who this guide is for	“Who should read this guide” on page 2
What this guide covers	“What this guide covers” on page 5
Finding firmware version	“About general settings” on page 277
Starting BenchWorks	“Starting BenchWorks” on page 34

What this guide covers

BenchWorks version

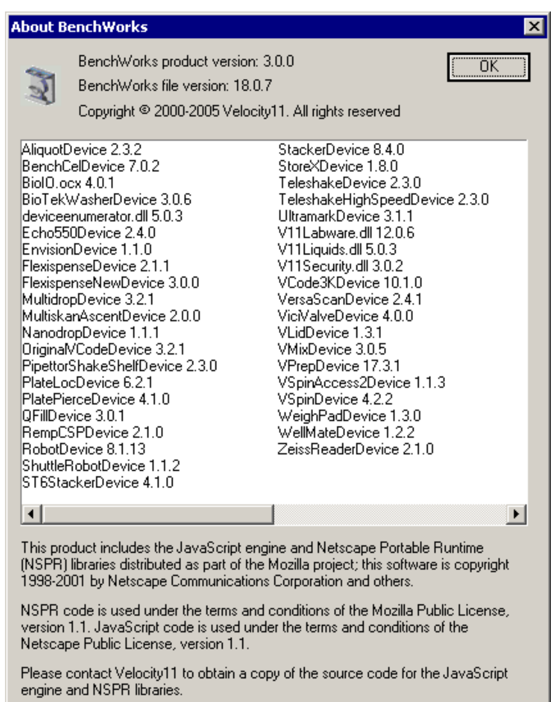
This guide covers the hardware and software operations of the BenchCel. This version of the guide is only for use with BenchWorks product version 3.x.x (file version 24.x) and the following major software components.

Note: Your BenchCel may not have all of the components shown in the following screen shot or may have components not shown here.

To find version information for your software:

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

   The **About** BenchWorks window lists the version number of the software and the version numbers of its major components.



Supported firmware version

This guide covers firmware version 2.2.8. The firmware version can be determined from the General Settings page of the BenchCel Diagnostics dialog box.

What this guide does not cover

This guide does not cover the operation of the following:

- ❑ Velocity11 instruments, such as the PlateLoc, VCode, and VPrep when used in stand-alone mode
- ❑ Other companies' devices, with the exception of the use of diagnostics software that is developed by Velocity11 for other companies' devices

For more information about these topics, see the user guides for the relevant instruments.

BenchCel versions

Because the BenchCel is available in different configurations and can be customized, your BenchCel may not be exactly as described in this guide.

Related information

For information about...	See...
Starting BenchWorks	"Starting BenchWorks" on page 34
Finding firmware version	"About general settings" on page 277

What's new in this version

About this topic

The following table describes the main new features of BenchWorks (version 24.x) and the BenchCel hardware that it supports since the last *BenchCel User Guide* was released.

New features

Feature	Description	See...
WellMate	Newly supported device	"Using WellMate Diagnostics" on page 293
QFill2	Newly supported device	"Using QFill2 Diagnostics" on page 291
MultiDrop	Newly supported device	"Using Multidrop diagnostics" on page 285
Nanodrop	Newly supported device	"Using Nanodrop Diagnostics" on page 288
JavaScript methods	task.repeat() method added	"The JavaScript task object and properties" on page 178
Reset script context protocol rule	New rule that changes the context of global variables	"Setting pre-protocol rules" on page 48
ActiveX controls	New	"BenchCel ActiveX controls" on page 323
Pre- and post-protocol processes	Ability to add a pre- and/or a post-protocol process to the run	"Setting up a pre-protocol or post-protocol process" on page 87
Save All command	New ability to save the current protocol file and device file at the same time	*Device Driver User Guide*
Bar code error handling	New tab on Task Parameters Toolbar of plate instance	"Setting up a plate instance" on page 88
Bar code task	Renamed to Apply label task	"Setting Apply Label task parameters" on page 102
Attempt to put plates away if deadlock occurs	If a deadlock occurs, the plates are moved to the positions they would be in at the end of a successful protocol run.	"Setting general error-handling options" on page 44
Change Instance pipette task	New task performs replicate dispense operations	"Setting Change Instance pipette task parameters" on page 146

Related information

For information about...	See...
Overview of the BenchCel	"BenchCel overview" on page 13

Finding your software versions

About this topic

This topic shows you some ways to find out your versions of BenchWorks and BenchCel Diagnostics.

From the software

You can open the software to find the version of BenchWorks or BenchCel Diagnostics.

To find the BenchWorks version number:

1. Start BenchWorks.
2. Navigate to **Help > About BenchWorks**.

To find the diagnostics version number:

1. Start BenchWorks.
2. Open BenchCel Diagnostics.
3. Read the version number on the title bar.

From the files

You can look at the version information in the executable files.

To find the BenchWorks or diagnostics version number:

1. Navigate to C:\Program Files\Velocity11\BenchCel.
2. Right-click BenchWorks.exe or BenchCel3.ocx (diagnostics) and select **Properties**.
3. Click the **Version** tab.

Related information

For information about...	See...
Getting help	"About Velocity11 user guides" on page 3
Opening Diagnostics	"Opening BenchCel Diagnostics" on page 253
Finding firmware version	"About general settings" on page 277
Starting BenchWorks	"Starting BenchWorks" on page 34

Reporting BenchCel problems

About this topic

If you find a bug in the software or have a technical or hardware problem that you can't resolve after reading the chapter on maintenance and troubleshooting, read the information in this topic for how to report problems.

Reporting software problems

If you find a problem in the Velocity11 software, let us know by:

- ❑ Sending a bug report from within BenchWorks
- ❑ Sending an email to service@velocity11.com or euroservice@velocity11.com
- ❑ Calling the Velocity11 Service Center at 1-800-979-4811 or 1-650-846-6611

Reporting hardware problems

If you have a problem with a hardware component of the BenchCel that you cannot resolve, let us know by:

- ❑ Sending an email to service@velocity11.com or euroservice@velocity11.com
- ❑ Calling the Velocity11 Service Center at 1-800-979-4811 or 1-650-846-6611

Sending files

When resolving software bugs or other problems, we may ask you to send device and protocols files and the Velocity11 registry file from the Windows registry.

Reporting user guide problems

If you find a problem with this user guide or have suggestions for improvement, please take a minute or two to give us your feedback using the feedback button in the online help. Your comments will be reviewed promptly and used to write the next version of the guide.

You can also send an email directly to documentation@velocity11.com.

Related information

For information about...	See...
Sending a bug report	"Sending a bug report" on page 321
Sending a registry file	"Moving or sending a registry file" on page 208

Safety information

About this topic

This topic gives general information about BenchCel safety hazards.

Before using

Before using a BenchCel, your organization should make sure that you are properly trained in:

- ❑ General laboratory safety
- ❑ The correct and safe operation of the BenchCel
- ❑ The correct and safe operation of other lab automation systems or components used in combination with the BenchCel

If you are the person in your organization responsible for training others on the BenchCel and you have a safety question, please take the time to ask us.

Safety labels

Pay attention to any safety labels printed on your BenchCel. A safety label, shown below, consists of a warning symbol. A description of the warning and information that will help you to avoid the safety hazard is located in the general hazard warning section of the user guide.

Chemical hazards

Some chemicals used when working with the BenchCel may be hazardous. Make sure that you follow the recommendations in the MSDS (Material Safety Data Sheet) for every chemical that you plan to use. The manufacturer of the chemical should provide you with the MSDS.

Waste disposal

Remember to dispose of chemicals in accordance with local, state, and federal regulations.

Product use

Velocity11's products must only be used in the manner in which we intend, as described in our user guides. Any other use may damage the product or injure you. Velocity11 is not responsible for damages caused, in whole or part, by unauthorized modifications, or by procedures that are not explicitly described in our user guides. Any modifications or changes to products not expressly approved in Velocity11 user guides could void the warranty.

Related information

For information about...	See...
General hazard warnings	"BenchCel hazard warnings" on page 11
Who should read this guide	"Who should read this guide" on page 2
Reporting a problem to Velocity11	"Reporting BenchCel problems" on page 9
Using the guide	"About Velocity11 user guides" on page 3

BenchCel hazard warnings

About this topic

This topic gives more information for some hazard warnings found in this guide. Be aware of these hazards at all times when using the BenchCel.

Intended use

!! INJURY HAZARD !! The BenchCel must be used for its intended purpose as specified in this user guide. Failure to do so will compromise the BenchCel safety features.

Moving parts injury hazard

!! INJURY HAZARD !! Never attempt to touch any of the moving parts or attempt to remove plates while the BenchCel is in operation. Possible pinching, piercing, or bruising can occur.

!! INJURY HAZARD !! Do not launch software while touching the machine. Automatic motion begins if the machine is turned on and connected to the PC.

!! INJURY HAZARD !! Keep your fingers, hair, clothing, and jewelry away from the machine while it is in motion.

It is possible to be hit by the robot head when it is moving. If this happens, the robot head is designed to immediately stop, minimizing injury, and the likely outcome is minor bruising. However, the robot is particularly powerful in the vertical z-axis and could pierce a hand with one of its grippers.

Not all circumstances can be foreseen and more serious injury is possible. It is the responsibility of every operator to follow warnings and safety labels and keep out of the robot's workspace whenever it is likely to move.

Removed safety guard injury hazard

!! INJURY HAZARD !! Operating the BenchCel without safety guards or enclosure covers increases risk of injury.

We recommend that you enclose the BenchCel in a light curtain or safety guard. This restricts access to the BenchCel while it is operating or shuts down the BenchCel if you breach the curtain or open the guard.

Safety interlock override hazard

!! INJURY HAZARD !! Operating the BenchCel without the safety interlock circuit connected to an enclosure increases injury risk.

The BenchCel has a safety interlock circuit that must be closed for the system to operate. A jumper can be used to close the safety interlock circuit, but European directives regarding the safe operation of machinery require that you always connect the safety interlock to a safety guard or light curtain.

User account passwords damage hazard

!! DAMAGE HAZARD !! Only fully trained BenchCel administrators should have access to, and use of, the user account passwords.

Improper use of the robot by untrained personnel can lead to damage to the machine. For example, the robot grippers could collide with a stacker if a teachpoint is not defined properly.

Improper cleaning hazard

!! DAMAGE HAZARD !! Do not use harsh abrasives, corrosive cleaning agents, or metal brushes to clean any BenchCel components or accessories. Do not use any concentration of bleach (sodium hypoclorite). Do not allow cleaning agents to contact any electrical or sensitive mechanical components.

2 BenchCel overview

This chapter provides an overview of the BenchCel platform; its operating requirements, available accessories, and integration options.

BenchCel description

About the BenchCel

BenchCel defined

The BenchCel is a microplate-processing automation platform that is used to store microplates and move them to and from separate instruments.

Instruments defined

Instruments are individual pieces of equipment that sit on either side of the BenchCel and process microplates.

For example, a VCode instrument can apply a barcode to a microplate, which can then be sealed by a PlateLoc instrument.

In this example, the BenchCel transports the microplate from one of its stacker racks to the VCode, then to the PlateLoc, and then to another stacker rack.

Basic BenchCel components

Every BenchCel has the following:

- Two to six stacker racks
- A plate-handling robot
- A connection panel
- BenchWorks software
- Computer

Standard BenchCel parts

Included with every standard BenchCel purchase are the following:

- Plate-handling robot
- Stacker racks
- BenchWorks software
- Computer (if ordered)
- One power cord
- One 1/4- to 1/8-inch air hose reducer
- One 4 foot 1/8-inch air hose
- If you plan to use the serial connection:
  - Serial cables
- If you plan to use the Ethernet connection:
  - Ethernet cables
  - Ethernet hub or switch

Related information

For information about...	See...
Available accessories	“BenchCel accessories and integration options” on page 17
Connecting the BenchCel to the computer	“Connecting the controlling PC and BenchCel” on page 24
Compatible devices	“BenchCel accessories and integration options” on page 17

BenchCel configurations

Available stacker configurations

The BenchCel may be factory assembled with 2, 4, or 6 plate stackers.

Package	Contents
BenchCel 2X	2 Std Stacker racks BenchWorks (includes OCX) Dell desktop PC
BenchCel 4X	4 Std Stacker racks BenchWorks (includes OCX) Dell desktop PC
BenchCel 6X	6 Std Stacker racks BenchWorks (includes OCX) Dell desktop PC

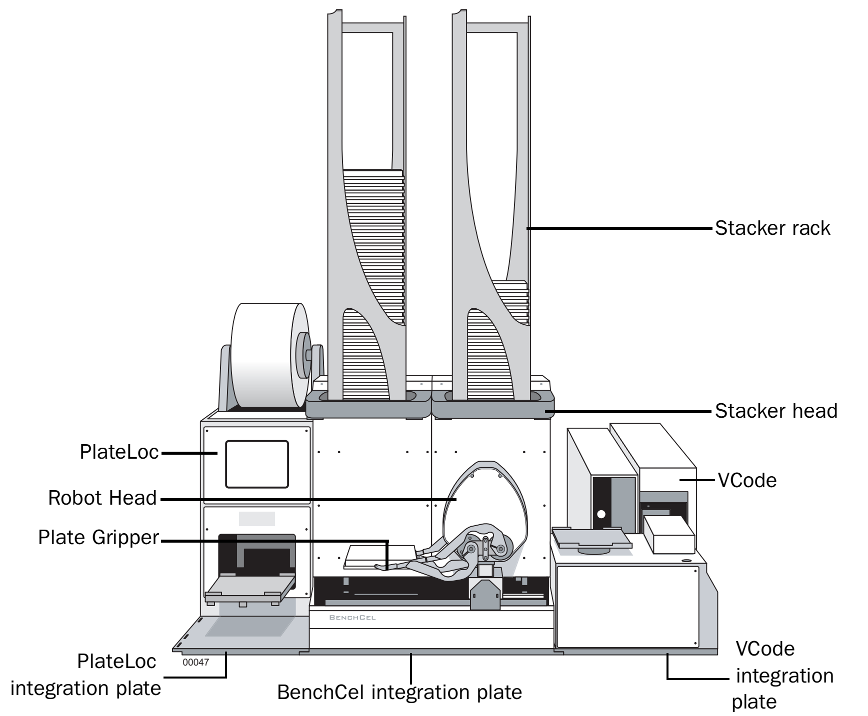
See “BenchCel accessories and integration options” on page 17 for additional accessories, options, and available integration packages.

BenchCel features

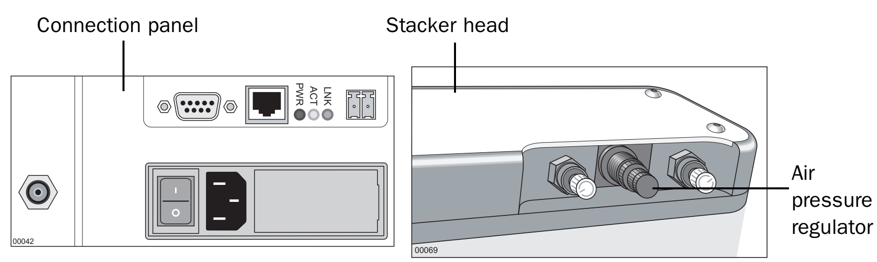
About this topic

This topic contains diagrams that display the main features of the BenchCel. These diagrams show a BenchCel with two stacker racks, connected to a PlateLoc and a VCode.

Front view



Rear view



BenchCel accessories and integration options

About this topic

Accessories provide a function without performing tasks themselves. For example, a platepad provides a temporary holding place for a plate.

Individual instruments that are mounted on integration plates on either side of the BenchCel can be used by the BenchCel robot to perform tasks with microplates. This topic lists the accessories that are currently available for the BenchCel.

If an instrument is not listed here

If there is a product not listed here that you would like to add to your lab automation system, please contact your Velocity11 sales representative or email your regional sales contact for customization information.

Regional sales contact	Email address
Asia	asiansales@velocity11.com
Europe	europeansales@velocity11.com
MidAtlantic	midatlanticsales@velocity11.com
Midwest	midwestsales@velocity11.com
Northeast	northeastsales@velocity11.com
Southeast	southeastsales@velocity11.com
Southwest	southwestsales@velocity11.com
West	westsales@velocity11.com

Compatible accessories

Accessory	Description
Stacker racks	Available in 4 sizes: ❑ Extra short (25.4 cm) ❑ Short (45.7 cm) ❑ Standard (66.0 cm) ❑ Tall (86.4 cm)
Laptop computer	Computer with a smaller footprint than the standard computer provided with the BenchCel.
Platepad	Provides a temporary storage place for a plate.
Plate hotel	Four platepads that are stacked vertically and configured as shelves in BenchWorks.
Auxiliary bar code reader	One or more bar code readers attached to a platepad.

Accessory	Description
Integration Plate	Each instrument and platepad requires a different integration plate to attach to the BenchCel.
ESTOP	An emergency stop button connected to the safety interlock circuit.

Compatible integrations

Velocity11 instruments

The following Velocity11 instruments can be integrated with the BenchCel:

Instrument	Description
PlateLoc	Thermal plate sealer
PlatePierce	Seal piercing station
VCode	Bar code print and apply station
VPrep	Liquid dispenser
VSpin with Access2	Microplate centrifuge station

Other companies' instruments

The following instruments from other companies can be integrated with the BenchCel. Before purchasing another company's instrument, ask Velocity11 about integration requirements.

This list is updated frequently with new instruments. Check with Velocity11 for the most recent information.

Instrument	Description
ABgene SEAL-IT 100	Adhesive plate sealer
Bio-Tek ELx405 Washer	Microplate washer
Genetix aliQuot	Low volume liquid dispenser
Genetix QFill2 Dispenser	Liquid dispenser
Innovadyne Nanodrop	Liquid dispenser
Labcyte Echo 550	Compound reformatter
Liconic STX (StoreX) and STR series	Plate incubators
Matrix Wellmate	Liquid dispenser
Molecular Devices instruments using SoftMax Pro software	Plate readers
PerkinElmer FlexDrop	Liquid dispenser
PerkinElmer ViewLux	Microplate imager

Instrument	Description
Thermo Electron Multiskan Ascent	Plate reader
Thermo Electron Multidrop	Microplate dispenser

Laboratory setup requirements

General bench requirements

Ensure that the bench for the BenchCel has the following:

- ❑ Proximity to utilities
- ❑ Enough space to accommodate the complete configuration of your BenchCel, which includes the number and size of stacks, computer, and any instruments
- ❑ Enough strength to support the BenchCel and instruments without excessive shaking or movement
- ❑ A fixed position (no wheels)
- ❑ Proper height for any users to comfortably operate the BenchCel and instruments

BenchCel space requirements

The minimum space requirements for your BenchCel depends on its stack and instrument configuration. The following table gives dimensions for a BenchCel with 2X, 4X, or 6X stack configuration.

Dimension	2X	4X	6X
Height w/short stack w/standard stack w/tall stack	43 cm 91.5 cm 107 cm 127 cm	43 cm 91.5 cm 107 cm 127 cm	143 cm 391.5 cm 107 cm 127 cm
Width	43 cm	65 cm	130 cm
Depth	20 cm	20 cm	20 cm
Weight	21.8 kg	28.1 kg	32.7 kg

Addition of instruments

If you are integrating your BenchCel with a Velocity11 device or another company's device, make sure to include adequate space to accommodate one or more instruments. The following table provides space requirements for some of the more typical configurations.

Instrument	Dimensions (cm) L x W x H
VCode	28.4 x 28.4 x 61
PlateLoc	39.9 x 58.4 x 21.6
VPrep	25.9 x 78.7 x 38.1
PlatePierce	20.3 x 35.6 x 34.4
VSpin with Access2	71.4 x 32.7 x 24.8

Utility and environmental requirements

Electrical requirements

The BenchCel requires access to the following source of power for the BenchCel, operating computer, and any integrated instruments:

Utility	Requirement
Electrical	100–206 VAC, 50/60 Hz

Compressed air requirements

The BenchCel requires access to the following source of compressed air.

Utility	Requirement
Compressed air	28 Lpm at 6 bar (1 cfm at 80 psi)

Environmental operating requirements

The BenchCel requires the following environmental conditions:

Parameter	Requirement
Temperature	5–40 ° C
Relative humidity	10–90%
Elevation	1–2000 meters

The computer and networking

About this topic

This topic describes the computer and network connection requirements for running a BenchCel.

Computer functions

You need a computer to control the BenchCel and run BenchWorks.

The computer you connect to the BenchCel is used to:

- ❑ Control the BenchCel and its instruments
- ❑ Input bar code labelling instructions
- ❑ Store and export log files
- ❑ Store and export data collected by plate readers
- ❑ Communicate with other computers on your company's network for exchange of files, reporting of errors and accessing email

Connecting to the BenchCel

The BenchCel can be controlled via Ethernet or serial cable.

If you connect the BenchCel to the host computer via Ethernet, and the host computer is already connected to a company LAN or other local network, you need a second, separate network card for the BenchCel. This lets the BenchCel and any of its ethernet controlled instruments operate on an isolated network.

Note: If you purchased your computer from Velocity11, it will have two network cards.

If you connect the BenchCel to the host computer with a serial cable, use the standard 9-pin RS-232 connector supplied with the BenchCel.

Minimum system requirements

If you are using your own computer to run the BenchCel, make sure that it has the following minimum requirements:

- ❑ PC system
  - ◆ Pentium 4, 2GHz or better
  - ◆ 256 MB RAM
  - ◆ Windows 2000 or XP Professional operating system
  - ◆ 50 GB free hard disk space
- ❑ Communication interface
  - ◆ Dedicated 10BaseT Ethernet port (two network cards if connecting to your company LAN)
  - ◆ RS-232 serial port

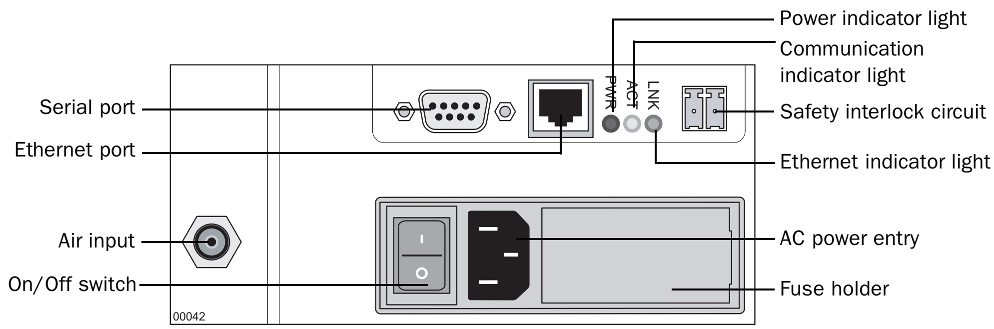
Connection panel description

About this topic

This topic uses an illustration to point out the specific connections of the connection panel at the rear of the BenchCel.

Connection panel for utilities

The diagram and the following table identify the inputs and outputs on the connection panel.



Item	Description
Serial port	A DB-9 RS-232 serial port that provides serial control of the BenchCel and firmware programming. If you connect to the BenchCel using the serial port, you do not need to connect through the Ethernet port.
Ethernet port	An RJ-45 Ethernet port that provides ethernet control of the BenchCel. Connect to the host computer with a shielded Cat-5 Ethernet cable. If you connect to the BenchCel using the Ethernet interface, you do not need to connect through the serial port.
Indicator lights	❑ Power indicator light is red when there is power to the BenchCel ❑ Communication indicator light flashes yellow when communicating to the software ❑ Ethernet indicator light is green when there is an Ethernet connection

Item	Description
Safety interlock circuit	The BenchCel is equipped with a safety interlock circuit that must be closed for the system to operate. Normally, the BenchCel is shipped with a relay jumper which allows the BenchCel to function without an interlock. An external circuit may be added to provide full ESPTOP (emergency stop) safety. Contact Velocity11 to obtain an ESTOP and connector. This circuit can also be fitted with a light curtain to shut off power to the BenchCel if the light boundary is breached. Light curtains are not currently available from Velocity11.
Fuse holder	Contains the main fuse. Use two 250 VAC, 5 A, fast-acting fuses.
AC power entry	Used to connect a power cord.
On/Off switch	Switches main power on (I) or off (O).
Air input	Used to connect a 1/8-inch air hose to supply the BenchCel with 28 Lpm at 6 bar (1.0 cfm at 80 psi). *Note:* Required air pressure depends on your platform configuration.

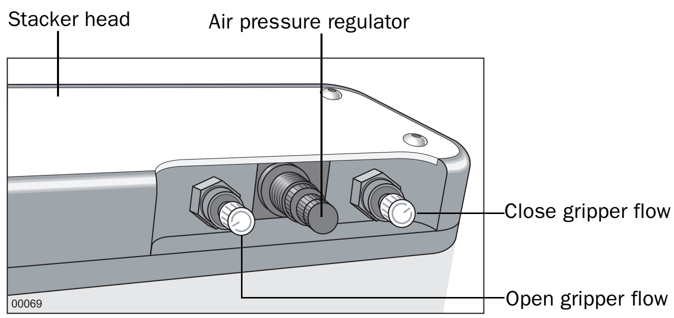
Stacker-gripper control description

About this topic

This topic describes the control knobs for the stacker grippers.

Controls for stacker grippers

On the back side of every stacker head are control knobs that regulate the stacker grippers. Refer to the diagram and table below for a description.



Item	Description
Air pressure regulator	Controls the air pressure to the stacker head.
Gripper flow control	Controls the air flow to the grippers to regulate the speed they grip and retract.

Connecting the controlling PC and BenchCel

About this topic

The following topic describes how to connect the BenchCel and controlling PC to a main power line and to each other with a serial or Ethernet connection.

Connecting to power

To connect the BenchCel and PC to power:

1. Plug one end of the power cord into the power entry on the back of the BenchCel and the other end into an appropriate electrical outlet.
2. Plug one end of the power cord into the power entry on the back of the operating computer and the other end into an appropriate electrical outlet.

Connecting by serial cable

To connect the computer to the BenchCel with a serial cable:

1. Plug in a standard 9–pin male serial cable to the RS-232 port on the BenchCel.
2. Plug the other end of the cable into one of the computer's serial ports.

Connecting by Ethernet cable

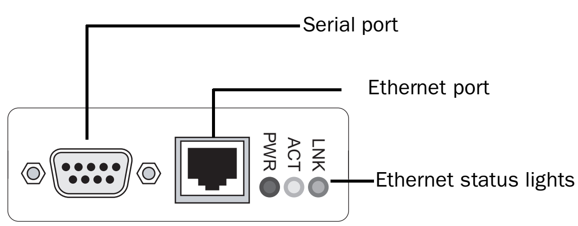
To connect the computer to the BenchCel with an Ethernet cable:

1. Connect one end of an Ethernet cable to the Ethernet port on the BenchCel.
2. Connect the other end of the Ethernet cable to your computer, hub or switch.

   !! IMPORTANT !! To connect the computer directly to the BenchCel, use a crossover Ethernet cable. If you are connecting the computer to a hub or switch, use a straight-through Ethernet cable.

3. After you connect the BenchCel to power and to the computer and turn the BenchCel on, verify the connections by checking the status lights.

Light	Color	Activity	Meaning
PWR	Red	Light turns on	The system has power
ACT	Yellow	Light is blinking	There is Ethernet activity
LNK	Green	Light turns on	An Ethernet link has been established



Connecting the BenchCel to compressed air

About this topic

This topic describes the information needed to connect the BenchCel to compressed air.

Air pressure

The following table lists the air pressure range required for the BenchCel and associated instruments.

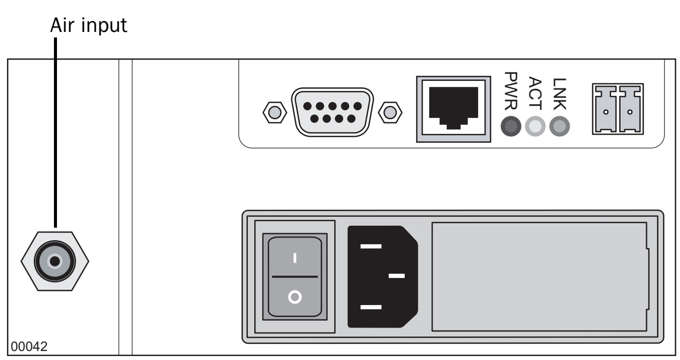
Unit	Air Pressure (psi)	Air Pressure (MPa)
BenchCel	50–90	0.34–0.62
VPrep	90–95	0.62–0.65
PlateLoc	85–90	0.56–0.62
VCode	70–75	0.56–0.62
VSpin with Access2	80–90	0.55–0.59

Procedure

To connect the BenchCel to the compressed air:

1. Connect one end of the air supply tubing to the source of compressed air.
2. Connect the other end to the air intake port on the BenchCel.

   If you have another instrument connecting to the BenchCel that requires compressed air, split the air supply tubing coming from the source.



Related information

For information about...	See...
Regulating stacker gripper pressure	“Adjusting the stacker gripper pressure” on page 320

About BenchWorks

About this topic

This topic gives an overview of BenchWorks, the software that runs the BenchCel.

What BenchWorks does

BenchWorks is the application that lets you control the BenchCel.

Through BenchWorks, you can quickly automate a series of tasks. BenchWorks lets you assemble a modular set of predefined tasks into processes.

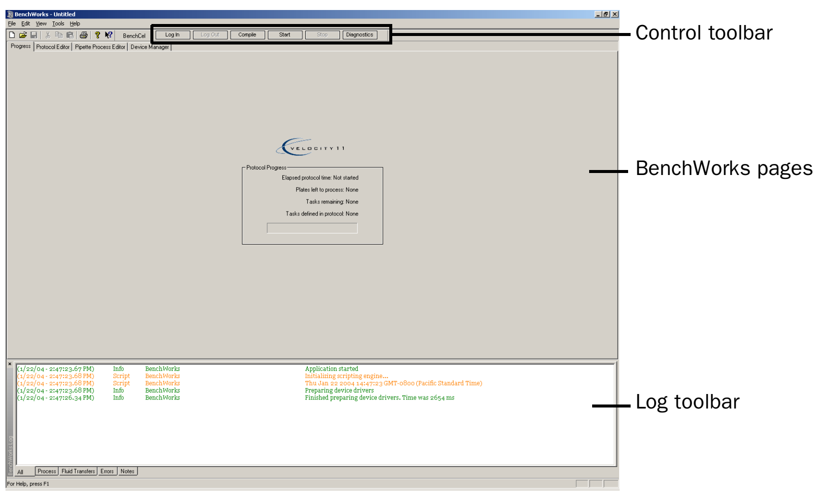
BenchWorks prerequisites

To control the BenchCel with BenchWorks, your computer must be connected to your BenchCel.

To connect your BenchCel to your computer, see “Connecting the controlling PC and BenchCel” on page 24.

BenchWorks main window overview

When you open BenchWorks, you are presented with the following window:



Element	Description
Control toolbar	Provides basic BenchWorks commands
BenchWorks page	Displays one of the six main BenchWorks pages
Log toolbar	Displays a log of BenchWorks actions

Control toolbar

There are seven buttons on the Control toolbar.

Button	Use to...
Log In	Logs a user into BenchWorks. This provides a level of security by controlling access to software security levels. See "Setting protocol rules" on page 49.
Log Out	Logs a user out of BenchWorks. See "Logging out of BenchWorks" on page 81.
Compile	Checks the protocols for errors. See "Compilation warnings and errors" on page 303.
Start	Begins a protocol or run. See "Starting a run from BenchWorks" on page 70.
Pause	Interrupts a protocol or run after it has been started. See "Pausing or stopping a run" on page 74.
Simulate	Helps you find errors that you would encounter during an actual run. See "Simulating a run" on page 97.
Diagnostics	Displays the BenchWorks Device list providing access to diagnostics for all of the installed devices. See "Using BenchCel Diagnostics" on page 251.

BenchWorks pages

Each BenchWorks page is accessed from a tab in the upper left corner of the page.

Page	Use to...
Progress	Display the status of the current protocol. See "Monitoring overall progress" on page 77.
Pre-Protocol Editor	Create a process to be executed before the protocol runs. See "Setting up a pre-protocol or post-protocol process" on page 87.
Protocol Editor	Create and edit protocols and processes. See the following topics: ❑ "Creating a protocol: basics" on page 83. ❑ "Creating a protocol: advanced topics" on page 163.

Page	Use to...
Post-Protocol Editor	Create a process to be executed after the protocol run is completed. See “Setting up a pre-protocol or post-protocol process” on page 87.
Pipette Process Editor	Create and edit pipette protocols and processes. This is only used if a VPrep is one of the configured peripheral devices. See “Adding and configuring a Pipette Process task” on page 140.
Device Manager	Manage and configure devices. Every device that is added to the platform must be added to the device file. See “Working with device files” on page 195.

Log toolbar

The Log toolbar can display different subsets of log data or display all log data. You can also add a note to a log.

There are five pages in the Log toolbar for different data. The five tabs to access these pages are in the lower left corner of the page.

Page	Description
All	Displays all logs
Process	Displays BenchCel process logs
Fluid Transfers	Displays VPrep fluid transfer logs
Errors	Displays run errors
Notes	Lets you enter a time stamped note into the log

See “Working with the Log toolbar” on page 77.

Relationships of BenchWorks components

About this topic

BenchWorks uses four file types to run the application. What they are and how they work together to operate the BenchCel is described below.

What you should know

It is important to understand the way each of the main configuration components in BenchWorks relate. Loading some components automatically loads others. For example, loading a protocol file loads a device file.

Definitions

Refer to the table below for information about these four BenchWorks configuration components:

Component	Definition	See
Protocol file	A file that contains instructions for performing a run.	"About protocol files" on page 84
Device file	A file that contains configuration information for configured devices.	"Working with device files" on page 195
Profile	A collection of settings, stored in the registry, that manages how you connect to devices.	"About profiles" on page 282
Teachpoint File	A file that saves your teachpoint settings.	"Managing teachpoint files" on page 274

Configuration component relationships

Refer to the table below to understand the relationships of these four BenchWorks configuration components:

Component	Extension	Opening this file loads...
Protocol	.bwl	❑ BenchWorks (if it is not already running) ❑ Device file ❑ Profile ❑ Teachpoint file
Device file	.dev	❑ Profile ❑ Teachpoint file
Profile	None	Teachpoint file (you are prompted)
Teachpoint file	.xml	Teachpoint definitions

Preparing for a run 3

This chapter describes how to setup the BenchCel and its instruments within BenchWorks to run an existing protocol. All of the procedures in this chapter can be performed by someone with operator privileges.

Workflow for preparing for a run

About this topic

This topic gives the order of recommended tasks before performing a run and tells you where to look for information and procedures for each task.

Workflow

The general workflow for starting the BenchCel and preparing for a run is listed in the following table:

Step	Topic
1	“Turning on the BenchCel” on page 33
2	“Logging into the operating system” on page 34
3	“Starting BenchWorks” on page 34
4	“Logging in to BenchWorks” on page 36
5	“Opening a protocol in BenchWorks” on page 40
6	“Setting general options” on page 41
7	“Setting general error-handling options” on page 44
8	“Setting up email error notification” on page 45
9	“Setting protocol error-handling options” on page 47
10	“Setting pre-protocol rules” on page 48
11	“Setting protocol rules” on page 49
12	“Setting log options” on page 54
13	“Understanding the protocol” on page 59
14	“Preparing plates, instruments, and accessories” on page 61
15	“Setting up a VSpin counterweight plate” on page 63
16	“Installing a rack” on page 64

Related information

For information about...	See...
Protocols	“About protocol files” on page 84
BenchWorks	“About BenchWorks” on page 27
BenchWorks components	“Relationships of BenchWorks components” on page 30

Turning on the BenchCel

About this topic

This topic describes how to turn on the BenchCel and integrated instruments if they are not already on.

Procedure

To turn on the BenchCel and configured instruments:

1. Make sure that the main power line, air line, and Ethernet or serial cables are plugged into the connection panel.

   !! INJURY HAZARD !! Keep your fingers, hair, clothing, and jewelry away from the machine while it is in motion.

2. Power-on any connected instruments. See their respective user guides for power switch locations.
3. On the connection panel, press the power switch to the **on (|)** position.

   The robot homes.

Note: If the BenchCel robot does not home or if the power indicator lights do not come on, turn off the BenchCel, check the connections and turn it on again.

Related information

For information about...	See...
Indicator lights	"Connection panel description" on page 22
Where this step fits in to the preparing for a run process	"Workflow for preparing for a run" on page 32
The next step	"Logging into the operating system" on page 34

Logging into the operating system

About this topic

After turning on the BenchCel, you may log in to the Windows operating system on the computer. This topic describes how to do that.

Procedure

To log in to the Windows operating system:

1. At the **Welcome to Windows** screen, press CTRL + ALT + DELETE.
2. At the **Log On to Windows** screen, type your user name and password.
3. If necessary, in the **Log on to** text box, type the name of the network domain to which your BenchCel is connected.

   Contact your network administrator if you do not know your network domain name. You should only need to type in this name once because the system will remember it.
4. Click **OK**.

Related information

For information about...	See...
The workflow this procedure belongs to	"Workflow for preparing for a run" on page 32
The next step	"Starting BenchWorks" on page 34

Starting BenchWorks

When to start BenchWorks

Start BenchWorks after you have turned on the BenchCel and computer and logged on to the computer operating system.

Starting BenchWorks

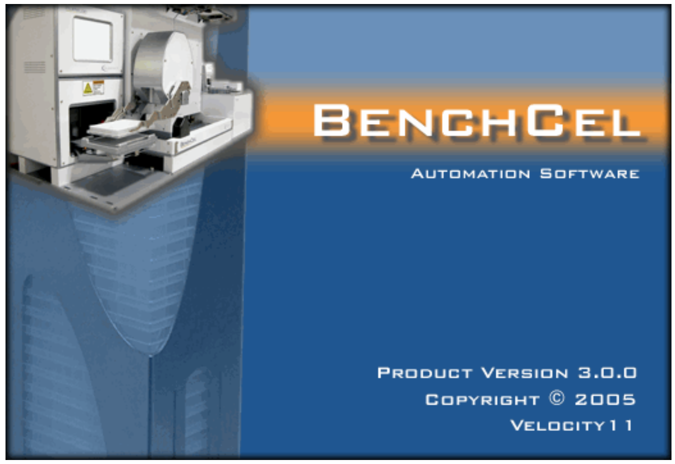
To start BenchWorks:

1. Make sure that everyone is clear of the BenchCel and that there are no objects that could obstruct the robot.

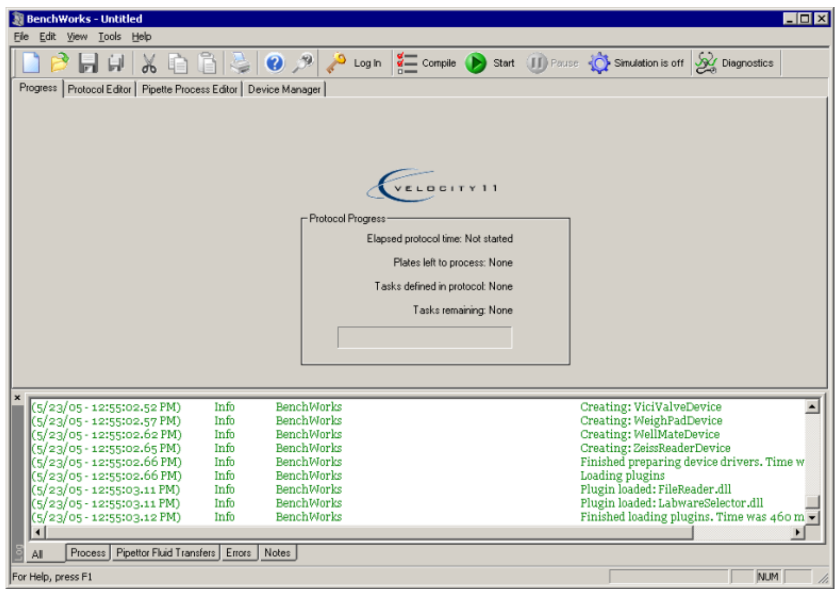
   !! INJURY HAZARD !! When BenchWorks starts, robot and instrument parts may unexpectedly move to their home positions.
2. Double-click the shortcut to BenchWorks on the Windows desktop.

   Note: If the shortcut has been deleted, open the folder C:\Program Files\Velocity11\BenchWorks and create a new shortcut from the executable file BenchWorks*version*.exe.

   The BenchWorks splash screen opens.



BenchWorks opens at the **Progress** tab, and you are logged in with guest privileges.



Related information

For information about...	See...
The workflow this procedure belongs to	"Workflow for preparing for a run" on page 32
The next step	"Logging in to BenchWorks" on page 36

Logging in to BenchWorks

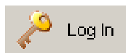
About this topic

To log in to BenchWorks you need a user account, created by an administrator. This topic describes how to log in and change your password once you have a user account.

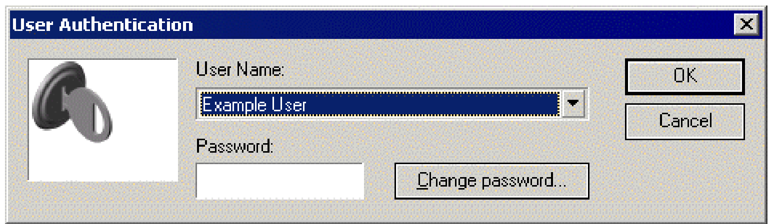
Logging in

To log in to BenchWorks:

1. Click **Login**.



2. Select your account **User Name** from the list box.



3. Enter your password and click **OK**.

Changing your password

You can change the password for your user account at any time.

To change your user account password:

1. Click the **Login** button.
2. Select your account **User Name** from the list box.
3. Click the **Change Password** button.
4. In the **Change Password** dialog box:
   a. Type your old password in the **Old** text box.
      If the **Old** text box field is gray if there is no password currently selected for the account.
   b. Type your new password in the **New** text box.
   c. Type your new password again in the **Confirm New** text box.
   d. Click **OK**.
5. Click **OK**.

Related information

For information about...	See...
Workflow this procedure belongs to	"Workflow for preparing for a run" on page 32
The next step	"About tasks, processes, and protocols" on page 37
Creating user accounts	"Adding and deleting a user account" on page 205
User permissions	"About user accounts and privileges" on page 204
Job roles and responsibilities	"Who should read this guide" on page 2

About tasks, processes, and protocols

About this topic

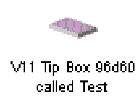
This topic defines some terms that you need to know before you can understand or create protocols.

Plate icon defined

A plate icon represents the basic information about a plate or collection of plates and is the first icon in a process displayed in the protocol editor. It has associated parameters that are defined in the Task Parameters toolbar.

The information it represents includes the type of labware used in the process, how many plates are available for processing at one time, whether the plates have lids, and so on.

The following example icon represents a plate icon for a tipbox.



Task defined

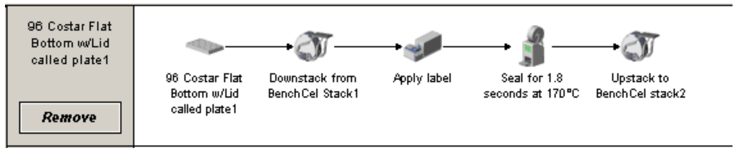
A task is an operation that is performed on one or more plates and is represented by an icon in the protocol editor. It has associated parameters that are defined in the Task Parameters toolbar.

The following example icon represents a task for downstacking plates.

Process defined

A process is a sequence of tasks that are performed on a plate icon.

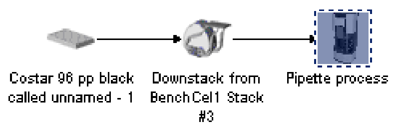
The following diagram shows a process with one plate icon and four task icons.



How a plate icon and task differ

Although the icon for a plate icon looks like a task icon, its function is different because it does not represent an action in a protocol. This distinction becomes important when you are interpreting error messages. Consider the following example protocol and error message.

The error refers to task 2, which is the Pipette Process task in the following diagram, and not the Downstack task. The first icon is the plate icon.

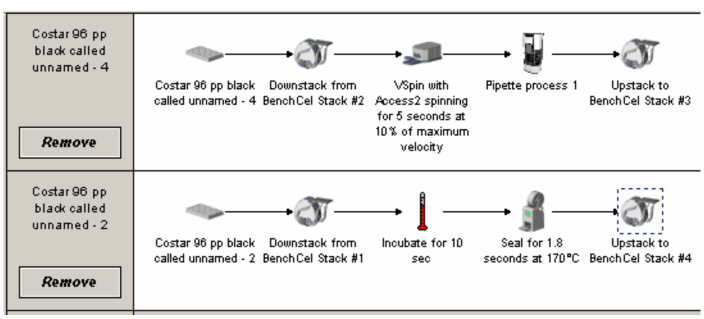


Error in Test, task 2: Pipette process 1 is required to process Test but does not

Protocol defined

A protocol is a collection of processes that run at the same time, as defined in the process editor.

The following diagram shows one protocol with two processes.



Pre-protocol defined

A pre-protocol is a collection of processes that are executed once, before the protocol. The processes are created in the pre-protocol editor which is accessed by clicking the Pre-Protocol Editor tab in the main BenchWorks window.

Priming reservoir pumps is an example of a pre-protocol task.

When you click Start, if there is a pre-protocol, it is executed first, followed by the protocol.

Post-protocol defined

A post-protocol is a collection of processes, which are executed once after the protocol. The processes are created in the Post-Protocol editor which is accessed by clicking the Post-Protocol Editor tab in the main BenchWorks window.

A post-protocol is typically used to clean the system after a protocol run.

When you click Start, a pre-protocol may run first, followed by the protocol and then any post-protocol processes.

Pipette process task defined

A Pipette Process task is the parent of a sequence of pipette tasks that perform liquid handling procedures using a VPrep or Multimek. A pipette process is sufficiently complex that it has to be defined by a group of separate sub-tasks, each with its own parameters.

An icon for a pipette process task is shown below.

Run defined

A run is a single protocol, which includes any pre-protocol and post-protocol processes, that is performed one or more times in a series.

When you start a run, you are prompted to enter the number of cycles that you want to run. The value that you enter represents the number of times the protocol executes before the run ends. For example, a value of four means that the protocol runs four times in the series.

Pipette process link icon

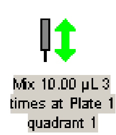
The pipette process link icon is always the first icon in a pipette process. The icon performs a similar function to a plate icon in that it does not represent a task itself but is the first icon in a pipette process.

The pipette process link icon's function is to link a pipette process task in the protocol editor to a pipette process in the pipette process editor.

Pipette task defined

A pipette task is an operation that is performed on one or more plates by a VPrep or a Multimek. It is represented by an icon in the pipette process editor.

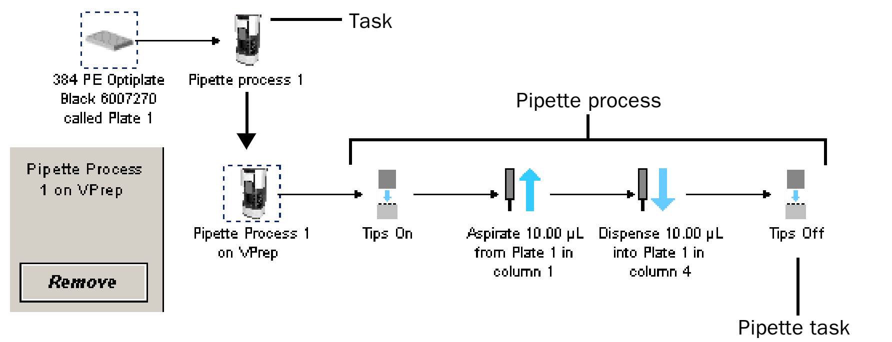
The following diagram shows a pipette task.



Pipette process defined

A pipette process is a sequence of pipette tasks that is performed on a specific plate, as defined in the process editor. It is a subroutine of a pipette process task.

The following diagram describes the relationship between a task, pipette process task, and a pipette process.



Related information

For information about...	See...
The workflow this procedure belongs to	"Workflow for preparing for a run" on page 32
The next step	"Opening a protocol in BenchWorks" on page 40

Opening a protocol in BenchWorks

About this topic

All runs on the BenchCel require a protocol. This topic describes how to open a protocol that has already been created.

Procedure

To open a protocol:

1. Select **File > Protocol File** > **Open**.
2. Navigate to the folder containing the protocol file.
3. Select the protocol file and click **OK**.

You can also open a protocol by navigating to the **.bwl** file in Windows and double-clicking it. This launches BenchWorks and opens the protocol.

Related information

For information about...	See...
The workflow this procedure belongs to	"Workflow for preparing for a run" on page 32
The next step	"Setting general options" on page 41

Setting general options

About this topic

This topic describes how to configure the general options in the BenchWorks Options dialog box.

General options include the following:

- Location of the bar code input or data file
- Maximum robot speed
- Visibility of task icons
- Release of BenchCel stackers

When to use

Review general options after you open a protocol but before starting a run.

!! IMPORTANT !! Protocols do not store Options dialog box settings as part of the protocol. This means that all protocols use the settings that are currently selected in the Options dialog box.

Procedure

You may decide to keep all or many of the options the same for every run.

!! IMPORTANT !! If you use bar code data files, remember to select the correct file for every run.

To set general options:

1. Select **Tools** > **Options**.
2. Make sure that the **Options** page of the **BenchWorks Options** dialog box is selected.
3. If you are using a bar code file, select its location:
   a. In the **Bar Code Settings** group box, click the ellipsis button (...) next to the appropriate type of bar code file.
   b. In the **Open** dialog box, navigate to the folder that contains the bar code file.
   c. Select the file (with a **.bar** filename extension for an input file and a **.dat** filename extension for a database file) and click **Open**.

4. In the **Robot Settings** group box, select the desired maximum speed of the robot movement.

   If the plate-specific robot speed (set in the **Maximum Robot Handling Speed** group box of the Labware editor) is different from the general robot speed, the slower of the two speeds is used.

   !! DAMAGE HAZARD !! If you are testing a new protocol or learning to use the BenchCel, run the robot at a slow or medium speed to reduce the risk of damage in the event of a crash.

5. Select an option in the **Protocol Editor Settings** group box, if desired.

   It is strongly recommended that you select the **Hide icons for non-configured device types** check box. This makes sure you choose the right icons for the devices available to you. This is especially important where similar icons are used for different tasks.

6. In the **BenchCel Settings** group Box, select **Release BenchCel stackers when filled during run** to be able to remove the stackers after they have been filled.

7. Click **OK** to close the **BenchWorks Options** dialog box.

Related information

For information about...	See...
The workflow this procedure belongs to	“Workflow for preparing for a run” on page 32
The next step	“About setting error options” on page 43

About setting error options

About this topic

This topic provides an overview the types of error handling options available when running a protocol with the BenchCel.

Review error options after you open a protocol, and review general options before starting a run. You may not need to do this for every run.

What error handling options include

- ❑ General error options
  - ◆ *Error reporting*. Choose from a list of error types that you want reported.
  - ◆ *Error handling*. Direct the BenchCel on how to handle errors encountered during the run.
  - ◆ *Scheduler error behavior*. Direct the BenchWorks scheduler on how to handle errors encountered during the run.
- ❑ *Error notification options*. Set up email notification in BenchCel to email or page you when a run error occurs.
- ❑ *Protocol error handling options*. Specify how the BenchCel should behave if it encounters an error while executing your protocol.

Related information

For information about...	See...
The workflow this procedure belongs to	“Workflow for preparing for a run” on page 32
The next step	“Setting general error-handling options” on page 44
Protocol error-handling options	“Setting protocol error-handling options” on page 47
Error notification	“Setting up email error notification” on page 45

Setting general error-handling options

About this topic

This topic describes the handling of general error options. See "Related information" on page 45 for information about protocol error options and email error notification.

Procedure

To set general error handling options:

1. Select **Tools > Options**.
2. Click the **Error Handling** tab of the **BenchWorks Options** dialog box.
3. In the **Error Reporting** group box, select the devices for which you want to report errors.

   Typically, all of the devices are selected for a run. If you want to perform a test run without plates, clear the options for devices used in the protocol that would otherwise report errors.
4. Select from the following options in the **Error Handling** group box:

Option	When checked...
Send Email when errors occur	The people listed in the **Email Setup** tab of the **BenchWorks Options** dialog box will receive an email for every error notification.
Halt on barcode misreads	The robot will halt the run if it encounters a bar code misread.
Halt on barcode database lookup errors	The robot will halt the run if the bar code found in the database does not match the plate bar code.
Launch program if error occurs...	A program that you specify is launched. You must specify the program by clicking the ellipsis button (...) and browsing to the program's executable. With the appropriate script, this function can be used to send a page or an email. If you want to pass the text of the error message to the program, select the **Add error text as command line argument** check box.
Halt if available disk space falls below...	The robot completes the currently scheduled step and then stops if the percentage of available hard disk space is less than the percentage specified in the text box.
Sound alarm on output...	Not for use with the BenchCel.

5. In the **Scheduler Error Behavior** group box, select one of the following options:

Option	When selected, in the event of an error...
Process as many plates as possible	As many tasks as possible, given the error, are completed.
Continue processing without starting any new plates	Tasks involving plates that are currently in the system continue. Other tasks are not scheduled.
Stop scheduler	The scheduler stops scheduling new tasks, even if plates are currently available to the robot. The current task continues to completion.

6. Click **OK** to close the **BenchWorks Options** dialog box.

Related information

For information about...	See...
The workflow this procedure belongs to	"Workflow for preparing for a run" on page 32
The next step	"Setting up email error notification" on page 45
Protocol error-handling options	"Setting protocol error-handling options" on page 47
Error options	"About setting error options" on page 43

Setting up email error notification

About this topic

This topic describes how to add an email address to BenchWorks so you can be notified by email or pager when there is a run error.

Email setup in BenchWorks enables you to do the following tasks:

- ❑ Automatically be notified by email or pager when errors occur during a protocol run
- ❑ Send a bug report to Velocity11

Requirements for email setup

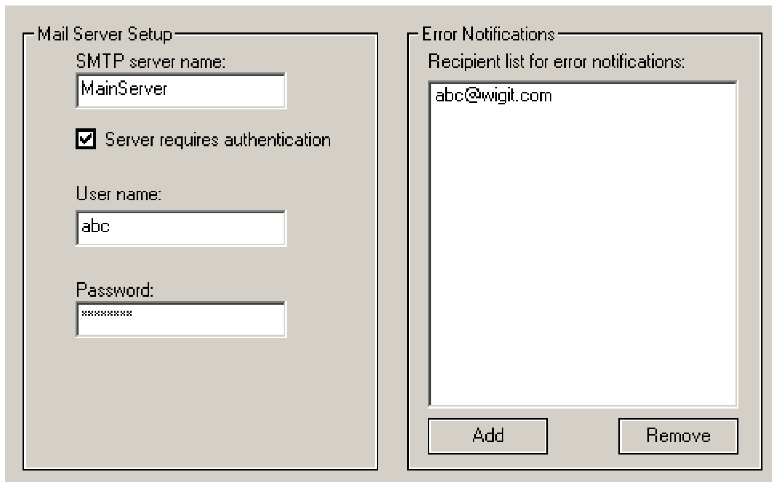
Before you can send an email from BenchWorks:

- ❑ The BenchCel computer must be connected to a network with internet access
- ❑ The outgoing email server must be set up on the system's computer

Setting up email

To set up the outgoing email server:

1. Select **Tools > Options**.
2. In the **BenchWorks Options** dialog box, click the **Email Setup** tab.
3. In the **Mail Server Setup** group box, enter the name of your **SMTP server name** (outgoing email server).
4. If the server requires a user name and password:
   a. Select the **Server requires authentication** check box.
   b. Enter the **User name** and **Password** that you use to access email on the network.



This information only needs to be set up once, provided the email account remains active. All email sent from BenchWorks is authenticated using this account.

Related information

For information about...	See...
The workflow that this procedure belongs to	"Workflow for preparing for a run" on page 32
The next step	"Setting protocol error-handling options" on page 47
How to setup the outgoing mail server	"Setting up email" on page 206
How to send a bug report using email	"Sending a bug report" on page 321

Setting protocol error-handling options

About this topic

This topic describes the error-handling options available in the Protocol Options dialog box.

There are four groups of protocol options.

Protocol Option	See...
Device File	"Working with device files" on page 195
Description/Notes	"Compiling and saving protocols" on page 93
Rules	"Using JavaScript in BenchWorks" on page 171 "Setting pre-protocol rules" on page 48 "Setting protocol rules" on page 49
Error Handling	This topic

Setting protocol error handling options

To set protocol error handling options:

1. Select **Tools > Protocol Options**.
2. In the **Error Handling** group box, select any or all of the following options:

Option	Description
Attempt to put plates away if deadlock occurs	If a deadlock occurs, any lids are replaced and the plates are moved to the positions they would be in at the end of a successful protocol run.
Abort run if stacker runs out of plates	If the stacker runs out of plates before the run is finished, the run is aborted. Aborting a run permanently terminates the run.

Related information

For information about...	See...
The workflow this procedure belongs to	"Workflow for preparing for a run" on page 32
The next step	"Setting pre-protocol rules" on page 48

Setting pre-protocol rules

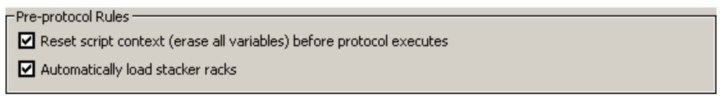
About this topic

This topic describes the pre-protocol rules available in the Protocol Options dialog box.

Pre-protocol rules include

There are two rules in the Pre-Protocol Rules group box:

- ❑ Reset script context (erase all variables) before protocol executes
- ❑ Automatically load stacker racks



Reset script context

Select this option if you want all variables to be cleared before the next protocol is executed.

Automatically load stacker racks

Select this option if you want all racks on stacks to automatically load before the next protocol is executed.

Related information

For information about...	See...
The workflow this procedure belongs to	“Workflow for preparing for a run” on page 32
The next step	“Setting protocol rules” on page 49
Using and resetting scripts	“Using JavaScript in BenchWorks” on page 171
General options	“Setting general options” on page 41

Setting protocol rules

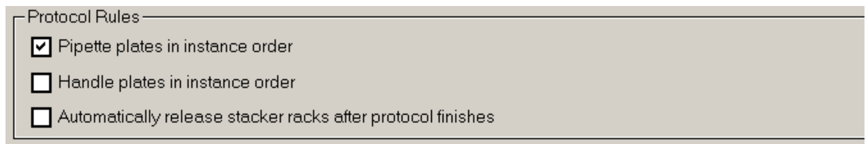
About this topic

This topic describes the protocol rules in the Protocol Options dialog box.

Protocol rules include

There are three rules in the Protocol Rules group box:

- ❑ Pipette plates in instance order
- ❑ Handle plates in instance order
- ❑ Automatically release stacker racks after protocol finishes



Pipette plates in instance order

Default setting

The default setting for the Pipette plates in instance order rule is for it to be checked, and it should be left checked for most protocols.

When to clear the rule

If your protocol has all of the following attributes, consider clearing this check box:

- ❑ The protocol has more than one pipette process that uses the same VPrep.
- ❑ The duration of one of the pipetting operations is much longer than another.

Explanation

Consider an example in which a protocol has two processes and both have a pipetting operation that uses the same VPrep. When the check box is selected, the pipetting operations for one process are completed before the pipetting operations begin for the other process.

Now consider what happens when the first pipetting operation takes significantly longer to complete than the second operation, and the protocol is run several times in succession. The overall time taken for the protocol to complete is much greater than it needs to be because during each cycle the system had to wait for the slower pipetting operations to complete for all the plates in the process before it could continue.

If the rule is turned off, a plate from the fast pipetting process can be delivered to the VPrep after a plate from the slow pipetting process, followed by another plate from the slow pipette process, and so on. This reduces the bottleneck at the VPrep because it allows the faster process to continue, and its second cycle in the series to start before the first cycle is complete.

Handle plates in instance order

Default setting

The default setting for the Handle plates in instance order rule is for it not to be selected.

When to select the rule

Use this rule if you need the plates to be handled in the order in which they enter the system.

Explanation

Consider a situation in which you are using BenchCel to seal plates using two PlateLocs and the first one runs out of seal, stopping on plate 5.

If this option is not selected, the second sealer continues sealing and upstacking plates but plate 5 would be omitted and thus out of order.

If this option is selected, the second sealer stops until you reload a new roll of seal and start the first sealer again and plate 5 is upstacked to its position in order.

Automatically release stacker racks after protocol finishes

Default setting

The default setting for the “Automatically release stacker racks after protocol finishes” setting is for it not to be checked.

When to select the rule

Use this rule if you have one or more stackers and you want all them to release their racks at the end of the protocol.

Related information

For information about...	See...
The workflow this procedure belongs to	“Workflow for preparing for a run” on page 32
The next step	“Setting log options” on page 54

About log and data files

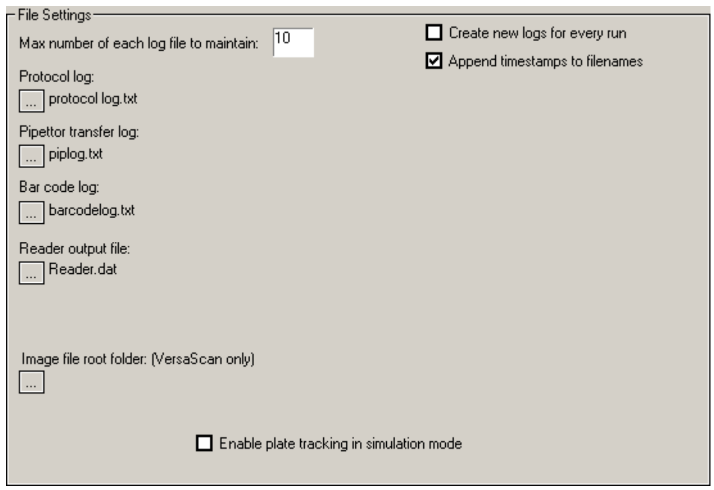
About this topic

This topic describes the different types of logs that BenchWorks creates.

Types of log and data files

Log files record event and error information in text files that can be useful for troubleshooting. Data files record data collected by devices such as plate readers.

The location of the three log files, a data file, and a folder are set in the Log Options page of the BenchWorks Options dialog box.



These are listed in the following table and described in more detail later in this topic:

File	Default file or folder name
Protocol log file	log.txt
Pipettor transfer log file	piplog.txt
Bar code log file	barcodelog.txt
Reader output file	Reader.dat
Image file root folder	A folder that you select

Note: You can change the default filenames to suit your own needs, but they are referred to using their default names throughout this guide.

About opening log files

You can open a log file in any text editor, but we recommend that you use the Windows application Notepad because with Notepad you can open the file during a run as the file is being written.

Protocol log file

The protocol log file records all available event and error information. The information recorded in the protocol log file cannot be modified.

```
protocol log(Monday, May 9, 2005 @ 4_38_58 PM).txt - Notepad
File  Edit  Format  Help
(5/9/05 - 4:39:06.69 PM) Info    BenchWorks  Creating: OriginalVCodeDevice
(5/9/05 - 4:39:06.70 PM) Info    BenchWorks  Creating: PipettorFilterShelfDevi
(5/9/05 - 4:39:06.70 PM) Info    BenchWorks  Creating: PipettorReagentShelfDev
(5/9/05 - 4:39:06.71 PM) Info    BenchWorks  Creating: PipettorServoShelfDevic
(5/9/05 - 4:39:06.71 PM) Info    BenchWorks  Creating: PipettorShakeShelfDevic
(5/9/05 - 4:39:06.81 PM) Info    BenchWorks  Creating: PipettorStandardShelfDe
(5/9/05 - 4:39:06.81 PM) Info    BenchWorks  Creating: PipettorTipChuteShelfDe
(5/9/05 - 4:39:06.81 PM) Info    BenchWorks  Creating: PipettorTipboxShelfDevi
(5/9/05 - 4:39:06.81 PM) Info    BenchWorks  Creating: PipettorVacuumShelfDevi
(5/9/05 - 4:39:06.81 PM) Info    BenchWorks  Creating: PlateLocDevice
(5/9/05 - 4:39:06.92 PM) Info    BenchWorks  Creating: PlatePierceDevice
(5/9/05 - 4:39:06.95 PM) Info    BenchWorks  Creating: QFillDevice
(5/9/05 - 4:39:07.29 PM) Info    BenchWorks  Creating: RempCSPDevice
(5/9/05 - 4:39:07.40 PM) Info    BenchWorks  Creating: RobotDevice
(5/9/05 - 4:39:07.61 PM) Info    BenchWorks  Creating: ST6StackerDevice
(5/9/05 - 4:39:07.66 PM) Info    BenchWorks  Creating: ShuttleRobotDevice
(5/9/05 - 4:39:07.73 PM) Info    BenchWorks  Creating: SpectrafluorDevice
(5/9/05 - 4:39:07.74 PM) Info    BenchWorks  Creating: StackerDevice
(5/9/05 - 4:39:07.80 PM) Info    BenchWorks  Creating: StandardPlatePadDevice
(5/9/05 - 4:39:07.80 PM) Info    BenchWorks  Creating: StorexDevice
(5/9/05 - 4:39:07.85 PM) Info    BenchWorks  Creating: StorexIOPadDevice
(5/9/05 - 4:39:07.85 PM) Info    BenchWorks  Creating: TeleshakeDevice
(5/9/05 - 4:39:07.89 PM) Info    BenchWorks  Creating: TeleshakeHighSpeedDevic
(5/9/05 - 4:39:07.96 PM) Info    BenchWorks  Creating: TrashDevice
(5/9/05 - 4:39:07.98 PM) Info    BenchWorks  Creating: UltramarkDevice
(5/9/05 - 4:39:08.07 PM) Info    BenchWorks  Creating: VCode3KDevice
(5/9/05 - 4:39:08.17 PM) Info    BenchWorks  Creating: VLidDevice
(5/9/05 - 4:39:08.20 PM) Info    BenchWorks  Creating: VLidStationDevice
(5/9/05 - 4:39:08.20 PM) Info    BenchWorks  Creating: VMixDevice
(5/9/05 - 4:39:08.25 PM) Info    BenchWorks  Creating: VPrepDevice
(5/9/05 - 4:39:08.34 PM) Info    BenchWorks  Creating: VSpinAccess2Device
```

During a run, you can type notes that are added to this log file.

Pipettor transfer log file

The pipettor transfer log file is a tab-delimited text file that records VPrep pipetting information on a BenchCel that has a VPrep.

Note: This log is not the same as the log generated by the VPrep.

Part of a pipettor transfer log file, opened in Notepad is shown in the following diagram.

```
)3 - 12:53:56.60 PM)     VPrep 1 (96 channels)   Q1Source 1     MasterA0004    No bar code     No bar code     N
 1      (5/22/03 - 12:54:04.71 PM)     IntermediateA 1 IntA101 IntA001 No bar code     No bar code     1       1

)3 - 12:54:12.78 PM)     VPrep 1 (96 channels)   Q1Source 1     MasterA0004    No bar code     No bar code     N
 1      (5/22/03 - 12:54:20.60 PM)     IntermediateB 1 IntB101 IntB001 No bar code     No bar code     1       1

)3 - 12:54:27.40 PM)     VPrep 1 (96 channels)   Q1Source 1     MasterA0004    No bar code     No bar code     N
 1      (5/22/03 - 12:54:40.76 PM)     Q1Source 1     MasterA0004    No bar code     No bar code     No bar co
 0.0    OK
)3 - 12:54:27.40 PM)     VPrep 1 (96 channels)   Q1Source 1     MasterA0004    No bar code     No bar code     N
 1      (5/22/03 - 12:54:49.03 PM)     IntermediateC 1 IntC101 IntC001 No bar code     No bar code     1       1
```

The pipettor transfer log contains the following information, separated by tabs:

- ❑ Aspiration timestamp
- ❑ Pipettor name
- ❑ Name of the plate aspirated from
- ❑ North bar code (of plate aspirated from)
- ❑ East bar code (of plate aspirated from)
- ❑ South bar code (of plate aspirated from)
- ❑ West bar code (of plate aspirated from)

- ❑ Quadrant of the plate aspirated from (number 1–16)
- ❑ Dispense timestamp
- ❑ Name of the plate dispensed to
- ❑ North bar code (of plate dispensed to)
- ❑ East bar code (of plate dispensed to)
- ❑ South bar code (of plate dispensed to)
- ❑ West bar code (of plate dispensed to)
- ❑ Quadrant of the plate dispensed to (number 1–16)
- ❑ Volume of liquid dispensed in microliters
- ❑ Status of the dispense

  Values are ERROR or OK. These refer to the status of the bar code verification and not the dispense itself.

Note: One log entry is created for every dispense task pair. For example, if 20 µL are aspirated and half is dispensed to one plate and half to another plate, two piplog entries are created. This example is treated as two dispense task pairs.

Bar code log file

The bar code log file contains the date and time at which each bar code is applied and the text of each field. Bar code fields are displayed in human readable form.

Reader output file

The reader output file defines the location of the data file (<*file name*>.dat) for a plate reader.

VersaScan image file root folder

The image root folder sets the folder in which images from a VersaScan are automatically stored.

Related information

For information about...	See...
The next step	“Setting log options” on page 54
Adding a note	“Working with the Log toolbar” on page 77
Bar code labeling	“Setting Apply Label task parameters” on page 102

Setting log options

About this topic

This topic explains how to configure the log options page of the BenchWorks Options dialog box. You may not need to configure these options for every run.

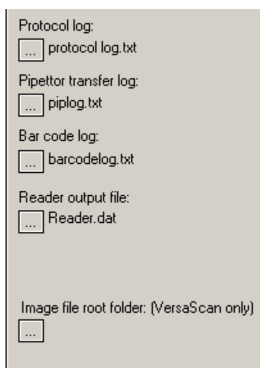
Available options

The following types of log options are available in BenchWorks:

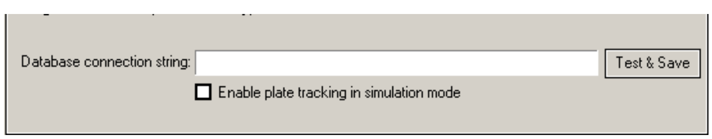
- ❑ The type of log information to show in the Log toolbar of BenchWorks

- ❑ General settings that pertain to log and data file maintenance

- ❑ The folders in which to store log and data files



- ❑ Checking the database connection and enabling plate tracking



How messages displayed in the Log toolbar are controlled

Event and error messages are displayed in the log toolbar. With all message options turned on, a large number of messages are displayed. For simplicity, you can hide types of messages that are not important to you.

The messages that are displayed during a run are controlled by:

- ❑ Selecting screen settings options (in the Tools > Options dialog box) before a run
- ❑ Clicking tabs in the Log toolbar during a run

Note: Screen display settings do not affect the information saved in log files. All error and event information is always saved.

Setting log options procedure

To set log file options:

1. Select **Tools** > **Options**.
2. Click the **Log Options** tab.
3. In the **Screen settings** group box, select one or more of the following options, as needed:

Log Option	Writes to screen and file...
Log “task begin” events on screen	Messages at the time that process tasks are scheduled (not at the time that they are performed). This applies only to the protocol log file.
Log “task complete” events on screen	Messages that confirm when process tasks are completed. This applies only to the protocol log file.
Log robot motions on screen	Robot motion events as they happen. This applies only to the protocol log file.
Log pipettor debug messages	Event messages that are generated by VPrep pipettors. This applies only to the protocol log file.

4. In the **File Settings** group box:

   a. In the **Max number of each log file to maintain** text box, type the maximum number of log files that you want to store.

      !! IMPORTANT !! After this number has been reached, each new log file replaces the oldest existing log file.

b. Select one or more of the following options, as needed:

Log options	Description
Create new logs for every run	A separate log file is created for every run. If the check box is cleared, each run appends data to the same log file and a new log file is created when BenchWorks is started. This affects all log files.
Append timestamps to log file names	The date and time of the run is appended to the name of the log file. This affects all log files.

c. Click the ellipsis button for a log file or folder.

d. Navigate to the folder to which you want to save the log file.

e. Click **Save**.

5. In the **BenchWorks Options** dialog box, click **OK**.

Note: Checking the database connection and enabling plate tracking, requires that you have an inventory management system. Please contact Velocity11 for more information.

Related information

For information about…	See
The workflow this procedure belongs to	"Workflow for preparing for a run" on page 32
The next step	"Understanding the protocol" on page 59
Log and data files	❑ "Setting pre-protocol rules" on page 48 ❑ "Working with the Log toolbar" on page 77

Importing a log file to Excel

About this topic

This topic describes how to import a log file into Microsoft Excel.

Because comma-delimited and tab-delimited text files contain structured data, you can quickly import them into Microsoft Excel, automatically organizing their data into columns.

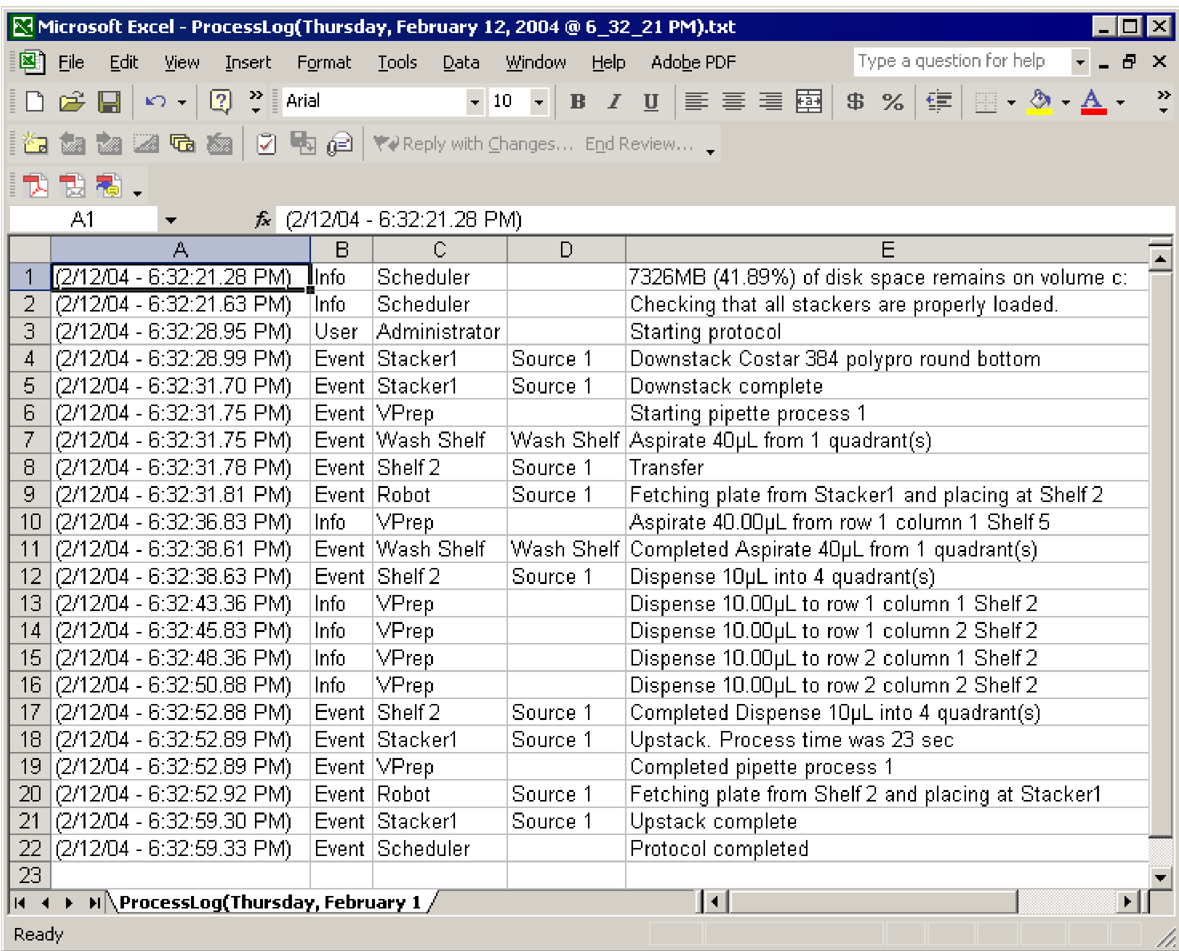
This feature is helpful for making it easier to analyze the data in log files.

Procedure

To import a log file to Excel:

1. Open Microsoft Excel.
2. Drag the file onto the Excel window.

   The data is imported.



	A	B	C	D	E
1	(2/12/04 - 6:32:21.28 PM)	Info	Scheduler		7326MB (41.89%) of disk space remains on volume c:
2	(2/12/04 - 6:32:21.63 PM)	Info	Scheduler		Checking that all stackers are properly loaded.
3	(2/12/04 - 6:32:28.95 PM)	User	Administrator		Starting protocol
4	(2/12/04 - 6:32:28.99 PM)	Event	Stacker1	Source 1	Downstack Costar 384 polypro round bottom
5	(2/12/04 - 6:32:31.70 PM)	Event	Stacker1	Source 1	Downstack complete
6	(2/12/04 - 6:32:31.75 PM)	Event	VPrep		Starting pipette process 1
7	(2/12/04 - 6:32:31.75 PM)	Event	Wash Shelf	Wash Shelf	Aspirate 40µL from 1 quadrant(s)
8	(2/12/04 - 6:32:31.78 PM)	Event	Shelf 2	Source 1	Transfer
9	(2/12/04 - 6:32:31.81 PM)	Event	Robot	Source 1	Fetching plate from Stacker1 and placing at Shelf 2
10	(2/12/04 - 6:32:36.83 PM)	Info	VPrep		Aspirate 40.00µL from row 1 column 1 Shelf 5
11	(2/12/04 - 6:32:38.61 PM)	Event	Wash Shelf	Wash Shelf	Completed Aspirate 40µL from 1 quadrant(s)
12	(2/12/04 - 6:32:38.63 PM)	Event	Shelf 2	Source 1	Dispense 10µL into 4 quadrant(s)
13	(2/12/04 - 6:32:43.36 PM)	Info	VPrep		Dispense 10.00µL to row 1 column 1 Shelf 2
14	(2/12/04 - 6:32:45.83 PM)	Info	VPrep		Dispense 10.00µL to row 1 column 2 Shelf 2
15	(2/12/04 - 6:32:48.36 PM)	Info	VPrep		Dispense 10.00µL to row 2 column 1 Shelf 2
16	(2/12/04 - 6:32:50.88 PM)	Info	VPrep		Dispense 10.00µL to row 2 column 2 Shelf 2
17	(2/12/04 - 6:32:52.88 PM)	Event	Shelf 2	Source 1	Completed Dispense 10µL into 4 quadrant(s)
18	(2/12/04 - 6:32:52.89 PM)	Event	Stacker1	Source 1	Upstack. Process time was 23 sec
19	(2/12/04 - 6:32:52.89 PM)	Event	VPrep		Completed pipette process 1
20	(2/12/04 - 6:32:52.92 PM)	Event	Robot	Source 1	Fetching plate from Shelf 2 and placing at Stacker1
21	(2/12/04 - 6:32:59.30 PM)	Event	Stacker1	Source 1	Upstack complete
22	(2/12/04 - 6:32:59.33 PM)	Event	Scheduler		Protocol completed
23					

Protocol log file information

The information given in a protocol log file is explained in the following table. Refer to the screenshot of the Excel file for the column letters.

Spreadsheet column	Information
A	Date and time that the entry was added
B	Type of information: ❑ Error ❑ Event ❑ Info ❑ Script ❑ User
C	Origin of the information
D	Either the: ❑ Name of the plate ❑ Name of the device, if it refers to a reagent
E	Description of the log entry

Related information

For information about...	See
The workflow for preparing to do a run	“Workflow for preparing for a run” on page 32
The next step	“Understanding the protocol” on page 59
Log and data files	❑ “Setting pre-protocol rules” on page 48 ❑ “Working with the Log toolbar” on page 77
Setting log options	“Setting log options” on page 54

Understanding the protocol

About this topic

This topic gives an overview of the information you should become familiar with when running a protocol.

What you should know

At a minimum, you should be familiar with the following:

- ❑ Which instruments you need to prepare
- ❑ Where you need to position the plates before the run and where they are moved to during the run
- ❑ Whether User Message tasks prompt you to perform certain actions after you start the run or whether you need to perform the actions on your own initiative before you start the run
- ❑ Whether you need to replace fluids and empty waste during the run
- ❑ Whether you need to remove and add plates during the run
- ❑ Which liquids you need to prepare, where they should be placed, and in what kinds of reservoir

Related information

For information about...	See
Where this topic fit into the workflow	"Workflow for preparing for a run" on page 32
The next step	"Preparing plates, instruments, and accessories" on page 61
Printing a protocol	"Printing a protocol" on page 60

Printing a protocol

About this topic

You can print a description of a protocol, which will help you to analyze the sequence of tasks. This topic describes how to print a copy of the protocol.

Printing a protocol

It may help you to refer to a printout of the protocol steps as you analyze the protocol.

To set up the printer:

1. Select **File > Print Setup**.
2. Select the printer you want to print to and configure the print dialog box as required.

To print a protocol:

1. Navigate to **File > Print Preview**.
2. View the preview and, if it is satisfactory, click **Print**.

Note: If you try to print a protocol before a network printer driver is installed on the BenchCel computer, you will receive an error. If this happens, contact your network administrator for help.

An example of a printed process that includes a pipette process is shown here.

```
BenchWorks: (6/5/03 - 9:42:30.48 AM)

    384 ABGene deepwell called Test Plate:
    Pipet process 1
    Downstack from Stacker 2

    ******************************

    Pipet Process 1:
    Aspirate 10.0 µL from Test Plate quadrant 1
    Dispense 10.0 µL to VPrep 1 Shelf 3 quadrant 1 using 384 Disposable Tip 01ul - 05ul
    Mix 10.00 µL 3 times at VPrep 1 Shelf 5 quadrant 1
```

Related information

For information about...	See...
Information you should know about protocols	“Understanding the protocol” on page 59

Preparing plates, instruments, and accessories

About this topic

This topic provides a quick reference for use when preparing instruments and accessories for use on the BenchCel. Do not use this topic as a substitute for the manuals for the individual instruments. This topic is intended for quick reference only.

The procedures in this topic assume that the instruments are already turned on and that you are familiar with the operation of the individual instruments connected to your BenchCel.

Prepare instruments and accessories after you have become familiar with the protocol.

Set up order

You would typically prepare the BenchCel instruments and accessories in the order presented in this topic. Using this order, you first set up instruments and accessories that have no time-sensitive elements and can therefore be set up the night before the run. Leave the more complex preparations, which may use expensive and unstable reagents, until last.

Note: If the protocol includes User Message tasks to prompt you to perform some setup steps, such as placing counterweight plates, don't include these steps in the setup.

PlateLoc set up process

The general process of preparing a PlateLoc is described here. For more detailed procedures, see the *PlateLoc User Guide*.

1. Make sure that there is enough seal stock on the roll for the run.
2. Make sure that the right support insert is installed for the plates you are using.
3. Select the profile that you want to use from the PlateLoc Diagnostics software.

VCode set up process

The general process for preparing a VCode for a run is described here. For more detailed procedures, see the *VCode User Guide*.

1. Make sure that there is enough label stock in the label source roll.
2. Make sure that there is enough media stock in the media source roll.
3. If the radius width of the label backing on the take-up roll is more than 1 inch (2.5 cm) wide, remove the label backing from the take-up roll, cut it, and reattach the end.

   Too much label backing on the take-up roll can interfere with the operation of the VCode.
4. Use VCode Diagnostics software to perform a plate labelling test label.

   For more information, see the *VCode User Guide*.

VPrep set up process

The general process for preparing a VPrep for a run is described here. For more detailed procedures, see the *VPrep User Guide*.

The BenchCel robot has access to VPrep shelves 2, 4, 6 and 8, meaning that only these shelves can hold sample plates. Shelves 1, 3, 5, and 7 are reserved for reservoirs and tipboxes.

1. If you are changing the reagent type in a particular reservoir:
   a. Remove and wash any reservoirs.
   b. Detach and wash reservoir tubing.
   c. Replace the tubing and reinstall the reservoir.
   d. Remove and wash any MicroWash trays.
   e. Detach and wash MicroWash tray tubing.
   f. Replace the tubing and reinstall the tray.
2. Make sure that the reservoir and tray tubing allows the shelves to move to their fully engaged position.
3. Install tubing into the correct pumps.
4. Fill the fluid reservoir bottle, replace the cap, and attach the fluid line to the cap connector.

   Make sure you attach the fluid line that pumps towards the VPrep.
5. Empty the waste container, replace the cap, and attach the fluid line to the cap connector.

   Make sure that you attach the fluid line that pumps away from the VPrep.
6. Use the VPrep diagnostics software to prime the fluid lines between the pump and VPrep reservoirs by filling the lines with the appropriate fluid.
7. Calibrate the Weigh Shelf if:
   - You are changing the liquid type used in the reservoir on the Weigh Shelf.
   - The pump module connected to the Weigh Shelf has been turned off.
   - If you haven’t calibrated it for a while.
8. If you are using disposable tips, you may need to attach them before starting the run. If so:
   a. Place a tipbox on the tipbox shelf.
   b. Use the VPrep diagnostics software to install the tips.
   c. Remove the tipbox.

   Attaching tips is an important step. Please refer to the *VPrep User Guide* for more information.

Related information

For information about...	See...
The workflow this procedure belongs to	“Workflow for preparing for a run” on page 32
Next step (if you are using a VSpin)	“Setting up a VSpin counterweight plate” on page 63
Next step (if you are not using VSpin)	“Installing a rack” on page 64
User message tasks	“Specifying task order across processes” on page 132

Setting up a VSpin counterweight plate

About this topic

The general process of preparing a Vspin with Access2 is described here. For more detailed procedures, see the *Access2 User Guide*.

About counterweight plates

If you are using a VSpin in a protocol, you need a counterweight plate to balance a sample plate when centrifuging in a VSpin. This means that only one sample plate can be centrifuged at one time in a protocol. (Although if you are using the VSpin as a stand-alone instrument, then two sample plates can be centrifuged together.)

Using a counterweight plate and centrifuging one sample plate at a time does not typically increase the length of a protocol run, compared to spinning two sample plates at one time. This is because the centrifuge task is not typically the time-limiting step of a protocol.

Counterweight plate defined

A counterweight plate is the same as a sample plate except that it is empty, or contains water to provide a more accurate balance. If a counterweight is not used, a load imbalance error will result. The load imbalance tolerance is 5 grams at 3000 rpm.

About loading counterweight plates

A counterweight plate is required for each plate type that is used in a protocol. For example, if three different types of plates are centrifuged during a protocol, then you need three different counterweight plates.

You must load the counterweight by hand, using VSpin Diagnostics software to open and close the centrifuge door and move the robot.

Manually loading counterweight plates

To load a counterweight by hand:

1. Open **VSpin Diagnostics**.

   The **VSpin Control** dialog box opens.

2. Follow the directions in the *VSpin User Guide* to open the VSpin door and load the counterweight.

   !! DAMAGE HAZARD !! To avoid a robot crash, always place the counterweight into bucket 2 and leave the door open with bucket 1 at the front. The sample plate will always be moved to bucket 1 by the robot.

Related information

For information about...	See...
The workflow this procedure belongs to	"Workflow for preparing for a run" on page 32
The next step	"Installing a rack" on page 64
Setting Access2 task parameters	"Setting VSpin with Access2 task parameters" on page 130
Using the VSpin with Access2	*Access2 User Guide*
Using Access2 diagnostics	*Access2 User Guide*

Installing a rack

About this topic

This topic explains how to install a rack.

Before you can perform a run, you load the BenchCel with racks containing plates specific to your application.

Procedure

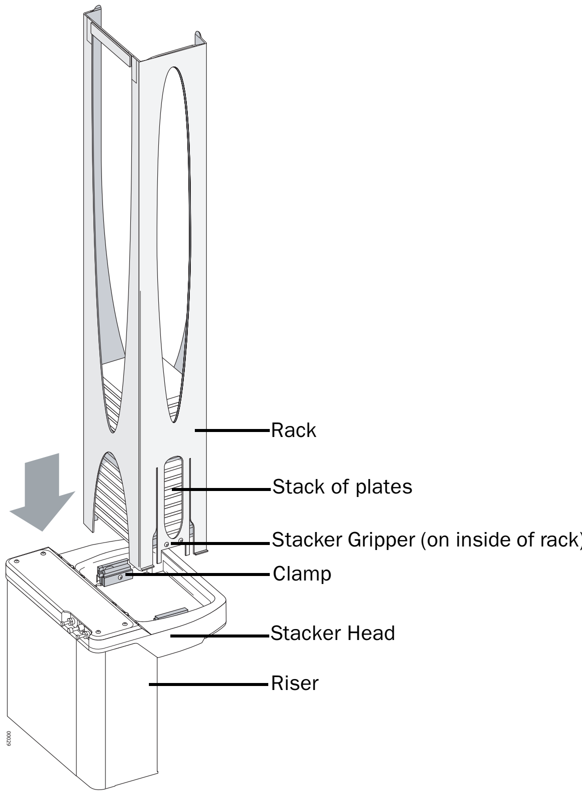
To install a rack:

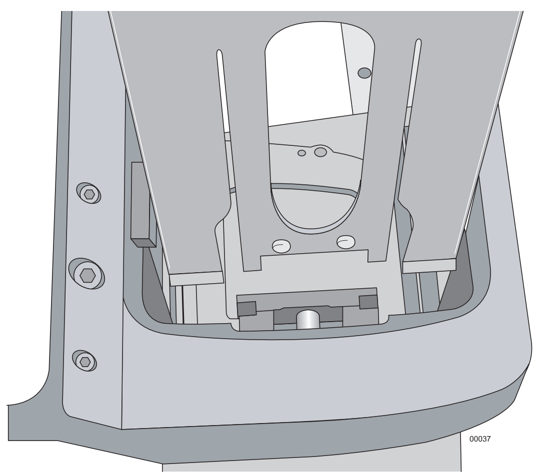
1. Verify that the clamps are extended in the stacker that you want to install the rack onto. You can do this by visual inspection of the stacker.

   If the rack does not easily fit onto the stacker head, then you may need to extend the clamps.

2. Place a rack on the base unit, lowering it so that the rectangular tabs drop into the slots in the gripper unit.

   The rack can be placed with the open corner at the front or back.


Rack
Stack of plates
Stacker Gripper (on inside of rack)
Clamp
Stacker Head
Riser


00037

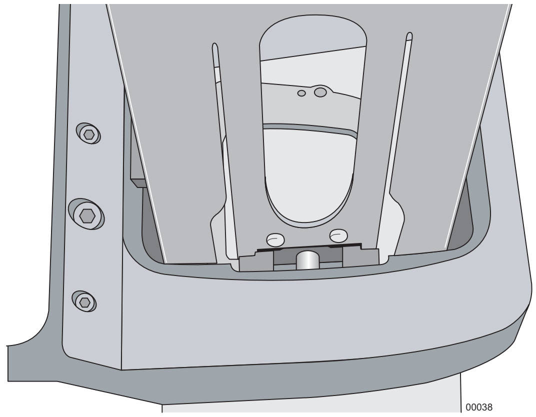
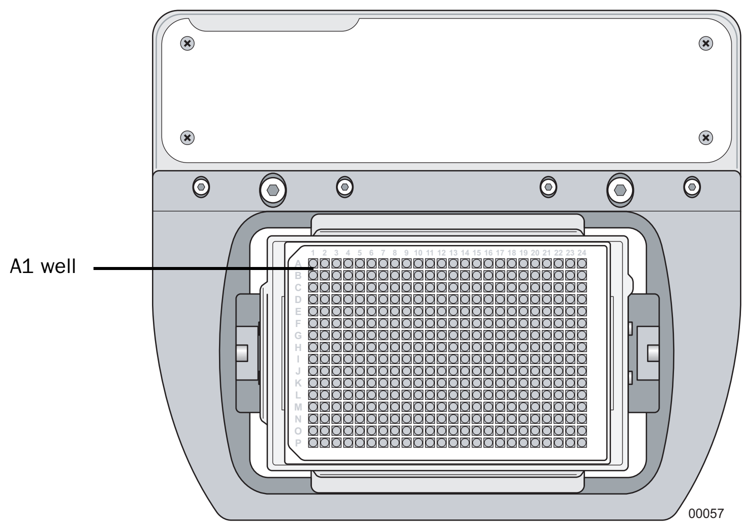


Plate orientation with stacker

Always place plates so that the A1 well is in the far left corner, as viewed by someone standing in front of the BenchCel.

Note: With some VPrep integrations, this orientation may be shifted such that A1 will be located on the lower right corner.



Related information

For information about...	See...
The workflow this procedure belongs to	“Workflow for preparing for a run” on page 32
Extending or retracting clamps	“Extending and retracting stacker clamps” on page 267

4

Performing a run

A run is a single protocol that is performed one or more times in a series. This chapter describes how to run an existing protocol on the BenchCel. All of the procedures in this chapter can be performed by someone with operator privileges.

About performing a run

About this topic

This topic tells you where to find the information about starting a run and performing tasks associated with running a protocol.

Ways to set up a run

There are two ways to start a run.

Method	See...
Set up a single protocol run and manually start it by clicking the Start button in the BenchWorks	"Starting a run from BenchWorks" on page 70
Set up a single protocol in advance and manually start it from a command line	"Starting a run from a command line" on page 73

Tasks associated with a run

Task	See...
Perform pre-run checks	"Performing pre-run checks" on page 69
Start a run	❑ "Starting a run from BenchWorks" on page 70 ❑ "Starting a run from a command line" on page 73
Monitor a run	❑ "Monitoring a run" on page 76 ❑ "Working with the Log toolbar" on page 77
Pause a run	"Pausing or stopping a run" on page 74
Clean up	"Cleaning up after a run" on page 80

Performing pre-run checks

About this topic

This topic describes how to perform a preliminary check of the equipment.

Procedure

To check that the BenchCel is ready for a run:

1. Make sure that there are no stray plates in robot-accessible places. These include:
   - Platepads
   - VCode plate stages
   - VPrep plate stages
   - PlateLoc plate stages
2. Make sure that the instrument air pressures lie within the ranges in the table below.

   To check the BenchCel air pressure, you need to read the values from each stacker in the **Stacker Sensors** group box of the **BenchCel Diagnostics Controls** page.

   You only need to do this occasionally, depending on the application. To check the air pressure for other modules, refer to the relevant manuals.

Unit	Air Pressure (psi)	Air Pressure (MPa)
BenchCel	50–90	0.34–0.62
VPrep	90–95	0.62–0.65
PlateLoc	85–90	0.56–0.62
VCode	70–75	0.48–0.52
VSpin with Access2	80–90	0.55–0.59

Related information

For information about...	See...
Checking stacker sensors	“Checking the stacker sensors” on page 314
Opening BenchCel Diagnostics	“Opening BenchCel Diagnostics” on page 253

Starting a run from BenchWorks

About this topic

This topic describes how to start a run using the Start button in BenchWorks. Start a run after performing pre-run checks and opening a protocol.

Procedure

To start a run:

1. Make sure that all instruments used in the protocol are in their home positions.

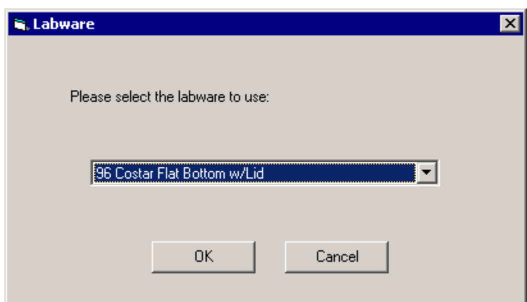
   Refer to the instrument user guides for more information about homing.

2. In BenchWorks, click **Start**.

   If this is the first run with this device file, BenchWorks confirms communication with all devices and instructs the devices to home.

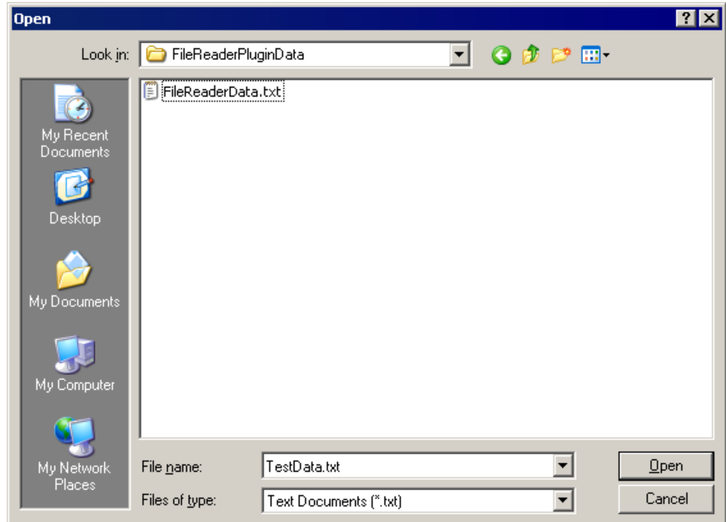
   If there are no protocol compilation errors, the **Number of Cycles** dialog box opens.

3. If you selected **from plug-in** as the plate type, the **Labware** dialog box opens asking you to select the labware type: select the appropriate labware type from the list.



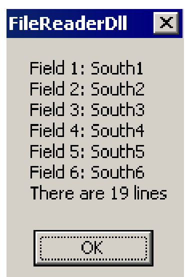
4. If you are running a VCode and are using the **FileReader** plug-in, the **Open** dialog box opens asking you to select the text file that contains the bar code label data.

   a. Select the plug-in text file that contains the data from the **Open** dialog box. and click **Open**.



b. In the new dialog box that opens, inspect the list of names of the columns of the text file and the total number of rows in the file.

In the following screenshot, the first six columns of the plug-in file are repeated for each side of the plate.

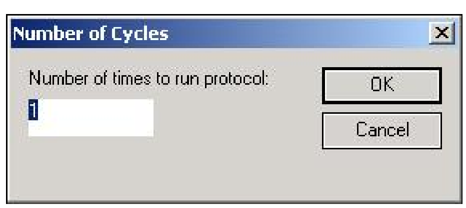


c. Click **OK**.

If the file is not the one you intend to use, you can cancel the run at the next step if needed.

5. In the **Number of Cycles** dialog box, type in the number of times that you want to execute the protocol during the run.

If you want to run an indefinite number of plates see “Running the protocol until you are done” on page 72 below.



6. Click **OK**.

The run starts.

7. If **User Message** tasks are included in the protocol, you are prompted to respond to them as they execute.

   If there are no **User Message** tasks to remind you to empty liquid waste containers and refill liquid source containers, set your own reminders using lab timers.

Running the protocol until you are done

For convenience, you can force the protocol to continue running by entering a sufficiently large value in the Number of times to run protocol field. When the downstack becomes empty, or the upstack becomes full, you will be prompted with the option of adding or removing plates. This feature saves you the time of having to restart the protocol when you are processing a large number of plates or when you are running plates intermittently.

What happens after a run starts

After you start the run, the following events occur:

- ❑ The Start button becomes unavailable and the Pause button becomes available.
- ❑ Log messages on the Log toolbar indicate the start of the run. Also, log files record events as they are performed on all plates in the run, according to the selected log file options.
- ❑ The first instructions of the protocol are executed.

If User Message tasks are included in the protocol, you are prompted to respond to them.

Related information

For information about...	See...
Pausing a run	“Pausing or stopping a run” on page 74
Monitoring the log	“Working with the Log toolbar” on page 77

Starting a run from a command line

About this topic

This topic describes how to start BenchWorks and initiate a run directly from a command line.

Procedure

To start BenchWorks from a command line:

1. In Windows, select **Start** > **Run.**
2. In the **Run** text box, type `cmd` and click **OK**.

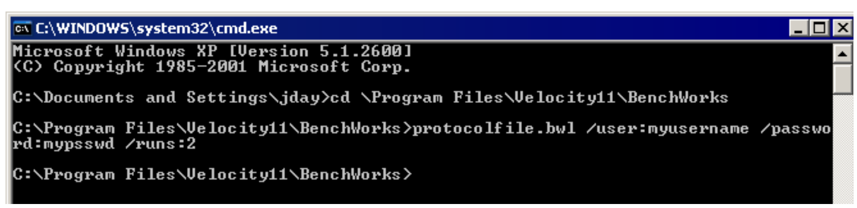
   The command prompt opens.
3. At the command prompt, change the current directory to BenchWorks workspace that contains the protocol file by typing:

   `cd \Program Files\Velocity11\BenchWorks`
4. Press ENTER.
5. Type the name of the BenchWorks protocol file followed by values for the switches that specify:
   - The protocol to run
   - Your user name
   - Your password
   - The number of cycles for which to run the protocol

   An example is:

   myprotocol.bwl /user:*your user name* /password:*your password* /runs:*number of run cycles*
6. Press **ENTER** to start the run.

   The following screenshot shows a generic examples of the completed command prompt.



Related information

For information about...	See...
Starting a run using the start button	“Starting a run from BenchWorks” on page 70
Pausing a run	“Pausing or stopping a run” on page 74
Monitoring a run	“Monitoring a run” on page 76

Pausing or stopping a run

About this topic

This topic describes when and how to stop or pause a protocol that is running.

Use this procedure to:

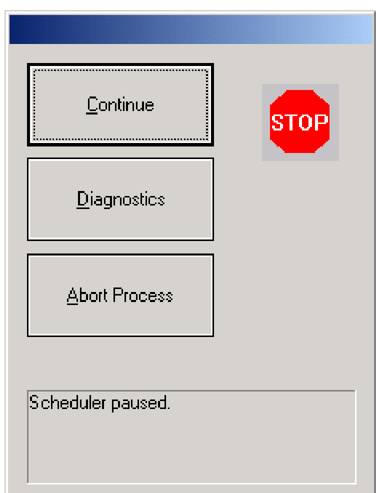
- ❑ Pause and continue a run, for example, when you want to:
  - ◆ Add or remove labware
  - ◆ Clean up a spill
  - ◆ Add buffer to a reservoir
  - ◆ Diagnose a problem that you notice
  - ◆ Perform an operation that is not part of the protocol
- ❑ Abort a run in a non-emergency situation

Procedure

To pause or stop a run using BenchWorks:

1. In BenchWorks, click **Pause**.

   The **Stop** dialog box opens and the currently scheduled task continues to completion. This may take a minute or more. After that, no more tasks are performed.



2. You now have three choices:

If you want to...	Then...
Continue with the run	Click **Continue**.
Troubleshoot a problem or perform a manual operation	Click **Diagnostics**, and select the module that caused the error. This opens the diagnostics software for that module, allowing you to troubleshoot the problem. For more information, see "Using BenchCel Diagnostics" on page 251.
Abort the protocol	Click **Abort Process**.

!! IMPORTANT !! Before continuing with a run, make sure that the system is in a valid state for the protocol.

Make sure that you have not made changes that will cause an error, such as moving a plate to a position that should not have a plate or causing samples to be switched around.

Related information

For information about...	See...
Starting a run	❑ "Starting a run from BenchWorks" on page 70 ❑ "Starting a run from a command line" on page 73
Monitoring a run	"Monitoring a run" on page 76
Using the Log toolbar	"Working with the Log toolbar" on page 77

Monitoring a run

About this topic

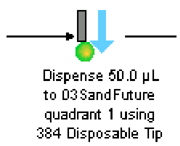
This topic describes what operations you need to monitor and how to view the progress of a run.

What to monitor

After starting a run, monitor the operation of the BenchCel. Exactly what you do to monitor a run depends on the protocol that you are using. For example, you might need to:

- ❑ Compare the motions of the robot to the protocol tasks being completed.

  You can identify the task that is currently being performed from the position of the green ball in the process panes. Because BenchWorks can schedule more than one task at a time, there may be more than one green ball displayed.

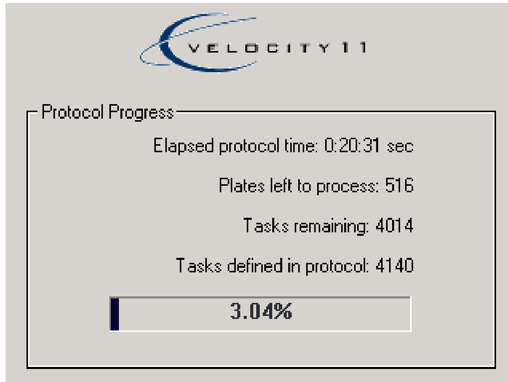
- ❑ Add and remove labware.
- ❑ Empty liquid waste containers.
- ❑ Fill liquid reservoirs.
- ❑ Replace an empty roll of PlateLoc seal or bar code labels.

!! IMPORTANT !! No errors are reported when a liquid waste container becomes full or a liquid reservoir becomes empty. (Exceptions to this are reservoirs on a VPrep Weigh Shelf.)

To guard against the problem of a full waste container or empty reservoir container, the protocol writer can incorporate User Message tasks into the protocol to remind the operator at the appropriate steps in the protocol. Alternatively, operators can set timer alarms to remind them to fill reservoirs and empty the waste container at the appropriate time.

Monitoring overall progress

You can monitor overall progress of the run on the Progress page of BenchWorks.



Closing unneeded toolbars

You can close unneeded toolbars to create more room on the screen for you to monitor a run by clicking the close box.

Related information

For information about…	See…
Pausing a run	“Pausing or stopping a run” on page 74
Starting a run	❑ “Starting a run from BenchWorks” on page 70 ❑ “Starting a run from a command line” on page 73
What to do when you get an error	“Maintenance and troubleshooting” on page 299

Working with the Log toolbar

About this topic

This topic gives an overview of what the Log toolbar does and how to use its features.

Event and error messages are displayed in the BenchWorks Log toolbar of the BenchWorks window.

With all message options turned on, a large number of messages are displayed during a run. For ease of use, you can hide types of messages that are not important to you. The messages displayed during a run are controlled by:

- ❑ Selecting Screen Setting options in the Log Options dialog box before a run
- ❑ Clicking tabs at the bottom of the display during a run

Viewing the Log toolbar

To view the Log toolbar:

1. If the toolbar is not showing, from the **View** menu, select **Toolbars** > **Log**.

   The toolbar opens at the bottom of the screen.

2. Refer to the table below to view different kinds of log data.

Log toolbar options

The following options are available for displaying screen messages.

BenchWorks log tab	Displays...
All	All event and error messages. This is the data that is recorded in the log.txt file.
Process	Event messages, including fluid transfer messages.
Pipettor Fluid Transfers	Event messages reported by VPrep. This is the same data that is recorded in the pipettor transfer log file, but is presented in sentence rather than tab-delimited format. Event messages can be disabled by deselecting the "Record in transfer log" option in the aspirate and dispense pipette task parameters.
Errors	Error messages that appear as alert boxes on the screen. This data is not saved in a separate log, but is included in the log.txt file.
Notes	Notes that you add.

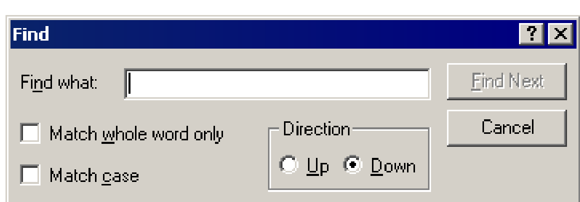
Searching the Log toolbar

You can search for specific text in the Log toolbar.

To perform a search in the Log toolbar:

1. Select the appropriate tab in the Log toolbar.
2. Click in the toolbar pane.
3. Click CTRL + F3 or CTRL + F.

   The **Find** dialog box opens.



Adding a note

You can type notes into the message display pane during a run. Any notes that you type are also incorporated into the log.txt file.

To add a note to the Log toolbar and log.txt file:

1. At the bottom of the message display pane, click the **Notes** tab.
2. Click in the display pane wherever you want to add the note.
3. Type the note.
4. Press **ENTER** on the keyboard

   A timestamp is appended to the note.

Message color coding

The BenchWorks log color-coding scheme is listed here.

Color	Meaning
Black	Standard events with a date stamp or user-added notes
Blue	Liquid transfer events
Red	Warnings
Lime green	General information

Related information

For information about...	See...
Setting screen message options before a run	“About log and data files” on page 51
Importing a log file	“Importing a log file to Excel” on page 57

Cleaning up after a run

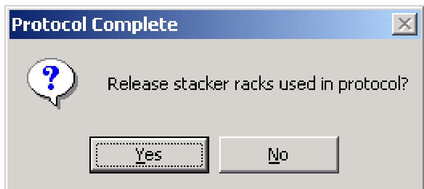
About this topic

This topic describes the tasks you should perform when you have finished a protocol run.

Procedure

To clean up after a run:

1. Click **Yes** in the dialog box that prompts you to release all stacker racks.



2. Remove used sample plates from the racks.
3. Remove manually placed plates from platepads, VPrep module shelves, and VSpin.
4. Wash items such as reservoirs, tubes, and manifolds that were contacted by liquid.
5. If necessary, use the VPrep diagnostics software to remove pipette tips.
6. Check run logs for errors.
7. If you have administrator or technician privileges and you have modified the protocol, including selected options, you can save the protocol.
8. Log out of BenchWorks.
9. Optionally shut down the BenchCel.

Related information

For information about...	See...
Run logs	"About log and data files" on page 51
Protocols	"Workflow for creating a protocol" on page 85
Logging out	"Logging out of BenchWorks" on page 81
Using the Log toolbar	"Working with the Log toolbar" on page 77
Shutting down	"Shutting down the BenchCel" on page 81

Logging out of BenchWorks

About this topic

This topic describes how to log out of BenchWorks.

Logging out of BenchWorks ensures that unauthorized users do not use your account to control the BenchCel or its devices. For example, an administrator should log out after making changes in the plate editor.

Procedure

To log out:

1. Click **Log Out**.

Related information

For information about...	See...
Managing users	"Adding and deleting a user account" on page 205
User privileges	"About user accounts and privileges" on page 204

Shutting down the BenchCel

About this topic

This topic describes when and how to shut down the BenchCel.

When to shut down

Shut down the BenchCel if you intend to:

- ❑ Move it
- ❑ Leave it unused overnight

Procedure

To shut down the BenchCel:

1. Make sure that the post-run clean up procedure was followed after the last run.
2. Select **File** > **Exit** to shut down the BenchWorks application.
3. If you use instruments that require a vacuum pump, turn off power at the pump if the pump module has an on/off switch.
4. Turn off the air supply to the BenchCel and other instruments.
5. On the connection panel, turn the main power switch to the off position.
6. Turn the power off on any instrument integrated with the BenchCel.

Related information

For information about...	See...
Clean up after a run	“Cleaning up after a run” on page 80
Power connections	“Connection panel description” on page 22

Creating a protocol: basics

5

This chapter is for people with technician and administrator privileges. It describes the process of creating a protocol and explains the parameters used to define each protocol task. Before reading this chapter you should be familiar with the concepts presented in "Performing a run" on page 67.

This chapter is not a tutorial on writing protocols—it provides the basic reference information you need to create protocols.

About protocol files

About this topic

This topic provides an overview of the protocol file format.

File format

Protocols are created in the drag-and-drop protocol editor. When they are saved, the information is written to a file in XML format. In XML, the elements indicate the protocol's properties, and text within the markup tags gives the properties' values.

Most users create and edit protocols in the protocol editor. Advanced users have the option of creating and editing protocols directly in XML. The XML files can also be useful for troubleshooting because you can, for example, see which device file is associated with the protocol.

XML example

A protocol file viewed in an XML editor that color-codes text is shown below.


```xml
<?xml version="1.0" encoding="ASCII"?>
<Velocity11 file="Protocol_Data" md5sum="751d03b4f669f58547085c6da8dd1706" version="1.0">
    <Device_File File_Name="C:\Documents and Settings\me\Desktop\ma.dev"/>
    <Description Description=""/>
    <Process Multiplier="1" Processes_To_Spawn="1" Tip_Box_Process="0">
        <Bar_Code_File_Info>
            <south Bar_Code_File=""/>
            <west Bar_Code_File=""/>
            <north Bar_Code_File=""/>
            <east Bar_Code_File=""/>
        </Bar_Code_File_Info>
        <Plate Has_Lid="0" Labware="Costar 96-well plate" Name="Plate 1" Sealed="0">
            <Bar_Code_Info>
                <south Bar_Code_File="No bar code" Has_Bar_Code="1"/>
                <west Bar_Code_File="No bar code" Has_Bar_Code="0"/>
                <north Bar_Code_File="No bar code" Has_Bar_Code="0"/>
                <east Bar_Code_File="No bar code" Has_Bar_Code="0"/>
            </Bar_Code_Info>
        </Plate>
        <Tasks>
            <Task0 Description="Downstack from stacker2" Device="" Enable_Timed_Release="0" Icon="0"
Object_Type="DownstackPlate" Release_Time_MS="10000">
                <Stackers_To_Use>
                    <Stacker Name="stacker2"/>
                </Stackers_To_Use>
            </Task0>
            <Task1 Description="Bar code plate" Device="vcode1" Icon="1" Object_Type="BarcodePlate" Type="BarcodePlate">
                <south Bar_Code_Names="0" Base="10" File_Name="" Format="1" Increment_Characters="3" Lookup_Key="2"
Max_Verification_Attempts="0" Number_Of_Fields="2" Plugin_Name="" Preprinted_Side="0" Print_Label="1"
Starting_Increment="1" Verify_Barcodes="0">
                    <Fields>
```


You can also open a protocol file in any browser that contains an XML parser, such as Internet Explorer version 4 or later.

XML schema

To create and edit protocols directly in XML, you need to have the XML schema that defines the logical rules of a BenchWorks protocol. If you want the schema, please contact the Velocity11 Service Center.

Related information

For information about...	See...
Workflow for creating a protocol	“Workflow for creating a protocol” on page 85
Using JavaScript in protocols	“Using JavaScript in BenchWorks” on page 171

Workflow for creating a protocol

About this topic

This topic gives the steps used to make a protocol and a cross-reference to the topic that describes each step.

Who creates protocols?

People who have technician and administrator level user accounts have the necessary privileges to create protocols.

Workflow

Step	Topic	See...
1.	Preparing back-end bar code label data to use with a plug-in (optional).	❑ “About the FileReader plug-in” on page 164 ❑ “Using the FileReader plug-in in a protocol” on page 168
2.	Writing JavaScript scripts to dynamically set task parameters (optional).	❑ “Using JavaScript in BenchWorks” on page 171 ❑ “The JavaScript task object and properties” on page 178
3.	Creating a pre-protocol process (optional).	“Setting up a pre-protocol or post-protocol process” on page 87

Step	Topic	See...
4.	Creating a protocol process which includes:	
	Setting up a plate instance.	“Setting up a plate instance” on page 88
	Deciding the number of simultaneous plates to set.	“Setting the number of simultaneous plates” on page 91
	Adding tasks.	“Adding and deleting tasks and pipette tasks” on page 92
	Setting task parameters.	❑ “About setting task parameters” on page 95 ❑ “Setting task parameters” on page 99
	Optionally, entering JavaScript scripts to dynamically set task parameters.	❑ “Using JavaScript in BenchWorks” on page 171 ❑ “The JavaScript task object and properties” on page 178
5.	Creating a pipette process (optional), which includes:	
	Linking a Pipette Process task icon to a pipette processes.	“About setting pipette task parameters” on page 96
	Adding pipette tasks.	“Adding and configuring a Pipette Process task” on page 140
	Setting pipette process task parameters.	“Setting pipette task parameters” on page 135
6.	Creating additional processes and pipette processes.	This table, step 4 and step 5
7.	Creating a post-protocol process (optional).	“Setting up a pre-protocol or post-protocol process” on page 87
8.	Adding user message tasks where needed.	“Setting User Message task parameters” on page 125
9.	Compiling and saving the protocol.	“Compiling and saving protocols” on page 93
10.	Testing the protocol with the simulator.	“Simulating a run” on page 97
11.	Running the protocol.	“Performing a run” on page 67

Related information

For information about...	See...
Protocols, including definitions of terms	“About tasks, processes, and protocols” on page 37
Privileges	“About user accounts and privileges” on page 204

Setting up a pre-protocol or post-protocol process

About this topic

This topic describes how to set up a pre-protocol and post-protocol process.

Pre-protocols and post-protocols are processes that are carried out before and after the protocol is executed.

Use the pre-protocol editor when you want a task performed, before the protocol runs. For example, if you are dispensing reagents, you may want to prime a pump with fluid.

Use the post-protocol editor when you want to perform a task, after the protocol runs. For example, you may want to flush lines with a buffer or cleaning agent.

Procedure

To set up a pre-protocol or post-protocol process:

1. Click the **Pre-Protocol** or **Post-Protocol Editor** tab.

   If the tab is not available:

   a. Select **Tools** > **Options**.

   b. Select the **Show Pre/Post Protocol Editor** check box.

   c. Click **OK**.

2. Click **Add**.

   A pre-protocol process icon appears in the **Pre-Protocol** or **Post-Protocol Editor** window.

3. If the **Task Parameters** toolbar is not showing, select **View** > **Toolbars** > **Protocol Parameters**.
4. Type in a name for the process in the **Process name** field.
5. Create the pre-protocol process as you would a protocol process, by adding tasks and then setting the task parameters.
6. When you are finished, compile and check your pre-protocol.

Related information

For information about...	See...
Creating a protocol process	❑ "Setting up a plate instance" on page 88 ❑ "Setting the number of simultaneous plates" on page 91 ❑ "Adding and deleting tasks and pipette tasks" on page 92
Compiling a protocol	"Compiling and saving protocols" on page 93
Running a protocol in simulation mode	"Simulating a run" on page 97
The workflow that this procedure belongs to	"Workflow for creating a protocol" on page 85

Setting up a plate instance

About this topic

This topic describes how to set up a plate instance when creating a protocol. See Related information at the end of this topic for where to find a definition of plate instance.

Procedure

To set up a plate instance:

1. Click the **Protocol Editor** tab.

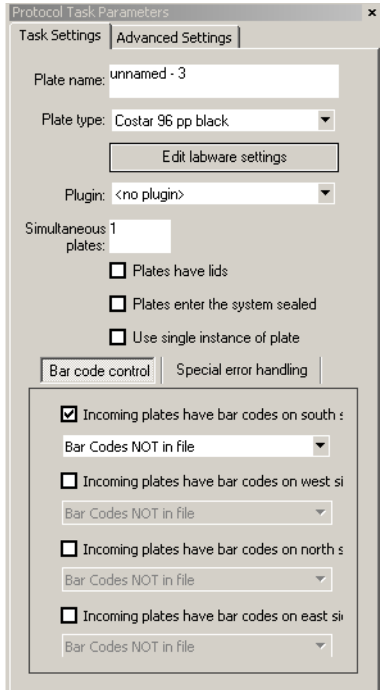
   This page is referred to as the protocol editor.

2. Click **Add**.

   A plate icon appears in the protocol editor window.

3. If the **Task Parameters** toolbar is not showing, select **View** > **Toolbars** > **Protocol Task Parameters**.

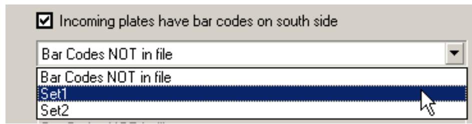


4. In the **Task Settings** page of the **Protocol Task Parameters** toolbar:
   a. Type a name for the plate in the **Plate name** text box.
   b. In the **Plate type** list box, select the type of plate you want to use.
   c. If you want to add or modify a plate definition, click **Edit labware settings** to open the labware editor.
   d. If you are using a plug-in, select it from the **Plug-in** list box.

      If the plug-in is not available for selection, it may be because the *file_name*.dll file is not in the plug-ins folder in the same folder as the BenchWorks executable.
   e. In the **Simultaneous plates** text box, type the maximum number of plates of this type that the system is allowed to operate on at any given time.
   f. If the plates entering the system have lids, select the **Plates have lids** check box.

      Note: This option is only available if the plate you selected is capable of using a lid, as defined in the labware editor.
   g. If the sample plates in the protocol have seals when they are loaded onto the BenchCel, select the **Plates enter the system sealed** check box.
   h. If you have only one instance of a plate type and want it to be used repeatedly, select the **Use single instance of plate** check box. For example, you may have one source plate from which you want to repeatedly aspirate a given volume and dispense it into a different receiving plates.

5. In the **Bar code control** sub-page:
   a. If the incoming plates have bar codes, select the appropriate **Incoming plates have bar codes...** check box.
   b. If you want to check the bar codes on the incoming plates against a series of bar codes in a bar code input file, select the name of the bar code series you want to use.

   For more information about bar codes input from file, see Related information at the end of this topic.
6. In the **Special error handling** sub-page:
   a. Select one or more devices that you want to use as quarantine stations and click **Add**.

      The device names are moved to the bottom quarantine list.

      A quarantine station is a place that plates will be placed if the plate has a bar code mismatch error.
   b. Check, or leave clear, the **Quarantine plate after process completed** check box, based on the following descriptions.

      If the check box is left clear, a plate that gives a bar code misread error is immediately moved to a quarantine station and a new plate takes the place of the quarantined plate in the processing sequence.

      If the check box is selected, plates that give a bar code misread are processed as normal, but are then moved to quarantine instead of moving to the final destination specified in the protocol.

Related information

For information about...	See...
Using plug-ins	"About the FileReader plug-in" on page 164
Definition of a plate instance	"About tasks, processes, and protocols" on page 37
Setting the number of simultaneous plates	"Setting the number of simultaneous plates" on page 91
Using bar codes	"About bar code reading and tracking" on page 197
The workflow that this procedure belongs to	"Workflow for creating a protocol" on page 85

Setting the number of simultaneous plates

About this topic

This topic describes the concept of simultaneous plates and how to set the simultaneous plate number.

The number of simultaneous plates is the maximum number of plates belonging to a single process that are in the system at one time.

Plates that are in the system could be in the robot gripper, on platepads, VPrep module shelves, plate hotels, and so on. Simultaneous plates do not include counterweight plates.

Setting the number of simultaneous plates is part of the process of setting up a plate instance.

Selecting the number

In general, the number of simultaneous plates to select for a plate instance depends on how many:

- Tasks there are in the protocol.

  In general, one simultaneous plate can be used for every task in the protocol. This is because, in general, each task uses one plate position. As an example, a protocol with three tasks can have three plates on the table at once. There are exceptions to this, though, such as cases where the same plate position is used for more than one of the tasks and when a VPrep is used. Several plates can be positioned on a VPrep at the same time.

- Positions there are in the system that will incubate plates.
- Plates there are in a restack task.

  If the protocol includes a restack task, the number of simultaneous plates must be equal to, or greater than, the number of plates in the restack task.

If the number of simultaneous plates is set too high, the protocol may be slowed down because the robot will move around to avoid a deadlock. In cases where too many plates are in the system and there is no way to move plates around further, this can result in a deadlock at which point the protocol stops.

If the number of simultaneous plates is set too low, the time for the protocol run could be extended. Optimizing the number of simultaneous plates is therefore critical for maximizing efficiency.

A typical BenchCel protocol has 5–30 simultaneous plates, but it could be many more than that. A safe number to start with when testing a new protocol is 1–2. Run the protocol in simulation to optimize the number of simultaneous plates.

Related information

For information about...	See...
Recovering from deadlock	"Resolving common problems and error messages" on page 304
Definition of a plate instance	"About tasks, processes, and protocols" on page 37
Setting up a plate instance	"Setting up a plate instance" on page 88
The workflow that this procedure belongs to	"Workflow for creating a protocol" on page 85

Adding and deleting tasks and pipette tasks

About this topic

This topic describes how to add and delete tasks and pipette tasks.

After you have set up a plate instance for a process, you can start adding tasks and pipette tasks.

Adding a task

You must always add a task to a process before you can define its task parameters.

To add a task:

1. If the **Protocol Tasks** toolbar is not showing, select **View** > **Toolbars** > **Protocol Tasks**.
2. There are various ways to add tasks to the protocol:
   - Click on a task icon and drag it from the task list to the protocol editor window until a vertical, dashed line appears.
   - Double-click the icon.
   - Copy (or cut) and paste task icons in the protocol.

Deleting a task

To delete a task:

1. In a protocol editor, select a task that is in a protocol process.
2. Press the DELETE key on the keyboard.
3. Click **Yes** in the **Delete Task** dialog box to delete the task.

Moving tasks

When editing a protocol, drag-and-drop or use cut-and-paste commands to move tasks and groups of tasks.

To move tasks in a protocol:

1. In a protocol editor, select a task or a group of tasks in a protocol process.
2. Do one of the following:
   - ◆ Drag and drop the tasks to a new location in the protocol.
   - ◆ Use the **Cut** or **Copy** and **Paste** commands on the **Edit** menu to move the tasks.

Related information

For information about...	See...
The workflow that this procedure belongs to	"Workflow for creating a protocol" on page 85
Setting up a plate instance	"Setting up a plate instance" on page 88
Setting the number of simultaneous plates	"Setting the number of simultaneous plates" on page 91

Compiling and saving protocols

About this topic

This topic describes how to compile and save a protocol.

When you compile a protocol, BenchWorks checks to make sure that your protocol makes logical sense.

Compiling a protocol

To compile a protocol:

1. Click **Compile**.

   Errors are reported in the Log toolbar.

   Note: Whenever you start a protocol, BenchWorks automatically compiles it and checks for errors.

Saving a protocol

To save a protocol you must be logged on with an administrator or technician user account.

!! IMPORTANT !! When you edit a protocol, the changes take effect immediately. However, unless you explicitly save the protocol, the changes are lost when you exit BenchWorks.

To save a protocol:

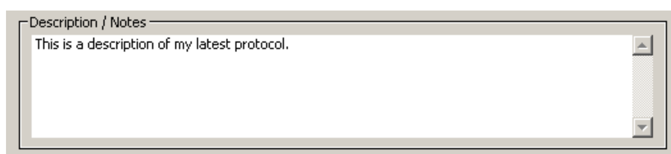
1. Select **File > Save As**.
2. In the **Save As** dialog box, navigate to the folder in which you want to save the protocol.
3. In the **File name** text box, replace the selected file name with a name of your choice.

Adding notes about the protocol

You can add a description of your protocol to remind yourself of its features or for BenchCel operators to review before performing a run.

To add notes about your protocol:

1. Click **Tools** and select **Protocol Options**.
2. Type your notes into the **Description/Notes** text box.



Note: When you open a new protocol, its associated description replaces the description of the previous protocol.

Related information

For information about...	See...
The workflow that this procedure belongs to	“Workflow for creating a protocol” on page 85
Setting up a plate instance	“Setting up a plate instance” on page 88
Resolving protocol compilation errors	“Compilation warnings and errors” on page 303
Setting the number of simultaneous plates	“Setting the number of simultaneous plates” on page 91

About setting task parameters

About this topic

This topic describes what task parameters are and what they do.

Task parameters defined

After you have added a task, you can set parameters for it. Most tasks require you to set parameters. The parameters specify the details of the task. As you set the parameters, the text underneath the task icons change to reflect the new parameters.

Related information

For information about the specific types of tasks, see the following topics:

For information about...	See
Pipette task parameters	"About setting pipette task parameters" on page 96
AliQuot task parameters	"Setting AliQuot task parameters" on page 100
Apply Label task parameters	"About setting Apply Label task parameters" on page 101
Delid/Relid task parameters	"Setting Delid/Relid task parameters" on page 108
Downstack and Upstack task parameters	"Setting Downstack and Upstack task parameters" on page 109
Incubate task parameters	"Setting Incubation task parameters" on page 112
Multidrop task parameters	"Setting Multidrop task parameters" on page 114
Nanodrop task parameters	"Setting Nanodrop task parameters" on page 116
Pierce task parameters	"Setting Pierce task parameters" on page 117
Place Plate task parameters	"Setting Place Plate task parameters" on page 120
QFill2 task parameters	"Setting QFill2 task parameters" on page 121
Restack task parameters	"Setting Restack task parameters" on page 122
Seal task parameters	"Setting Seal task parameters" on page 124
User Message task parameters	"Setting User Message task parameters" on page 125

For information about...	See
Waitfor and Signal task parameters	"Setting Waitfor task and Signal task parameters" on page 127
WellMate task parameters	"Setting WellMate task parameters" on page 128
VSpin with Access2 task parameters	"Setting VSpin with Access2 task parameters" on page 130
Setting the order of tasks	"Specifying task order across processes" on page 132
Checking for protocol errors	"Simulating a run" on page 97
Individual modules that perform tasks	"About BenchWorks" on page 27

About setting pipette task parameters

About this topic

This topic describes the differences between tasks and pipette tasks and provides cross references to topics about specific pipette tasks.

Pipette tasks versus tasks

Pipette tasks differ from tasks in the following ways:

- ❑ Pipette tasks are added in the pipette process editor and not the protocol editor
- ❑ Pipette tasks refer to settings in the liquid library editor
- ❑ Pipette tasks may require you to configure a VPrep shelf in the device manager

Related information

For information about the specific types of pipette tasks, see the following topics:

For information about...	See...
Aspirate task parameters	"Setting Aspirate pipette task parameters" on page 144
Change instance pipette task parameters	"Setting Change Instance pipette task parameters" on page 146
Change tips pipette task parameters	"Setting Change Tips pipette task parameters" on page 148
Dispense pipette task parameters	"Setting Dispense pipette task parameters" on page 150

For information about...	See...
Dry tips pipette task parameters	"Setting Dry Tips pipette task parameters" on page 153
Loop pipette task parameters	"Setting Loop pipette task parameters" on page 153
Mix pipette task parameters	"Setting Mix pipette task parameters" on page 155
Pump reagent pipette task parameters	"Setting Pump Reagent pipette task parameters" on page 157
Wash tips pipette task parameters	"Setting Wash Tips pipette task parameters" on page 159

Simulating a run

About this topic

This topic provides suggestions on how to check for errors in a protocol after it is compiled.

Simulating the run

After making sure that there are no compiler errors in the protocol, you can check for other types of problems by running the protocol through the simulator. The simulator allows you to confirm that steps are done and sequenced correctly and to find problems such as:

- Deadlocks
- Periods of inefficiency, such as when the robot is not being used
- Plates spending different times at critical steps when they should be run under identical conditions
- A number of simultaneous plates that is too high or too low

The simulator does not move any instruments. It performs a "virtual run" based on the estimated task execution times displayed for each task. You can change the execution times for tasks to make the simulation more accurate for your protocol.

One approach you can use for testing is the following:

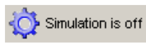
1. Run the simulator with the default task execution times and the same number of plates expected for a run to identify deadlocks and rate limiting tasks.
2. Resolve any major problems with the protocol.
3. Perform a real, dry run with a plate.
4. Use the times recorded in the Log toolbar to edit the task execution times for each task.

5. Run the simulator with the more accurate task execution times.
6. Fine-tune the protocol based on the results of the simulation.

Running the simulator

To run the simulator:

1. Turn the **Simulator** on by clicking the **Simulation is off** button on the toolbar.

   The button changes to read **Simulation is on**.

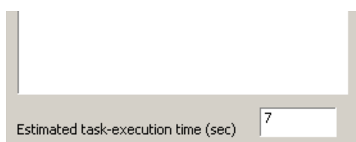
2. Click the **Start** button to run the protocol.

Changing execution times

Not all tasks have adjustable execution times, but for those that do, follow this procedure to change them.

To change the default task execution times:

1. Select a task in a protocol sequence.
2. In the **Protocol Task Parameters** toolbar, click the **Advanced Settings** tab.
3. At the bottom of the toolbar, enter a new value for the **Estimated task-execution time**.

Related information

For information about...	See
Compiling and saving a protocol	"Compiling and saving protocols" on page 93
Resolving errors	"Compilation warnings and errors" on page 303
The workflow that this procedure belongs to	"Workflow for creating a protocol" on page 85

6 Setting task parameters

This chapter gives the procedures for configuring the parameters for individual tasks in a protocol.

Setting AliQuot task parameters

About this topic

This topic describes how to set the AliQuot task parameters.

Aliquot task defined

The AliQuot task moves a plate to an AliQuot where liquid is dispensed into the plate.

Procedure

To set AliQuot task parameters:

1. Add the **AliQuot** task to the protocol.
2. In the **Task Settings** page of the **Protocol Task Parameter** toolbar, type the volume of liquid that you want to dispense in the **Enter dispense volume** text box.
3. In the **Available devices** list, select the AliQuot device that you want to use for this task and click **Add**.

   The name of the device moves to the **Devices to use** list.
4. If you intend to use a filter plate:
   a. Select the **Use filtration** check box.
   b. Complete the **Begin filtration...** sentence to enter the remaining parameters.

      The **After filtration delay** is the time after the filtration has stopped before the plate can be picked up.

      !! DAMAGE HAZARD !! **If the After filtration delay is too short, the robot may crash.**

Related information

For information about...	See...
Adding and deleting tasks	"Adding and deleting tasks and pipette tasks" on page 92
Pipette task parameters	"About setting pipette task parameters" on page 96
Creating protocols	"Workflow for creating a protocol" on page 85

About setting Apply Label task parameters

About this topic

This topic provides some information about the Apply Label task to help you make choices about bar code labeling.

Apply Label task defined

The Apply Label task uses a VCode to print a bar code label and apply it to a plate.

Bar code labelling decisions

Before you add an Apply Label task to a protocol, consider your record-keeping and automation needs. For example:

- ❑ Do you need each bar code label to be unique within a run, but not from run to run?
- ❑ Do you need each bar code label to be unique across all runs?
- ❑ Will you be using a Laboratory Information Management System (LIMS) for barcoding information and record-keeping?
- ❑ What human-readable fields do you want to include on the label?
- ❑ Do you want to use a bar code input file?
- ❑ Do you want to use a bar code data file?

Bar code format

When setting Apply Label task parameters, you need to select a bar code format, which specifies the type, number, properties, and location of fields that are printed on bar code labels.

Some formats are provided with the VCode, but you can define others according to your needs. Each format is identified by a number, which you enter when setting up an Apply Label task.

Related information

For information about...	See...
Using bar code modifiers	"About combining bar code modifiers" on page 106
Creating a bar code input file	"Using bar code input files" on page 198
Using bar code data files	"Using bar code data files" on page 200
General options	"Setting general options" on page 41
Creating a plate instance	"Setting up a plate instance" on page 88
Using plug-ins	"About the FileReader plug-in" on page 164
Log files	"About log and data files" on page 51

Setting Apply Label task parameters

About this topic

This topic describes how to set the parameters for the Apply Label task.

Read this topic if you are:

- ❑ An administrator or technician who writes protocols that uses a VCode
- ❑ An operator who needs to specify parameters for one or more of the apply label tasks

Before you start

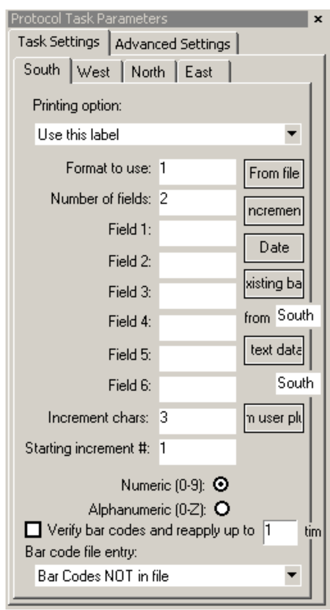
Before working with Apply Label tasks, read the *VCode User Guide* to learn about bar codes and label formats.

Procedure

To set Apply Label task parameters:

1. Add the **Apply Label** task to a protocol process.
2. In the **Protocol Task Parameters** toolbar, click a tab corresponding to a side of the plate.

   The options are **South**, **West**, **North**, and **East**.



3. Select one of the options in the **Printing Option** list box.

If...	Then...
You do not want to place a label on this side of the plate	Select **No Label** and return to step 2.
You want to define a bar code label and place it on this side of the plate	Select **Use this label** and continue with step 4.
You want to print a label that is the same as a label already set up for another side of the plate	Select **Use *side* label** and continue with step 4. All other parameters on the page are ignored.

4. In the **Format to use** text box, type a number that corresponds to the bar code format that you want.
5. In the **Number of Fields** text box, type the number of fields that you want to print on the bar code.

   The maximum number of fields you can print is limited by the number of fields in the format you selected. For example, if the format specifies three fields you cannot print a bar code with four fields. In this example, if you do enter the number four into the text box, the last field is ignored.
6. Click in the **Field** text box for the first field that you want to use and enter one or more of the following types of information that you want to print in that position on the label:

If you want to print...	Then...
Text field that does not increment	Type the text in the text box.
A field that is identical to a bar code field on another side of the plate	To use this option, your VCode must have an attached bar code reader. a. Select the side of the plate from the **from *side* side** list box, below the **Use existing bar code button**. b. Click **Use existing bar code**. This places the code [BC] in the text box. *Note:* This option copies a single field from another side of the plate. The similar option selected in the **Printing Options** list box copies an entire bar code from another side of the plate.

If you want to print...	Then...
A series of bar codes from a bar code input file	a. Make sure that you have set up a bar code input file and selected it in BenchWorks general options. b. Make sure that the number of bar codes specified in the series of the bar code input file that you want to use is equal to or greater than the number of labels that you want to print. c. Select the series in the **Bar code file entry** list box at the bottom of the toolbar. *Note:* If there are no entries in the **Bar code file entry** list box other than the default text, you need to set the location of the bar code input file. d. Click **From File**. This places the code [FILE] in the text box. **!! IMPORTANT !! If you use this option in two fields, the same data will be printed in both fields. You cannot enter different data into fields using this method.**
A series of bar codes by referencing bar codes on another side of the plate, using a bar code data file	a. Make sure that you have created a bar code database file and selected it in BenchWorks general options. b. Make sure that the incoming plates have bar code labels on the south or west side, or a previous Apply Label task is set up to print labels on the south or west side. c. If the incoming plates are labelled, make sure that the system verifies the labels by setting up bar code control on the plate icon. d. Click **From text database**. This places the code [DB] in the text box. e. In the **use *side* side** list box under the **From text database** button, select the side of the plate that has the bar codes you want to use as a reference. **!! IMPORTANT !! Although you can select any side, only the south and west sides can currently be used.**

If you want to print...	Then...
A series of bar codes that increment, but which are *not* specified by a bar code input file	a. Type the root data that you want in the **Field 1** text box. b. Click **Increment**. This adds the code [INC] to the root data. c. In the **Increment chars** text box, type the number of alphanumeric characters that you want to be appended to the root data. For example, if you want the series to increment from 01 enter 2. If you want it to increment from 001, enter 3. d. In the **Starting increment #** text box, type the number that you want to be printed on the first label, for example, 100. e. Select either **Numeric** or **Alphanumeric** depending on the increment style you prefer. Alphanumeric increments use 0–9, A–Z, whereas numeric increments use 0–9.
From a plug-in that you have developed	Select **From user plug-in**.

7. Return to step 6 and fill out another field until all required fields are completed.

   Note: If you enter information in a field that does not exist in the format you have chosen, the information is ignored.

8. Return to step 3 and define labels to put on other sides of the plate.

Related information

For information about...	See...
Using bar code modifiers	"About combining bar code modifiers" on page 106
Creating a bar code input file	"Using bar code input files" on page 198
Using bar code data files	"Using bar code data files" on page 200
General options	"Setting general options" on page 41
Creating a plate instance	"Setting up a plate instance" on page 88
Using plug-ins	"About the FileReader plug-in" on page 164
Log files	"About log and data files" on page 51

About combining bar code modifiers

About this topic

This topic describes how to combine bar code modifiers for the Apply Label task.

Bar code modifiers are text/numerical strings that are appended to the bar code. Typically they are used to add a readable text/numerical string to a bar code, which increments for each plate, giving each plate a unique label.

Read this topic if you are:

- ❑ An administrator or technician who writes protocols that uses a VCode
- ❑ An operator who needs to specify parameters for one or more of the Apply Label tasks

Before you start

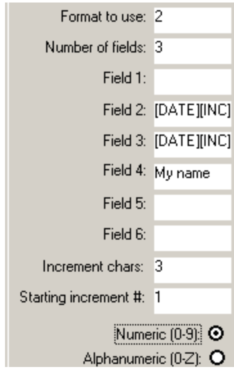
Before you modify the bar codes, read the *VCode User Guide* to become familiar with bar codes and label formats.

Combining bar code modifiers

You can combine bar code modifiers with text in a single field. This section shows how to do this by using an example. The example has five fields and for each field the symbology or font used has been defined as follows in the VCode.

Format field (valid for pre-2003 VCodes)	BenchWorks/ VCode field	Symbology/Font
Field 0	Field 1	Swiss Mono 721 Bold
Field 1	Field 2	Code 39
Field 2	Field 3	Dutch Roman 801 prop
Field 3	Field 4	Dutch Roman 801 prop
Field 4	Field 5	Dutch Roman 801 prop

In the screenshot below, you can see that three of the BenchWorks fields are used: Field 2, Field 3, and Field 4.



Field 2 prints a Code 39 symbology bar code that is incremented for each printing. Field 3 prints the same bar code in the human readable Dutch Roman font. The text entered into Field 3 is the same as that entered into Field 2.

The information entered in Field 3, combined with the information entered into the Increment chars and Starting increment # fields, creates the following human readable bar code sequence:

*Date*001

*Date*002

*Date*003

*Date*004

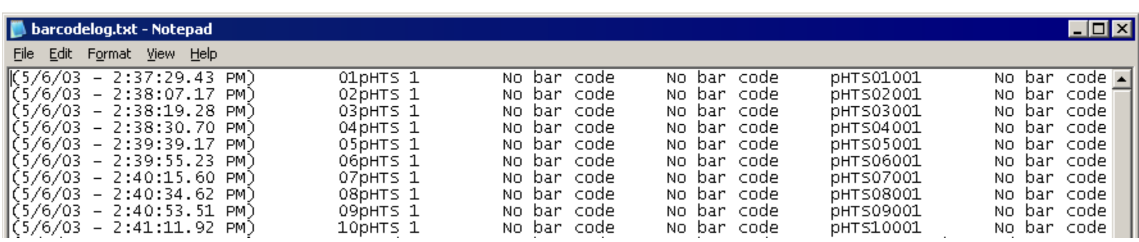
*Date*005

*Date*006

*Date*007...

The bar code log file

Information about the bar codes that are applied during a run is saved in a bar code log file. Because this is a tab-delimited text file, the data can easily be imported into a spreadsheet program.


barcodelog.txt - Notepad

File Edit Format View Help

(5/6/03 - 2:37:29.43 PM)	01pHTS 1	No bar code	No bar code	pHTS01001	No bar code
(5/6/03 - 2:38:07.17 PM)	02pHTS 1	No bar code	No bar code	pHTS02001	No bar code
(5/6/03 - 2:38:19.28 PM)	03pHTS 1	No bar code	No bar code	pHTS03001	No bar code
(5/6/03 - 2:38:30.70 PM)	04pHTS 1	No bar code	No bar code	pHTS04001	No bar code
(5/6/03 - 2:39:39.17 PM)	05pHTS 1	No bar code	No bar code	pHTS05001	No bar code
(5/6/03 - 2:39:55.23 PM)	06pHTS 1	No bar code	No bar code	pHTS06001	No bar code
(5/6/03 - 2:40:15.60 PM)	07pHTS 1	No bar code	No bar code	pHTS07001	No bar code
(5/6/03 - 2:40:34.62 PM)	08pHTS 1	No bar code	No bar code	pHTS08001	No bar code
(5/6/03 - 2:40:53.51 PM)	09pHTS 1	No bar code	No bar code	pHTS09001	No bar code
(5/6/03 - 2:41:11.92 PM)	10pHTS 1	No bar code	No bar code	pHTS10001	No bar code

Related information

For information about...	See...
Creating a bar code input file	"Using bar code input files" on page 198
Using bar code data files	"Using bar code data files" on page 200
General options	"Setting general options" on page 41
Creating a plate instance	"Setting up a plate instance" on page 88
Using plug-ins	"About the FileReader plug-in" on page 164
Log files	"About log and data files" on page 51

Setting Delid/Relid task parameters

About this topic

This topic describes how to set the Delid/Relid task parameters. These tasks are used by the BenchCel to remove and replace labware lids.

Read this topic if you are:

- ❑ An administrator or technician who writes protocols that uses this task
- ❑ An operator who needs to specify parameters for the Delid/Relid tasks

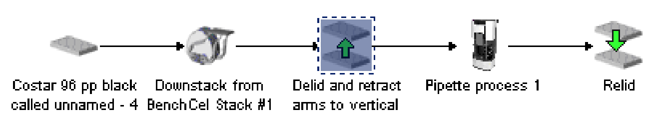
Delid/Relid task defined

The Delid task removes a plate's lid using a device such as a plate hotel or vacuum delidding station. The Relid task replaces the lid.

Neither the Delid nor Relid tasks have configurable parameters.

Example

The Delid/Relid tasks are used to remove and replace a plate lid. A typical use for this task is shown below:



In the example, a plate is downstacked and the lid is removed. The plate is pipetted at the VPrep and the lid is replaced. If your system has a trash chute and you want to remove a plate's lid and put the lid in the trash, use the Delid task and do not add a subsequent Relid task.

Procedure

Setting Delid/Relid task parameters

There are no task parameters for these processes.

Related information

For information about...	See...
Defining lidded plates	“Defining labware” on page 213
Adding and deleting tasks	“Adding and deleting tasks and pipette tasks” on page 92
Creating a protocol process	“Setting up a plate instance” on page 88
Specifying task order	“Specifying task order across processes” on page 132

Setting Downstack and Upstack task parameters

About this topic

This topic describes how to set the Downstack and Upstack task parameters.

Read this topic if you are:

- ❑ An administrator or technician who writes protocols using the Downstack and Upstack tasks
- ❑ An operator who needs to specify parameters for these tasks

Stacker task defined

The Downstack and Upstack tasks move plates into or out of BenchCel racks.

The process of moving a plate out of a stacker rack is called downstacking. The process of moving a plate into a stacker rack is called upstacking.

Plates may be returned to the same or different stacker racks.

You can make a single task upstack to, or downstack from, more than one stacker rack. For example, in a downstacking task, when all of the plates are removed from one stacker rack, the robot will begin to pick plates from a second stacker rack. The two stacker racks are referred to as pooled downstackers.

Procedure

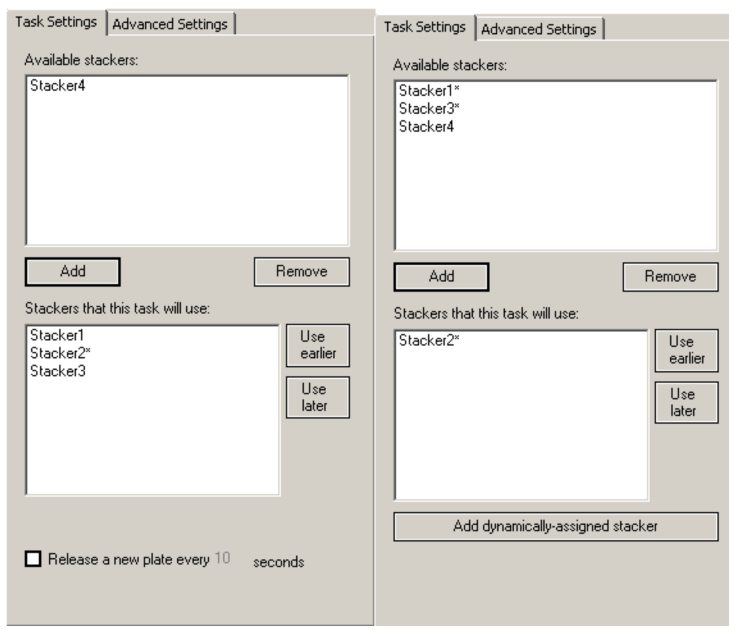
To set Downstack and Upstack task parameters:

1. Add the **Downstack** or **Upstack** task to a protocol process.

2. In the list of available stacker racks in the **Protocol Task Parameters** toolbar, select a stacker rack to downstack from or upstack to and click **Add**.

   To select more than one stacker rack, SHIFT-click or CTRL-click before clicking **Add**.

   An asterisk next to a stacker in the list means that the stacker rack is currently assigned to a task that uses the same labware.



3. To remove a stacker rack from your list of available stacker rack devices, select it and click **Remove**.
4. If you have added more than one stacker rack, you can change the order in which particular stackers racks are used:
   a. Select a stacker rack.
   b. Click **Use earlier** to increase the priority of the stacker rack or **Use later** to decrease the priority of it.

5. To specify a time interval for when plates are made available to the system, select the **Release a new plate** check box and type in an interval time.

   You can use this feature to avoid a plate processing bottleneck that results in plates having different incubation times.

   Consider a simplified example process in which plates are downstacked, labels applied, liquid dispensed into, and then incubated for 10 minutes.

   Applying the label only takes a few seconds while subsequent tasks take longer. This creates a processing bottleneck.

   If the dispense task takes 2 minutes, plates that are ready for the pipetting step would have to wait. In this example, the first plate would incubate for approximately 5 minutes, the second plate for approximately 7 minutes, the third plate for approximately 9 minutes, and so on. To avoid these different incubation times, you could downstack one plate every 2 minutes. The plates are then incubated sequentially and not simultaneously.

6. If you want to dynamically assign an upstacking stacker racks, click **Add dynamically-assigned stacker.**

   Note: This option is only available for upstacking tasks.

   With dynamic assignment you do not have to specifically assign every stacker rack that will receive plates because assignments are made automatically.

   When stacker racks are dynamically assigned, the text "TBD," meaning "To Be Determined", is added to the stacker task icon.

   Note: When using dynamically assigned stackers, you need to have a dynamic stacker in the Device Manager for each stack you expect to require.

   You can determine which stackers contain which plates at the end of the run by consulting the run log.

Related information

For information about...	See...
Adding and deleting tasks	"Adding and deleting tasks and pipette tasks" on page 92
Creating a protocol process	"Setting up a plate instance" on page 88
Specifying task order	"Specifying task order across processes" on page 132
Restacking	"Setting Restack task parameters" on page 122

Setting Incubation task parameters

About this topic

This topic describes how to set the Incubation task parameters.

Read this topic if you are:

- ❑ An administrator or technician who writes protocols using the Incubation task
- ❑ An operator who needs to specify parameters for this task

Incubate task defined

The Incubate task performs a timed incubation of a plate. It is typically used for short incubations.

The number of plates that can be incubated simultaneously is limited by the number of platepads that are available for holding plates.

Process overview

The overall process for a typical incubation is as follows:

1. Incubation of the plate starts with the addition of an initiating reagent.

   This would be performed by a liquid-handling task, such as a Pipette Process task.
2. The plate is moved to a platepad.

   When the plate arrives at the platepad, the incubation time parameter that you specify starts timing.
3. The plate is moved from the platepad to a reader.

   The plate is moved when the incubation time parameter that you specify ends.

Incubation time error

The time parameter that you set for the incubation period is not the actual time of incubation. It represents the minimum time that the plate sits on the platepad where the incubation task is carried out.

The actual incubation period starts when the initiating reagent is added and continues until the plate is transferred to the next step in the process. This means that the actual incubation is longer than the time parameter that you set, by an amount that depends on the scheduling and operating speed of the robot.

If your assays require greater precision in plate incubation times than this method supports, you can adjust the rate at which plates enter the system.

Procedure

To set incubation parameters:

1. Add the **Incubate** task to a protocol process window.
2. In the **Task Settings** page of the **Protocol Task Parameter** toolbar, type the length of time for which you want the plate to incubate on the platepad.

3. If you need to control the rate at which plates are delivered to a particular instrument, incubate the plates on a plate hotel or similar short-term storage device and select **Start timer when previous plate finishes incubating**.

   A situation can arise in which plates move through a protocol too fast for one of the steps. An example is a pipetting step in which a reagent that starts a timed reaction is added to the plate. The result is that plates queue up at the pipettor.

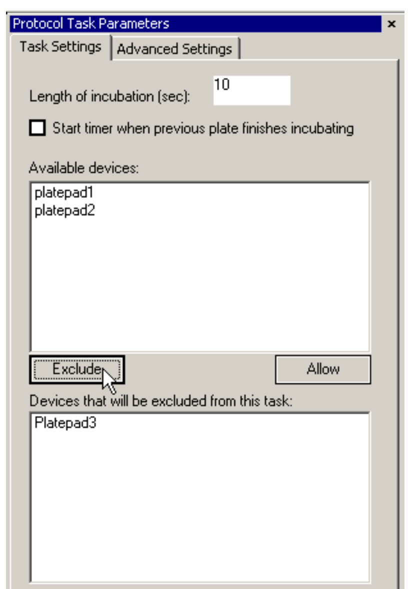
   Without selecting the **Start timer when previous plate finishes incubating** check box, plates are delivered to the plate hotel as fast as the robot can deliver them, are incubated for the time specified in the **Length of incubation text box**, and then leave as fast as the robot can remove them. The robot may remove the plates as frequently as every 10 seconds.

   With the **Start timer when previous plate finishes incubating** check box selected, the time separation between each plate being picked up is specified by the value in the **Length of incubation text box**. Using this function, plates can leave the plate hotel and delivered to the pipettor every minute instead of every 10 seconds.

   An alternative way to control the time at which plates are delivered to an instrument is to use the **Release a new plate every *x* seconds** parameter for the Downstack task.

4. If you want to restrict a device so it is not used as the place where the incubation takes place, in the **Available devices** box, select the device and click **Exclude**.

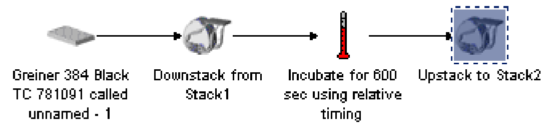
   This may be useful if a particular platepad is used during multiple steps of the protocol.



   The name of the platepad moves to the lower box.

Example

In the following example, a plate is downstacked, moved to an incubator, and then upstacked.



Related information

For information about...	See...
Adding tasks to protocols	"Adding and deleting tasks and pipette tasks" on page 92
Pipette tasks	"About setting pipette task parameters" on page 96
Other tasks	"About setting task parameters" on page 95

Setting Multidrop task parameters

About this topic

This topic describes how to set the Multidrop task parameters for the BenchCel.

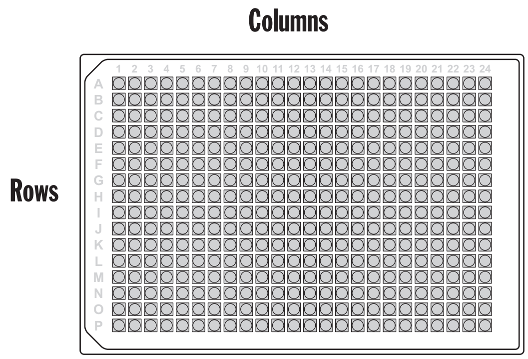
Read this topic if you are:

- ❑ An administrator or technician who writes protocols using the Multidrop instrument
- ❑ An operator who needs to specify parameters for the Multidrop task

Multidrop task defined

The Multidrop task moves a plate to a Thermo Labsystems Multidrop 384 or 96 dispenser and instructs the dispenser to fill the plate. You can specify whether to include prime, shake or purge operations.

The Multidrop is capable of dispensing up to eight different liquids into one plate. During a dispense, the plate stage moves in the left-right axis and the wells are filled in columns, starting with column 1. In a 96-well plate there are eight rows, so every well in a column is filled at the same time. In a 384-well plate there are 16 rows, requiring two dispense passes. During the first pass, every second well in a column is filled. The plate stage then moves over by one well so that the dispensers sit above empty wells. In the second pass, the stage returns to the starting position and the remaining wells are filled.



Procedure

To set Multidrop parameters:

1. Add a Multidrop task to a protocol process.
2. If the BenchCel has more than one Multidrop, select the one you want to use for the task from the **Select the Multidrop to work with** list box.
3. If you want to prime the Multidrop, enter the volume of liquid to prime with in the **Prime volume** text box.

   Priming moves liquid through the dispensing system into waste to make sure that there is no air in the lines.
4. In the **Dispense volume** text box, enter the volume of liquid that you want to dispense into each well.
5. To shake the plate after the dispense, enter a value into the **Shake time** text box.
6. To purge the lines after the dispense, select the **Purge x times when complete** check box and enter the number of times to purge.

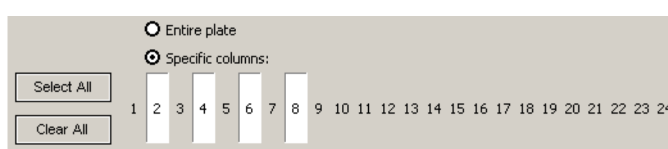
   Purging moves liquid back into the liquid reservoir or reservoirs. If you want to return all liquid, the value of *x* may need to be greater than one.
7. To dispense into all wells, make sure that the **Entire plate** option is selected.
8. To only dispense into some of the wells:

   a. Select the **Selected columns** option.

   b. Click **Select none**.

   c. In the number matrix, click the numbers corresponding to the columns to dispense into.

   In the following example liquid will be dispensed into columns 2, 4, 6, and 8 of a 384-well plate.



Related information

For information about...	See...
Adding tasks to protocols	"Adding and deleting tasks and pipette tasks" on page 92
Pipette tasks	"About setting pipette task parameters" on page 96
Other tasks	"About setting task parameters" on page 95

Setting Nanodrop task parameters

About this topic

This topic describes how to set the Nanodrop parameters on a BenchCel.

Read this topic if you are:

- ❑ An administrator or technician who writes protocols using the Nanodrop instrument
- ❑ An operator who needs to specify parameters for the Nanodrop task

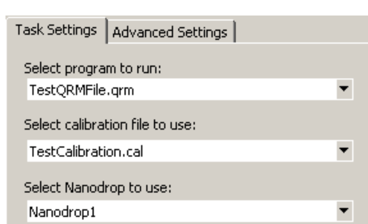
Nanodrop task defined

The Nanodrop task moves a plate to a Nanodrop liquid dispenser and runs a pre-defined program, using a calibration file for more accurate dispensing.

Procedure

To set Nanodrop task parameters:

1. Add the Nanodrop task to a protocol process.
2. In the **Task Settings** page of the **Protocol Task Parameter** toolbar, select the following from the list boxes:
   a. The program to run.
   b. The calibration file to use.
   c. The Nanodrop to use, if your BenchCel is connected to more than one.
   The name of the Nanodrop is defined in the device manager.



Related information

For information about...	See...
Adding tasks to protocols	"Adding and deleting tasks and pipette tasks" on page 92
Pipette tasks	"About setting pipette task parameters" on page 96
Other tasks	"About setting task parameters" on page 95

Setting Pierce task parameters

About this topic

This topic describes how to set the Pierce task parameters for the BenchCel.

Read this topic if you are:

- ❑ An administrator or technician who writes protocols using the Pierce task
- ❑ An operator who needs to specify parameters for the Pierce task

Pierce task defined

The Pierce task pierces a plate seal using a PlatePierce.

Procedure

To set the Pierce task parameters:

1. Add the **Pierce** task to a protocol process.
2. In the **Task Settings** page of the **Protocol Task Parameter** toolbar, enter a value in the **Pierce Pressure** text box.

   If you are unsure of the best piercing pressure to use for your application, contact the Velocity11 Service Center.

Related information

For information about...	See...
Adding tasks to protocols	"Adding and deleting tasks and pipette tasks" on page 92
Pipette tasks	"About setting pipette task parameters" on page 96
Other tasks	"About setting task parameters" on page 95

Setting Place Labware task parameters

About this topic

This topic describes how to set the Place Labware task parameters.

When to use

The Place Labware task is designed for labs that use multiple types of labware for a protocol. It is useful if you have devices where you want to use one type of labware on the device for some runs and another type for other runs of the same protocol. This saves you from having to create a device file for each protocol that uses a different type of labware.

If you are always using the same labware, you may not want to use this task.

The Place Labware task is only used in pre-protocol processes.

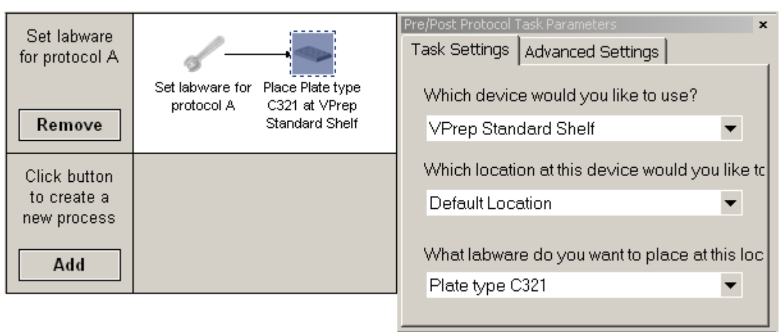
The Place Labware task can only be used with labware that is stationary throughout the running of the protocol.

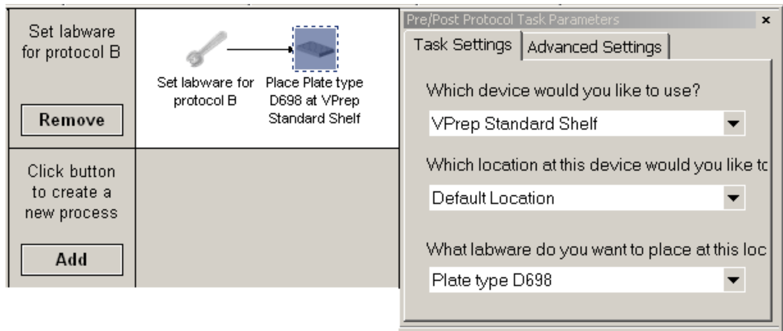
The Place Labware task allows you to associate a labware entry with a device just before starting a run instead of in the device file.

Usage example

For example, if in one protocol you restrict use of a device to *labware1* only, and in another protocol you restrict use of the same *device* to *labware2* only, there are two ways to handle this:

1. Create two device files, one called *device* for *labware1* and another called device for *labware2*, where the labware type is specified in the "Allowed / prohibited labware" device property.
2. Create one device file where labware type is not specified, then create a pre-protocol for each protocol where the Place Labware task specifies the labware that must be used for the protocol.





Procedure

To set task parameters for the Place Labware pre-protocol task:

1. Click on the **Place Labware** icon and drag it from the pre-protocol task list to the protocol editor window until a vertical, dashed line appears.
2. In the **Task Settings** page of the **Pre/Post Protocol Task Parameters** toolbar select items from appropriate boxes:
   - The device you want to use with this task
   - The location of the labware on the device you are using
   - Labware type you are using in this protocol

Related information

For information about...	See...
Working with device files	"Working with device files" on page 195
Pre-protocol and post-protocol processes	"Setting up a pre-protocol or post-protocol process" on page 87
Creating protocols	"Workflow for creating a protocol" on page 85

Setting Place Plate task parameters

About this topic

This topic describes how to set the Place Plate task parameters for the BenchCel.

Read this topic if you are:

- An administrator or technician who writes protocols using the Place Plate task
- An operator who needs to specify parameters for the Place Plate task

Place Plate task defined

The Place Plate task moves a plate to any location. This task has many uses.

The next task in the process after a Place Plate task moves the plate to another location.

Place Plate task and bar codes

The Place Plate task can be used in combination with a platepad bar code reader to read a bar code. The requirements for this are as follows:

- The platepad must be set up in the device manager as a bar code reader for the platepad.
- The plate icon for the plate must indicate that the plate has a bar code on the side that the bar code reader reads. See Related information at the end of this topic for how to do this.

Whenever a plate is placed on this platepad, the scheduler tells the bar code reader to read the plate's bar code.

Procedure

To set the Place Plate task parameters:

1. Add the **Place Plate** task to the protocol process.
2. In the **Task Settings** page of the **Protocol Task Parameter** toolbar, select the device to which you want to move the plate.

   You may need to scroll down to make your selection.
3. If the **Place Plate** task is the first task in the protocol and you want to require that the operator confirms the bar code on the plate that is placed, check the **Manually confirm bar code** check box.

   This can prevent the wrong plate from being used in the protocol.

   To confirm the bar code, when the plate is picked up, the operator is prompted to enter the bar code of the plate that should be in the placed position. If the two codes do not match, an error is generated.

Related information

For information about...	See...
Indicating bar codes on plate icon	“Setting up a plate instance” on page 88
Adding tasks to protocols	“Adding and deleting tasks and pipette tasks” on page 92
Pipette tasks	“About setting pipette task parameters” on page 96
Other tasks	“About setting task parameters” on page 95

Setting QFill2 task parameters

About this topic

This topic describes how to set the QFill2 task parameters for the BenchCel.

Read this topic if you are:

- ❑ An administrator or technician who writes protocols using the QFill2 instrument
- ❑ An operator who needs to specify parameters for the QFill2 task

Note: In some versions of BenchWorks, you will see “QFill” instead of “QFill2”.

QFill2 task defined

The QFill2 task dispenses liquids into plate wells and offers the options of vacuum filtration.

If you want to use vacuum filtration, your QFill2 must include the vacuum station option developed by Velocity11.

For more information about setting up vacuum filtration as a separate task, contact the Velocity11 Service Center.

Procedure

To set the QFill2 task parameters:

1. Add a **QFill2** task to a protocol process.
2. In the **Task Settings** page of the **Protocol Task Parameter** toolbar, enter the volume of liquid that you want to dispense into each plate well.
3. In the **Available devices** box, select the QFill2 that you want to use for this task and click **Add**.

   The name of the device moves to the **Devices to use** list.
4. If you want to use vacuum filtration, in the **Filtration** group box:
   a. Select the **Use filtration** check box.

b. Complete the **Begin filtration...** sentence, specifying how long before or after filling you want the filtration to start.

c. Type a filtration duration in the **Filter for** text box.

d. If you want to add a post-filtration delay, type the delay time in the remaining text box.

   The delay is the time that elapses before the robot will pick up the plate, allowing the vacuum under the plate to dissipate.

   !! IMPORTANT !! If this value is too small, the robot may not pick up the plate correctly, causing a robot crash.

Related information

For information about...	See...
Adding tasks to protocols	"Adding and deleting tasks and pipette tasks" on page 92
Pipette tasks	"About setting pipette task parameters" on page 96
Other tasks	"About setting task parameters" on page 95

Setting Restack task parameters

About this topic

This topic describes how to set the Restack task parameters on the BenchCel.

Read this topic if you are:

- ❑ An administrator or technician who writes protocols using the Restack task
- ❑ An operator who needs to specify parameters for the Restack task

Restack task defined

The Restack task collects plates in a stack, moves them to another stack in a definable time and in a way that maintains the proper order of the plates, so that they are ready to be passed to another task.

The Restack task can be used as part of a larger process that carries out simultaneous, timed incubations of more than one plate, where the goals are the following:

- ❑ Time between the start of the incubation and the reading of the plate to be approximately the same for each plate
- ❑ Evaporation from the plates is to be minimized

Note: Two or more racks are required to use the Restack task.

Procedure

To set Restack task parameters:

1. Add the **Restack** task to a protocol process.
2. In the **Protocol Task Parameters** toolbar either:
   - Select the empty racks that you want to use for the task and click **Add.**
   - Click **Add dynamically-assigned stacker**.

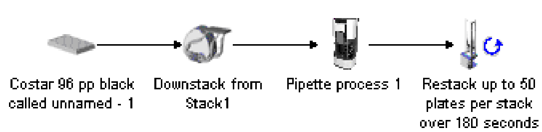
     With this option, the scheduler automatically assigns racks during a run, and there must be at least two racks available for this operation.
3. If you have added more than one rack, you can change the order in which particular racks are used:
   a. Select a rack.
   b. Click **Use earlier** to increase the priority of the rack or **Use later** to decrease the priority of it.
4. In the **Store up to** text box, type a number equal to the number of plates you intend to incubate.

   This value is important because it can affect the timing of the incubation. For example, if the time taken to move all plates to the first restack rack is greater than the time specified for the incubation, the first plate cannot be moved to the next task in time. This problem can be resolved by lowering the number of plates in a restack operation and adding more racks.
5. In the **Incubate plate for** text box, type the time interval between when a plate enters the first restack rack and leaves the second restack rack.

!! IMPORTANT !! When you start a run that includes a Restack task, you must type in a number that is equal to the total number of plates you want to restack in the Number of Runs dialog box.

Usage example

The following screen shot shows one example of how to use the Restack task. The plates are first downstacked and delivered to a VPrep for a pipette process. After the pipette process is completed, the plates are restacked to their original order, ready for the next step in the assay protocol.



Related information

For information about...	See...
Adding tasks to protocols	"Adding and deleting tasks and pipette tasks" on page 92
Pipette tasks	"About setting pipette task parameters" on page 96
Other tasks	"About setting task parameters" on page 95

Setting Seal task parameters

About this topic

This topic describes how to set the Seal task parameters for the BenchCel.

Read this topic if you are:

- ❑ An administrator or technician who writes protocols using the Seal task
- ❑ An operator who needs to specify parameters for the Seal task

Seal task defined

The Seal task places a seal on a plate using a PlateLoc.

If you are sealing more than one type of plate that requires different sealing temperatures, we recommend that you use a separate PlateLoc for each temperature. This avoids time delays as the PlateLoc heats and cools between different plate types.

When you open a protocol containing one or more Seal tasks, the PlateLoc immediately starts adjusting to the temperatures defined in the task parameters.

Procedure

To set Seal task parameters:

1. Add the **Seal** task to a protocol process.
2. In the **Protocol Task Parameters** toolbar, enter a seal time and seal temperature.

   When you enter a seal temperature, the PlateLoc immediately starts adjusting to that temperature.
3. Select the PlateLoc to use for the sealing operation from the **Select PlateLoc to use** list box.

   If you are using more than one PlateLoc, make sure that you select the device with the correct seal type temperature and time for the plate.

Related information

For information about...	See...
Adding tasks to protocols	"Adding and deleting tasks and pipette tasks" on page 92
Pipette tasks	"About setting pipette task parameters" on page 96
Other tasks	"About setting task parameters" on page 95

Setting User Message task parameters

About this topic

This topic describes the parameters for the User Message task.

Read this topic if you are:

- ❑ An administrator or technician who writes protocols for the BenchCel
- ❑ An operator who runs protocols and may need to set the parameters for this task

User Message task defined

The administrator or technician who creates a protocol can add User Message tasks to provide reminders to the operator. The reminders are in the form of messages that appear on the screen at the appropriate time and pause the protocol until acknowledged by the operator.

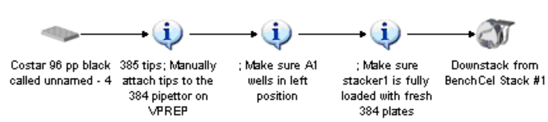
User message tasks can, for example, be used to remind the operator to empty the waste container, fill a reservoir, or remove plates.

!! IMPORTANT !! Remember to remove all user messages from protocols that you intend to run unattended.

Example

In the example shown below, the protocol has four user messages that remind the operator to perform final run-preparation tasks. The user messages appear before the plates are downstacked.

The intended sequencing of the User Messages tasks can be ensured by the addition of Signal tasks to the other processes in the protocol.



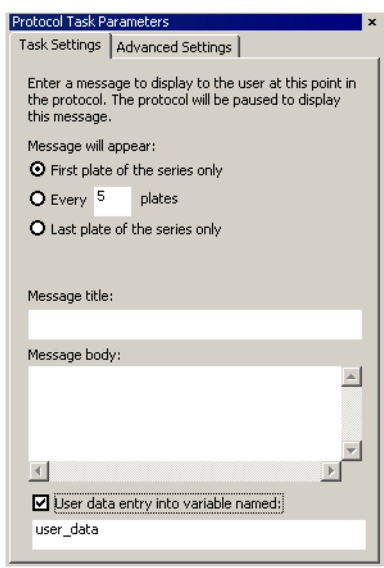
Procedure

To set User Message task parameters:

1. Add the **User Message** task to a protocol process.
2. In the **Protocol Task Parameters** toolbar, select a **Message will appear** option using the following table as a guide.

Option	Description
First plate of the series only	Displays the message the first time it is encountered for that process during the run.
Every *x* plates	Displays the message the first time it is encountered for that process, and then every *x* number of times it is encountered for that process during the run. For example, if the value of *x* is 3, the first plate and the fourth plates in the protocol will trigger the message.
Last plate of the series only	Displays the message the last time it is encountered for that process during the run.
User data entry into variable named	Displays a message that asks for input from the user. The message is generated from a script that is added to one of the tasks.

3. Type the message that you want to display:
   a. Type the message title in the top text box.
      This appears in the process line.
   b. Type the message body in the bottom text box.
4. If using a variable, enter the name in the text box.



Related information

For information about...	See...
The workflow that this procedure belongs to	"Workflow for creating a protocol" on page 85
Adding tasks to protocols	"Adding and deleting tasks and pipette tasks" on page 92
Using JavaScript with BenchWorks	"Using JavaScript in BenchWorks" on page 171
Signal tasks	"Setting Waitfor task and Signal task parameters" on page 127
Pipette tasks	"About setting pipette task parameters" on page 96
Other tasks	"About setting task parameters" on page 95

Setting Waitfor task and Signal task parameters

About this topic

This topic describes how to set the Waitfor and Signal task parameters for the BenchCel.

Read this topic if you are:

- ❑ An administrator or technician who writes protocols for the BenchCel
- ❑ An operator who runs protocols and may need to set the parameters for this task

Waitfor and Signal tasks defined

The Waitfor task and a Signal task work together to specify the order in which tasks are performed across processes.

You must first set the Waitfor task and then set the Signal task.

Procedure

To set Waitfor task parameters:

1. Add a **Waitfor** task to a protocol process.
2. In the **Protocol Task Parameters** toolbar, type a name for the task.

To set Signal task parameters:

1. Add a **Signal** task to a protocol process.
2. In the **Available waitfors** text box of the **Protocol Task Parameters** toolbar, select the **Waitfor** task that you want to reference.

3. Click **Add**.

   The task moves to the lower box.

Related information

For information about...	See...
Example usage of Waitfor and Signal tasks	“Specifying task order across processes” on page 132
Adding tasks to protocols	“Adding and deleting tasks and pipette tasks” on page 92
Specifying task order	“Specifying task order across processes” on page 132
User message tasks	“Setting User Message task parameters” on page 125

Setting WellMate task parameters

About this topic

This topic describes how to set the WellMate task parameters in BenchWorks.

Read this topic if you are:

- ❑ An administrator or technician who writes protocols for the BenchCel
- ❑ An operator who runs protocols and may need to set the parameters for this task

For general information about using the WellMate, see the Matrix Technologies WellMate user documentation.

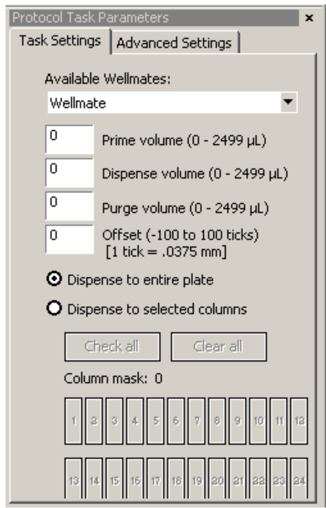
WellMate task defined

The WellMate task moves a plate to a Matrix Technologies WellMate and dispenses liquid into the plate.

Procedure

To set the WellMate task parameters:

1. Add the WellMate task to a protocol process.
2. If your system has more than one WellMate in the **Available Wellmates** list box, select the one you want to use.

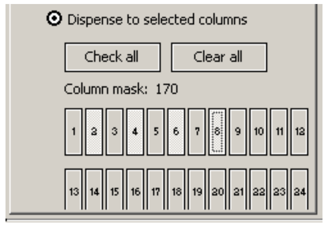


3. Enter values for one or more of the following parameters.

Parameter	Description
Prime volume	The volume of liquid to move through the tubing cartridge before dispensing liquid into a plate
Dispense volume	The volume of liquid to dispense into a plate
Purge volume	The volume of liquid to move through the tubing from the tips back to the reservoir
Offset	Offset defines how far away from the center of the well to dispense

4. To dispense into all wells, make sure that the **Dispense to entire plate** option is selected.
5. To only dispense into some of the wells:
   a. Select the **Dispense to selected columns** option.
   b. Click **Clear All**.
   c. In the number matrix, click the numbers corresponding to the columns to which you want to dispense.

      In the following example, liquid will be dispensed into columns 2, 4, 6, and 8 of a 384-well plate.



Related information

For information about...	See...
Workflow for creating protocols	"Workflow for creating a protocol" on page 85
Adding tasks to protocols	"Adding and deleting tasks and pipette tasks" on page 92
Pipette tasks	"About setting pipette task parameters" on page 96
Other tasks	"About setting task parameters" on page 95

Setting VSpin with Access2 task parameters

About this topic

This topic describes how to set the Access2 task parameters.

Read this topic if you are:

- ❑ An administrator or technician who writes protocols
- ❑ An operator who may need to change the Access2 task parameters

VSpin with Access2 task defined

This task moves a plate to an Access2 and centrifuges it.

Setting VSpin with Access2 task parameters

To set VSpin with Access2 task parameters:

1. Add the **VSpin with Access2** task to a protocol process.
2. In the **Task Settings** page of the **Protocol Task Parameter** toolbar, select a VSpin from the **Select a VSpin with Access2 to use** list box.
3. Set the desired options in the **Load Plate** group box:
   a. Enter the **Gripper Z offset (mm)** for the plate you are using. This value is the distance from the bottom of the plate to where the Access2 grippers will grab the plate. Typically, this value is the same as the robot gripper offset.
   b. If you don't want to use the plate sensor on the Access2, select the **Ignore optical plate sensor** check box.

c. If you are using flexible plates, such as PCR plates, select the **Grip gently** check box.

4. Set the spinning parameters in the **Spin Plate** group box. Enter a value for the **Velocity (%)**, **Acceleration (%)** and **Deceleration (%)**.

   These parameters are calculated as a percentage of the maximum value possible. Maximum speed is 3000 rpm. For more information, see the *VSpin User Guide*.

5. Set the spin duration.

   a. Choose a **Timer mode** option.

      Total time includes the time it takes the VSpin to accelerate to the target speed and decelerate to a stop.

      Time at speed only includes the time that the rotor is at the target speed.

   b. Enter the spin duration.

      The format for the timer is hours:minutes:seconds.

Related information

For information about...	See...
Setting up counterweights	"Setting up a VSpin counterweight plate" on page 63
Adding tasks to protocols	"Adding and deleting tasks and pipette tasks" on page 92
Pipette tasks	"About setting pipette task parameters" on page 96
Other tasks	"About setting task parameters" on page 95

Specifying task order across processes

About this topic

If you are running more than one process in a protocol, you may want to specify that a particular task in one process is performed before a particular task in another process. You do this when creating a protocol by using the combination of Waitfor and Signal tasks.

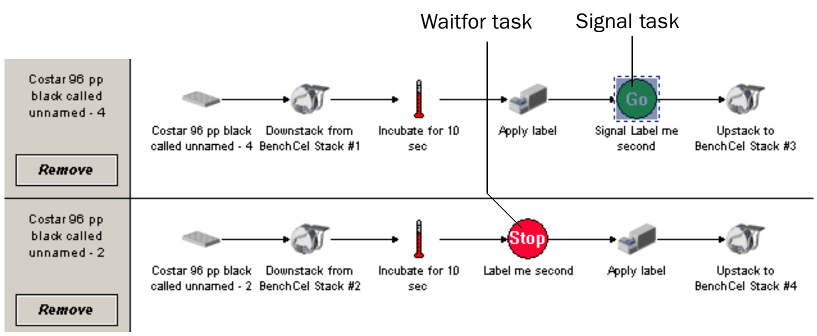
This topic describes how to use this task by providing an example. Read this topic if you are:

- An administrator or technician who writes protocols for the BenchCel
- An operator who runs BenchCel protocols

Example

In this example, the administrator or technician creating the protocol wants to make sure that a bar code is applied to plate 1 before plate 2.

The operator creates the protocol shown in the following diagram.



The order in which plate 1 and plate 2 are processed is selected by the scheduler program. Without the Waitfor and Signal tasks, this would also be the case for the bar coding operation. However, in this protocol, the Waitfor task, called “Label Me Second”, is included for Plate 2, and this causes the process to wait until it receives an instruction to continue. Meanwhile, the bar code is applied to Plate 1.

After the bar code has been applied to Plate 1, the Signal task called “Label Me Second” releases the wait condition on Plate 2 and the bar code is applied.

Related information

For information about...	See...
Workflow for creating protocols	"Workflow for creating a protocol" on page 85
Adding tasks to protocols	"Adding and deleting tasks and pipette tasks" on page 92
Pipette tasks	"About setting pipette task parameters" on page 96
Other tasks	"About setting task parameters" on page 95

7

Setting pipette task parameters

This chapter gives the procedures for configuring the parameters for individual pipette tasks in a protocol.

Configuring a pipette process: example

About this topic

This topic gives an example of how to construct a protocol sub-process that performs a simple pipetting operation. See Related information at the end of this topic for more information about creating a pipette process.

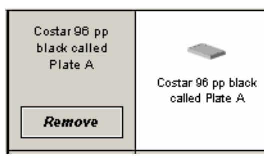
The example

The goal of the pipetting operation in this example is to pipette 20 µL of 1X TE buffer from a reservoir on a particular VPrep into a Costar 96-well plate.

Creating a protocol

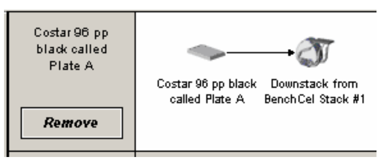
The first step is to create a new protocol by setting up a plate icon for the plate you want to pipette into.

Name the plate icon “Plate A” and select the “Costar 96-well plate” as the plate type.



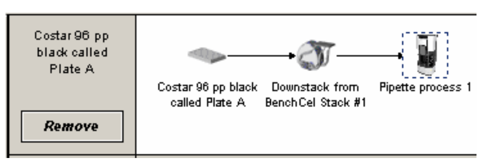
Downstacking a plate

The next step is to add a Downstack task that downstacks a plate from an appropriate stacker.



Adding a Pipette Process task

Next, you add a Pipette Process task.

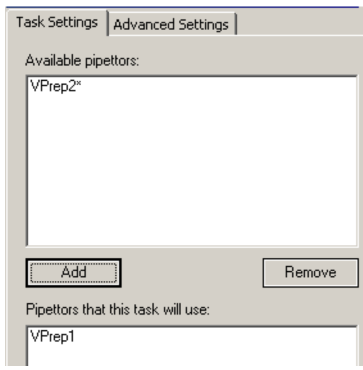


Because there can be a number of Pipette Process tasks in a protocol, this particular pipetting operation is identified by the name selected from the list box in the Pipette Task Parameters toolbar. In this simple case there is only one name to select, which is “Process1.”

Selecting a VPrep

In this example, there are two VPrep pipettors configured so you need to associate the one you want to use with the pipette process.

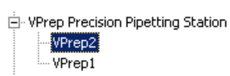
You do this in the Pipette Process Editor page. In the following example, the task will use VPrep1. VPrep2 remains available but is not used for this process.



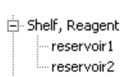
Configuring the VPrep shelf

You want to aspirate 20 µL of 1X TE buffer from a reservoir. This means that you have to configure a shelf of the VPrep to hold the reservoir that will contain the buffer. You decide to use a 384 V11 Reservoir (manual fill) 21.5 deep plate type for the reservoir.

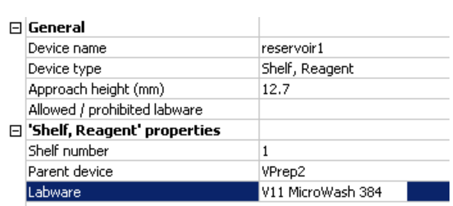
We know that the VPrep itself has already been configured in BenchWorks as a device because you were able to select it in the previous step. We can also see it listed as a device in the device manager. To open the device manager you click the Device Manager tab.



Each shelf that you want to use on the VPrep must also be configured as a device. The shelves are already set up as devices, but you need to make sure that shelf 1, where you want to place the buffer reservoir, is configured correctly. Again looking at the device manager you see that there are two shelves configured as reagent shelves, which can hold reservoirs.



On selecting reservoir 1 you see that it is assigned to shelf 1 of VPrep 2, which is what you want. However, the labware associated with the shelf is the wrong type.



If you leave it as V11 MicroWash 384, the pipette tips may crash into the reservoir because the task will be performed on the assumption that the tips are moving into a 384 V11 Reservoir (manual fill) 21.5 deep reservoir.

So, you change the labware association to 384 V11 Reservoir (manual fill) 21.5 deep.

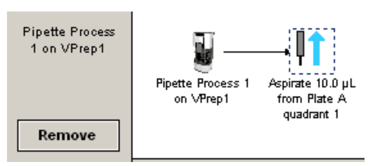
Note: To save the changes in the device manager you need to have administrator login privileges.

Note that when you associate a type of labware with the VPrep shelf, you are also associating all of the parameters for that type of labware stored in the labware database. The VPrep references the labware database parameters so that the pipette tips move to the right depth, position, and so on, as they enter the reservoir.

After configuring a VPrep shelf, compile the current protocol to check for errors.

Adding the Aspirate pipette task

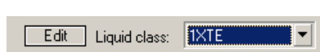
The next step is to add the Aspirate pipette task.



Associating the task with a liquid class

In the Pipette Task Parameters toolbar of the Aspirate pipette task, you need to tell the system what class of liquid it is aspirating. The system then uses the parameters stored in the liquid library database for that class during the aspiration operation.

In this case, you select the class 1XTE.

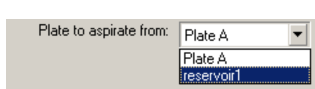


You can see the parameters used for the class by clicking the Edit button, which opens the liquid library editor.

Associating the task with the VPrep shelf

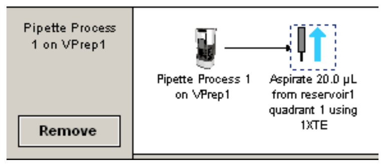
In the Pipette Task Parameters toolbar of the Aspirate pipette task, you need to tell the system from what type of labware to aspirate.

In this case there are two choices. If you select Plate A, the Costar 96-well plate you downstacked will be moved to the VPrep and the volume aspirated from it. Instead, you select reservoir1, which is the name of the device that holds the buffer reservoir.



Finally, you specify that we want to aspirate 20 µL.

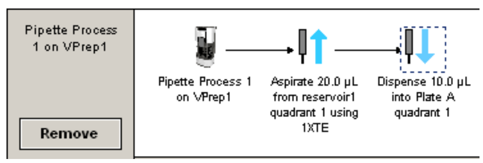
The modified task is shown in the following diagram:



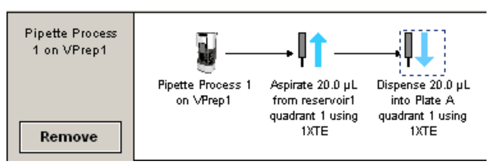
Adding a Dispense task

To complete the pipetting operation we have to add a Dispense task and set the parameters.

Drag the Dispense task into the pipette process pane.



The task correctly defaults to dispense into Plate A, but the dispense volume is incorrect and there is no associated liquid class. After editing the task parameters, the task is shown in the following diagram:



The protocol is now complete.

Related information

For information about...	See...
Setting up a plate instance	"Setting up a plate instance" on page 88
Liquid library editor	"About the liquid library editor" on page 244
Creating a pipette process task	"Adding and configuring a Pipette Process task" on page 140

Adding and configuring a Pipette Process task

About this topic

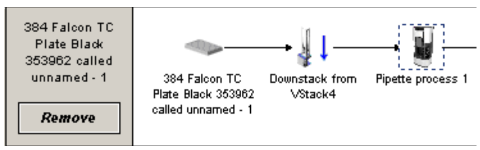
This topic describes how to configure a Pipette Process task. This task is used when creating a BenchWorks protocol that uses a Velocity11 VPrep.

Read this topic if you are:

- ❑ An administrator or technician who writes protocols
- ❑ An operator who may need to change Pipette Process parameters

Adding a Pipette Process task

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



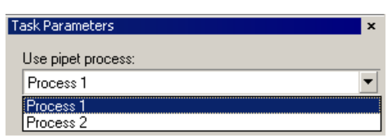
Setting Pipette Process parameters

When you add the Pipette Process task, a new pipette process is started in the pipette process editor. The pipette process is identified by its Pipette Process link icon.

Because you can have more than one pipette process in a protocol, you must link the Pipette Process task to the pipette process by setting the Pipette Process parameters.

To set the Pipette Process parameters:

1. In the **Protocol Editor** window, add a **Pipette Process** task to the protocol and then select it in the protocol sequence.
2. In the **Pipette Task Parameters** toolbar, select the pipette process that you want to use for this pipetting task.



3. If the pipette process is for a replicate pipetting series, so that the same plate can be used over and over again, select the **Use single instance of plates** check box in the **Task Setting** page of the plate icon.

   If the plate is a tip box, when this option is selected, the tips will be picked up and the tip box will be moved from the VPrep. At the end of the pipetting series, the tip box will be returned to the VPrep and the tips replaced in it.

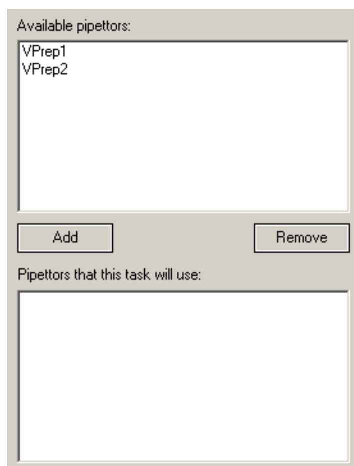
Associating the link icon

Because you can have more than one VPrep on a lab automation system, you must link each pipette process link icon with one or more VPreps that you want the pipette task to be able to use. You do this by setting the parameter for the pipette process link icon.

To link a Pipette Process task to a pipette process:

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

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



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

Related information

For information about...	See...
Creating a pipette process	“Configuring a pipette process: example” on page 136
The workflow that this procedure belongs to	“Workflow for creating a protocol” on page 85
Configuring VPrep shelves	“Configuring a VPrep shelf as a device” on page 142

Configuring a VPrep shelf as a device

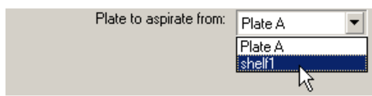
About this topic

All VPrep shelves have to be configured in BenchWorks as devices before they can be used in a protocol. All shelves on your VPrep were set up as devices at the factory. This topic shows you how to modify the existing settings for a shelf.

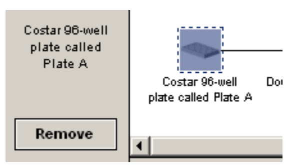
Read this topic if you are an administrator or technician who writes protocols that uses a VPrep.

Use this procedure when creating a protocol that uses a type of reservoir on the VPrep that is not currently set up or when a new type of shelf device is used in a protocol for the first time.

When you set parameters for some pipette tasks, you have to select the type of labware or device used in the task. The following screenshot shows both a plate and a device in the list box of an Aspirate pipette task.



The plate in the list box refers to Plate A in the associated process.



With Plate A selected, the robot will move the plate to the VPrep and liquid will be aspirated from it.

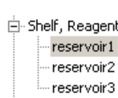
The device in the list box refers to a reagent shelf on the associated VPrep. With shelf 1 selected, the VPrep head will move to shelf 1 and the VPrep will aspirate from whatever type of labware sits on the shelf.

Before you can run the protocol, you have to associate a type of labware with the VPrep shelf. You do this in the device manager.

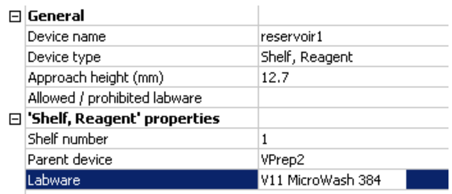
Procedure

To configure a VPrep reagent shelf as a device:

1. Click the **Device Manager** tab.
2. Select a reagent shelf in the **Device List**.

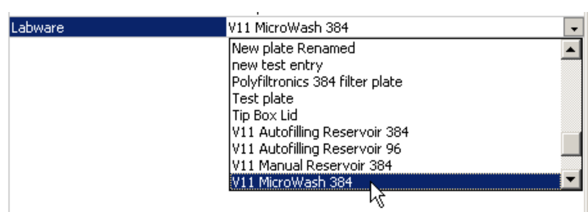


3. Make sure that:
   - The **Parent device** is the VPrep you intend to use.
   - The **Shelf number** is the shelf you intend to use (reagent shelves have odd numbers, with shelf 1 being at the top left).



   If these are not correct, select another reagent shelf in the **Device List**.

4. Optionally, change the **Device name** to one that describes the type of liquid being used by typing over the existing name.
5. Select the type of labware that will contain the reagent from the **Labware** list box.



6. Click the blank column to the right of **Allowed/prohibited labware**.
7. Click the ellipsis button.

   The **Labware Classes** dialog box opens.

8. Make sure that the labware you intend to use on this VPrep shelf is in the **Labware classes allowed to use this device** column and not in the prohibited column.
9. Select **File** > **Device File** > **Save** to save the device file.

Related information

For information about...	See...
Working with Device files	“Working with device files” on page 195
Labware editor	“About the labware editor” on page 215

Setting Aspirate pipette task parameters

About this topic

This topic describes how to set the Aspirate pipette task parameters. This task is used when creating a BenchWorks protocol that uses a Velocity11 VPrep.

Read this topic if you are:

- ❑ An administrator or technician who writes protocols
- ❑ An operator who needs to specify parameters for the Aspirate task

Aspirate pipette task defined

An Aspirate pipette task is used with a VPrep to draw up liquid from a plate or reservoir.

Before you start

Before you start setting the Aspirate task parameters, you need to associate a VPrep shelf with the labware type that will be used for the aspirating.

Procedure

To set Aspirate pipette task parameters:

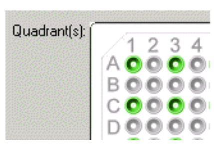
1. Add an **Aspirate** pipette task icon to the pipette process editor window.
2. If you have defined a liquid class for the liquid you intend to aspirate, select it from the **Liquid class** list box at the bottom of the **Pipette Task Parameters** toolbar.
3. Complete the following fields:

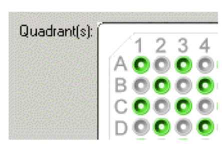
Field	Description
Aspirate volume	The volume of liquid to be drawn up into each pipette tip.
Aspirate velocity	The rate at which to draw up liquid. If you selected a liquid class, this value is entered automatically from the liquid library editor and cannot be edited here.
Aspirate acceleration	The rate of increase in velocity before the maximum aspirate velocity is reached. If you selected a liquid class, this value is entered automatically from the liquid library editor and cannot be edited here.
Distance from well bottom	The distance between the bottom of the pipette tips and the bottoms of the plate wells or MicroWash tray chimneys. If you are using dynamic tip retraction this value sets the lowest point to which the tips will travel.

Field	Description
Tip retract distance	The distance that the tips should move downwards per unit volume of liquid being aspirated. This value allows the tips to move downwards during aspiration to maintain a certain height below the surface of the liquid. You will need to determine an appropriate value by trial-and-error for each type of plate you use. You might want this value to be the same as the Tip Retract Distance for the Dispense pipette task.
Pre-aspirate volume	The volume of air to be drawn up before the pipette tips enter the liquid.
Post-aspirate volume	The volume of air to be drawn up after the liquid is drawn up.

4. If the VPrep head has fewer tips than the plate has wells, select a quadrant configuration from the **Quadrant(s)** diagram to indicate which well quadrant of the plate you want to aspirate from.

   To select a quadrant, click a representative well. Two possible examples are shown below.





5. In the **Plate to Aspirate from** list box, select the type of labware or device from which to aspirate.
6. If you do not want to record this dispense in the transfer log, clear the **Record in transfer log** check box.

   You might do this, for example, if you are running a casual test protocol.

Related information

For information about...	See...
Configuring VPrep shelves	"Configuring a VPrep shelf as a device" on page 142
Defining liquid handling parameters	"About the liquid library editor" on page 244
Labware editor	"About the labware editor" on page 215

For information about...	See...
Configuring a pipette process	❑ "Configuring a pipette process: example" on page 136 ❑ "Adding and configuring a Pipette Process task" on page 140

Setting Change Instance pipette task parameters

About this topic

This topic describes how to set the Change Instance task parameters. This task is used when creating a BenchWorks protocol that uses a Velocity11 VPrep.

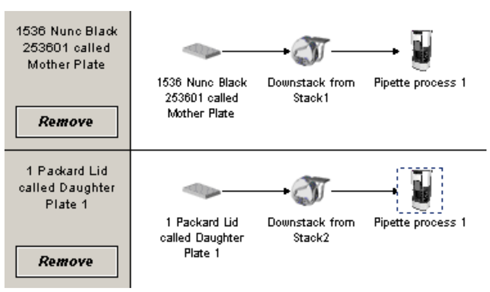
Read this topic if you are:

- ❑ An administrator or technician who writes protocols
- ❑ An operator who needs to specify parameters for the Change Instance task

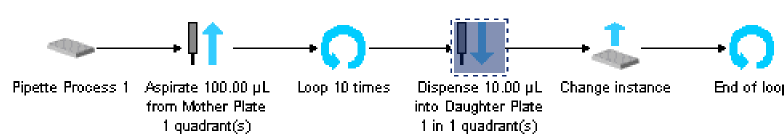
Change Instance pipette task defined

The Change Instance pipette task is used in combination with a pipette loop to perform replicate dispense operations from a mother plate into a series of daughter plates.

Part of an example protocol, which contains a mother plate and the first of 10 daughter plates, is shown below.



An example pipette process for the mother plate is shown below.



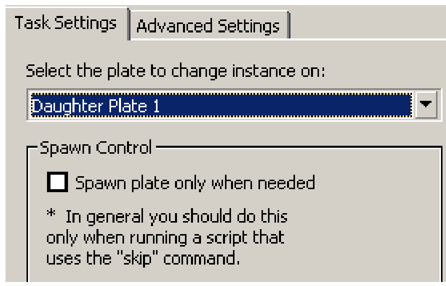
Note: This example has been simplified for the purposes of explanation.

When the protocol runs, 100 µl from each well of the mother plate are aspirated and 10 µl are dispensed into the first daughter plate (Daughter Plate 1). When the Change Instance task is reached, Daughter Plate 2 is moved into the system and 10 µl are dispensed into it from the mother plate. This loop repeats until all 10 daughter plates have been dispensed into.

Procedure

To set Change Instance pipette task parameters:

1. Set up a protocol and pipette processes following the example shown above.
2. In the **Pipette Task Parameters** toolbar, from the list box, select the first daughter plate for the plate icon to change instance on.



3. If you are using a JavaScript in the protocol that involves task skipping, check the **Spawn control** check box to force BenchCel to only bring in the next plate when the Change Instance task is reached.

   The script may otherwise override this flow.

Related information

For information about...	See...
Setting up a plate instance	"Setting up a plate instance" on page 88
Setting Loop task parameters	"Setting Loop pipette task parameters" on page 153
Configuring a pipette process	❑ "Configuring a pipette process: example" on page 136 ❑ "Adding and configuring a Pipette Process task" on page 140

Setting Change Tips pipette task parameters

About this topic

This topic describes how to set the Change Tips task parameters. This task is used when creating a BenchWorks protocol that uses a Velocity11 VPrep.

Read this topic if you are:

- ❑ An administrator or technician who writes protocols
- ❑ An operator who needs to specify parameters for the Change Tips task

Change Tips pipette task defined

The Change Tips pipette task uses the robot to apply or remove VPrep pipette tips.

Pairs of Change Tips pipette tasks are usually used together. For example, if the protocol starts with tips already on the VPrep, the first Change Tips pipette task would remove the tips and the second Change Tips pipette task would install new tips.

Change Tips pipette tasks are always entered in a process created solely for changing tips; an example is described in this topic.

Before you start

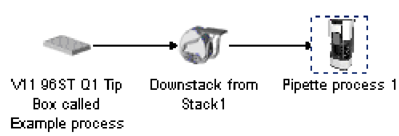
Before you start, you need to configure a shelf on the VPrep to use tip boxes and have ready a tip box, containing tips, with the tip box lid removed and placed in a robot-accessible position.

Also, if you are removing tips first, you need an empty tip box on the tip box shelf of the VPrep that you intend to use for the operation.

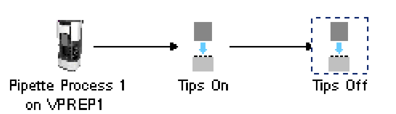
Overall process

Changing tips with a VPrep requires you to create the following processes:

1. In the protocol editor, create a process for the tip box, such as in the following example.



2. In the pipette process editor, create a pipette process to change tips, such as in the following simple example.



Creating the process for the tip box

Before you can add a Change Tips pipette task, you must create a process for the tip box that will contain the tips.

Creating a process for the tip box:

1. Click the **Protocol Editor** tab.
2. Click **Add**.

   A plate definition icon appears in the **Protocol Editor** window.
3. In the **Protocol Task Parameters** toolbar:
   a. Type a name for the tip box in the **Plate name** text box.

   b. In the **Plate type** list box, select the tip box that has already been configured for use on the shelf of the VPrep that you intend to use.

      Note: Make sure that you select and use a tip box without lids. If the correct tip box is selected, the Plates have lids check box is unavailable.

   c. In the **Simultaneous Plates** text box, type the maximum number of tip boxes of this type that you want to be available to the system at one time.

   d. If the tip boxes have a bar code on the south side or west side, select the appropriate **Incoming plates have a bar code**... check box and list box option.
4. Add the **Downstack** task to the Protocol Process window.
5. Set the **Downstack** task parameters.
6. Add the **Pipette Process** task to the Protocol Process window.
7. Set the **Pipette Process** task parameters.
8. Add any other tasks that you want to for the tip box.

   You could, for example, add an **Apply Label** task to place a bar code on the tip box.

Creating the pipette process for changing tips

After you have created a process for the tip box, create a pipette process for the Change Tips pipette task.

To create a pipette process for changing tips:

1. Click the **Pipette Process Editor** tab.
2. Add a **Change Tips** pipette task to the pipette process window.
3. In the **Pipette Task Parameters** toolbar, select either:

Option	With this option, during the protocol...
❑ Press On New Tips	Puts tips on to a VPrep head.
❑ Tips Off	Removes tips from a VPrep head.

4. From the **Select the tip box to use** list box, select the name of the tip box that you have assigned to the tip box process.

5. Add other pipette tasks, including a second **Change Tips** pipette task, as required.

Related information

For information about...	See...
How to set the number of simultaneous plates	"Setting the number of simultaneous plates" on page 91
Configuring VPrep shelves	"Configuring a VPrep shelf as a device" on page 142
Labware editor	"About the labware editor" on page 215
Applying labels	"Setting Apply Label task parameters" on page 102
Configuring a pipette process	❑ "Configuring a pipette process: example" on page 136 ❑ "Adding and configuring a Pipette Process task" on page 140

Setting Dispense pipette task parameters

About this topic

This topic describes how to set the Dispense pipette task parameters. This task is used when creating a BenchWorks protocol that uses a Velocity11 VPrep.

Read this topic if you are:

- ❑ An administrator or technician who writes protocols
- ❑ An operator who needs to specify parameters for the Dispense task

Dispense pipette task defined

A Dispense pipette task is used with a VPrep to dispense liquid into a plate.

Dispense value limits

You cannot dispense more volume than you aspirated.

If you enter a total dispense volume that is greater than the total aspirate volume you will get an error message when you compile the protocol. More specifically, the Aspirate Volume + Pre-Aspirate Volume + Post-Aspirate Volume must be greater than or equal to the Dispense Volume + Blowout Volume + Post Dispense Volume.

Before you start

Before you can set the Dispense pipette task parameters, you need to associate a VPrep shelf with the labware type from which you will aspirate.

Procedure

To set Dispense pipette task parameters:

1. In the pipette process window, click the **Dispense** pipette task icon.
2. If you have defined a liquid class for the liquid you intend to dispense, select it from the **Liquid class** list box at the bottom of the **Pipette Task Parameters** toolbar.
3. Either:
   - Type the volume that you want to move out of each pipette tip in the **Dispense Volume** text box.
   - Select the **Empty tips** check box.

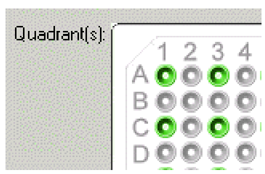
     You may need to drag the toolbar to widen it so you can see the check box. Select this option if you want to empty the tips rather than deliver a specific volume of liquid.
4. Complete the remaining fields:

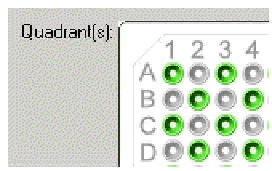
Field	**Description**
Dispense velocity	The rate at which to dispense the liquid. If you selected a liquid class, this value is entered automatically from the liquid library editor and cannot be edited here.
Dispense acceleration	The rate of increase in velocity before the Dispense Velocity is reached. If you selected a liquid class, this value is entered automatically from the liquid library editor and cannot be edited here.
Distance from well bottom	The distance between the bottom of the pipette tips and the bottoms of the plate wells or MicroWash tray chimneys. If you are using dynamic tip retraction this value sets the lowest point to which the tips will travel.
Retract distance	The distance that the tips should move upwards per unit volume of liquid being dispensed. This value allows the tips to move upwards during dispensing to maintain a certain height above the surface of the liquid. You will need to determine an appropriate value by trial-and-error for each type of plate you use. You might want this value to be the same as the Tip Retract Distance for the Aspirate pipette task.

Field	Description
Blowout volume	The volume of air to blow out when the tips are in the liquid. This is typically the same as the pre-aspirate volume. *Note:* Blowout only occurs in the last quadrant dispensed into for a given Dispense task.
Post-dispense volume	The volume of air to blow out when the tips are out of the liquid.

5. If the VPrep head has fewer tips than the plate has wells, select a quadrant configuration from the **Quadrant(s)** diagram to indicate which well quadrant of the plate you want to dispense to.

   To select a quadrant, click a representative well. Two possible examples are shown below.





6. If you want the tips to touch one or more sides of the plate wells:
   a. Select the **Enable tip touch** check box.
   b. Type a value for the **tip touch rise height**.

      This is the height that the tips should move upwards before touching the side of the wells.
   c. Type a value for the **Tip touch horiz distance**.

      When the value for this parameter is zero, the tips will move horizontally one well radius. The well radius is defined in the labware database for the type of plate you are using. If you want the tips to touch harder, increase this value. If you want the tips to touch more lightly, enter a negative value.
   d. In the **Number of sides to touch** text box, type a value for number of sides of the wells that you want the tips to touch.
7. In the **Plate to dispense to** list box, select the plate or device to dispense to.
8. If you do not want to record this dispense in the transfer log, clear the **Record in transfer log** check box.

   You might do this, for example, if you are running a casual test protocol.

Related information

For information about...	See...
Configuring VPrep shelves	"Configuring a VPrep shelf as a device" on page 142
Defining liquid handling parameters	"About the liquid library editor" on page 244
Labware editor	"About the labware editor" on page 215
Configuring a pipette process	❑ "Configuring a pipette process: example" on page 136 ❑ "Adding and configuring a Pipette Process task" on page 140

Setting Dry Tips pipette task parameters

Special note

The Dry Tips task is associated with the VPrep tip dryer. This tip dryer hardware is no longer available. If you have a tip dryer, contact the Velocity11 Service Center before using the Dry Tips task.

Setting Loop pipette task parameters

About this topic

This topic describes how to set the Loop task parameters. This task is used when creating a BenchWorks protocol that uses a Velocity11 VPrep and is illustrated with an example in which an Aspirate/Dispense pair of tasks is looped four times.

Read this topic if you are:

- ❑ An administrator or technician who writes protocols
- ❑ An operator who needs to specify parameters for the Loop task

Loop task defined

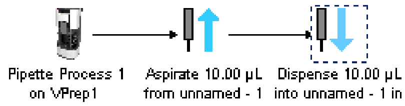
The Loop pipette task allows you to repeat a set of tasks within a process.

Procedure

To set Loop pipette task parameters:

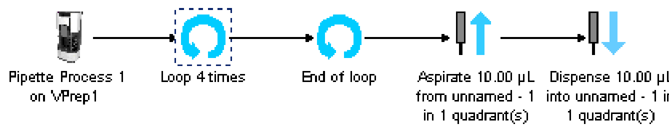
1. Open the **Pipette Process Editor** to display the process in which you want to add the **Loop** pipette task.

   An example is shown below.

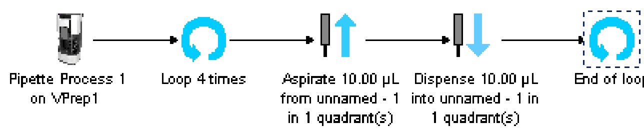


2. In the pipette process window, click to select the **Loop** pipette task icon and drag it into the process in front of the first task that you want to be in the loop.

   A **Loop** pipette task icon and **End of loop** pipette task icon are added to the process.



3. Drag the **End of loop** pipette task icon to the other side of the last task you want to be in the loop.



4. Select the **Loop** task pipette icon and in the **Pipette Task Parameters** toolbar, enter the number of times you want the tasks inside the loop to run.

Related information

For information about...	See...
The loop task in an example	“Setting Change Instance pipette task parameters” on page 146
Configuring a pipette process	❑ “Configuring a pipette process: example” on page 136 ❑ “Adding and configuring a Pipette Process task” on page 140

Setting Mix pipette task parameters

About this topic

This topic describes how to set the Mix pipette task parameters. This task is used when creating a BenchWorks protocol that uses a Velocity11 VPrep.

Read this topic if you are:

- ❑ An administrator or technician who writes protocols
- ❑ An operator who needs to specify parameters for the Mix task

Mix pipette task defined

The Mix pipette task is used with a VPrep to mix reagents by aspirating and then dispensing.

Before you start

Before you start, you must associate a VPrep shelf with the labware you will be using for the aspiration step of the mixing.

Procedure

To set Mix pipette task parameters:

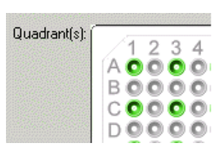
1. Add the **Mix** pipette task to the pipette process.
2. If you have defined a liquid class for the liquid you intend to mix, select it from the **Liquid class** list box at the bottom of the **Pipette Task Parameters** toolbar.
3. Complete the following properties:

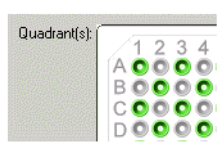
Property	Description
Mixing volume	The volume of liquid to be aspirated and dispensed to each plate well.
Number of mixing cycles	The number of aspirate/dispense operations.
Aspirate velocity	The rate at which to draw up liquid. If you selected a liquid class, this value is entered automatically from the liquid library editor and cannot be edited here.
Aspirate acceleration	The rate of increase in velocity before the maximum aspirate velocity is reached. If you selected a liquid class, this value is entered automatically from the liquid library editor and cannot be edited here.
Dispense velocity	The rate at which to dispense the liquid. If you selected a liquid class, this value is entered automatically from the liquid library editor and cannot be edited here.
Dispense acceleration	The rate of increase in velocity before the Dispense Velocity is reached. If you selected a liquid class, this value is entered automatically from the liquid library editor and cannot be edited here.

Property	Description
Distance from well bottom	The distance between the bottom of the pipette tips and the bottoms of the plate wells or MicroWash tray chimneys. If you are using dynamic tip retraction this value sets the lowest point to which the tips will travel.
Retract distance	The distance that the tips should move upwards or downwards per unit volume of liquid being dispensed or aspirated. This value allows the tips to move upwards or downwards during dispensing or aspirating to maintain a certain height below or above the surface of the liquid. You will need to determine an appropriate value by trial-and-error for each type of plate you use.
Pre-aspirate volume	The volume of air to be drawn up before the pipette tips enter the liquid, and before mixing begins.
Last-cycle blowout volume	The volume of air to blow out when the tips are in the liquid once the mixing is complete. This is typically the same as the pre-aspirate volume.

4. If the VPrep head has fewer tips than the plate has wells, select a quadrant configuration from the **Quadrant(s)** diagram to indicate in which well quadrant of the plate you want to mix.

   To select a quadrant, click a representative well. Two possible examples are shown below.





5. If you want the tips to touch one or more sides of the plate wells:

   a. Select the **Enable tip touch** check box.

   b. Type a value for the **Tip touch rise height**.

   This is the height that the tips should move upwards before touching the side of the wells.

   c. Type a value for the **Tip touch horizontal distance**.

   When the value for this parameter is zero, the tips will move horizontally one well radius. The well radius is defined in the labware database for the type of plate you are using. If you want the tips to touch harder, increase this value. If you want the tips to touch more lightly, enter a negative value.

d. In the **Number of sides to touch** text box, type a value for the number of sides of the wells that you want the tips to touch.

6. In the **Plate to mix** list box, select the type of labware or device to mix in.

Related information

For information about...	See...
Defining labware	"About the labware editor" on page 215
Configuring VPrep shelves	"Configuring a VPrep shelf as a device" on page 142
Configuring a pipette process	❑ "Configuring a pipette process: example" on page 136 ❑ "Adding and configuring a Pipette Process task" on page 140
Defining liquid handling parameters	"About the liquid library editor" on page 244

Setting Pump Reagent pipette task parameters

About this topic

This topic describes how to set the Pump Reagent pipette task parameters. This task is used when creating a BenchWorks protocol that uses a Velocity11 VPrep.

Read this topic if you are:

- ❑ An administrator or technician who writes protocols
- ❑ An operator who needs to specify parameters for the Pump Reagent task

Pump Reagent task defined

The Pump Reagent pipette task is used on a VPrep to pump liquid into an installed autofilling reservoir. Reservoirs are typically filled with washing buffer or water, and drained through the gravity drain.

!! IMPORTANT !! If you run an empty reservoir step and a fill reservoir step in the same protocol, check the protocol to make sure that it will not lead to an overflow.

Procedure

To fill a VPrep reservoir:

1. Add the Pump Reagent task to the pipette process.
2. In the **Pipette Task Parameters** toolbar, select **Fill reservoir**.

   The **Fill reservoir** and **Empty reservoir** values determine whether the pumps will fill or empty the reservoir.

   To empty the reservoir you must complete the **Autofill Configuration** information on the **Shelves** tab of the VPrep Diagnostics.
3. In the list box, select the shelf on which the reservoir is located.
4. In the **for** text box, type the pumping duration.

   This is the time in seconds that the pumps pump.
5. In the **at** text box, type the percentage of maximum pumping rate.

   This, combined with the pumping duration, determines the volume of fluid moved.
6. In the **every** text box, type a number that controls how frequently the liquid is pumped.

   For example, if you type `3`, the pump will run every third time the task runs.
7. If you are using a Weigh Shelf, in the **If liquid is below this level** text box, enter the minimum percentage of liquid that you want the reservoir to contain.

   A typical value is 45%.
8. If you are using a Weigh Shelf, in the **then fill reservoir to this level** text box, enter the maximum percentage of liquid that you want the reservoir to contain.

   A typical value is 60%.

Related information

For information about...	See...
Defining labware	“About the labware editor” on page 215
Configuring VPrep shelves	“Configuring a VPrep shelf as a device” on page 142
Configuring a pipette process	❑ “Configuring a pipette process: example” on page 136 ❑ “Adding and configuring a Pipette Process task” on page 140
Defining liquid handling parameters	“About the liquid library editor” on page 244

Setting Wash Tips pipette task parameters

About this topic

This topic describes how to set the Wash Tips pipette task parameters. This task is used when creating a BenchWorks protocol that uses a Velocity11 VPrep.

Read this topic if you are:

- ❑ An administrator or technician who writes protocols
- ❑ An operator who needs to specify parameters for the Wash Tips task

Wash Tips pipette task defined

A Wash Tips pipette task is used with a VPrep to wash pipette tips.

Before you start

Before you start, you need to have a VPrep shelf associated with the labware type with which you want to wash.

Procedure

To wash pipette tips:

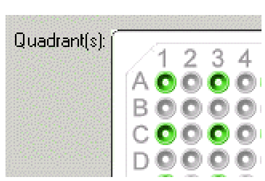
1. Add the Wash Tips task to the pipette process.
2. If you have defined a liquid class for the liquid you intend to use for washing, select it from the **Liquid class** list box at the bottom of the **Task Parameters** toolbar.
3. Complete the following properties:

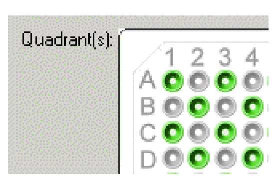
Property	Description
Wash volume	The volume of liquid to be aspirated and dispensed to each plate well.
Dispense only	Select this if you want to dispense the wash liquid to waste instead of dispensing it back into the reservoir of washing liquid. *Note:* The **Dispense to waste at height of** check box must be selected for this option to be available. Enter the amount to dispense in the **Wash Volume** field.
Empty tips	This option is only available if you select **Dispense only**. Select this if you want to empty the tips, regardless of the volume.
Number of wash cycles	The number of aspirate/dispense operations.
Aspirate velocity	The rate at which to draw up the liquid. If you selected a liquid class, this value is entered automatically from the liquid library editor and cannot be edited here.

Property	Description
Aspirate acceleration	The rate of increase in velocity before the aspirate velocity is reached. If you selected a liquid class, this value is entered automatically from the liquid library editor and cannot be edited here.
Dispense velocity	The rate at which to dispense the liquid. If you selected a liquid class, this value is entered automatically from the liquid library editor and cannot be edited here.
Dispense acceleration	The rate of increase in velocity before the dispense velocity is reached. If you selected a liquid class, this value is entered automatically from the liquid library editor and cannot be edited here.
Distance from well bottom	The distance between the bottom of the pipette tips and the bottoms of the MicroWash tray chimneys.
Retract Distance	The distance that the tips should move upwards or downwards per unit volume of liquid being dispensed or aspirated. This value allows the tips to move upwards or downwards during dispensing or aspirating to maintain a certain height below or above the surface of the liquid. You will need to determine an appropriate value by trial-and-error.
Pre-aspirate volume	The volume of air to be drawn up before the pipette tips enter the liquid, and before mixing begins.
Last-cycle blowout volume	The volume of air to blow out when the tips are in the liquid once the mixing is complete. This is typically the same as the pre-aspirate volume.

4. If the VPrep head has fewer tips than the plate has wells, select a quadrant configuration from the **Quadrant(s)** diagram to indicate which well quadrant of the plate you want to dispense to.

   To select a quadrant, click a representative well. Two possible examples are shown below.





5. Type a value for the **Inflow pump**, which is the relative rate of liquid flow into the MicroWash tray manifold.

   This value should be high enough for the washing liquid to just bubble over the tops of the chimneys.

6. Type a value for the **Outflow pump**, which is the relative rate of liquid flow out of the MicroWash tray manifold.

   This value should be zero.

7. If you want to dispense the wash liquid to waste instead of dispensing it back into the reservoir of washing liquid:

   a. Select the **Dispense to waste at height of** check box.

   b. Type a value into the associated text box for the height above the chimney from which you want the liquid to be dispensed.

      The value can be a positive or negative number.

      The pipette tips move up and sideways to dispense the wash liquid between the chimneys into waste.

8. If you want the tips to touch the outside tops of the chimneys to remove drops from the tips, select the **Enable tip touching** check box:

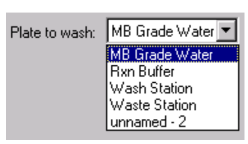
   a. Type a value for the **Tip touch rise height**.

      This is the height that the tips should move upwards before touching the side of the wells.

   b. Type a value for the **Tip touch horizontal distance**.

      When the value for this parameter is zero, the tips will move horizontally one well radius. The well radius is defined in the labware database for the type of plate you are using. If you want the tips to touch harder, increase this value. If you want the tips to touch more lightly, enter a negative value.

9. In the **Plate to wash** list box, select the VPrep and shelf position of the MicroWash tray.



   If the name that you chose does not appear in this list, you probably associated the VPrep shelf with the labware type *after* adding the Wash tips pipette task to the pipette process. Remove the task and add it again for the choice to appear.

Washing the MicroWash tray manifold

To wash the MicroWash tray manifold:

1. Create a protocol process that contains only a pipette process.
2. Add a **Wash Tips** task to the pipette process.
3. Enter the following values for the task:

Property	Value
Wash volume	0
Number of wash cycles	0
Inflow pump	Typically set below 100%. The actual rate of inflow depends on the viscosity of the liquid and the height of the reservoir above the Microwash tray. It is best to observe the height of the fluid in the tray and set the inflow pump value so that there is an even flow of liquid and the height does not rise to cause an overflow.
Outflow pump	Typically set to 100%

4. Run the process.

Related information

For information about...	See...
Defining labware	"About the labware editor" on page 215
Configuring VPrep shelves	"Configuring a VPrep shelf as a device" on page 142
Configuring a pipette process	❑ "Configuring a pipette process: example" on page 136 ❑ "Adding and configuring a Pipette Process task" on page 140
Defining liquid handling parameters	"About the liquid library editor" on page 244

Creating a protocol: advanced topics

8

This chapter is intended for people with technician or administrator privileges. It provides the background information necessary to set up certain tasks.

Before reading this chapter you should be familiar with the concepts presented in the following chapters:

- “Creating a protocol: basics” on page 83
- “Setting task parameters” on page 99
- “Setting pipette task parameters” on page 135

Note: This chapter is not a tutorial on writing protocols—it provides the basic reference information you will need to write protocols.

Setting up the LabwareSelector plug-in

About this topic

The LabwareSelector plug-in allows you to assign a plate type at the time you start your run.

Use this plug-in if your laboratory is using many plate types with the same protocols.

Procedure

To set up the LabwareSelector plug-in:

1. Select the plate icon (the first icon in a protocol process).
2. In the **Task Settings** page of the **Protocol Task Parameters** toolbar, select **LabwareSelector.dll** from the **Plugin** list box.
3. Select **<From Plugin>** from the **Plate type** list box.

Note: The software defaults to whatever is selected in the **Plate type** field. For example, if you have the **LabwareSelector.dll** selected as a **Plugin** and **Costar 99 pp black** selected as the **Plate type**, the software will execute the run with **Costar 99 pp black** as the plate type and will ignore the plugin.

Related information

For information about...	See...
The plate icon	"About tasks, processes, and protocols" on page 37

About the FileReader plug-in

About this topic

Velocity11 has created a plug-in for BenchWorks, called FileReader.dll. This topic describes this plug-in.

The FileReader plug-in lets BenchWorks read from a tab-delimited or a CSV (comma separated values) file to specify the content of label fields printed with a VCode.

Read this topic if you are a technician or an administrator who writes protocols with Apply Label (VCode) tasks and who wants the FileReader plug-in to read and process the label text.

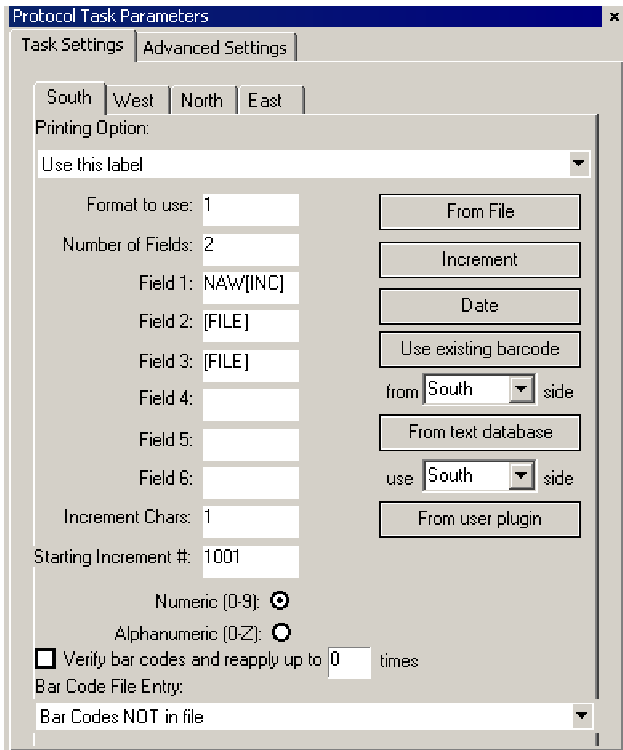
Defining label field content

The Apply Label task provides several ways to define the contents of the label fields printed on a series of plates during a run. However, without plug-ins and scripting, for each label in a given run, the Apply Label task *cannot* apply:

- ❑ Two different pieces of data from the same file
- ❑ Two different increments for two fields

❑ Numeric increments for one field and alphanumeric increments for another different field

In the following screenshot, Field2 and Field3 use the same label input file for the data in the fields. However, this function is limited because there is no way to specify two different locations in the same file for the same label.



The FileReader plug-in allows BenchWorks to input the data in the label fields from a text file. Functionally, the text file is similar to a label data file except that it includes data for all fields in a label instead of just one field. This allows two fields on the same label to contain different data from the same file in the same run, without using prefixes and suffixes.

Related information

For information about...	See...
Setting up the file reader plug-in	"Using the FileReader plug-in in a protocol" on page 168
Using JavaScript with BenchWorks	"Using JavaScript in BenchWorks" on page 171
The workflow that this procedure belongs to	"Workflow for creating a protocol" on page 85

About the FileReader file format

About this topic

This topic describes the format of the text file that is read by the FileReader plug-in.

Read this topic if you are a technician or administrator who writes protocols with Apply Label (VCode) tasks and who want to use the FileReader plug-in to read and process the label text.

The header row

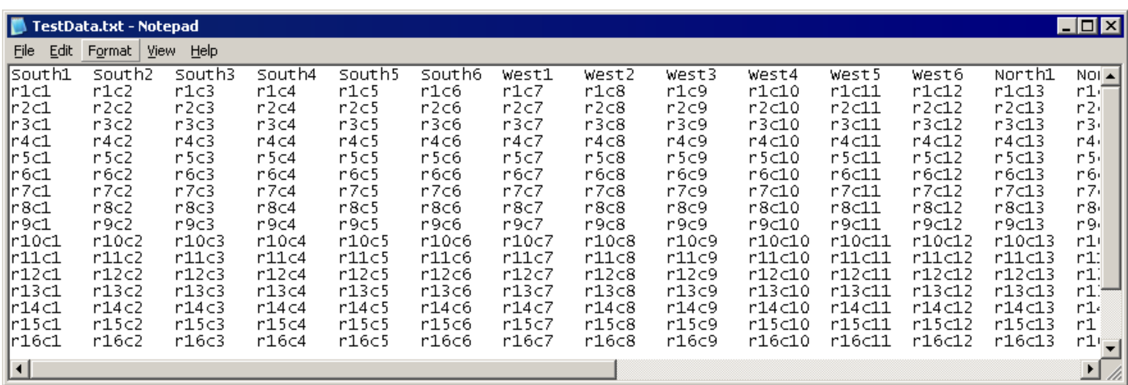
The first row in the text file must contain a header row.

The header row is a human-readable guide to show where the contents of each column will print.

The FileReader plug-in will assume that the rest of the text contains the same text separation method as the header row.

For example, if the header row uses a CSV format but the rest of the file uses a tab-delimited format, the FileReader plug-in expects commas to separate the values in the file. Having found none, it assumes that the entire text for each row after the header row is one field.

A screenshot of a FileReader plug-in text file is shown below. This example is a tab-delimited text file, but a CSV text file could also be used.



South1	South2	South3	South4	South5	South6	West1	West2	West3	West4	West5	West6	North1	No
r1c1	r1c2	r1c3	r1c4	r1c5	r1c6	r1c7	r1c8	r1c9	r1c10	r1c11	r1c12	r1c13	r1
r2c1	r2c2	r2c3	r2c4	r2c5	r2c6	r2c7	r2c8	r2c9	r2c10	r2c11	r2c12	r2c13	r2
r3c1	r3c2	r3c3	r3c4	r3c5	r3c6	r3c7	r3c8	r3c9	r3c10	r3c11	r3c12	r3c13	r3
r4c1	r4c2	r4c3	r4c4	r4c5	r4c6	r4c7	r4c8	r4c9	r4c10	r4c11	r4c12	r4c13	r4
r5c1	r5c2	r5c3	r5c4	r5c5	r5c6	r5c7	r5c8	r5c9	r5c10	r5c11	r5c12	r5c13	r5
r6c1	r6c2	r6c3	r6c4	r6c5	r6c6	r6c7	r6c8	r6c9	r6c10	r6c11	r6c12	r6c13	r6
r7c1	r7c2	r7c3	r7c4	r7c5	r7c6	r7c7	r7c8	r7c9	r7c10	r7c11	r7c12	r7c13	r7
r8c1	r8c2	r8c3	r8c4	r8c5	r8c6	r8c7	r8c8	r8c9	r8c10	r8c11	r8c12	r8c13	r8
r9c1	r9c2	r9c3	r9c4	r9c5	r9c6	r9c7	r9c8	r9c9	r9c10	r9c11	r9c12	r9c13	r9
r10c1	r10c2	r10c3	r10c4	r10c5	r10c6	r10c7	r10c8	r10c9	r10c10	r10c11	r10c12	r10c13	r1
r11c1	r11c2	r11c3	r11c4	r11c5	r11c6	r11c7	r11c8	r11c9	r11c10	r11c11	r11c12	r11c13	r1
r12c1	r12c2	r12c3	r12c4	r12c5	r12c6	r12c7	r12c8	r12c9	r12c10	r12c11	r12c12	r12c13	r1
r13c1	r13c2	r13c3	r13c4	r13c5	r13c6	r13c7	r13c8	r13c9	r13c10	r13c11	r13c12	r13c13	r1
r14c1	r14c2	r14c3	r14c4	r14c5	r14c6	r14c7	r14c8	r14c9	r14c10	r14c11	r14c12	r14c13	r1
r15c1	r15c2	r15c3	r15c4	r15c5	r15c6	r15c7	r15c8	r15c9	r15c10	r15c11	r15c12	r15c13	r1
r16c1	r16c2	r16c3	r16c4	r16c5	r16c6	r16c7	r16c8	r16c9	r16c10	r16c11	r16c12	r16c13	r1

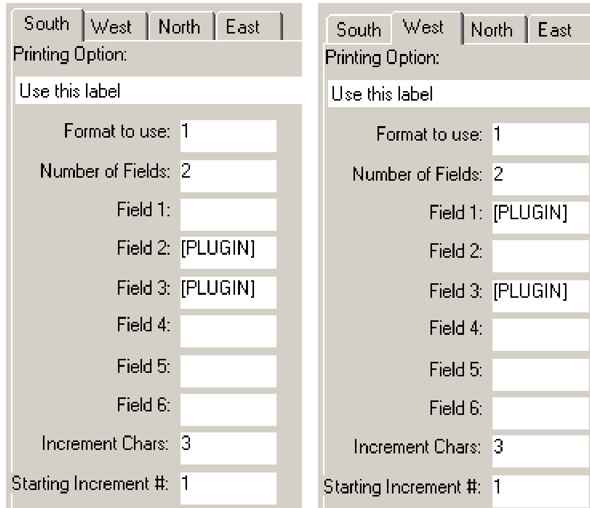
The order of the columns

Note how the order of the columns references the order of the tabs in the Apply label task Protocol Task Parameters toolbar.

The six columns of a particular side correspond to the six fields in the protocol task parameters.

The column “South3” is the column that contains the root data that will be printed on the south side of the plate in field 3. Each row of the table represents a different cycle in the run. The first row contains the root data that will be printed during the first run cycle, and so on.

The following screenshots show how the same plug-in is used in a total of four fields, on two sides of the plate.



Using these parameters, data from the above plug-in file would print the following labels:

Field	Printed labels in run cycle 1
South, Field 2	r1c2
South, Field 3	r1c3
West, Field 1	r1c7
West, Field 3	r1c9

Repeating columns

A text file with data for labels on only one side of the plate is sufficient to print the same labels on other sides of the plate.

To understand this, think of the columns as occurring in six column sets. If there are fewer than six columns, the remaining columns will be left blank.

If there is only one set in the file, the set is repeated for the other sides of the plate where you have selected “Use this label” from the list box.

Related information

For information about...	See...
Setting up the file reader plug-in	“Using the FileReader plug-in in a protocol” on page 168
Using JavaScript with BenchWorks	“Using JavaScript in BenchWorks” on page 171

Using the FileReader plug-in in a protocol

About this topic

The FileReader plug-in is installed and registered during the standard BenchWorks installation. You should see the FileReader.dll file in your Velocity11\BenchWorks\plugins directory.

This topic describes how to modify the task parameters in the protocol so that the Apply Label task uses the FileReader plug-in.

Read this topic if you are a technician or administrator who writes protocols with Apply Label (VCode) tasks and who wants the FileReader plug-in to read and process the label text.

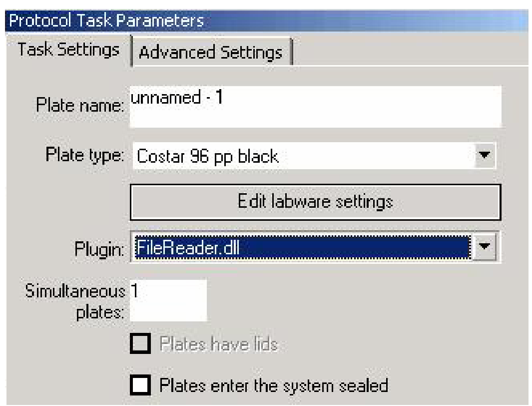
Setting up the protocol

Create a protocol, and then modify the plate icon and Apply Label task parameters as described here.

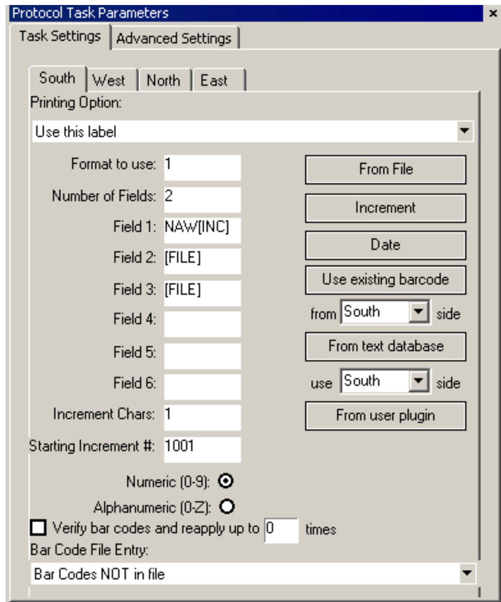
To set up the protocol to use the File Reader plug-in:

1. Select a plug-in to use:
   a. Click the plate icon in the **Protocol Editor**.
   b. Select FileReader.dll from the **Plugin** list box of the **Protocol Task Parameters** group box.

      If the FileReader.dll is not available for selection, it is because the FileReader.dll file is missing from the plug-ins folder.

2. Populate the **Apply label** task fields:
   a. Click the **Apply label** icon in the **Protocol Editor**.
   b. Populate the **Apply label** task fields with the **From user plugin** button.



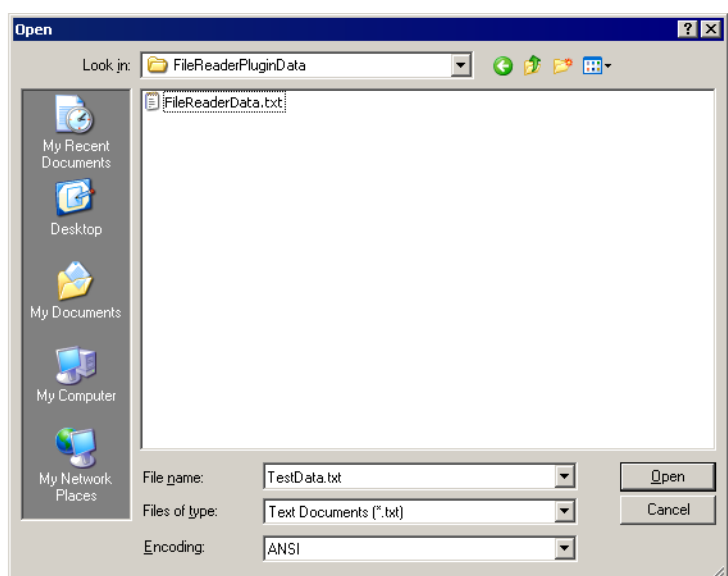
c. If you would like to use prefixes or suffixes, enter them before or after **[PLUGIN]** in the relevant **Field** text box.

An example field with a prefix is shown in the following screenshot.

Running the protocol

To run a protocol with a FileReader plug-in file:

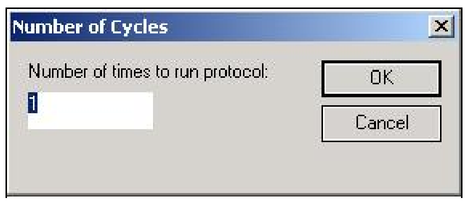
1. Open the protocol that you created to use the FileReader plug-in.
2. Click **Start** from BenchWorks to start the protocol.
3. Select the plug-in text file from the dialog box that opens.



4. In the **FileReader** dialog box that opens, check the list of names of the columns of the text file and the total number of rows in the file.

5. Click **OK**.

   The **Number of Cycles** dialog box opens.



6. Enter the number of cycles to run.

   If the file is not the one you intend to use, you can cancel the run.

   If the number of rows in the file is greater than the number of cycles that are run, the extra rows will be unused.

   If the number of rows is less than the number of cycles that are run, an error occurs when the rows are executed. If the error is ignored, additional labels are left blank.

   The input file does not change during a run, so if you perform another run with the same file, the same labels will print.

   If a power outage occurs during a run, and you are unable to resume the run, delete the first few rows after the header row. Then run the protocol again.

7. Click **OK**.

Related information

For information about...	See...
Using JavaScript with BenchWorks	“Using JavaScript in BenchWorks” on page 171
The workflow that this procedure belongs to	“Workflow for creating a protocol” on page 85

Using JavaScript in BenchWorks

About JavaScript in BenchWorks

In BenchWorks, JavaScript programs (scripts) can be used to:

- ❑ Configure tasks in ways that task parameters do not allow
- ❑ Change the parameters of a protocol task immediately before it is scheduled

This extends the capability of BenchWorks because the parameters can be changed dynamically during a run, based on:

- ❑ 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

Scripts can be run as part of pre-protocols, protocols, and post-protocols.

About JavaScript

JavaScript is a general-purpose programming language that requires an interpreter to run its programs.

You are probably most familiar with JavaScript where it is used to create dynamic effects in Web pages. This form of JavaScript is made up of a core language plus Web browser-specific language. It is processed by the JavaScript interpreter that is built into modern browsers.

The core JavaScript language can be used to write scripts that have nothing to do with web pages. These scripts can be used for any application that includes a JavaScript interpreter. BenchWorks is an example of such an application—it uses a JavaScript 1.5 interpreter.

JavaScript resources

There are many JavaScript resources available online and in print. If you want to learn more about JavaScript for use in BenchWorks, look for resources that cover the core JavaScript language separately from the browser, client-side language, and Document Object Model.

Web references

You can find useful information at:

http://www.mozilla.org/js/

Print reference

A good print reference is *JavaScript: The Definitive Guide*, Fourth Edition, published by O'Reilly.

Examples of use

You can use JavaScript to:

- ❑ Print the parameters of a task to the BenchWorks log
- ❑ Run a command line that launches an external application, such as a batch file or database updating program
- ❑ Simplify protocol writing, for example, by incrementing pipetting volumes each cycle of a protocol to perform a dilution series

Where scripts are written

Scripts can be written in two *places*:

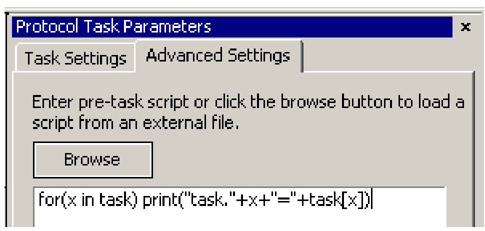
- ❑ Pre/post protocol scripts
- ❑ The Advanced Settings tab

Scripts can be written in two *ways*:

- ❑ Directly into the text box
- ❑ As an external file that is located by clicking Browse

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

The following screenshot shows 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 text box.



About BenchWorks-defined functions and objects

The BenchWorks interpreter supports the JavaScript 1.5 core functions and objects. Velocity11 has also defined its own functions and objects that can be used in BenchWorks scripts.

BenchWorks-defined global functions

The following BenchWorks-defined functions are available globally, meaning that they are not restricted to a particular object or programming context.

Function	Description
print()	Prints time-stamped messages to the BenchWorks log. Parameter: Text string Example: `print(plate.name)`
open()	Opens a file. Parameter: Text string Example: `open("c:\benchworks workspace\benchworks3\text.txt")`

Function	Description
run()	Runs a program as though it is being called from a command line. Parameters: ❑ Text string. Required. Allows you to initiate a command that you could otherwise enter into the Windows Run dialog box, such as `notepad text.txt` (opens a file named text.txt in Windows Notepad). ❑ Boolean True/False. Optional. Default is False. If True, BenchWorks waits for the function to complete before continuing (blocking).

BenchWorks-defined objects

The BenchWorks JavaScript interpreter provides two objects that can be accessed by a script. They are the plate object and task object.

Plate object

The plate object provides access to properties of the plate that the current task is operating upon.

Properties

The plate object has the following properties:

Property	Data type	Description
plate.name	String	Name of the plate.
plate.instance	Integer	Plate instance number.
plate.labware	String	Name of the labware type.
plate.barcode	Array	Array of four strings corresponding to SOUTH=0, WEST=1, NORTH=2, EAST=3. Example: `plate.barcode[SOUTH]= "mybarcode"`
plate.volume	Array of arrays	An array of floating point numbers. The array size depends on the number of wells in the labware (96, 384, or 1536), arranged in row, column format. This property is only enabled on BenchCel systems that have the volume-tracking database option.

Methods

Methods are JavaScript functions invoked through an object.

The plate object has the following methods, available on those systems with the volume-tracking database option.

Method	Comments
plate.setUserData(string key, string value)	Stores 'value' under the key 'key' in a database record associated with this plate
plate.getUserData(string key)	Returns the 'value' stored earlier using plate.setUserData

Task object

The task object is a BenchWorks-defined generic object that refers to the currently executing task. It allows the properties of the task to be accessed using a standard syntax. Depending on which task is executing, a different set of properties may be available.

Properties

The task object provides a comprehensive set of properties that can be read/write, or read-only. These properties specifically affect the behavior of the task that is about to be executed by the BenchWorks scheduler.

For example, the Aspirate task has a property called "volume". To store this property in a variable you would write:

```
x = task.volume
```

To set the volume property of the Aspirate task to the value stored in the variable 'x', you would write:

```
task.volume = x
```

In this example, the run-time interpreter determines through the context that 'task.' refers to the currently executing Aspirate task.

Attempting to access properties that are inappropriate for the current task will result in a scripting syntax error, but will not halt the execution of your protocol.

Methods

The task object has the following methods:

Method	Comments

task.skip()	Skips execution of the current task. Use this function to conditionally execute a task, such as in this example which skips the task if the simulator is not running:
	`if (!isSimulatorRunning())`
	`task.skip()`

task.pause()	Pauses the protocol and opens a dialog box that asks you whether you want to continue or abort the run. Use this function if you need to pause the protocol to, for example, replenish the fluid in a static reservoir. You could use the print() function to add a note to the log toolbar describing the action to take when the BenchCel has paused.
task.isSimulatorRunning()	Returns true if this is a simulated run. Has no arguments.
task.repeat()	Schedules the task to be repeated.

None of the task object methods accept any parameters.

These are generic methods that are the same regardless of the task that is executing them. The properties of these methods are specific to the current task.

About variables

The default behavior of BenchWorks JavaScript is that the values of all variables are cleared (set to undefined) before the next protocol is run.

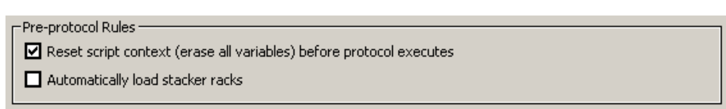
You have the option to change this so that the value of a variable assigned in a script is held in memory until BenchWorks is closed. This means that if you assign the value to a variable in one protocol, the same value will be used in the next protocol with the same variable. This is the reset script context feature.

To clear the reset variables default option:

1. Select **Tools** > **Protocol Options**.

   The **Protocol Options** dialog box opens.

2. In the **Pre-Protocol Rules** group box, clear the **Reset script context (erase all variables) before protocol executes** check box.

Quadrant representation

In JavaScript, an array is a built-in object that stores a collection of like values, called elements. Each element is accessed by an index value that is enclosed in square brackets. Index values can be non-negative integers or strings.

The following example script declares an array with three elements:

```
var vehicle_type = new Array(3);
vehicle_type[0] = “car”;
vehicle_type[1] = “truck”;
vehicle_type[2] = “van”;
```

Because a plate is already an array of wells, locations on plates (quadrants) are represented in Velocity11 JavaScript as an array of arrays. for example, the quadrant property of the task object for one task is represented as:

```
[[1,1]]
```

In this representation, the first number refers to the plate row and the second number refers to the plate column. These numbers can be represented by variables in a script, as shown in the following statement.

```
task.quadrants = [[disp_row,disp_column]]
```

For two quadrants, the representation would be:

```
[[1,1],[1,2]]
```

Cautions

When you run a script that dynamically changes the values of task properties, there is a risk that a value will be set that causes a problem. We therefore recommend that before using a script, you run the simulator with each set of values that will be set by the script. Running scripts cannot cause robot crashes, because scripts cannot modify teachpoints. However, an incorrect task.tipOffset property (Distance from well bottom parameter) on a VPrep could cause the pipette tips to crash into the bottom of the wells resulting in loss of sample and damage to plates.

In addition, be aware that when a protocol is being compiled, it uses the values displayed in the Protocol Task Parameters toolbar screen and not the values that will be set by any scripts. This means that there may be errors in the protocol that are not detected during compilation. The values that appear in the Task Parameter toolbar do not change to reflect the effect of any script.

Also, scripts do not check pipetting volumes before the run begins, so you must make sure that the pipetting steps make logical sense. For example, you will not be alerted beforehand if a script will attempt to aspirate 1 mL from a plate well that can only hold 0.5 mL.

Example scripts

Example 1

This script prints the word “hello” to the log toolbar and log.txt file.

```
print("hello");
```

Example 2

This one-line script opens an external file that could contain another script. The new script is run immediately.

```
open("C:\scripts\script1.txt")
```

Example 3

This script prints a list of the properties for the task to the log toolbar. It is an essential part of determining the names of properties when creating JavaScripts.

```
for(x in task) {
   print("task[" + x + "]=" + task[x]);
}
```

Script-writing service

Velocity11 offers a custom script-writing service for BenchWorks and other applications. Please contact us for more information.

Related information

For information about...	See...
The properties for each available task in BenchWorks, version 18	“The JavaScript task object and properties” on page 178
The workflow that this procedure belongs to	“Workflow for creating a protocol” on page 85

The JavaScript task object and properties

About this topic

The BenchWorks JavaScript interpreter includes a task object that is defined by Velocity11.

This topic lists the properties for the Velocity11 JavaScript task object. One of the properties is for the Apply Label task. The other properties are all for pipette tasks.

Task properties

Properties available to all tasks

The following properties can be used for any task.

Property	Data type	Description
task.name	String	Name of the task, for example, "Aspirate"
task.description	String	Description of the task that is given under the icon in the protocol editor. For example, a downstack task that has the script `print(task.description)` will send the following text to the protocol log: Downstack from stacker2

Apply Label task

The JavaScript Apply Label task properties are listed below, along with the data type of the property and the names of the corresponding Apply Label task parameters.

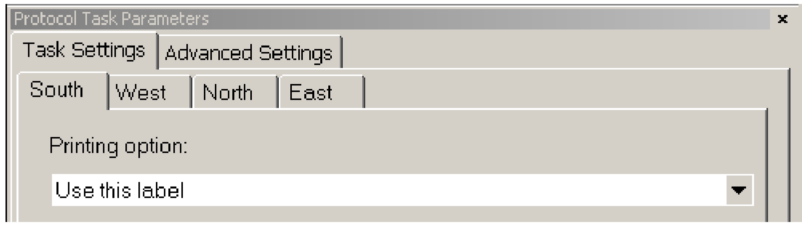
The task.side property is an array of four label_data objects:

- task.side[SOUTH]
- task.side[EAST]
- task.side[NORTH]
- task.side[WEST]

Each of these task.side properties has nine properties, representing the fields on the Protocol Task Parameters toolbar for the Apply Label task.

In the table below, *point* can be replaced by SOUTH, EAST, NORTH, or WEST. For example, the Printing option field for the south label (see screenshot) is represented as:

task.side[SOUTH].printLabel



Property	**Data type**	**Task parameter(s)**	**Comments**
task.side[*point*].field	An array of six strings	Field 1, Field 2, Field 3, Field 4, Field 5, Field 6	For example, task.side[WEST].field[2] matches the Field 2 text box in the West tab of the Protocol Task Parameters toolbar.
task.side[*point*].format	Integer	Format to use	A number that corresponds to the bar code format that you want. For information about formats, see the *VCode User Guide*.
task.side[*point*].incrementChars	Integer	Increment chars	The number of alphanumeric characters that you want to be appended to the root data.
task.side[*point*].startingIncrement	Integer	Starting increment #	The number that you want to be printed on the first label.
task.side[*point*].base	Integer	Numeric (0-9) Alphanumeric (0-Z)	0 for numeric increments 1 for alphanumeric increments
task.side[*point*].verifyBarcode	Integer	Verify bar codes	0 for no bar code verification 1 for bar code verification
task.side[*point*].maxVerifyAttempts	Integer	Reapply up to __ times	The number of attempts made to verify a bar code.
task.side[*point*].sourceBarcodeSide	Integer	Use existing bar code from ____ side	Copies the bar code from this side of the plate 0 = South 1 = West 2 = North 3 = East
task.side[*point*].printLabel	Integer	Printing option	0 = No label 1 = Use this label 2 = South 3 = West 4 = North 5 = East

Aspirate task

The JavaScript Aspirate task properties are listed below, along with the data type of the property, the names of the corresponding aspirate properties in the Protocol Task Parameters toolbar, and a reference to more information.

Property	Data type	Task parameter	Comments
task.plateName	String	Plate name	The name of the plate. Read only.
task.acceleration	Float	Aspirate acceleration	The rate of increase in velocity before the maximum aspirate velocity is reached. If you selected a liquid class, this value is entered automatically from the liquid library editor.
task.liquidClass	String	Liquid class list box	The name of the liquid class.
task.postAspirateVolume	Float	Post aspirate volume	The volume of air to be drawn up after the liquid is drawn up.
task.preAspirateVolume	Float	Pre aspirate volume	The volume of air to be drawn up before the pipette tips enter the liquid.
task.quadrants	An array of pairs of integers	Quadrant selection diagram	A quadrant is an evenly spaced array of locations that is addressable by the tips on a pipette head. A 96-well head can dispense into a 96-well plate, four quadrants of a 384-well plate, and 16 quadrants of a 1536-well plate. A 384-well head can dispense into a 384-well plate or the four quadrants of a 1536-well plate.

Property	Data type	Task parameter	Comments
task.retract	Float	Tip retract distance	The distance that the tips should move downwards per unit volume of liquid being aspirated. This value allows the tips to move downwards during aspiration to maintain a certain height below the surface of the liquid. You will need to determine an appropriate value by trial-and-error for each type of plate you use. You might want this value to be the same as the Tip Retract Distance for the Dispense pipette task.
task.tipOffset (Distance from well bottom)	Float	Distance from well bottom	The distance between the bottom of the pipette tips and the bottoms of the plate wells or MicroWash tray chimneys. If you are using dynamic tip retraction this value sets the lowest point to which the tips will travel.
task.velocity	Float	Aspirate velocity	The rate at which to draw up liquid. If you selected a liquid class, this value is entered automatically from the liquid library editor.
task.volume	Float	Aspirate volume	The volume of liquid to be drawn up into each pipette tip.

Change Tips task

These properties are the same as the properties described for the Change Tips task.

Property	Data type	Task parameter	Comments
task.plateName	String	Plate name	The name of the plate.
task.quadrants	An array of pairs of integers	Quadrants (diagram)	A quadrant is an evenly spaced array of locations that is addressable by the tips on a pipette head. A 96-well head can dispense into a 96-well plate, four quadrants of a 384-well plate, and 16 quadrants of a 1536-well plate.
task.action	Integer	❑ Press On New Tips (integer = 1) ❑ Tips off (integer = 2)	Puts tips on to a VPrep head or removes tips from a VPrep head.

Dispense task

These properties are the same as the properties described for the Dispense task.

Property	Data type	Task parameters	Comments
task.plateName	String	Plate name	The name of the plate.
task.acceleration	Float	Dispense acceleration	The rate of increase in velocity before the Dispense Velocity is reached. If you selected a liquid class, this value is entered automatically from the liquid library editor.
task.blowoutVolume	Float	Blowout volume	The volume of air to blow out when the tips are in the liquid. This is typically the same as the pre-aspirate volume.
task.liquidClass	String	Liquid class (list box)	The name of the liquid class.
task.postDispenseVolume	Float	Post dispense volume	The volume of air to blow out when the tips are out of the liquid.

Property	Data type	Task parameters	Comments
task.quadrants	An array of pairs of integers	Quadrants (diagram)	A quadrant is an evenly spaced array of locations that is addressable by the tips on a pipette head. A 96-well head can dispense into a 96-well plate, four quadrants of a 384-well plate, and 16 quadrants of a 1536-well plate. A 384-well head can dispense into a 384-well plate or the four quadrants of a 1536-well plate.
task.retract	Float	Retract distance	The distance that the tips should move upwards per unit volume of liquid being dispensed. This value allows the tips to move upwards during dispensing to maintain a certain height above the surface of the liquid. You will need to determine an appropriate value by trial-and-error for each type of plate you use. You might want this value to be the same as the Tip Retract Distance for the Aspirate pipette task.
task.tipOffset	Float	Distance from well bottom	The distance between the bottom of the pipette tips and the bottoms of the plate wells or MicroWash tray chimneys. If you are using dynamic tip retraction this value sets the lowest point to which the tips will travel.
task.tipTouch	Boolean	Enable tip touching	Whether you want the tips to touch the sides of the plate wells or not.
task.tipTouchHorizontalDistance	Float	Tip touch horizontal distance	When the value for this parameter is zero, the tips will move horizontally one well radius. The well radius is defined in the labware database for the type of plate you are using. If you want the tips to touch harder, increase this value. If you want the tips to touch more lightly, enter a negative value.

Property	Data type	Task parameters	Comments
task.tiptouchRiseHeight	Float	Tip touch rise height	The height that the tips should move upwards before touching the side of the wells.
task.tipTouchSides	Integer	Number of sides to touch	The number of sides of the wells that you want the tips to touch.
task.velocity	Float	Dispense velocity	The rate at which to dispense the liquid. If you selected a liquid class, this value is entered automatically from the liquid library editor.
task.volume	Float	Dispense volume	The volume of liquid to be moved out of each pipette tip.
task.enableShake	Boolean	Enable shaking	Turns on shelf shaking during dispense.
task.shakeRPM	Integer	Shake RPM	Shaking speed (in RPM).
task.shakeDelay	Integer	Post-dispense delay	Wait time (in milliseconds) before shaking starts.
task.shakeTime	Integer	Shake time	Duration (in milliseconds) of shaking.

Mix task

These properties are the same as the properties described for the Mix task.

Property	Data type	Task parameters	Comments
task.plateName	String	Plate name	The name of the plate.
task.aspirateAcceleration	Float	Aspirate acceleration	The rate of increase in velocity before the maximum aspirate velocity is reached. If you selected a liquid class, this value is entered automatically from the liquid library editor.
task.aspirateVelocity	Float	Aspirate velocity	The rate at which to draw up liquid. If you selected a liquid class, this value is entered automatically from the liquid library editor.

Property	Data type	Task parameters	Comments
task.blowoff	Float	Last cycle blowout volume	The volume of air to blow out when the tips are in the liquid. This is typically the same as the pre-aspirate volume.
task.cycles	Integer	Number of mixing cycles	The number of aspirate/ dispense operations.
task.dispenseAcceleration	Float	Dispense acceleration	The rate of increase in velocity before the Dispense Velocity is reached. If you selected a liquid class, this value is entered automatically from the liquid library editor.
task.dispenseVelocity	Float	Dispense velocity	The rate at which to dispense the liquid. If you selected a liquid class, this value is entered automatically from the liquid library editor.
task.liquidClass	String	Liquid class (list box)	The name of the liquid class.
task.preAspirateAirGap	Float	Pre-aspirate volume	The volume of air to be drawn up before the pipette tips enter the liquid.
task.quadrants	An array of pairs of integers	Quadrants (diagram)	A quadrant is an evenly spaced array of locations that is addressable by the tips on a pipette head. A 96-well head can dispense into a 96-well plate, four quadrants of a 384-well plate, and 16 quadrants of a 1536-well plate. A 384-well head can dispense into a 384-well plate or the four quadrants of a 1536-well plate.

Property	Data type	Task parameters	Comments
task.retract	Float	Retract distance	The distance that the tips should move upwards or downwards per unit volume of liquid being dispensed or aspirated. This value allows the tips to move upwards or downwards during dispensing or aspirating to maintain a certain height below or above the surface of the liquid. You will need to determine an appropriate value by trial-and-error for each type of plate you use.
task.tipOffset	Float	Distance from well bottom	The distance between the bottom of the pipette tips and the bottoms of the plate wells or MicroWash tray chimneys. If you are using dynamic tip retraction this value sets the lowest point to which the tips will travel.
task.tipTouch	Boolean	Enable tip touching	Whether you want the tips to touch the sides of the plate wells or not.
task.tipTouchHorizontalDistance	Float	Tip touch horiz. dist	When the value for this parameter is zero, the tips will move horizontally one well radius. The well radius is defined in the labware database for the type of plate you are using. If you want the tips to touch harder, increase this value. If you want the tips to touch more lightly, enter a negative value.
task.tiptouchRiseHeight	Float	Tip touch rise height	The height that the tips should move upwards before touching the side of the wells.
task.tipTouchSides	Integer	Number of sides to touch	The number of sides of the wells that you want the tips to touch.
task.volume	Float	Mixing volume	The volume of liquid to be aspirated and dispensed to each plate well.

Loop task

These properties are the same as the properties described for the Loop task.

Property	Data type	Task parameter	Comments
task.numberOfLoops	Integer	Number of times to loop	Number of times to loop.

Pump Reagent task

These properties are the same as the properties described for the Pump Reagent task.

Property	Data type	Task parameter	Comments
task.action	Integer	❑ Fill reservoir (value = 0) ❑ Empty reservoir (value = 1)	The Fill reservoir and Empty reservoir values determine whether the pumps will fill or empty the reservoir. To empty the reservoir you must complete the Autofill Configuration information on the Shelves tab of the VPrep Diagnostics software. For more information, see the *VPrep User Guide*.
task.howOften	Integer	Every	The number that controls how frequently the liquid is pumped. For example, if you type 3, the pump will run every third time the task runs.
task.maxLevel	Integer	Max level	The maximum percentage of liquid that you want the reservoir to contain.
task.minLevel	Integer	Min level	The minimum percentage of liquid that you want the reservoir to contain.
task.plateName	String	Plate name	The name of the plate.
task.shelf	String	(unnamed list box)	The shelf on which the reservoir is located.
task.speedPercent	Integer	at	The percentage of maximum pumping rate.

Property	Data type	Task parameter	Comments
task.time	Integer	for	The time in seconds that the pumps pump.

Wash Tips task

These properties are the same as the properties described for the Wash Tips task.

Property	Data type	Task parameters	Comments
task.plateName	String	Plate name	The name of the plate.
task.aspirateVelocity	Float	Aspirate velocity	The rate at which to draw up liquid. If you selected a liquid class, this value is entered automatically from the liquid library editor.
task.aspirateAcceleration	Float	Aspirate acceleration	The rate of increase in velocity before the maximum aspirate velocity is reached. If you selected a liquid class, this value is entered automatically from the liquid library editor.
task.blowoff	Float	Last cycle blowout volume	The volume of air to blow out when the tips are in the liquid. This is typically the same as the pre-aspirate volume.
task.cycles	Integer	Number of wash cycles	The number of aspirate/ dispense operations.
task.dispenseAcceleration	Float	Dispense acceleration	The rate of increase in velocity before the Dispense Velocity is reached. If you selected a liquid class, this value is entered automatically from the liquid library editor.
task.dispenseToWaste	Boolean	Dispense to waste at height of (check box)	The tips will dispense outside the MicroWash tray chimneys.

Property	Data type	Task parameters	Comments
task.dispenseVelocity	Float	Dispense velocity	The rate at which to dispense the liquid. If you selected a liquid class, this value is entered automatically from the liquid library editor.
task.heightAboveWaste	Float	Dispense to waste at height of (text box)	The height, in millimeters, above the MicroWash chimneys at which the tips will dispense. Used in combination with the dispense to waste property. Enter a negative number to ensure that the tips are below the tops for the chimneys.
task.inFlowPercent	Integer	Inflow pump	The relative rate of liquid flow into the MicroWash tray manifold. This value should be high enough for the washing liquid to just bubble over the tops of the chimneys.
task.liquidClass	String	Liquid class (list box)	The name of the liquid class.
task.outFlowPercent	Integer	Outflow pump	The relative rate of liquid flow out of the MicroWash tray manifold. This value is typically zero because the fluid is drained by gravity.
task.preAspirateAirGap	Float	Pre-aspirate volume	The volume of air to be drawn up before the pipette tips enter the liquid.

Property	Data type	Task parameters	Comments
task.quadrants	An array of pairs of integers	Quadrants (diagram)	A quadrant is an evenly spaced array of locations that is addressable by the tips on a pipette head. A 96-well head can dispense into a 96-well plate, four quadrants of a 384-well plate, and 16 quadrants of a 1536-well plate. A 384-well head can dispense into a 384-well plate or the four quadrants of a 1536-well plate.
task.retract	Float	Retract distance	The distance that the tips should move upwards or downwards per unit volume of liquid being dispensed or aspirated. This value allows the tips to move upwards or downwards during dispensing or aspirating to maintain a certain height below or above the surface of the liquid. You will need to determine an appropriate value by trial-and-error for each type of plate you use.
task.tipOffset	Float	Distance from well bottom	The distance between the bottom of the pipette tips and the bottoms of the plate wells or MicroWash tray chimneys. If you are using dynamic tip retraction this value sets the lowest point to which the tips will travel.
task.tipTouch	Boolean	Enable tip touch	Whether you want the tips to touch the sides of the plate wells or not.

Property	Data type	Task parameters	Comments
task.tipTouchHorizontalDistance	Float	Tip touch horiz. dist.	When the value for this parameter is zero, the tips will move horizontally one well radius. The well radius is defined in the labware database for the type of plate you are using. If you want the tips to touch harder, increase this value. If you want the tips to touch more lightly, enter a negative value.
task.tiptouchRiseHeight	Float	Tip touch rise height	The height that the tips should move upwards before touching the side of the wells.
task.volume	Float	Wash volume	The volume of liquid to be drawn up into each pipette tip.

Related information

For information about...	See...
Using JavaScript in BenchCel	"Using JavaScript in BenchWorks" on page 171
The JavaScript plate object and properties	"Using JavaScript in BenchWorks" on page 171
Apply Label task parameters	"About setting Apply Label task parameters" on page 101
Change Tips task parameters	"Setting Change Tips pipette task parameters" on page 148
Dispense task parameters	"Setting Dispense pipette task parameters" on page 150
Dry Tips task parameters	"Setting Dry Tips pipette task parameters" on page 153
Loop pipette task parameters	"Setting Loop pipette task parameters" on page 153
Mix task parameters	"Setting Mix pipette task parameters" on page 155
Pump Reagent task parameters	"Setting Pump Reagent pipette task parameters" on page 157
Wash Tips task parameters	"Setting Wash Tips pipette task parameters" on page 159

For information about...	See...
The workflow that this procedure belongs to	"Workflow for creating a protocol" on page 85

Understanding the configuration settings in a pipette task

About this topic

Dozens of separate pieces of data are required by BenchWorks to complete a single pipetting task. These range from the velocity at which to aspirate a liquid to the height at which the gripper should approach a plate that is sitting on a platepad. These data are the configuration settings. With the right privileges, you can modify and add configuration settings to fine-tune and extend the capabilities of your BenchCel.

How you can modify configuration settings is discussed in other topics. This topic uses the example of an Aspirate pipette task to provide the conceptual background you need to understand:

- ❑ The relationships between the device manager, labware editor, liquid library editor, task parameters, and protocol file
- ❑ What information is stored in a protocol
- ❑ Why you perform certain steps when creating protocols
- ❑ How to resolve protocol errors

Before reading this topic, it may help to read "Configuring a pipette process: example" on page 2.

This topic presents the same information from a different perspective.

Information flow diagram

The following diagram explains how information flows from the various data stores in BenchWorks into a single Aspirate pipette task that is stored as part of a protocol file.

Four main kinds of information are associated with a pipette task in a protocol file. These are:

- ❑ General task properties set in the Pipette Task Parameters toolbar
- ❑ Device information
- ❑ Labware information
- ❑ Liquid properties information

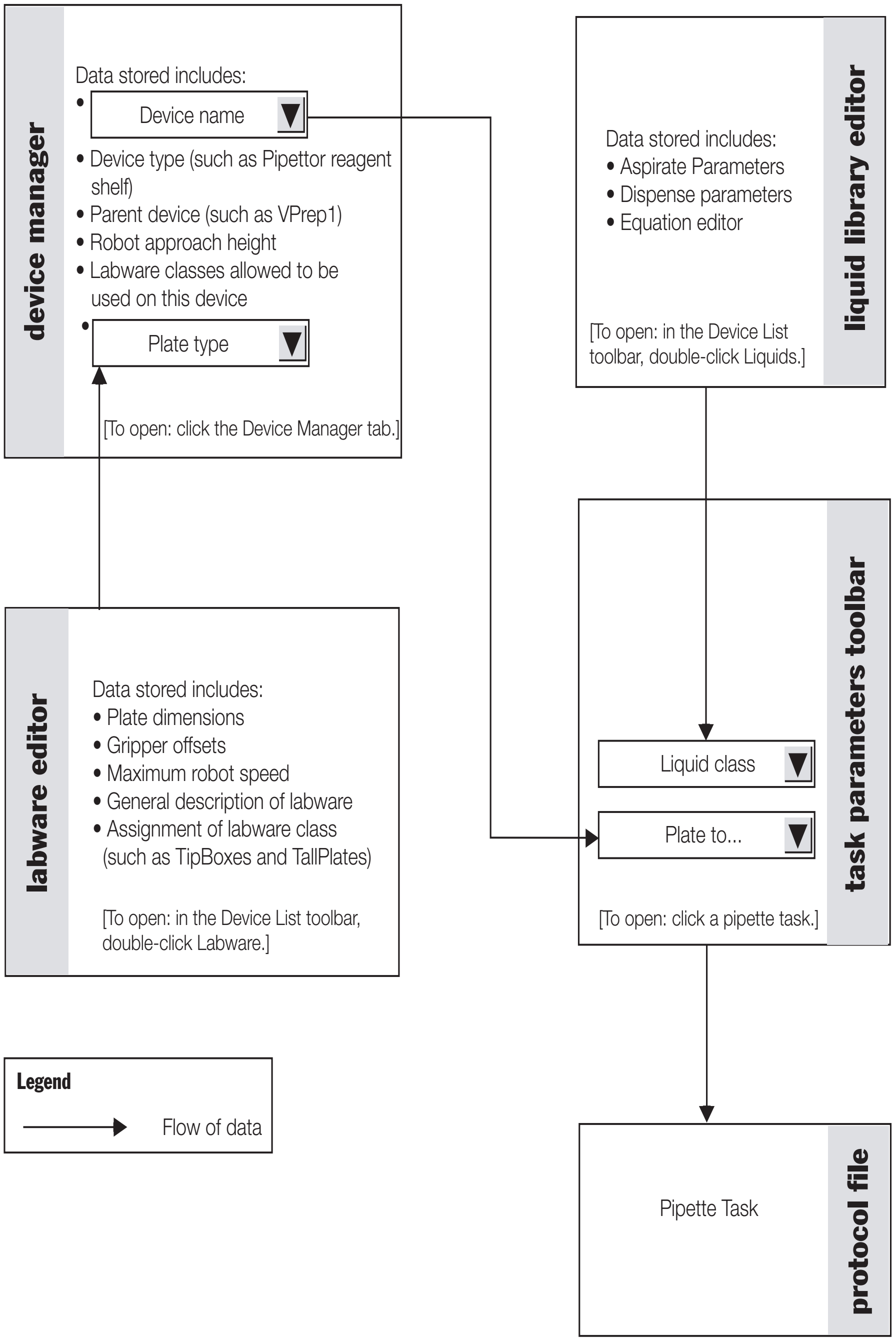
This information is included in the pipette task through a cascade of associations.

In the Aspirate pipette task example, the association between the type of liquid and the pipette task is made by selecting a liquid class for the pipette task.

All of the settings from the three data categories are stored in their respective databases and entered through their respective interface.

Data categories	Databases	Data interfaces
device files	device manager	device manager
plate types	labware database	labware editor
liquid classes	liquid library	liquid library editor

When a protocol is saved, the names of the three data categories that it uses are saved and associated with it, so that you don't have to select them every time. During a run, these data categories are referenced from the protocol they were associated with.



Related information

For information about...	See...
Liquid library editor	"About the liquid library editor" on page 244
Labware editor	"About the labware editor" on page 215

For information about...	See...
BenchWorks	“Relationships of BenchWorks components” on page 30

Working with device files

Device defined

A device is an item on the BenchCel that has an entry in the device manager. A device can be an instrument or a location on the BenchCel that can hold a piece of labware.

Examples of devices:

- ❑ Platepad
- ❑ VPrep shelf
- ❑ Stacker-rack

Device configuration

Every device on your BenchCel system has been properly configured for you by Velocity11 personnel. However, if you have a VPrep integrated with your BenchCel system and want to add a shelf or if you wish to add new instruments to your BenchCel system, administrators in your organization need to configure the relevant device settings.

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.

Saving a device file

After you make changes in the device manager, you must save the changes for them to be available the next time the protocol is opened.

To save a device file:

1. Make sure you are logged in as an administrator.
2. Select **File** > **Device File** > **Save As**.
3. Navigate to the folder in which you want to save the file.

   If you want to save the file in the current folder, skip this step.
4. Click **Save**.

   The path of the device file specified in the protocol file and **Protocol Options** dialog box is changed to reflect the different location.

The next time you compile or run a protocol that references the device file, the new devices are registered with the Windows operating system.

Loading a device file

When you open a protocol file, the device file associated with it is automatically loaded for you.

If you need to load a new device file for your current protocol, use one of the following methods.

Method 1: Load a device file from Windows

To load a device file from Windows:

1. In Windows, navigate to the device file.
2. Double-click the file.

   BenchWorks is launched if it is not already running and the device file is loaded.

Method 2: Load a device file from within BenchWorks

To load a device file from within BenchWorks:

1. Select **File** > **Device File** > **Open.**

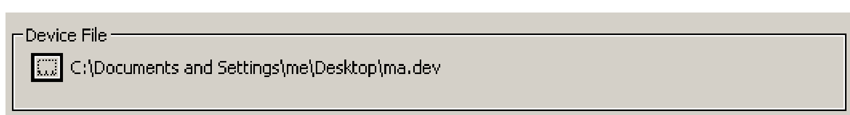
   If you want to open a recently opened device file, select it from the list of device files at bottom of the menu, and the device file is loaded.
2. Navigate to the folder that contains the file to load.
3. Click **Open**.

Method 3: Load a device file from Protocol Options

If you load a device file using the following method, the device file will be saved with the protocol when you save the protocol.

To load a device file from protocol options:

1. Click **Tools** and select **Protocol Options**.
2. In the **Protocol Options** tab of the dialog box, click the device file ellipsis button.



3. Navigate to the folder that contains the file to load.
4. Click **Open**.

Note: The file path displayed in the **Protocol Options** dialog box is updated if you save a protocol to a different folder.

Related information

For information about...	See...
Modifying the device file for a VPrep shelf	“Configuring a VPrep shelf as a device” on page 142

For information about...	See...
How the device file relates to other BenchWorks components	"Relationships of BenchWorks components" on page 30

About bar code reading and tracking

About this topic

This topic gives an overview of the bar code reading and tracking abilities of the BenchCel.

Read this topic if you are a technician or administrator who writes protocols with bar code reader tasks or Apply Label tasks.

Bar code readers

A customized BenchCel can have bar code readers installed that can read bar codes on one side of a plate (the side varies, depending on your application needs). These may be installed on VPrep shelves or platepads. Every time a plate is moved to one of these devices, the bar code is read.

Note: To read a bar code at a platepad bar code reader or VCode, use the Place Plate task.

VCode: bar code printer and optional reader

If your BenchCel has a VCode, you have the ability to print and apply bar code labels.

If the VCode includes an optional reader, bar code labels can be read on any side of the plate, because the VCode can rotate the plate in a full circle.

Related information

For information about...	See...
Planning to add bar code labels to plates	"Using bar code input files" on page 198
Adding the Apply bar code task to a protocol	"Setting Apply Label task parameters" on page 102
Reading plates at a platepad	"Setting Place Plate task parameters" on page 120

Using bar code input files

About this topic

This topic describes how to create and use bar code input files.

Read this topic if you are a technician or an administrator who writes protocols with bar code reader or Apply Label (VCode) tasks.

Bar code fields

When setting up a VCode, you set task parameters that specify the content of bar code fields.

An example of a bar code field is:

NAW1001

Bar code fields can be imported from bar code input files.

For detailed information about bar code fields and formats, see the *VCode User Guide*.

Bar code input files

Filename and location

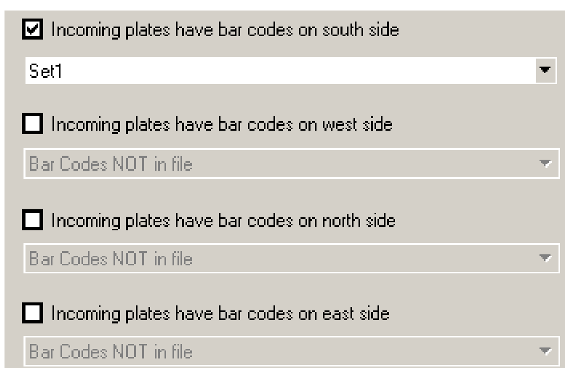
Bar code input files are text files with the naming convention (*filename*.bar). They are stored in the location specified in the general BenchWorks options.

When to use

You can use bar code input files to do the following:

- Verify the bar codes on incoming plates, which are plates that are downstacked into the system.

  This function is set in the parameters for the plate icon of the incoming plates:

- Specify each field of a bar code that is printed on a plate.

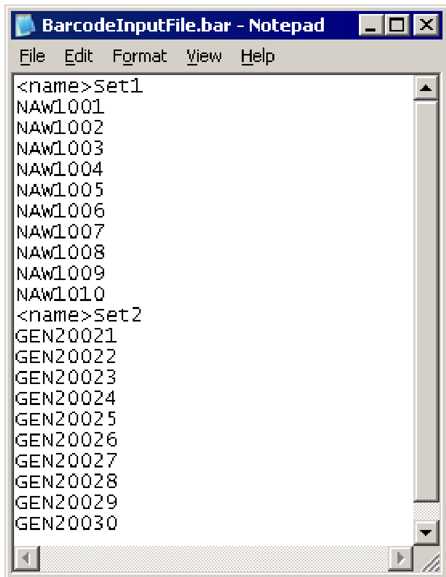
How they are created

Bar code input files are typically generated by a LIMS system, although you can create them manually.

File structure

Bar code input files contain lists of bar code fields, or parts of fields, that are grouped together in series. In the following example, there are two series:

- ❑ Set1
- ❑ Set2



Each series could be used to label a different side of the same plate or label plates during different runs.

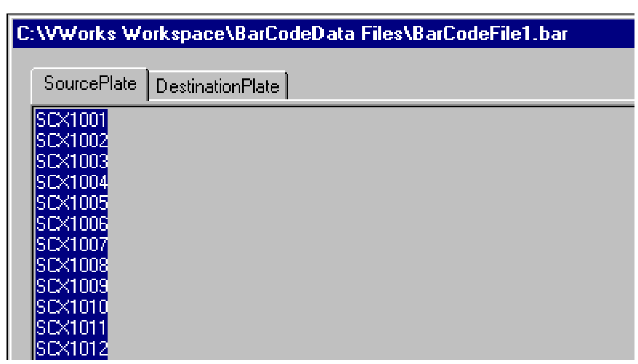
Viewing a bar code file in BenchWorks

You can view the bar code file that is currently associated with BenchWorks as follows.

To view the associated bar code input file:

1. Select **Tools** > **Show Bar Code File**.

   This opens a view of the bar code input file.



2. Click a tab to show a different series of data.
3. To close the window, click the close box.

Updating a bar code input file

If you change a bar code input file while a protocol is running, you must reload the file for the change to be registered.

To reload a bar code input file:

1. Select **Tools** > **Reload Bar Code File**.

   The bar code input file is reloaded.

Related information

For information about...	See...
Selecting bar code files	"Setting general options" on page 41
Using the FileReader	"About the FileReader plug-in" on page 164

Using bar code data files

About this topic

This topic describes how to create and use bar code data files in collaboration with bar code input files.

Read this topic if you are a technician or an administrator who writes protocols with Apply Label (VCode) tasks.

Bar code data files

File format and location

Bar code data files are tab-delimited text files with the name *filename*.dat. They are stored in a location specified in the general BenchWorks options.

How they work

A bar code data file acts as a lookup table that specifies what bar code fields to print on other sides of a plate. The typical sequence of events is as follows:

1. A plate with a south-side bar code is downstacked into the system.
2. The robot picks up the plate, reads the bar code, and verifies it against a bar code input file.
3. The robot moves the plate to a VCode.
4. The bar code that was read is used as a key to look up the bar code fields to print on the other sides of the plate, using the bar code data file as the lookup table.
5. The VCode prints a bar code on the north-side, east-side, and west-side of the plate.

Bar code data files can also be used with incoming plates that have west-side bar codes. In this case, the bar code must be read at the VCode or platepad and not by the robot's bar code reader.

!! IMPORTANT !! Bar code data files cannot currently be used with incoming bar codes on the north or east sides.

Bar code data files can still be used if the downstacked plate has no bar code, provided that incoming bar code verification is turned off. The plate could be moved to the VCode and labelled on its south or west side. That label could then be read and used with a bar code data file to specify the labels to be printed on the other sides of the plate.

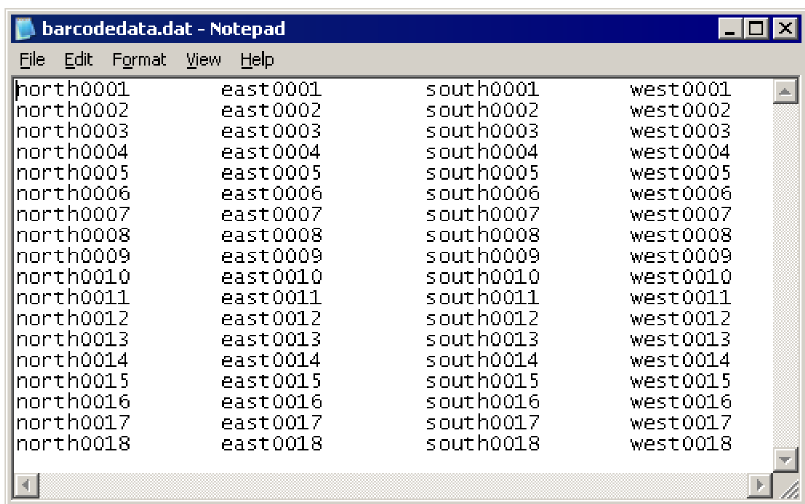
Where they are specified

The use of bar code data files is specified when configuring task parameters for an Apply Label task.

File structure

An example of a bar code data file is shown below. The columns are separated by tabs.

!! IMPORTANT !! The columns must be in the order north, east, south and west, from left to right.



north0001	east0001	south0001	west0001
north0002	east0002	south0002	west0002
north0003	east0003	south0003	west0003
north0004	east0004	south0004	west0004
north0005	east0005	south0005	west0005
north0006	east0006	south0006	west0006
north0007	east0007	south0007	west0007
north0008	east0008	south0008	west0008
north0009	east0009	south0009	west0009
north0010	east0010	south0010	west0010
north0011	east0011	south0011	west0011
north0012	east0012	south0012	west0012
north0013	east0013	south0013	west0013
north0014	east0014	south0014	west0014
north0015	east0015	south0015	west0015
north0016	east0016	south0016	west0016
north0017	east0017	south0017	west0017
north0018	east0018	south0018	west0018

Related information

For information about...	See...
Selecting bar code data files	"Setting general options" on page 41
Bar code input files	"Using bar code input files" on page 198
Applying bar codes	"Setting Apply Label task parameters" on page 102

Administrator procedures

9

This chapter is for people with administrator login privileges. It describes assorted administrative tasks that include how to:

- ❑ Administer user accounts
- ❑ Import and export BenchWorks data from the registry of your computer
- ❑ Set up email so that you can send bug reports from within BenchWorks

About user accounts and privileges

About this topic

You must have a user account to log in to BenchWorks. Your user account is associated with a user role that determines the privileges you have to perform particular functions.

This topic describes the privileges associated with different user roles.

The effect of privileges

Privileges have the following effects:

- ❑ If you do not have the privilege to perform a function associated with a particular menu command, the text of the command is gray.
- ❑ If you do not have the privilege to perform the functions accessed from a particular tabbed page, the tab is not visible to you.
- ❑ In some cases, if you do not have the privilege to perform an operation, when you attempt the operation you get an error message telling you that your privileges are insufficient.

User roles and privileges

User roles enforce the following privileges:

User role	Has privileges to...
Guest	Run existing protocols.
Operator	❑ Perform guest functions (see above). ❑ Operate devices in real time using diagnostics software.
Technician	❑ Perform operator functions (see above). ❑ Create and save protocols. ❑ Edit the labware database and liquid library database.
Administrator	❑ Perform technician functions (see above). ❑ Manage devices through the device manager. ❑ Create and delete user accounts. ❑ Run a protocol that contains compiler errors.
Velocity11	Perform any function. Used by Velocity11 personnel only.

Related information

For information about...	See...
Adding and deleting accounts	"Adding and deleting a user account" on page 205
Setting up email	"Setting up email" on page 206
Sending a registry file	"Moving or sending a registry file" on page 208

Adding and deleting a user account

About this topic

We recommend that BenchCel administrators create an account for every user. The privileges set for the account should be appropriate for the users' job role.

This topic explains how to add and delete user accounts.

About blank passwords

!! IMPORTANT !! If you do not assign a password to a user, a blank password is automatically assigned. This means that anybody can log in.

Adding a user account

You must be logged in with an administrator user account to add a user account.

To add a user account:

1. Select **Tools** > **Manage Users**.
2. In the **User Editor** dialog box, click **Add**.
3. In the **User name** text box, type a name for the user.
4. From the **Access level** list box, select the privilege that you want to assign to the user.
5. Assign a password:
   a. Click **Change password**.
   b. Enter a password in the **New** text box.
   c. Re-type the password in the **Confirm New** text box.
   d. Click **OK**.
6. Click **OK**.

Deleting a user account

You must be logged in with an administrator user account to delete a user account.

To delete a user account:

1. Select **Tools** > **Manage Users**.
2. In the **User name** text box, select the user you want to delete.
3. Click **Delete**.

   If you attempt to delete the account that is currently in use, you receive an error message.
4. In the **Confirm Delete** dialog box, click **Yes**.

   The password is deleted.

Related information

For information about...	See...
User accounts and privileges	"About user accounts and privileges" on page 204
Setting up email	"Setting up email" on page 206

Setting up email

About this topic

This topic describes how to add an email address to BenchWorks so you can be notified by email or pager when there is a run error.

Email setup in BenchWorks enables you to do the following tasks:

- ❑ Automatically be notified by email or pager when errors occur during a protocol run
- ❑ Send a bug report to Velocity11

Requirements for email setup

Before you can send an email from BenchWorks:

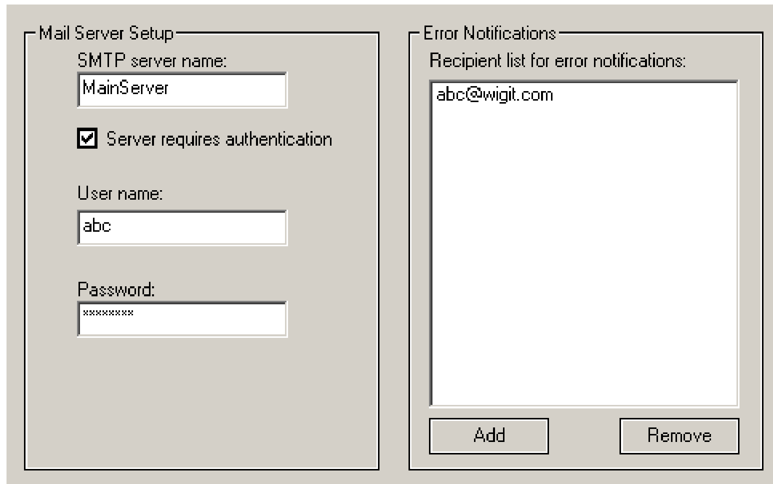
- ❑ The BenchCel computer must be connected to a network with internet access
- ❑ The outgoing email server must be set up on the system's computer

Setting up email

To set up the outgoing email server:

1. Select **Tools** > **Options**.
2. In the **BenchWorks Options** dialog box, click the **Email Setup** tab.
3. In the **Mail Server Setup** group box, enter the name of your **SMTP server name** (outgoing email server).

4. If the server requires a user name and password:
   a. Select the **Server requires authentication** check box.
   b. Enter the **User name** and **Password** that you use to access email on the network.



This information only needs to be set up once, provided the email account remains active. All email sent from BenchWorks is authenticated using this account.

Related information

For information about...	See...
Sending a bug report	"Sending a bug report" on page 321
Sending a registry file	"Moving or sending a registry file" on page 208

Moving or sending a registry file

About this topic

This topic provides instructions on how to export a Windows registry file for import to another computer or for emailing to Velocity11.

When to do this

- ❑ To propagate a labware or liquid library database to other devices using a different controlling computer
- ❑ To email a labware or liquid library database when requested by personnel at Velocity11

About moving data

The labware and liquid libraries are maintained in the Windows registry of the controlling computer.

When you use BenchWorks to make a change to the labware or liquid libraries, the change is seen when accessing that information from the networked instrument's software. This is because BenchWorks and instrument software run on the same controlling computer and access the same databases.

If you make a change to the labware or liquids database, you can use a two-step process to propagate the change to another computer.

1. Export the Windows registry key containing the data to a file.
2. Import the file to the other computer's registry.

Important

!! IMPORTANT !! Making a mistake when editing the registry may cause critical failures with your operating system.

Exporting a registry key

To export a registry key:

1. From the Windows **Start** menu, select **Run**.
2. In the **Open** text box, type `regedit`.
3. Click **OK**.

   The Windows registry editor opens.
4. Expand folders to display the following folder:

   HKEY_LOCAL_MACHINE\SOFTWARE\Velocity11\Shared
5. Expand the **Shared** folder and select either:
   - ◆ Labware
   - ◆ Liquid Library
6. From the **Registry** (or **File**) menu, select **Export**.

   The **Export Registry File** browser box opens.
7. Navigate to the folder in which you want to save the file.
8. In the **File name** text box, type a name of your choice for the file.

9. If you are moving the file to a computer with a different Windows operating system, select the appropriate recipient operating system from the **Save as type** list box, if one is available.
10. Click **Save**.
11. Select **Registry** > **Exit** (or **File** > **Exit)** to close the registry editor.

Importing a registry key

Before you start

You must have Windows Administrator access to perform this task.

To import a registry key:

1. Copy the registry file to any location on the recipient computer.
2. On the recipient computer, double-click the registry file.

   The information in the file will be written automatically to the registry.

Emailing a registry file

Occasionally, you may be asked to send a registry file to Velocity11.

To email a registry file:

1. Export the Windows registry key containing the data to a file.
2. Zip the file or change its extension to .re_ (This is necessary because many Exchange servers do not allow *.reg files to be emailed.)
3. Email the file.

Related information

For information about...	See...
Sending a bug report	"Sending a bug report" on page 321
Setting up email for error notification	"Setting up email" on page 206

Obtaining information about the BenchCel network cards

About this topic

You may need to provide some of the information to your network administrator for your BenchCel to be connected to your organization's network.

Your BenchCel has two network cards. The network connections for these cards are named WAN and LAN.

This topic describes the BenchCel network cards and how to obtain their network IP addresses.

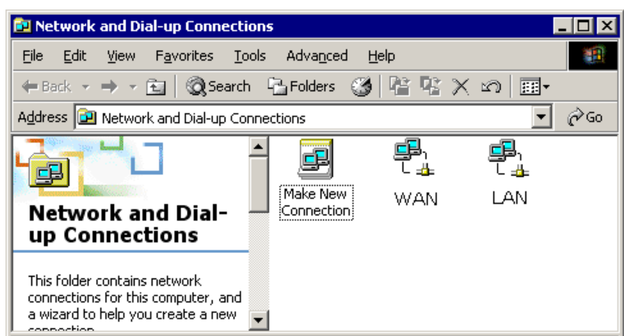
BenchCel computer network connections

Note: Depending on how you have personalized your operating system, you may need to use a slightly different procedure from the one below.

To see the network connections in Windows:

1. In Windows, from the **Start** menu select **Settings > Control Panel**.
2. Double-click the **Network and Dial-Up Connections** icon.

   The **Network and Dial-up Connections** window opens.



LAN network card

The LAN network card is used for communication between the computer and devices that use Ethernet communication cable. This network is considered to be the local area network.

The LAN network card has a fixed IP address, which is 192.168.0.1.

WAN network card

The WAN network card is for networking with your organization's network. The settings for this card should be configured by your network administrator in the same way that any PC would be configured to make it available to your network.

The WAN network card has an IP address that is dynamically assigned by your domain name server when you start you BenchCel's computer.

Obtaining network IP addresses

To see the IP address of the two network cards:

1. In BenchWorks, click the **Device Manager** tab.
2. Select a stack from the **Device List**.
3. Click **Device Diagnostics**.
4. In the **Discovered BioNet Devices** dialog box, click the drop-down arrow for the **Select the Ethernet adapter to use...** list box.

   !! IMPORTANT !! Make sure that you do not change the selected IP address when you do this. VStacks should always be connected to the LAN network with the 192.168.0.1 IP address.

Related information

For information about...	See...
Sending a bug report	"Sending a bug report" on page 321
Setting up email for error notification	"Setting up email" on page 206
Users and privileges	"About user accounts and privileges" on page 204

Defining labware 10

This chapter is for people with administrator or technician login privileges. It describes the use of the labware editor dialog box and the labware parameters group box, which are used to enter information about labware.

About defining labware in BenchWorks

Labware defined

Labware is a physical object such as a plate, lid, or tip box.

Labware entry defined

A labware entry is the collection of property values used to describe the properties of a piece of labware. This information is stored in the BenchWorks database and is used by the robot to perform tasks with the object.

Entering labware parameters

Labware parameters may be entered into BenchWorks by two methods:

- ❑ Using the Labware Editor dialog box in the labware diagnostics to add new or edit existing labware types
- ❑ Using the Labware Parameters group box in the BenchCel robot diagnostics to edit existing labware definitions

Each method has its own purpose:

- ❑ Use the Labware Editor dialog box to add new or to edit labware definitions in the database
- ❑ Use Labware Parameters group box when you want quick access to basic labware parameters to edit existing labware definitions

Related information

For information about...	See...
Using the labware editor	"About the labware editor" on page 215
Labware parameters group box	"About the Labware Parameters group box" on page 236
Opening the labware editor	"Opening the labware editor" on page 219

About the labware editor

Labware editor defined

The labware editor is the BenchWorks interface through which users enter information about labware.

Note: The labware database cannot be used by other companies' systems into which Velocity11 devices are integrated.

You must be logged on with an administrator or technician user account to use the labware editor.

Types of information stored

Two main types of information are stored in the labware database:

- ❑ Information about the labware properties
- ❑ Information about labware classes

About labware properties

Labware has physical properties such as width, length, and number of wells as well as non-physical properties such as robot handling speed, offsets, and plate handling options.

Without the labware editor, you would have to enter values for the many properties associated with a plate each time you set up a protocol. With the labware editor, all you have to do is select the type of labware to use.

About labware classes

Labware classes are sets of labware entries, grouped so they are easier to manage than many individual labware entries.

Labware classes are used in combination with the device manager of BenchWorks to restrict which types of labware can be used on which devices during a protocol run. This helps to prevent wasted runs and damage to the devices on the platform. An example of how damage can be prevented by labware restriction is where a tipbox that is too tall for a device crashes into the device as the robot delivers it.

Related information

For information about...	See...
Moving the labware database to another computer	"Moving or sending a registry file" on page 208
Defining labware	"About the defining labware process" on page 218
Opening the labware editor	"Opening the labware editor" on page 219
Editing labware parameters	"Changing labware parameters" on page 238

Labware editor overview

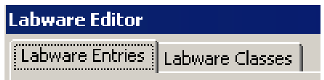
About this topic

This topic gives an overview of the organization of the labware editor's user interface.

Labware Editor pages

The labware editor has two tabbed pages:

- ❑ Labware Entries—where labware definitions are entered
- ❑ Labware Classes—where defined labware is assigned to classes



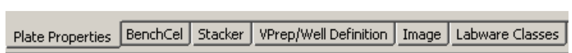
Labware Entries page

Sub-pages

The Labware Entries page has the following sub-pages:

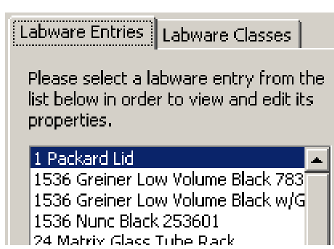
- ❑ Plate Properties
- ❑ BenchCel
- ❑ Stacker
- ❑ VPrep/Well Definition
- ❑ Image
- ❑ Labware Classes

The sub-page tabs are located at the bottom of the page. (Some tabs may be hidden if they are not relevant.)



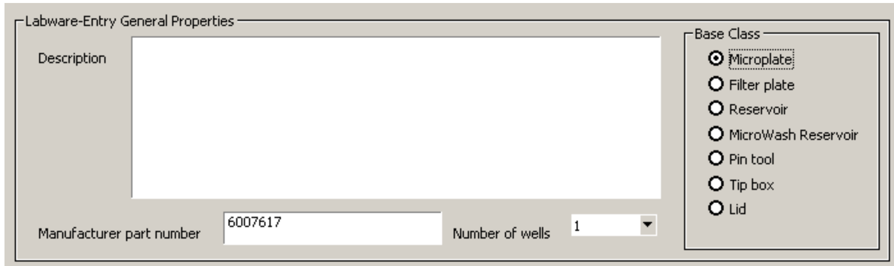
Labware selection box

The labware selection box, which is the left-hand column, is used to select a labware entry that you want to edit.



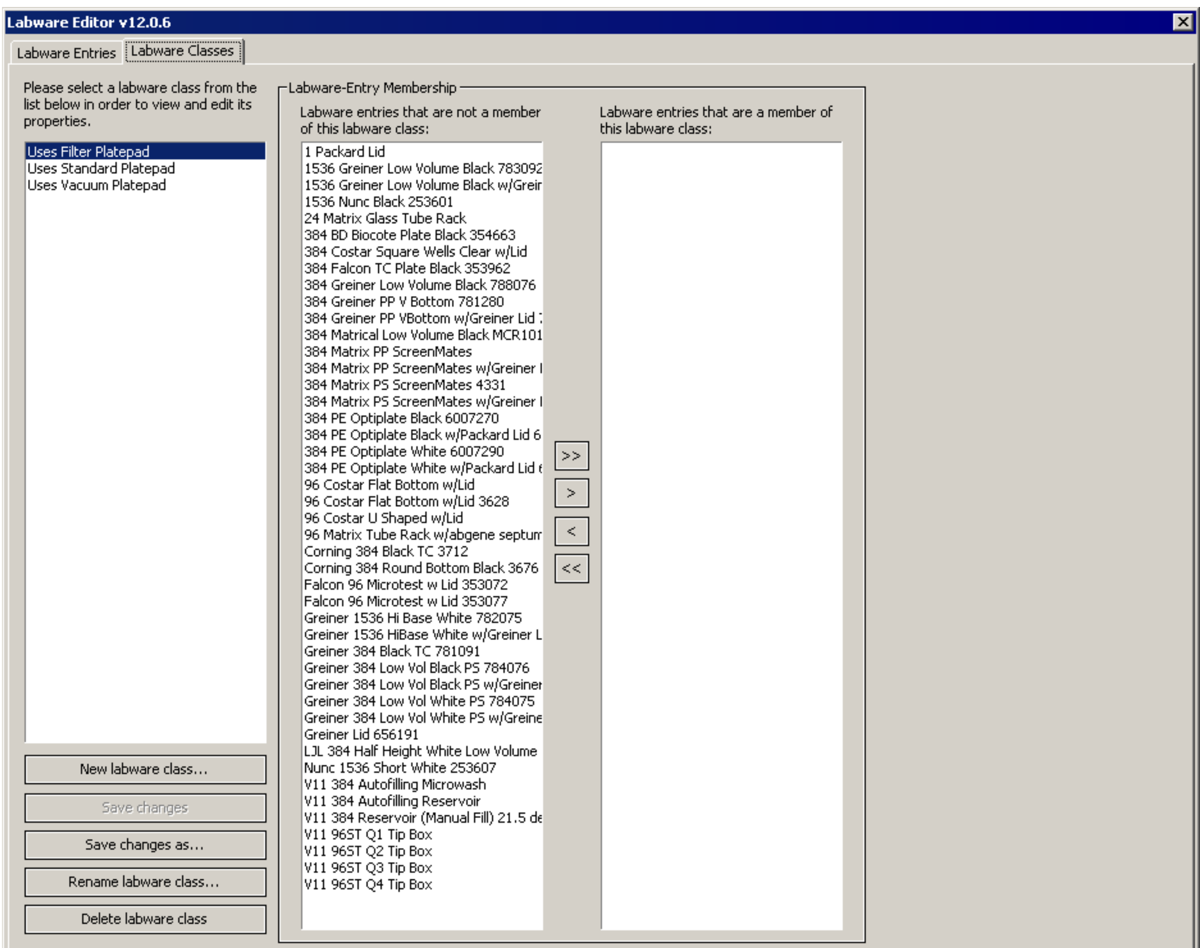
Labware-Entry General Properties group box

The Labware-Entry General Properties group box displays the labware-entry general properties whose selections applies across all sub-pages.



Labware Classes page

In the Labware Classes page, you create labware classes and assign defined labware to a labware class.



Related information

For information about...	See...
Moving the labware database to another computer	"Moving or sending a registry file" on page 208
Defining labware	"About the defining labware process" on page 218
Opening the labware editor	"Opening the labware editor" on page 219
Editing labware parameters	"Changing labware parameters" on page 238

About the defining labware process

About this topic

This topic:

- ❑ Describes how to see what labware is already defined
- ❑ Gives an overview of how to define labware

Labware standards

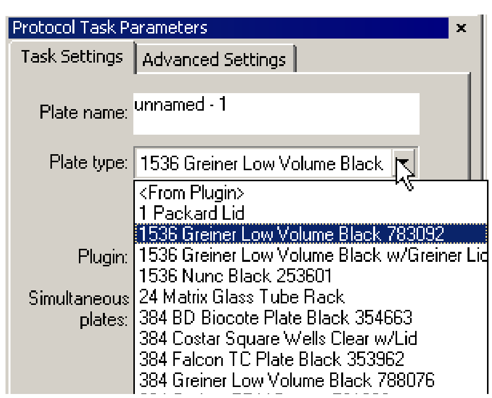
!! IMPORTANT !! All labware used with Velocity11 products must conform to the SBS microplate standards. This includes deepwell and PCR plates. Visit www.sbsonline.org **for current SBS standards, or contact your labware's manufacturer.**

Finding out if a piece of labware is defined

Before defining a piece of labware yourself, you should make sure that it has not already been defined.

To find out if a type of plate is already defined:

1. In BenchWorks, click the **Protocol Editor** tab, and select the plate icon in a process.
2. In the **Protocol Task Parameters** toolbar, see if the labware is among the available plate types.



If there is no entry for the plate, it is not defined.

Overall process

The overall process of defining a piece of labware is to:

1. Add a labware entry.
2. View each sub-page of the **Labware Entries** page in turn, entering values that are appropriate for the product you are using.
3. Use the **Labware Classes** page to optionally associate the labware with one or more labware classes.

Related information

For information about...	See...
Moving the labware database to another computer	"Moving or sending a registry file" on page 208
Defining labware	"About the defining labware process" on page 218
Opening the labware editor	"Opening the labware editor" on page 219
Editing labware parameters	"Changing labware parameters" on page 238

Opening the labware editor

About this topic

This topic explains how to open the labware editor.

You open the labware editor when you want to:

- ❑ View existing labware entries or classes
- ❑ Edit labware entries or classes
- ❑ Add new labware entries or classes
- ❑ Delete labware entries or classes
- ❑ Rename labware entries or classes

Before you start

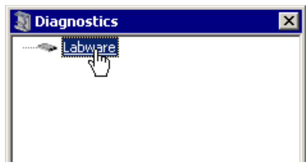
You must be logged on with an administrator or technician user account to open the labware editor.

Procedure

To open the labware editor:

1. In BenchWorks, click **Diagnostics**.

2. In the **Diagnostics** pop-up window, expand the system icon if necessary.



3. Click **Labware** to select it.
4. Click **Device Diagnostics**.

Related information

For information about...	See...
Defining labware	❑ "Labware editor overview" on page 216 ❑ "About the defining labware process" on page 218
Editing labware parameters	"Changing labware parameters" on page 238

Adding a labware entry

About this topic

The first step in defining a new piece of labware is to add a labware entry for it.

You must be logged on with an administrator or technician user account to perform this procedure.

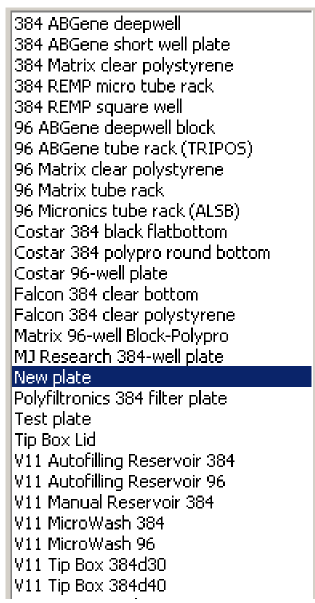
Procedure

To add a labware entry:

1. Open the labware editor.
2. Under the labware selection box on the left of the window, click **New labware entry**.

3. In the **New Labware Entry** dialog box, enter a name for the plate and click **OK**.

   The entry appears in the labware selection box.



Related information

For information about...	See...
Opening the labware editor	“Opening the labware editor” on page 219
Defining general properties of your new labware entry	“Defining general properties” on page 225
Defining the Plate Properties	“Defining plate properties” on page 226
Defining BenchCel properties	“Defining BenchCel properties” on page 229

Deleting a labware entry

About this topic

If there is a labware entry that you no longer need, you can delete it. This topic describes how to delete a labware entry,

You must be logged on with an administrator or technician user account to perform this procedure.

Before you start

Make sure that the entry you are deleting is not referenced in protocols.

!! IMPORTANT !! If you delete a labware entry that is already referenced in protocols, the link between the protocol and the labware data will be broken and the protocol will not run.

Procedure

To delete a labware entry:

1. Open the labware editor.
2. In the labware selection box on the left of the window, select the labware entry to be deleted.
3. Click **Delete labware entry**.
4. In the **V11Labware** dialog box, click **Yes** to delete the entry.

Related information

For information about...	See...
Opening the labware editor	"Opening the labware editor" on page 219
Defining labware	❑ "Labware editor overview" on page 216 ❑ "About the defining labware process" on page 218
Editing labware parameters	"Changing labware parameters" on page 238
Adding a labware entry	"Adding a labware entry" on page 221

Renaming a labware entry

About this topic

You can change the name of a labware entry. In general, this is something you might do if you just named a labware type and decided to give it a different name.

You must be logged on with an administrator or technician user account to perform this procedure.

Before you start

Make sure that either:

- ❑ The entry you are renaming is not already referenced in protocols, or
- ❑ If the entry is referenced in protocols, you update those protocols

!! IMPORTANT !! If you rename a labware entry that is already referenced in protocols, the link between the protocol and the labware data is broken and the protocol will not run until the protocols are updated.

Procedure

To rename a labware entry:

1. Open the labware editor.
2. In the labware selection box on the left of the window, select the labware entry to be renamed.
3. Click **Rename labware entry**.
4. In the **V11Labware** dialog box, click **OK** to confirm that you want to rename this entry.
5. In the **Rename Labware Entry** dialog box, enter the new name for the plate and click **OK**.

Related information

For information about...	See...
Opening the labware editor	“Opening the labware editor” on page 219
Defining labware	❑ “Labware editor overview” on page 216 ❑ “About the defining labware process” on page 218
Editing labware parameters	“Changing labware parameters” on page 238
Adding a piece of labware	“Adding a labware entry” on page 221
Deleting a labware entry	“Deleting a labware entry” on page 222

Copying a labware entry

About this topic

To save time when creating a new entry that is similar to an existing one, you can copy an existing labware entry. This topic describes how to copy an existing labware entry.

You must have an administrator or technician user account to perform this procedure.

Procedure

To copy a labware entry:

1. Open the labware editor.
2. In the labware selection box on the left of the window, select a labware entry.
3. Click **Save changes as**.
4. In the **Save Labware Entry As** dialog box, type a name for the new entry that is different from the selected one, and click **OK**.

   The copied entry appears in the labware selection box.

Related information

For information about...	See...
Opening the labware editor	"Opening the labware editor" on page 219
Renaming labware	"Renaming a labware entry" on page 223
Editing labware parameters	"Changing labware parameters" on page 238
Adding a piece of labware	"Adding a labware entry" on page 221
Deleting a labware entry	"Deleting a labware entry" on page 222

Defining general properties

About this topic

After adding a labware entry, define the general properties of the labware. The general properties describe the type of labware that is being entered into the database and are visible on all of the sub-pages of the labware editor. This topic describes how to define the labware's general properties.

You must have a technician or administrator user account to perform this procedure.

Before you start

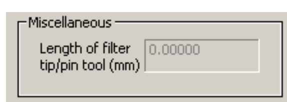
You must have added a labware entry that you want to define.

Procedure

To define the general properties of a piece of labware:

1. Open the labware editor.
2. In the **Description** text box, type in a description of the labware, if desired.
3. For your reference, in the **Manufacturer part number** text box, enter the part number for the labware.
4. In the **Number of wells** list box, select the number of wells in the plate.
5. In the **Base Class** group box, select one of the options.

   The option you select determines which labware editor properties are available. For example, when a base class of **Microplate** is selected, the **Length of filter tip/pin tool (mm)** property is unavailable.



Related information

For information about...	See...
Opening the labware editor	"Opening the labware editor" on page 219
Defining labware	❑ "Labware editor overview" on page 216 ❑ "About the defining labware process" on page 218
Editing labware parameters	"Changing labware parameters" on page 238

Defining plate properties

About this topic

This topic describes the parameters used to define a plate or other piece of labware in the Plate Properties sub-page of the labware editor.

You must have a technician or administrator user account to perform this procedure.

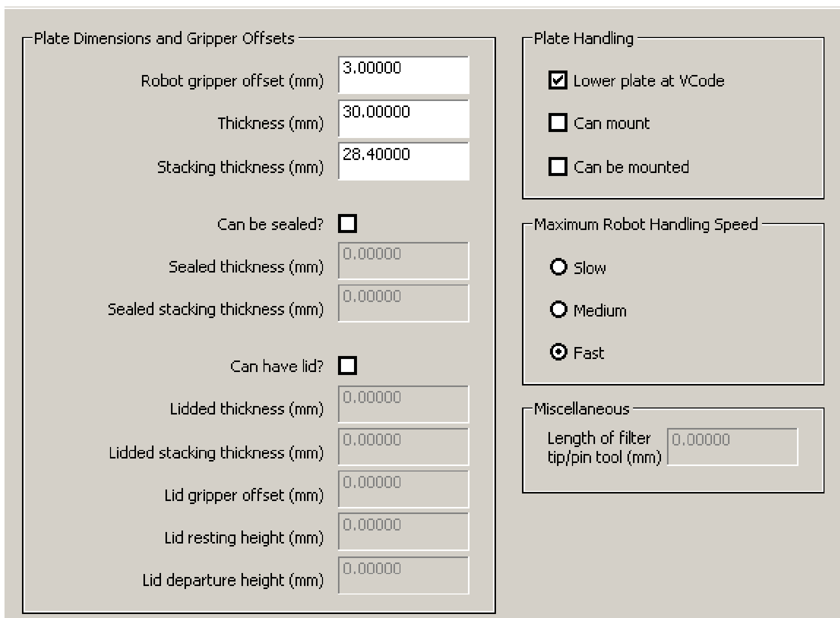
Defining plate properties

To define plate properties:

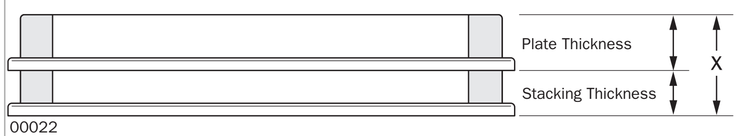
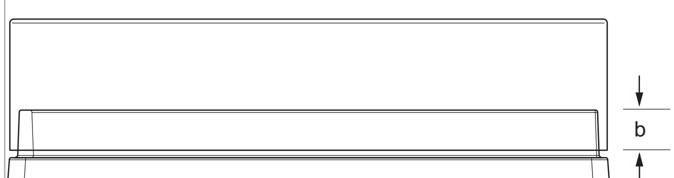
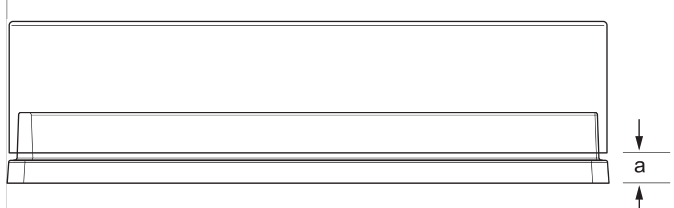
1. Click the **Plate Properties** sub-page tab of the labware editor.
2. Enter the values for the available parameters according to the type of plate or labware you are defining.

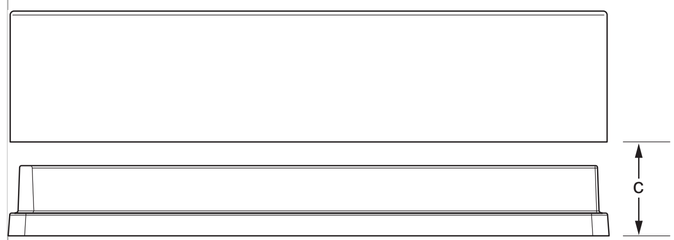
   The parameters on the **Plate Properties** sub-page are described in the following screenshot and table.

   Note: Only the parameters associated with the **Base Class** you selected in the **General Properties** group box will be available.



Property	Description
Robot gripper offset	This refers to the offset for a Velocity11 robot. The comparable parameter for the BenchCel robot is located on the BenchCel tab.
Thickness	The distance, in millimeters, from the bottom surface of the plate to the top surface of the plate. Measure using calipers.

Property	Description
Stacking thickness	The thickness, in millimeters, of two stacked plates minus the thickness of one plate. Measure using calipers. Example: Thickness of two stacked plates (*x*) = 23.14 mm Thickness of one plate = 14.14 mm Stacking thickness: 23.14 mm - 14.14 mm = 9.00 mm 
Can be sealed?	Select if the plate can be sealed.
Sealed thickness	Thickness of the plate with a seal in place. Available only if **Can be sealed?** is selected.
Sealed stacking thickness	Stacking thickness of the plate with the a seal in place. Available only if **Can be sealed?** is selected.
Can have lid?	Select if the plate can have a lid.
Lidded thickness	Thickness of the plate with a lid in place. Available only if **Can have lid?** is selected.
Lidded stacking thickness	Stacking thickness of the plate with the lid in place. Available only if **Can have lid?** is selected.
Lid gripper offset	Height above the lid resting height at which to grip the lid. (Shown as *b* below.) 
Lid resting height	Height above the bottom of the plate at which the bottom of a plate lid rests. (Shown as *a* below.) 

Property	Description
Lid departure height	Height above the bottom of the plate to which the lid is lifted. 
Lower plate at VCode	Select if the plate has a thick skirt and must be lowered when on the stage of the VCode. This allows the VCode to place the label above the thick skirt.
Can mount	Select if the plate can be placed on top of another plate. This property is for filter plates that are placed on top of waste plates during filtration steps of a protocol.
Can be mounted	Select if another plate can be placed on top of this plate. This property is for collection plates that collect filtrate from filter plates during the filtration steps of a protocol. Many different plates may be able to fit under any one type of filter plate. **!! IMPORTANT !! The wells of the waste plate must have a large enough diameter that the filter plate does not stick on the waste plate. The robot must be able to pick up the filter plate without the waste plate lifting up with it.**
Maximum robot handling speed	Defines the maximum speed at which this type of plate should be moved. The general robot speed is set in BenchWorks. If the plate-specific robot speed (set here) is different from the general robot speed, the slower of the two speeds is used.
Length of filter tip/pin tool	Use calipers to measure the length (in millimeters) of the filter tip or pin tool in filter and pin tool plates.

Related information

For information about...	See...
Defining labware	❑ “Labware editor overview” on page 216 ❑ “About the defining labware process” on page 218
Editing labware parameters	“Changing labware parameters” on page 238
Adding a piece of labware	“Adding a labware entry” on page 221
Deleting a labware entry	“Deleting a labware entry” on page 222

Defining BenchCel properties

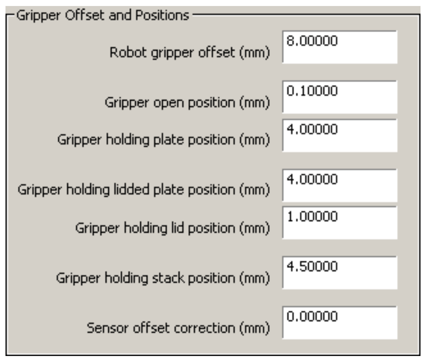
About this topic

This topic describes the properties on the BenchCel sub-page of the Labware Editor dialog box.

Note: You must also define Stacker properties because they apply to the built-in stackers of the BenchCel.

Properties

The properties on the BenchCel sub-page are described in the following sceenshot and table. All of the properties on this sub-page only apply to the BenchCel.



Property	Description
Robot gripper offset	The distance, in millimeters, from the bottom of the plate to the point where the grippers grip the plate. Typically this value is 6–10 mm. *Note:* The robot gripper offset that appears on the Plate Properties page applies to Velocity11 robots used in BioCels, not BenchCel robots.
Gripper open position	Distance between the widest possible robot gripper position and the position at which the grippers are considered to be open.
Gripper holding plate position	Distance between the widest possible robot gripper position and the position at which the grippers hold a plate.
Gripper holding lidded plate position	This value is currently not used; if you enter a value, it will have no effect.
Gripper holding lid position	Distance between the widest possible robot gripper position and the position at which the grippers close to hold a lid.

Property	Description
Gripper holding stack position	Distance between the widest possible robot gripper position and the position at which the grippers close to when they are holding a stack of plates.
Sensor offset correction	Leave this value at 0 (zero).

Defining stacker properties

About this topic

This topic describes how to define the stacker properties of the robot.

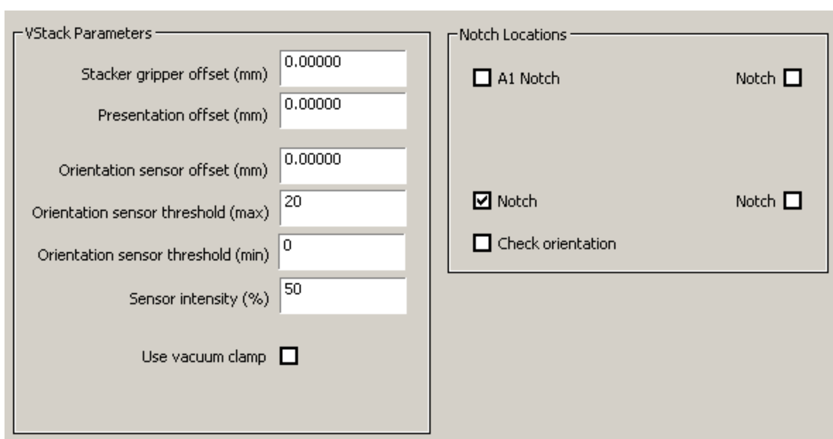
Read this topic if you are an administrator or technician who is responsible for adding and editing labware entries.

Defining properties

To define stacker properties:

1. Click the **Stacker** tab of the labware editor.
2. Enter the values for the available parameters according to the type of plate or labware you are defining.

   The properties on the **Stacker** sub-page are described in the following screenshot and table.



Property	Description
Stacker gripper offset	Adjusts the height at which the plate stage stops for the grippers to grip the plate, with respect to the stacker's Grip teachpoint. Change this value only if the stacker is not gripping the plates correctly.

Property	Description
Presentation offset	Adjusts the height of the stack. This parameter does not apply to the BenchCel and therefore does not affect the BenchCel.
Orientation sensor offset	Adjusts the height at which the orientation checking sensors “view” the plate, with respect to the bottom of the plate. If the orientation sensor offset is 0 mm, the bottom of the plate should be in the same plane as the orientation sensors.
Orientation sensor threshold (max)	Determines the highest intensity of the detected light at which the stacker senses a notch when the plate stage is in the orientation sensor position. If the stacker does not sense a notch when it should, adjust the sensor threshold value. The maximum value is 255.
Orientation sensor threshold (min)	Determines the lowest intensity of the detected light at which the stacker senses a notch when the plate stage is in the orientation sensor position. If the stacker does not sense a notch when it should, adjust the sensor threshold value.
Sensor intensity	Sets the percentage of maximum sensor intensity for all sensors. If the sensor intensity is set too low, a plate will not be detected even though one is present. If it is set too high, the sensors may become saturated, causing failure to detect the orientation of a plate. This property adjusts for the fact that clear, black, and white plates reflect light differently. For example, white plates generally reflect more light so the sensor intensity should be set lower.
Use vacuum clamp	Obsolete.
Notch locations	With the A1 well of your plate positioned in the far, left corner as you face the BenchCel, select the corresponding notch or notches for your plate in the Notch Locations group box.
Check orientation	When checked, turns on plate orientation checking.

Related information

For information about...	See...
Defining labware	❑ “Labware editor overview” on page 216 ❑ “About the defining labware process” on page 218
General properties	“Defining general properties” on page 225
Plate properties	“Defining plate properties” on page 226
BenchCel properties	“Defining BenchCel properties” on page 229

Inserting an image

About this topic

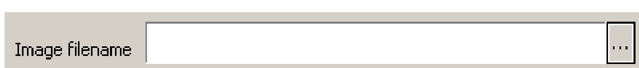
To make it easier for operators to identify a plate type, you can insert an image of each plate type in the labware editor. This topic describes how to insert an image into the labware editor.

Image files must be in the JPG, GIF or BMP format.

Procedure

To insert an image:

1. Click the **Image** sub-page tab of the labware editor.
2. Click the ellipsis button (...), and browse to the folder location of the image file.



3. Double-click the image file.

   The image appears below in the user interface.

Related information

For information about...	See...
Defining labware	❑ "Labware editor overview" on page 216 ❑ "About the defining labware process" on page 218
General properties	"Defining general properties" on page 225
Plate properties	"Defining plate properties" on page 226
BenchCel properties	"Defining BenchCel properties" on page 229

Defining labware classes

About this topic

This topic explains how to set up labware classes. Labware classes contain labware entries.

When you set up a device, you can associate labware classes with the device to indicate what labware can (and cannot) be used with the device.

Before you create labware classes, consider what labware you want used or prohibited on each of your devices.

Out-of-the-box, BenchWorks is provided with three labware classes already defined:

- ❑ Uses Filter Platepad
- ❑ Uses Standard Platepad
- ❑ Uses Vacuum Platepad

Two places to define classes

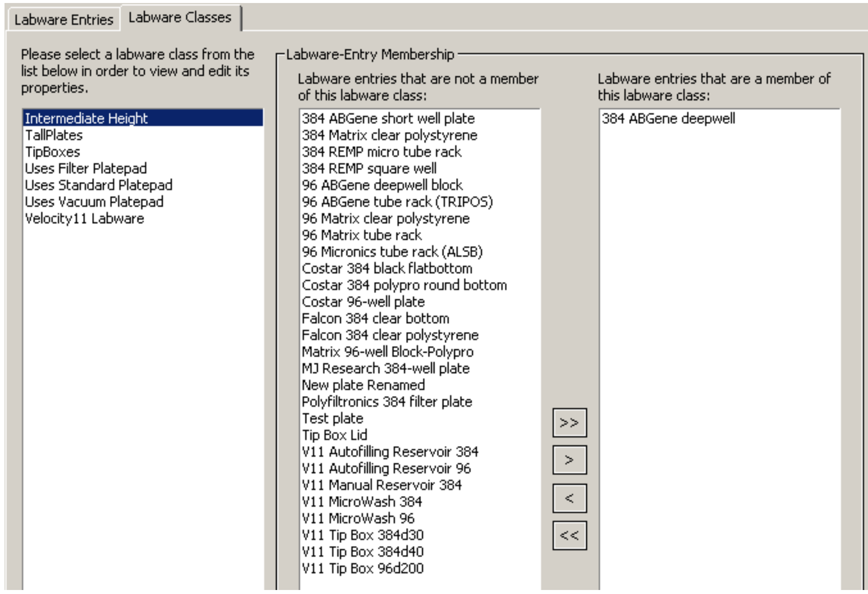
You can view and define which plate types are associated with which labware classes in:

- ❑ The Labware Classes page
- ❑ The Labware Classes sub-page of the Labware Entries page

These views present the same information in different ways.

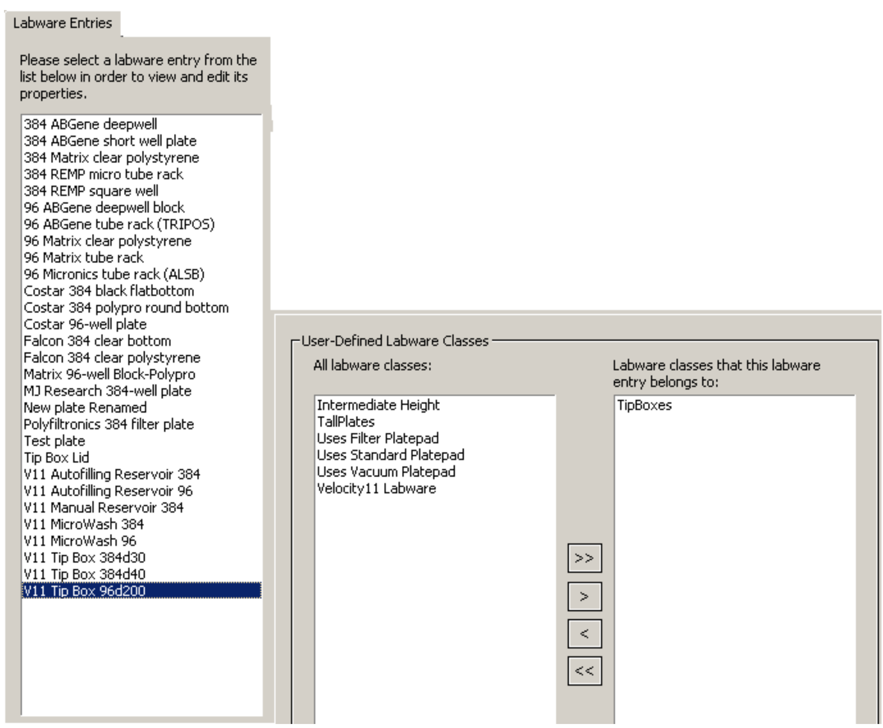
About the Labware Classes page

In the Labware Classes page, if you select a class in the labware selection box on the left, the plate types that are members of that class are displayed in the far right-hand column.



About the Labware Classes sub-page

In the Labware Classes sub-page, if you select a type of plate in the labware entry box on the left, the far right-hand column displays the classes that it is a member of.



Procedure

To add a labware class:

1. Open the labware editor.
2. In the **Labware Classes** page, click **New labware class**.
3. In the **New Labware Class** dialog box, enter a name for the labware class and click **OK**.

   The class appears in the list of labware classes.

To associate a type of plate with a labware class:

1. Open the labware editor.
2. In the **Labware Classes** page or **Labware Classes** sub-page, select an item from the middle column.

   To select more than one item, use SHIFT-click or CTRL + click. If you want to move all entries, click **> >**.
3. Click **>** to move the labware entries or labware classes to the right-hand column.
4. Click **Save Changes** to save your changes. Or, if you are in the **Labware Classes** page, you can click **Save Changes As** to save as a new labware class.

Related information

For information about...	See...
Defining labware	❑ "Labware editor overview" on page 216 ❑ "About the defining labware process" on page 218
General properties	"Defining general properties" on page 225
Plate properties	"Defining plate properties" on page 226
BenchCel properties	"Defining BenchCel properties" on page 229

About the Labware Parameters group box

About this topic

This topic briefly describes the Labware Parameters group box.

Labware parameters defined

You may want to make quick changes to the labware parameters without opening the labware editor. You can access the labware parameters directly from the Labware Parameters group box on the BenchCel Diagnostics Controls page.

The Labware Parameters group box contains the plate and BenchCel labware parameters found in the labware editor. It does not contain the parameters relevant to the any of the instruments that are integrated with the BenchCel. For editing labware classes and labware associated with the VPrep, we recommend that you use the labware editor.

When to use

Typically, the Labware Parameters group box is used when you are testing a new plate type or troubleshooting plate sensor errors.

Related information

For information about...	See...
BenchCel Diagnostics	"About the Controls page" on page 254
Labware parameters	❑ "Opening the Labware Parameters group box" on page 237 ❑ "Changing labware parameters" on page 238

Opening the Labware Parameters group box

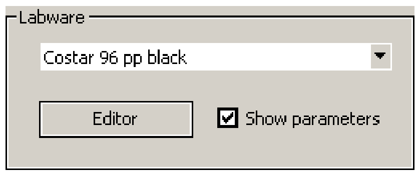
About this topic

This topic describes how to access the Labware Parameters group box so you can make changes to some of the labware settings.

Procedure

To show the Labware Parameters group box:

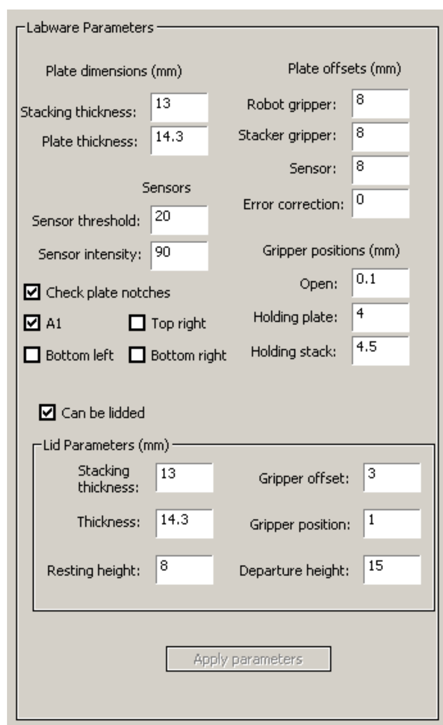
1. Open **BenchCel Diagnostics**.
2. In the **Controls** page of the **BenchCel Diagnostics** dialog box, select a labware type from the list box.



3. Select the **Show parameters** check box.

   The **Labware Parameters** group box is displayed on the right side of the **Controls** page.

   The values displayed in the **Labware Parameters** group box are for the plate selected.



Related information

For information about...	See...
The BenchCel Controls page	“About the Controls page” on page 254
Opening BenchCel Diagnostics	“Opening BenchCel Diagnostics” on page 253
Changing labware parameters	“Changing labware parameters” on page 238

Changing labware parameters

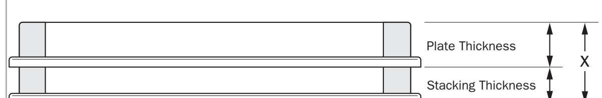
About this topic

When adding new labware or troubleshooting existing labware you may need to make adjustments to the labware parameters. This topic describes how to edit labware parameters using the Labware Parameters group box.

Procedure

To change labware parameters:

1. Ensure that you select the desired plate type from the list box.
2. Enter new values or edit existing values for the **Plate Dimensions**, using the following table as a guide.

Parameter	Comment
Stacking thickness	The thickness, in millimeters, of two stacked plates minus the thickness of one plate. Measure using calipers. Example: Thickness of two stacked plates (x) = 23.14 mm Thickness of one plate = 14.14 mm Stacking thickness: 23.14 mm - 14.14 mm = 9.00 mm 
Plate thickness	The distance, in millimeters, from the bottom surface of the plate to the top surface of the plate. Measure using calipers.

3. Enter new values or edit existing values for the **Sensors**, using the following table as a guide.

Parameter	Comment
Sensor threshold	Sets the sensor threshold value that determines the presence of a notch. If the sensor value is below this threshold, the device reports a notch in the corner corresponding to the orientation sensor. If the sensor value is above this threshold, no notch is reported. Increase this value if the orientation notch sensor readings are too high with a notch present. The default value is 100. To check the sensors, see "Checking the stacker sensors" on page 314.
Sensor intensity	This setting changes the intensity of the light emission that the sensors use to determine the presence of a notch or plate. Some plates are more reflective than others. Increase the sensor intensity if plate sensor readings are too low (not significantly higher than the corresponding threshold value) when either a plate is present, or a notch is absent. The default value is 50. Changing this setting affects the numeric value of the four notches sensors in the Stacker Sensors group box. See "Checking the stacker sensors" on page 314.
Check plate notches	Turns on plate orientation checking. Select the appropriate notches for the type of plate you are using. These are called A1 (for the top left notch), Top right, Bottom left, and Bottom right.

4. Enter new values or edit existing values for the **Plate Offsets**, using the following table as a guide

   Each offset is measured from the bottom of the plate in millimeters.

Parameter	Comment
Robot gripper	Determines where the robot grips the plate
Stacker gripper	Determines where the stacker grips the plate
Sensor	Determines where the plate sensors read the plates for orientation checking
Error correction	Leave this value at 0 (zero).

5. Enter new values or edit existing values for the **Gripper position parameters**, using the following table as a guide:

   !! DAMAGE HAZARD !! Set the Holding plate and Holding stack parameters so the grippers securely grasp plates without damaging them.

   For each of these offsets:

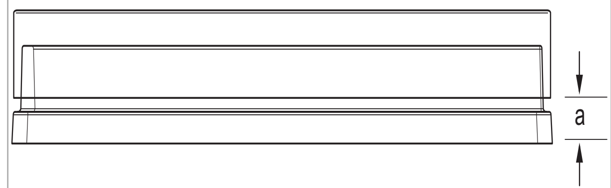
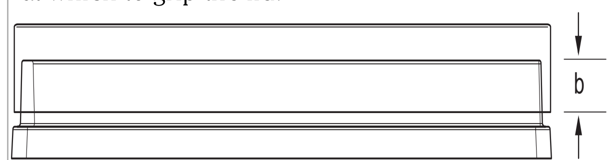
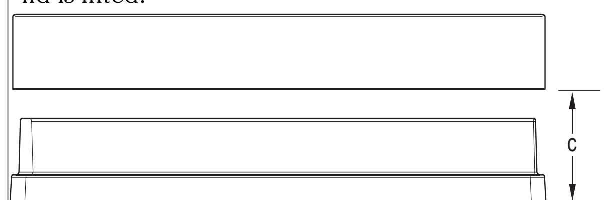
   - Increasing the value narrows the distance between the robot grippers.
   - Decreasing the value widens the distance between the robot grippers.
   - The value should never be less than zero.
   - This value adjusts the position of a single gripper and is not a point to point adjustment between grippers.

Parameter	Comment
Open	Adjusts the distance between the robot grippers when open. Set this parameter so that the grippers release the plate when opened.
Holding plate	Adjusts the distance between the robot grippers when holding a plate.
Holding stack	Adjusts the distance between the robot grippers when holding a stack of plates. This value is usually larger than the holding plate value because the grippers need to grip a stack of plates harder than a single plate to support the weight of the stack.

6. If your plates can be lidded, select the **Can be lidded** check box.

   You can now configure the lid parameters, using the following table as a guide:

Parameter	Comment
Can be lidded	Select if the plate can be lidded. The following settings are only available if this setting is selected.
Stacking thickness	Stacking thickness of the plate with a lid in place.
Thickness	The distance, in millimeters, from the bottom surface of the of a lidded plate to the top surface of the lid. Measure using calipers.

Parameter	Comment
Resting height	Height, in millimeters, above the bottom of the plate at which the bottom of a plate lid rests. 
Gripper offsets	Height, in millimeters, above the lid resting height at which to grip the lid. 
Gripper position	Adjusts the distance between the robot grippers when holding a lid.
Departure height	Height above the bottom of the plate to which the lid is lifted. 

7. Click **Apply parameters** for the changes to take effect.

Related information

For information about...	See...
The BenchCel Controls page	"About the Controls page" on page 254
Opening BenchCel Diagnostics	"Opening BenchCel Diagnostics" on page 253

Setting liquid-handling definitions 11

This chapter is for people with administrator or technician login privileges. It describes the liquid library editor, which is used to set parameters that affect pipetting speed, accuracy and precision, and to save the parameters as classes for convenient reuse.

About the liquid library editor

Accessing the liquid library editor

The liquid library editor is only available when you have a VPrep or other liquid handling device connected to the BenchCel. When you have a liquid handling device connected to the BenchCel, you can access the liquid library editor through the diagnostic page of the VPrep or liquid handling device.

Liquid library editor defined

The liquid library editor is a dialog box through which users with technician or administrator privileges can enter values for properties that affect pipetting speed, accuracy and precision.

Liquid classes defined

The values entered into the liquid library editor can be saved as a collection, known as a liquid class. Using liquid classes saves time when writing protocols because you do not have to enter values for the liquid properties every time you create a protocol.

Liquid library database defined

The data that represents a liquid class is saved to the liquid library database, which is maintained in the Windows registry.

Using a liquid class

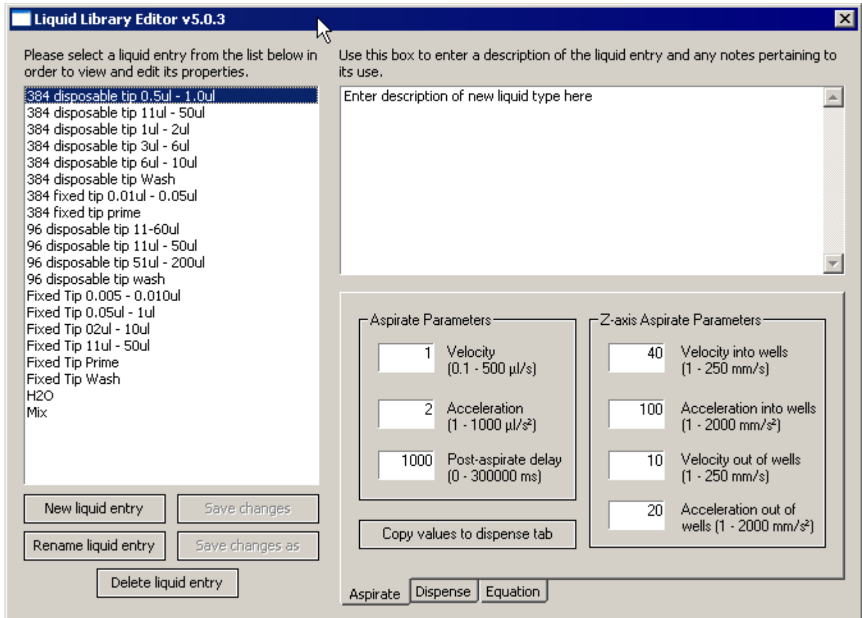
When preparing for a protocol run, you select the liquid class that you want to use. Then, during the run, the liquid class values are referenced for pipetting operations.

Calibrating the VPrep

The liquid library editor also has an equation editor that can be used to calibrate the VPrep.

Liquid Library Editor dialog box

A screenshot of the liquid library editor is shown below.



Related information

For information about...	See...
Opening the liquid library editor	“Opening the liquid library editor” on page 246
Creating a new liquid class	“Creating a liquid class” on page 248

Opening the liquid library editor

About this topic

This topic explains how to open the liquid library editor.

You open the liquid library editor when you want to:

- ❑ View the properties that are defined for a liquid class
- ❑ Edit the properties that are defined for a liquid class
- ❑ Add new liquid classes

You must be logged on with an administrator or technician user account to open the liquid library editor.

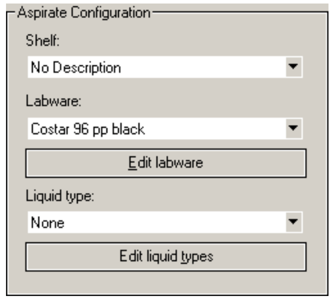
Procedure

To open the liquid library editor from BenchWorks:

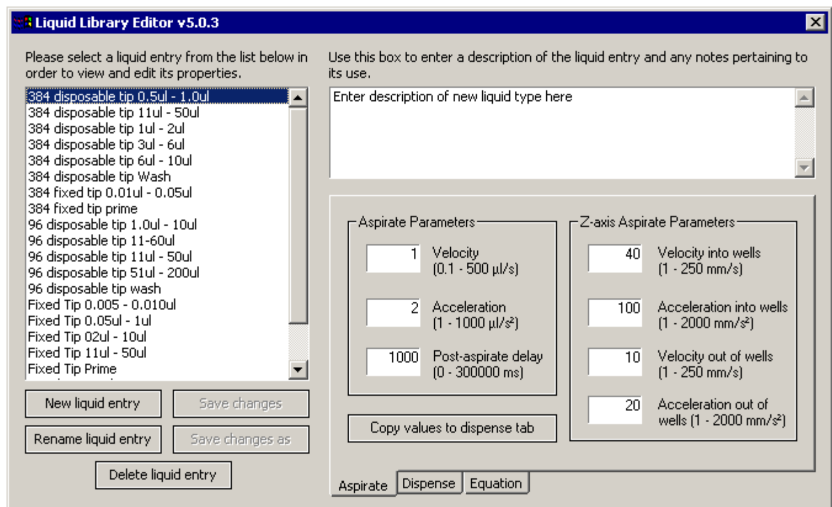
1. Click the **Device Manager** tab.
2. In the **Device List**, select **the VPrep device**.
3. Click **Device Diagnostics**.

   The **VPrep Diagnostics** dialog box opens.
4. Click the **Processes** tab.
5. Click on one of the **Process** buttons.

6. In the **Configuration** group box, which in this case is **Aspirate**, click **Edit liquid type**.



The **Liquid Library Editor** dialog box opens.



Related information

For information about...	See...
The liquid library editor	"About the liquid library editor" on page 244
Creating a new liquid class	"Creating a liquid class" on page 248

Creating a liquid class

About this topic

This topic describes how to create a liquid class using the liquid library editor.

You must be logged on with an administrator or technician user account to perform this procedure.

Types of liquid classes to create

You may want to create different classes for different:

- ❑ Types of liquids

  For example, water versus DMSO
- ❑ Volumes of liquids

  For example, 1 µL versus 200 µL
- ❑ Liquid operations

  For example, washing versus mixing

Liquid compatibility

!! INJURY HAZARD !! Velocity11 products are intended to be used with non-hazardous aqueous liquids. Please contact Velocity11 before using any non-aqueous solvents, or solvents generally considered to be hazardous.

Procedure

To create a liquid class:

1. Open the liquid library editor.
2. Click **New liquid entry**.
3. In the **New Liquid Entry** dialog box, enter a name for the liquid class and click **OK**.
4. In the list box at the top left, replace the text **New Liquid Type** with a name for the new liquid.

   This is the name of the liquid class.
5. In the **Note** text box at the top right, type a note describing the liquid library entry for your records.
6. Enter values for the aspirate properties.

   The following table describes these properties.

Aspirate property	Definition
Velocity	Specifies the speed of the aspiration stroke, in microliters per second.
Acceleration	*VPrep only*. Specifies acceleration during the aspiration stroke, in microliters per second squared.
Z-axis velocity into wells	Specifies how fast the pipettor moves as the tips enter the wells, in millimeters per second.

Aspirate property	Definition
Z-axis acceleration into wells	*VPrep only.* Specifies the acceleration of the pipettor as the tips move into the wells, in millimeters per second squared.
Z-axis velocity out of wells	Specifies how fast the tips leave the wells, in millimeters per second.
Z-axis acceleration out of wells	*VPrep only.* Specifies the acceleration of the pipettor as the tips move out of the wells, in millimeters per second squared.
Post-aspirate delay	Specifies the time the pipettor waits after aspiration is complete before moving the tips out of the wells, in milliseconds.

7. Enter values for the dispense properties.

   The following table describes these properties.

Dispense property	Definition
Velocity	Specifies the maximum speed of the dispensing stroke, in microliters per second.
Acceleration	*VPrep only.* Specifies acceleration during the dispensing stroke, in microliters per second squared.
Z-axis velocity into wells	Specifies how fast the pipettor moves as the tips enter the wells, in millimeters per second.
Z-axis acceleration into wells	*VPrep only.* Specifies the acceleration of the pipettor as the tips enter the wells, in milliliters per second squared.
Z-axis velocity out of wells	Specifies how fast the pipettor moves as the tips leave the wells, in millimeters per second.
Z-axis acceleration out of wells	*VPrep only.* Specifies the acceleration of the pipettor as the tips leave the wells, in millimeters per second squared.
Post-dispense delay	Specifies the time the pipettor waits after the dispense stroke before moving the tips out of the wells, in milliseconds.

8. Click **OK**.

   The changes are now stored in the liquid library database.

Related information

For information about...	See...
Opening the liquid library editor	"Opening the liquid library editor" on page 246
The liquid library editor	"About the liquid library editor" on page 244

Using BenchCel Diagnostics

12

This chapter explains how to use diagnostics software to control the robot and modules.

We recommend that only administrators and experienced personnel use the procedures in this chapter to diagnose errors with the BenchCel.

Velocity11 has also developed diagnostics software for other companies' modules that can be used with the BenchCel. Instructions for using this software are given in this guide.

About diagnostics

Background

The BenchCel can be controlled in real time directly through the BenchWorks diagnostics software using simple commands.

Diagnostics software is used for:

- Troubleshooting
- Setting teachpoints
- Performing manual operations outside a protocol

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

Types of diagnostics software

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

Related information

For information about...	See...
Opening diagnostics	"Opening BenchCel Diagnostics" on page 253
Controls page	"About the Controls page" on page 254
General Settings page	"About general settings" on page 277
Profiles page	"About profiles" on page 282

Opening BenchCel Diagnostics

About this topic

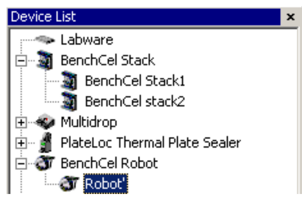
This topic describes how to open BenchCel Diagnostics.

Procedure

To open BenchCel diagnostics:

1. Click the **Device Manager** tab.
2. Select the BenchCel robot 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 driver's **Diagnostics** dialog box opens.

Related information

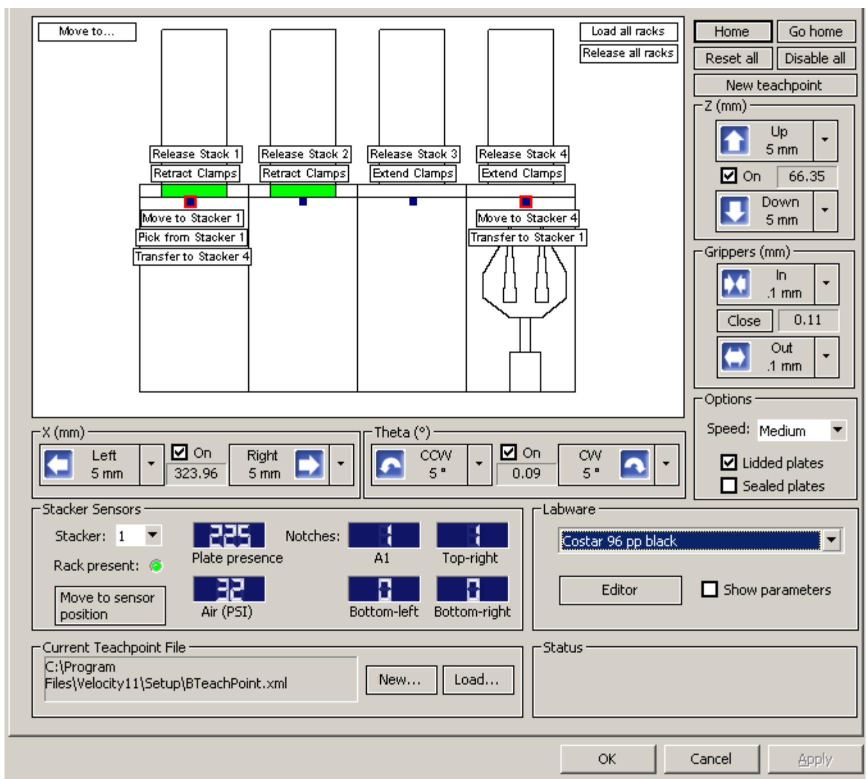
For information about...	See...
Diagnostics	"About diagnostics" on page 252
Controls page	"About the Controls page" on page 254
General Settings page	"About general settings" on page 277
Profiles page	"About profiles" on page 282

About the Controls page

Contents of the Controls page

The Controls page is the first page displayed when you open the BenchCel Diagnostics dialog box. It provides:

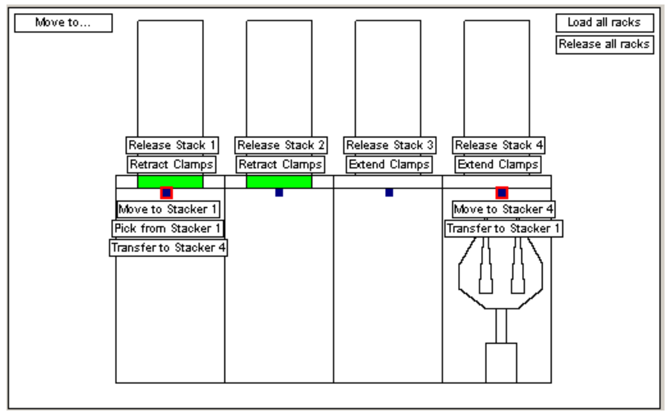
- ❑ An interface for controlling the robot and defining teachpoints
- ❑ An interface for creating and loading teachpoint files
- ❑ Controls for homing the robot, sending it home, and adjusting its speed
- ❑ Controls for moving the robot incrementally in each of its axes
- ❑ Stacker sensor readouts
- ❑ An interface for changing labware settings



The BenchCel animated display

The BenchCel animated display provides a visual way to represent teachpoints, control the BenchCel and monitor the robot's movement. The graphical representation of the robot moves in real time, mirroring the movement of the actual robot.

Typically, the display is used for defining and managing teachpoints. See "About teachpoints" on page 269 for more information.



Robot control procedures

The following are procedures you can perform from the Controls page:

Procedure	Control	See...
Home the robot	Home	"Homing the robot" on page 257
Send the robot home	Go home	"Sending the robot home" on page 257
Reset all robot servos	Reset all	"About stacker controls" on page 264
Disable all robot servos	Disable all	"About teachpoints" on page 269
Add a new teachpoint	New teachpoint	"About teachpoints" on page 269
Check the stacker sensors	Stacker Sensors	"Checking the stacker sensors" on page 314
Change labware parameters	Labware	"About the Labware Parameters group box" on page 236
Change the speed of the robot	Speed: Medium	"Changing the robot speed" on page 261
Jog the robot		"About the Controls page" on page 254

Robot axes

You can control the robot's movement in three directions:

Axis	Description
X	The horizontal distance, in millimeters, along the rail from the home position.
Theta	The angle, in degrees, that the grippers rotate from their home position
Z	The vertical distance, in millimeters, from the home position to a height 20 millimeters above the base of the column.

Related information

For information about...	See...
Diagnostics	"About diagnostics" on page 252
Opening diagnostics	"Opening BenchCel Diagnostics" on page 253
General Settings page	"About general settings" on page 277
Profiles page	"About profiles" on page 282

Homing the robot

About this topic

This topic describes when and how to home the BenchCel robot.

Homing

Home the robot to make sure that all of its axes are calibrated correctly. For example, if you notice the robot is not accurately picking up or placing plates, home the robot.

Sending home

Send the robot to its home position when you want to quickly move it to a central location, or out of the way.

Homing the robot

When you command the robot to home, it executes a sequence that resets the home position in the *x*, *y*, gripper, and *theta* axes.

!! INJURY HAZARD !! Keep away from the robot when it is moving, especially in the *z*-axis direction. The robot's *z*-axis motor is particularly powerful. It might not stop immediately in a collision, and a gripper could pierce your hand.

To home the robot:

1. Open **BenchCel Diagnostics** dialog box to access the **Controls** page.
2. Click **Home**.

   The robot homes.

   The axis order of homing is gripper —> *z* —> *x* —> *theta*.

Sending the robot home

When you command the robot to go home, it moves to the zero position of the *x*, *z*, and *theta* axes.

To send the robot to the home position:

1. Open **BenchCel Diagnostics** dialog box to access the **Controls** page.
2. Click **Go Home**.

   Note: If you have cleared one or more **On** check boxes or have clicked **Disable all** to disable the servos for robot axes, a dialog box will display, "Some of the Robot's servos are inactive. Enable servos?"

   You must click **Yes** to enable the servo motors so the robot can go home.

Related information

For information about...	See...
Opening diagnostics	"Opening BenchCel Diagnostics" on page 253

Moving the robot manually

About this topic

This topic describes how to move the robot in manual mode. You can move the robot manually in the *x*, *z*, and *theta* axes, but first you must disable the robot's servo motors.

Move the robot manually when you want to position the robot by eye exactly where you want. This is useful when you are creating a new teachpoint.

For example, you could move the robot manually to a platepad and position its grippers to where they would grip a plate. Then you could create a teachpoint there by pressing Use current position on the Teachpoint Details dialog box.

Procedure

To move the robot manually:

1. Open **BenchCel Diagnostics** dialog box to access the **Controls** page.

   !! DAMAGE HAZARD !! **Moving the robot in any of its axes without first disabling the servos could damage the robot's motors.**

2. Disable the robot servo motors for the *x*, *z*, and *theta* axes:

   a. Click **Disable all** on the **Controls** page of the **BenchCel Diagnostics** dialog box.

      If the robot is above its lowest *z*-axis height, it will slowly sink until it reaches the lowest *z*-axis height.

   b. Alternatively, you can clear the **On** check boxes for the *x*, *z*, and *theta* axes in the **Controls** page of the **BenchCel Diagnostics** dialog box.

      Note: The robot head will begin to fall to its lowest z-axis position when the **On** check box is cleared.

3. Move the robot's *x*, *z*, and *theta* axes manually.

   Any subsequent commands will trigger a dialog box that states, **"Some of the Robot's servos are inactive. Enable servos?"**

   You must click **Yes** before the robot can move autonomously again.

Related information

For information about...	See...
Teachpoints	"About teachpoints" on page 269
Opening diagnostics	"Opening BenchCel Diagnostics" on page 253

Jogging the robot

About this topic

Jogging is the process of moving the robot in small increments.

You can jog the robot head in the *z*, *x*, or *theta* axes and jog the grippers open and closed.

Jog the robot when you need to move it in small increments.

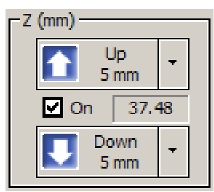
Jogging is useful when creating teachpoints. For example, you could move the robot manually to a position you want to set as a teachpoint. After you have re-enabled the robot's servos, you can fine tune the robot's position by jogging it.

Jogging the robot in the *z*-axis

!! DAMAGE HAZARD !! Before you jog the robot, make sure that there is nothing that the robot can crash into. Move the robot slowly and keep the jog increment small until you are certain that there is no obstruction.

To jog the robot in the z-axis:

1. Open **BenchCel Diagnostics** dialog box to access the **Controls** page.
2. Select the increment that you want the robot to move by:

   a. Click the list box down arrow for either the **Up** or **Down** movement controls

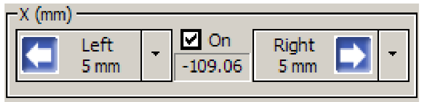
   b. Select one of the increment options. (These are listed in millimeters.)
3. Click a blue **Up** or **Down** arrow to move the robot head in the *z*-axis (vertically).
4. Monitor the current position of the robot by checking the number listed between the blue **Up** or **Down** arrows. This shows the number of millimeters the robot is from the home position.

   !! INJURY HAZARD !! Keep away from the robot when it is moving, especially in the *z*-axis direction. The robot's *z*-axis motor is particularly powerful. It might not stop immediately in a collision, and a gripper could pierce your hand.

Jogging the robot in the *x*-axis

To jog the robot in the x-axis:

1. Select the increment that you want the robot to move by:



   a. Click the list box down arrow for either the **Left** or **Right** movement controls.

   b. Select one of the increment options. These are listed in millimeters.

2. Click a blue **Left** or **Right** arrow to move the robot head in the *x*-axis (horizontally).

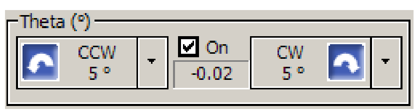
3. Monitor the current position of the robot by checking the number listed between the blue **Left** or **Right** arrows. This shows the number of millimeters the robot is from the home position.

   !! INJURY HAZARD !! Keep away from the robot when it is moving. The robot is designed to stop if it contacts an object, but movement along the *x*-axis can build up considerable momentum resulting in injury.

Jogging the robot in the *theta*-axis

To jog the robot in the theta-axis:

1. Select the increment that you want the robot to move by:



   Click the list box down arrow for either the clockwise (**CW**) or counterclockwise (**CCW**) movement controls and select one of the increment options. These are listed in millimeters.

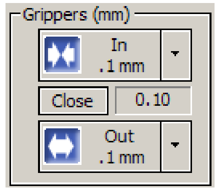
2. Click a blue **CW** or **CCW** arrow to rotate the robot head grippers around the *theta*-axis.

3. Monitor the current position of the robot by checking the number listed between the blue **CW** or **CCW** arrows. This shows the angle that the grippers are pointing by degrees.

Jogging the robot gripper position

To jog the robot gripper position:

1. Select the increment that you want the robot to move by:



   Click the list box down arrow for either the **In** or **Out** movement controls and select one of the increment options.
2. Click a blue **In** or **Out** arrow to change the spacing between the gripper points.
3. Monitor the current position of the robot by checking the number listed between the blue **In** or **Out** arrows. This shows the number of millimeters the robot is from the home position.

Related information

For information about...	See...
Teachpoints	"About teachpoints" on page 269
Opening diagnostics	"Opening BenchCel Diagnostics" on page 253
Controls page	"About the Controls page" on page 254

Changing the robot speed

About this topic

It is useful to change the robot speed when creating new protocols or teachpoints.

For example, change the robot speed to Slow when you are experimenting with new teachpoints and protocols so that you can monitor the robot's accuracy more closely.

After you have observed the robot move to a new teachpoint or complete a protocol without errors, change the robot speed to Medium.

Change the robot speed to High when the robot is able to perform a protocol without errors and you have eliminated inaccuracies in new teachpoints and protocols.

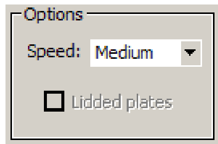
If the robot starts to have errors, set the robot speed lower until you can diagnose the problem.

Note: It is safe to move at high speed with default Velocity11 teachpoints and protocols, because they have been extensively tested for accuracy.

Procedure

To change the robot speed:

1. Open **BenchCel Diagnostics** dialog box to access the **Controls** page.
2. Click the down-arrow of the **Speed** list box in the **Options** group box.



3. Select a speed from the drop-down menu:
   - Slow
   - Medium
   - Fast
4. Click **OK** at the bottom of the **Controls** page for the changes to take effect.

Related information

For information about...	See...
Opening diagnostics	"Opening BenchCel Diagnostics" on page 253

Using Move-To-Position commands

About this topic

The BenchCel Move-to-Position commands let you quickly move the robot to a set of *x*, *z*, and *theta* coordinates that you can specify with sliders. Each slider control represents a point on a spectrum that spans the full range of movement for a robot axis.

The Move-to-Position commands can be accessed from the Move to... button, displayed in the upper left corner of the BenchCel animated display.

After you have become familiar with jogging the robot and you have a sense for how the robot moves, use the Move-to-Position commands to move the robot much more quickly than you can with jogging commands.

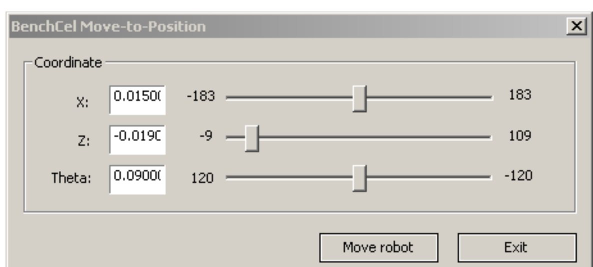
!! DAMAGE HAZARD !! Make sure the robot is clear of any obstacles that you have not defined. BenchWorks will display a dialog box warning you of a possible collision if the requested movement overlaps a known teachpoint. The Move-to-Position commands present a higher damage risk to the robot, because you command it to move in large increments.

Procedure

To move the robot with the Move-to-Position command:

1. Make sure that you have selected a **Plate Type** to use from the **Labware** list box.
2. In the BenchCel animated display, click **Move to...**

   The **BenchCel Move-to-Position** dialog box appears.

3. Adjust the sliders controls for each axis you want to change.

   !! INJURY HAZARD !! Make sure everyone is clear of the BenchCel before you command the robot to move.
4. Click **Move robot**.

   The robot axes will move to their commanded positions.
5. To close the dialog box, click **Exit**.

Related information

For information about...	See...
Animated display	"About the Controls page" on page 254

About stacker controls

Stacker clamps

The stacker controls are located in the BenchCel animated display. Each stacker has two clamps that perform two functions:

- Anchor the stacker rack to the stacker head
- Grip and release plates

You can command the BenchCel stacker clamps to grip plates or release plates with the **Extend Clamps** and **Retract Clamps** controls.

You can command the BenchCel stack clamps to load and release stacker racks with the **Load Stack** and **Release Stack** controls.

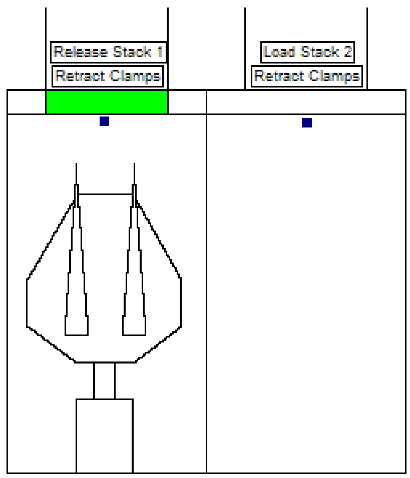


Plate-present indicator

The plate-present indicator above each stacker teachpoint marker will change color depending on the presence of a stack with plates.

Color	Meaning
Green	A stack with plates is loaded correctly, and the BenchCel is ready for you to start a protocol.
Yellow	A stack with plates is present, but the robot grippers are blocking the stacker sensor.
Blank	There are no plates present at the stacker sensor level, regardless of the presence of a stack.

Related information

For information about...	See...
Teachpoints	"About teachpoints" on page 269
The animated display	"About the Controls page" on page 254

Loading and releasing stacks

About this topic

This topic describes when and how to load and release BenchCel stacks.

When to use

Stacker rack loading and unloading normally take place automatically in the course of performing a run. After you place the stacker racks onto the stacker heads and start a run, the stacker racks are automatically secured. After the run has completed, the software asks if you want to release the stacks.

When diagnosing problems, you may want to manually load and release stacker racks at your convenience. Do this from the BenchCel animated display.

Loading and releasing defined

The process of loading stacker racks is the securing of the stacker onto a stack head so that the BenchCel robot can access the stack of plates. When you start a run, the BenchCel automatically secures the stack and prepares the plates. This process involves both the stacker clamps and the robot.

The BenchCel stacker will not operate until the stacker rack is loaded.

The process of releasing involves the BenchCel repositioning the stack of plates so that the stack is free to move and the plates are secured within the stack. The stack becomes free to move after the robot moves the bottom plate up so it rests on top of the clamp pads.

After a run has completed, BenchWorks will query whether you want to release the stacks or not.

Two ways to load and release

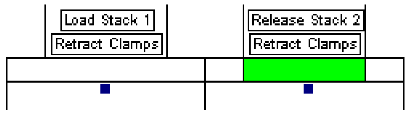
There are two ways to load and release stacks in the BenchCel animated display:

- ❑ Load or release stacks individually with the **Load stack** or **Release stack** buttons above each stacker representation.
- ❑ Load or release all stacks simultaneously with the **Load all stacks** or **Release all stacks** buttons.

Loading a stack using the display

To load a stack:

1. Verify that the button displays **Load Stack** in the BenchCel animated display. Refer to the button at the top left of the following screenshot.



2. Click **Load Stack** once to command the robot to move to the stacker and grab the plate on the bottom of the stack.

   The name of the button changes to **Release Stack**.

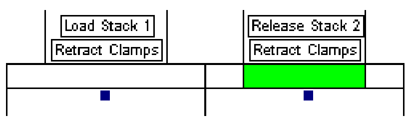
   The stacker clamps retract and release the stack and the robot moves the stack down. Then the stacker clamps extend and grab the plate stack by the stacker grippers. The robot then releases the stack.

   The rack is loaded.

Releasing a stack using the display

To release a stack:

1. Verify that the button displays **Release Stack** in the BenchCel animated display. Refer to the button at the top right of this screenshot.



2. Click **Release Stack** once to command the robot to move to the stacker and hold the plate on the bottom of the stack.

   The name of the button changes to **Load Stack**.

   The stacker grippers retract and release the stack, while the robot moves the stack up. Then the stacker clamps extend and the robot releases the stack so that it is resting above the stacker grippers.

   The rack is ready to remove from the stacker.

   !! DAMAGE HAZARD !! Always remove stacks from the stackers before attempting to move the BenchCel.

3. Lift the stacker rack up and off the base.

Loading and releasing all stacks

To speed up your workflow, you may want to load or release all of your stacks simultaneously.

To load all stacks simultaneously:

1. Load the stacks onto their stack base.
2. Click **Load all stacks** in the upper right corner of the BenchCel animated display.

   The stacker grippers will take hold of the plate clamps on the stack.

To release all stacks simultaneously:

1. Click **Release all stacks**.
2. Unload the stacks from the stacker heads.

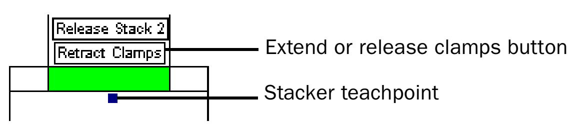
Related information

For information about...	See...
Loading or releasing all stacker racks simultaneously	"Loading and releasing stacks" on page 265
The animated display	"About the Controls page" on page 254

Extending and retracting stacker clamps

About this topic

Above each stacker teachpoint marker in the BenchCel animated display, there is a button for extending or releasing the clamps.



It is important to understand the difference between extending and releasing stacker clamps and loading and releasing a stack.

Unlike the load and release stack commands, the extend and retract stacker clamp commands do not use the robot. The clamp actuators are the only mechanisms commanded to move.

- ❑ Extending clamps moves them away from the walls of the stacker head
- ❑ Retracting clamps draws them in against the walls of the stacker head

When to use

Extend or retract clamps when you need to diagnose a problem you have performing a run.

For example, if you want to load a stack, but the clamps are retracted, use the **Extend Clamps** command to extend the clamps.

If there is a single plate in the stacker grippers which you are using to diagnose the BenchCel with, it may be faster to **Retract Clamps** and catch the plate with your hand than to wait for the robot to move the plate after you issue the **Release Racks** command.

Do not use the extend and retract clamps controls to load or release a stack. To load and release stacks, use the **Load Racks** and **Release Racks** commands.

!! DAMAGE HAZARD !! Releasing the clamps when the robot is not supporting a stack of plates will drop the plates.

Extending clamps

To extend clamps:

1. Verify that the button displays **Extend Clamps.**
2. Click **Extend Clamps** once to extend the stack clamps.

   The name of the button changes to **Retract Clamps**.

   The clamps extend.

   If a stack is present at the stacker you commanded, the stack is released by the clamps. If the robot is holding a plate within the stacker, the clamps grab the plate.

Retracting clamps

To retract clamps:

1. Verify that the button displays **Retract Clamps**.
2. Click **Retract Clamps** once to retract the stack clamps.

   A confirmation dialog box opens, “Doing this might cause plates to drop. Are you sure you want to open the clamps?” Plates might drop because the clamp actuators grab the stack, but release any plates that might be present in the stack.
3. If there are plates in the stack, make sure the robot is holding the plate on the bottom of the stack.
4. Click **OK.**

   The clamps retract.

Related information

For information about...	See...
The animated display	“About the Controls page” on page 254
Loading and releasing stacks	“Loading and releasing stacks” on page 265

About teachpoints

Teachpoints defined

A teachpoint is a point in space that the robot can move its grippers to and is defined by a set of axial coordinates. A teachpoint is where the robot moves plates to or from.

Teachpoints also represent physical objects. This is very important for collision avoidance.

After you create a teachpoint, the robot will automatically navigate around the teachpoint during a run. If you tell the robot to move, and the teachpoint is in the robot's path, the software will ask you before moving the robot through it.

Teachpoints are shown graphically as square dots called teachpoint markers. They are displayed and manipulated in the BenchCel animated display on the Controls page of the BenchCel Diagnostics dialog box.

Related information

For information about...	See...
Teachpoints	❑ "About teachpoints" on page 269 ❑ "Editing a teachpoint" on page 272 ❑ "Deleting a teachpoint" on page 273 ❑ "Managing teachpoint files" on page 274 ❑ "Moving between teachpoints" on page 275 ❑ "Selecting two teachpoints" on page 276

Adding a teachpoint

About this topic

This topic describes how to add a new teachpoint.

A teachpoint is a set of coordinates that the robot can move its grippers to, for the purpose of picking up or dropping off plates. Each teachpoint is represented by a teachpoint marker in the BenchCel animated display.

New teachpoints are automatically saved to your current teachpoint file.

Add a new teachpoint to define where the robot can pick up and drop off plates on an instrument or platepad.

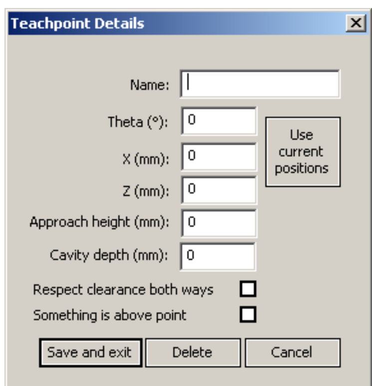
You can also add teachpoints for objects that you want the robot to avoid colliding with.

Procedure

To add a new teachpoint:

1. Click **New teachpoint** on the **Controls** page of the **BenchCel Diagnostics** dialog box.

   The **Teachpoint Details** dialog box is displayed.



2. Complete the dialog box, using the following table as a guide:

Setting	Comment
Name	A name for the teachpoint. This name will appear in the BenchCel animated display.
Theta	Sets the value of *theta*, or the angle that the grippers will rotate from their home position, in degrees.
X	Sets the value of *x*, or the horizontal distance from the home position, in millimeters.
Z	Sets the value of *z*, or the vertical distance from the home position to a height 20 millimeters above the base of the column, in millimeters.
Use current positions	This will create a teachpoint from the robot's current axes values.

Setting	Comment
Approach height	As the robot approaches a teachpoint such as a platepad or module, this is the height the robot maintains above that teachpoint to avoid colliding with it. This setting only applies when the robot is carrying a plate. When the robot is not carrying a plate, it will approach the teachpoint along the *x*-axis at the height of the teachpoint, unless the Respect clearance both ways check box is selected. This height is the same as the clearance above the teachpoint and is measured in millimeters.
Cavity depth	Specifies the height above the normal grip height that the robot must grip the plate. Some platepads prevent the robot from gripping a plate at the normal grip height. The cavity depth is measured in millimeters.
Respect clearance both ways	When selected, the robot moves to and from the teachpoint at the approach height, even if it is not carrying a plate. If not selected, the robot approaches and retreats from the teachpoint at the height of the teachpoint when it is not carrying a plate.
Something is above point	Prevents the robot from moving in the space above the teachpoint.

3. Click **Save and exit** to save the teachpoint and close the **Teachpoint Dialog** box.

Related information

For information about	See...
Teachpoints	"About teachpoints" on page 269
Editing teachpoints	"Editing a teachpoint" on page 272
Deleting teachpoints	"Deleting a teachpoint" on page 273
Managing teachpoints	"Managing teachpoint files" on page 274
Moving between teachpoints	"Moving between teachpoints" on page 275
Selecting teachpoints	"Selecting two teachpoints" on page 276

Editing a teachpoint

About this topic

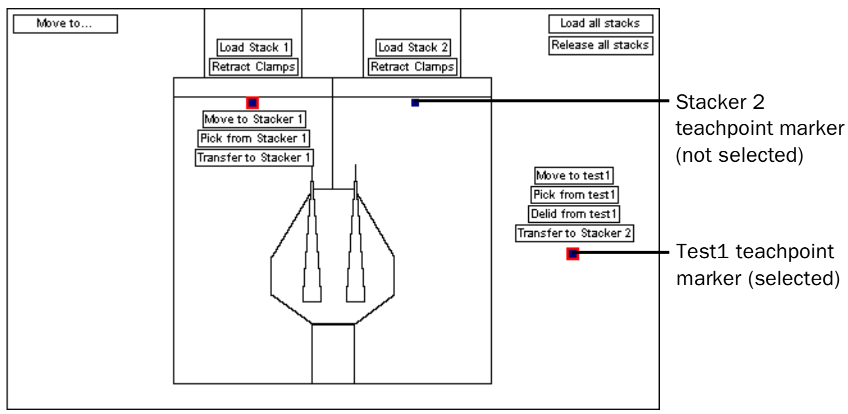
After you have created a teachpoint, you can edit it.

Edits to your teachpoints are automatically saved to your current teachpoint file.

Procedure

To edit a teachpoint:

1. Double-click the teachpoint marker that corresponds to the teachpoint you want to change. Teachpoint markers are little square dots in the BenchCel animated display.



   The **Teachpoint Details** dialog box is displayed with values of the teachpoint in the text boxes.

2. Do one of the following:

   ◆ Modify the contents of the text boxes or check boxes using the information from “About teachpoints” on page 269.

   ◆ Follow the procedure in “Moving the robot manually” on page 258, then click **Use current positions**.

Related information

For information about	See...
Teachpoints	“About teachpoints” on page 269
Adding teachpoints	“Adding a teachpoint” on page 270
Deleting teachpoints	“Deleting a teachpoint” on page 273
Managing teachpoints	“Managing teachpoint files” on page 274
Moving between teachpoints	“Moving between teachpoints” on page 275
Selecting teachpoints	“Selecting two teachpoints” on page 276

Deleting a teachpoint

About this topic

Occasionally, you may need to delete a teachpoint. For example, if you want to replace one of your modules with another, you would first delete the teachpoint for the old module, and then add a teachpoint for the new one.

Procedure

To delete a teachpoint:

1. Double-click the teachpoint marker that represents the teachpoint you want to delete.

   The **Teachpoint Details** dialog box is displayed.
2. Click **Delete**.
3. Your current teachpoint file will automatically be updated when you delete a teachpoint.

Related information

For information about	See...
Teachpoints	"About teachpoints" on page 269
Editing	"Editing a teachpoint" on page 272
Adding	"Adding a teachpoint" on page 270
Managing teachpoints	"Managing teachpoint files" on page 274
Moving between teachpoints	"Moving between teachpoints" on page 275
Selecting teachpoints	"Selecting two teachpoints" on page 276

Managing teachpoint files

About this topic

Teachpoint files store teachpoint parameters for all of the defined teachpoints using the Teachpoint Details dialog box. This includes all of the teachpoints that are visible as teachpoint markers on the BenchCel animated display.

Teachpoint files are written in XML and have an .xml extension.

By default, your teachpoint files are saved in the directory C:\Program Files\Velocity11.

Your default teachpoint file name is your BenchCel serial number followed by your company name.

In the Current Teachpoint File group box on the Controls page of the BenchCel Diagnostics dialog box, you can create a new teachpoint file or load an existing one.

In most cases you will not need to create a new teachpoint file, but will instead modify your existing one.

Create a new teachpoint file when you are adding or adjusting teachpoints and you do not want to corrupt your original teachpoint file.

To restore your original teachpoint file, use the load procedure.

Creating a new teachpoint file

To create a new teachpoint file:

1. Click **New**.

   The **Save As** dialog box is displayed.
2. Select a name and location for your teachpoint file.
3. Click **Save**.

Loading a teachpoint file

To load a teachpoint file:

1. Click **Load**.

   The **Open** dialog box is displayed.
2. Select the location and name of your teachpoint file.
3. Click **Open**.

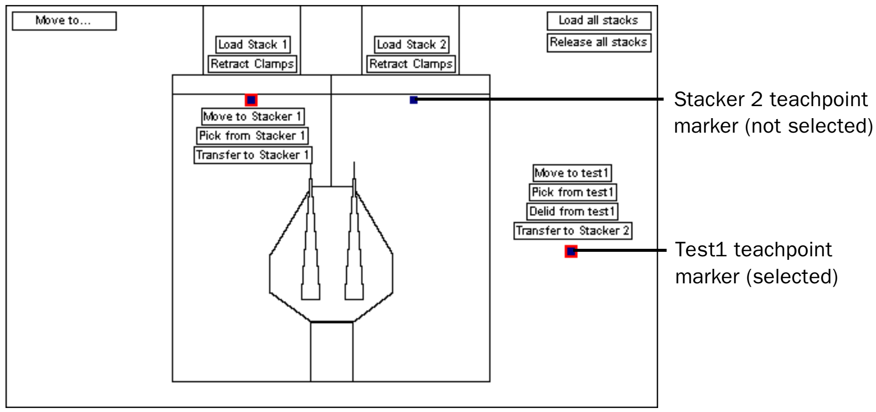
Related information

For information about...	See...
Teachpoint details dialog box	"About teachpoints" on page 269

Moving between teachpoints

About this topic

Each teachpoint is symbolized by a square dot called a teachpoint marker in the BenchCel animated display.



In this illustration, there are three available teachpoints. There is a teachpoint for each of the two stackers and one teachpoint called Test1.

Both Stacker1 and test1 teachpoints are selected, as indicated by the enlarged border around each teachpoint marker.

Up to two teachpoint markers can be selected at one time.

Procedure

To move the robot using teachpoint control:

1. Make sure you have selected the desired **Plate Type** from the **Labware** list box.
2. Click once on a teachpoint marker.

   Several boxed **command** buttons appear. See the **test1** teachpoint marker in the illustration above.
3. Click a button, using the following table as a guide:

Command button	Description
Move to *teachpoint*	Move the robot to the teachpoint.
Pick from *teachpoint*	Pick or place a plate at the teachpoint.
Delid from *teachpoint*	Tells the robot to delid. This option is not available for stacker teachpoints.
Transfer to *teachpoint*	Transfer from the currently selected teachpoint to another selected teachpoint. This command only becomes available after you have selected two teachpoints.

!! DAMAGE HAZARD !! The robot will move to a teachpoint using the most direct path. If there is an object in the way, the robot will collide with it.

If you have defined a teachpoint for an object that is in the path of the robot, the robot will compensate for this and avoid a collision.

Selecting two teachpoints

In the BenchCel animated display, you can select one or two teachpoints at a time. It is useful to select a pair of teachpoints so that you can conveniently move the robot between them. When you click a teachpoint marker, controls appear permitting you to perform additional functions.

Procedure

To select two teachpoints:

1. Click one teachpoint marker.

   A red border appears around the teachpoint marker, indicating that the teachpoint is selected.

2. Click a second teachpoint marker.

   A red border appears around the second teachpoint marker, indicating that its teachpoint is selected.

   These two markers will remain selected until you choose another teachpoint. At that moment, the first teachpoint marker you chose is deselected.

Note: Occasionally command buttons on the animated display obscures a teachpoint you want to select. Expose the obstructed teachpoint by deselecting the teachpoint whose command buttons are in the way.

Related information

For information about...	See...
Creating a teachpoint	"About teachpoints" on page 269
The animated display	"About the Controls page" on page 254

About general settings

About this topic

The General Settings page of the BenchCel Diagnostics dialog box contains offsets and corrections that are set at Velocity11. You should rarely, if ever, need to change them.

When adding new labware, you may have to make some adjustments after initially defining the labware.

!! IMPORTANT !! Be very careful making changes to the general settings. Incorrect settings can cause robot crashes resulting in gripper misalignment. If you are unsure, consult with a Velocity11 engineer before changing these settings to ensure that you input them correctly. Velocity11 keeps records of the settings that were factory set for your machine.

When you make a service call to Velocity11, you may be asked to provide some information about your BenchCel. This information is contained in the General Settings page.

BenchCel information

The **BenchCel** group box at the top of the General Settings page contains the firmware version and the MAC address of the BenchCel. You may be asked for this information when you make a service call.

Related information

For information about...	See...
Adjusting labware settings	“Maintenance and troubleshooting” on page 299

Changing general settings

About this topic

This topic describes how to edit the general settings.

!! IMPORTANT !! Be very careful making changes to the general settings. Incorrect settings can cause robot crashes resulting in gripper misalignment. If you are unsure, consult with a Velocity11 engineer before changing these settings to ensure that you input them correctly. Velocity11 keeps records of the settings that were factory set for your machine.

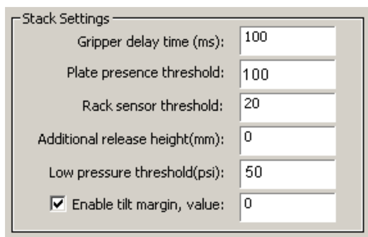
Procedure

To change general settings:

1. Click the **General Settings** tab.

   The BenchCel Diagnostics **General Settings** dialog box appears.

2. Configure the **Stack Settings**.

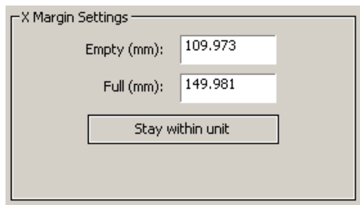


   Type the new value into the text box for the parameter you want to change using the following table as a guide.

Setting	Comment
Gripper delay time	The length of time the robot pauses to wait for the stacker gripper to open or close.
Plate presence threshold	Any value at or higher than this means a plate is present. Below this value means there is no plate.
Rack sensor threshold	Similar to the Plate presence threshold, but higher because the sensor is closer to the metal rack, which reflects more light than the plates do.
Additional release height (mm)	This setting lets you compensate for the stacker grippers if they grip the plates too low. The setting is adjusted in millimeters.
Low pressure threshold (psi)	Sets the threshold for the low pressure warning. This value is usually 50 psi.

Setting	Comment
Enable tilt margin, value	When checked, the sensors will check for plate tilt using the value entered (0–5 mm) as the amount of allowable tilt. If tilt is detected, the BenchCel will automatically restack the plates.

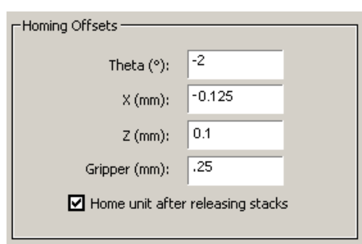
3. Configure the **X Margin Settings**.



Type the new value into the text box for the parameter you want to change using the following table as a guide.

If...	Then...
Empty	The distance the robot needs to travel from its farthest horizontal points to safely move its arms when not holding a plate.
Full	The distance the robot needs to travel from its farthest horizontal points to safely move its arms when holding a plate.
Stay within unit	This button restores the default values to the previous settings.

4. Check the **Homing Offsets**.



!! IMPORTANT !! These offsets are set at Velocity11 and should not need to be changed. They are used to calibrate all axes relative to their true home position.

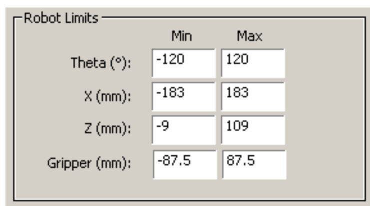
Type the new value into the text box for the parameter you want to change, or check the appropriate check box, using the following table as a guide.

Setting	Comment
Theta	This offset changes the angle (in degrees) that the grippers home to. It is measured from a line perpendicular to the *x*-axis.

Setting	Comment
X	Changes the distance from the center of the robot's horizontal travel. Find this distance by measuring the distance the robot travels from an arbitrary point to the far left and far right positions and averaging them.
Z	Changes the distance between the bottom of the robot head and the column. Ideally, this should be 20 mm.
Gripper	Changes the distance between the opposing gripper points. This distance should be 87.5 mm.
Home unit after releasing stacks	Checking this box tells the robot to home itself every time it releases a stack.

5. Check the **Robot Limits**.

   These values define the range of the *theta*, *x* and *z* axes for the robot.



   !! IMPORTANT !! These offsets are set at Velocity11, and should not need to be changed.

6. Click **OK** for the settings to take effect.

   !! IMPORTANT !! If you change the general settings and click OK, the values are changed in BenchWorks, which will affect protocols that you run afterwards.

Related information

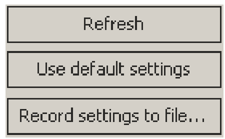
For information about...	See...
Adjusting labware settings	"Maintenance and troubleshooting" on page 299
Global settings	"Making global general settings changes" on page 281

Making global general settings changes

About this topic

In the bottom right of the **General Settings** page are three buttons that affect changes to the whole page. This topic provides a description of these buttons.

You should rarely need to use these buttons.



Button descriptions

For information about each button, consult the following table.

Button	Description
Refresh	Pressing the refresh button does not change any settings, but verifies that your computer is displaying the correct values. This is useful when you have lost connection to the machine, or you want to make sure your settings are reflecting the settings that the BenchCel is currently using.
Use default settings	This will restore the factory default values. This is useful if you have made changes that do not work correctly, and you want to get back to normal functionality.
Record settings to file	This creates an XML file with the current general settings values. This is used for sending information to a Velocity11 service engineer to help diagnose problems on your machine.

Related information

For information about...	See...
Adjusting labware settings	"Maintenance and troubleshooting" on page 299
General settings	"Changing general settings" on page 278

About profiles

Profiles defined

A profile contains the initialization settings associated with a particular BenchCel. The data in a profile is used by BenchWorks to identify each BenchCel on the network.

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

Because profiles identify each BenchCel device, every BenchCel on the network 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 BenchCel is connected using serial or Ethernet
- ❑ If the BenchCel is connected using Ethernet, the Device ID of the BenchCel on the network
- ❑ If the BenchCel is connected using serial, the COM port that the controlling computer uses for communication

Related information

For information about...	See...
Creating a profile	“Creating a BenchCel profile” on page 283
Managing profiles	“Managing profiles” on page 284

Creating a BenchCel profile

About this topic

Your BenchCel was delivered to you with the correct profiles already created. However, at times you may need to create a new profile if, for example, to set up unique communications settings or use a different teachpoint file than you have with other profiles.

This topic describes how to create a new BenchCel profile.

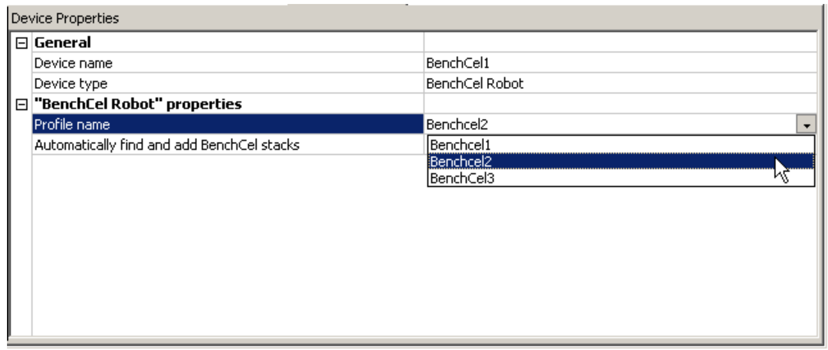
Procedure

To create a BenchCel profile:

1. Make sure you have a teachpoint file that you want to associate with your new profile.
2. In the **BenchWorks** main window, click the **Device Manager** tab.
3. Select the desired BenchCel from the **Device List**.
4. Click **Device Diagnostics** located at the bottom of the **Device List** toolbar.
5. In the **Device Diagnostics** window, click the **Profiles** tab.
6. Click **Create a new Profile**.
7. Enter a name, and click **OK**.
8. In the **Profiles** group box, select the **Connection** type from the list.

   Select **Ethernet** if your BenchCel is connected to your computer with an Ethernet connection.

   If you have a serial connection, select the **COM** port the connection is using. If your BenchCel supports flow control, select the **Use flow control** check box.
9. If you selected an Ethernet connection, select the device to associate with the profile. Click **Find available device**.
10. In the **Discovered BioNet Devices** dialog box, select the appropriate BenchCel from the list and then click **OK**.
11. Select a teachpoint file:
    a. Click the ellipsis (...) button.
    b. In the **Select a Teachpoint File** dialog box, navigate to the location of the file, select it and click **Open**. The file path appears in the **Profiles** group box.
12. Click **Initialize this profile** to initiate communication to the BenchCel using the new profile.
13. Click **OK** to save the changes and close the dialog box.
14. In the **Device Properties** window, select the new profile from the list box next to the **Profile name** property.



Related information

For information about...	See...
Profiles	❑ "Relationships of BenchWorks components" on page 30 ❑ "About profiles" on page 282
Teachpoint files	"Managing teachpoint files" on page 274

Managing profiles

About this topic

After creating a profile, you can edit, delete, or rename it. This topic describes how to do these tasks.

Procedure

To edit a profile:

1. In the **BenchWorks** main window, click the **Device Manager** tab.
2. Select the desired BenchCel from the **Device List**.
3. Click **Device Diagnostics** located at the bottom of the **Device List** toolbar.
4. In the **Device Diagnostics** window, click the **Profiles** tab.
5. Select one of the options, using the following table as a guide:

Option	Description
Create a copy of this profile	Click to create a copy of the profile currently displayed in the **Profile name** field.
Rename this profile	Click to rename a profile. Type in the desired name in the **Rename Profile** dialog box.

Option	Description
Delete this profile	Click to delete a profile. A dialog message appears asking if you are sure you want to delete this profile.
Update this profile	Click to save changes made to a profile, for example if you have changed the Teachpoint file for that profile.
Initialize this profile	Click to instruct the software to connect to the BenchCel using the current profile.

Related information

For information about...	See...
Profiles	“Relationships of BenchWorks components” on page 30
Creating a profile	“Creating a BenchCel profile” on page 283

Using Multidrop diagnostics

About this topic

The Thermo Electron Multidrop microplate dispenser can be used either in stand-alone mode or as a device within a protocol using the BenchCel.

This topic explains how to manipulate the dispenser independently of BenchWorks, using Multidrop diagnostics software created by Velocity11.

To use the dispenser within a BenchWorks protocol, set the Multidrop parameters in the Protocol Task Parameter toolbar.

Terms defined

The Multidrop has priming and purging functions that can be controlled through the Multidrop Diagnostics.

Priming

Priming moves liquid through the dispensing system into waste to make sure that there is no air in the lines.

Purging

Purging moves liquid back into the liquid cassettes.

Before you start

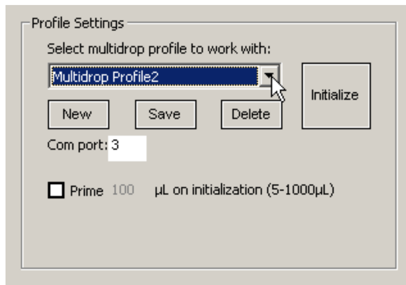
Before you can manipulate a Multidrop, make sure that it is correctly initialized.

Operating the Multidrop

To initialize a Multidrop:

1. In the **Profile Settings** group box, select the desired Multidrop from the list box.

   The value in the **Com port** changes to match the Multidrop you selected.



2. If you want to prime the fluid lines when you initialize the Multidrop, select the **Prime** check box and enter the volume of liquid you want to prime with.
3. Click **Initialize**.

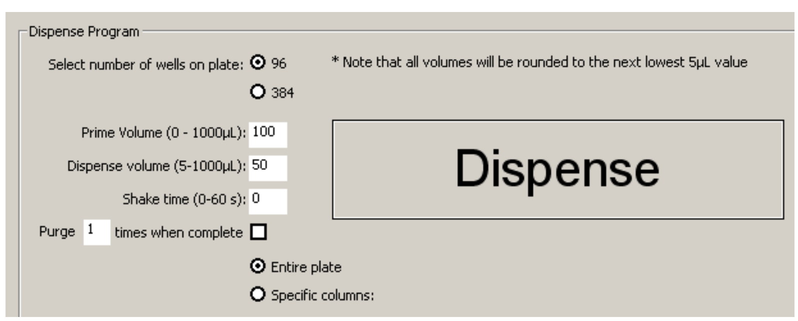
Creating and deleting devices

You do not need to use the **New**, **Save**, or **Delete** buttons in the **Profile Settings** group box. These are for Velocity11 use only.

Filling a plate

To fill a plate:

1. Refer to the *Multidrop 384 User Manual* for information about preparing to fill a plate.
2. In the **Dispense Program** group box, select the number of wells in the plate.



3. If you want to prime the Multidrop, enter the volume of liquid to prime with in the **Prime volume** text box.
4. In the **Dispense volume** text box, enter the volume of liquid that you want to dispense into each well.
5. To shake the plate after the dispense, enter a value into the **Shake time** text box.

6. To purge the lines after the dispense, select the **Purge *x* times when complete** check box, and enter the number of times to purge.

   If you want to return all liquid, the value of *x* may need to be greater than one.

7. To dispense into all wells, make sure that the **Entire plate** option is selected.

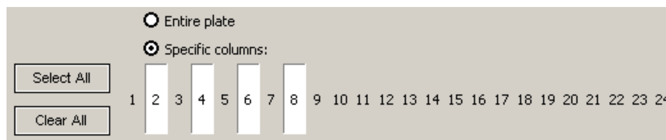
8. To only dispense into some of the wells:

   a. Select the **Selected columns** option.

   b. Click **Clear All**.

   c. In the number matrix, click the numbers corresponding to the columns to dispense into.

      In the following example liquid will be dispensed into columns 2, 4, 6, and 8 of a 384-well plate.



9. Click **Dispense**.

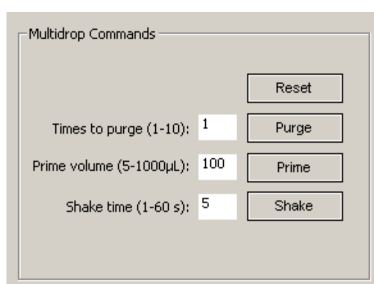
Performing non-dispense operations

You can perform basic non-dispense operations using the buttons in the **Multidrop Commands** group box.

To purge the liquid lines:

1. Enter a value in the **Times to purge** text box and click **Purge**.

   If you want to return all liquid, the value of *x* may need to be greater than one.



To prime the liquid lines:

1. Enter a value in the **Prime volume** text box and click **Prime**.

To shake the plate:

1. Enter a value in the **Shake time** text box and click **Shake**.

Related information

For information about...	See...
Diagnostics	"About diagnostics" on page 252
Using the Multidrop	*Multidrop 384 User Manual*

For detailed information about the Multidrop, refer to Thermo Electron Corporation's *Multidrop 384 User Manual*.

Using Nanodrop Diagnostics

About this topic

The Innovadyne Nanodrop dispenser is an 8-channel pipettor that dispenses 0.1–40 µL of liquid into each well.

It can be used either in stand-alone mode or as a device within a protocol using the BenchCel.

This topic describes how to:

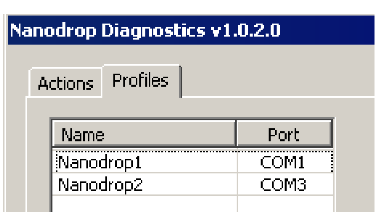
- ❑ Manage Nanodrop profiles, which are used to communicate with the computer.
- ❑ Perform an individual operation, such as a dispense or wash, using the parameters set in a Quick Run Method (QRM), and referencing a calibration file for increased accuracy.

For detailed information about the Nanodrop, see Innovadyne's *Nanodrop User Manual*.

About Nanodrop profiles

A profile is one or more settings that are remembered after you exit BenchWorks. In the case of the Nanodrop, the only setting stored in a profile is the COM port, which identifies the serial port to be used for communication between the device and the computer.

You need to have one profile for every Nanodrop on your BenchCel, as illustrated in the following screenshot of a system that has two Nanodrop pipettors.



All profiles should have been set at the factory so you should not need to create a new one unless you are adding another Nanodrop to your system.

If you have more than one Nanodrop on your BenchCel, select the profile for the one with which you want to communicate.

To select a Nanodrop profile:

1. In the **Actions** tab, select the profile from the **Profile** list box.
2. Click **Connect**.

To create a profile

You only need to create a new profile if you are adding a Nanodrop to your BenchCel.

To create a Nanodrop profile:

1. Click the **Profiles** tab.
2. Click **Add**.
3. In the **Profile Details** dialog box, type a name for the profile.
4. Select an available communications port from the **Serial port** list box.
5. Click **OK**.

Managing profiles

To modify a Nanodrop profile:

1. Open **Nanodrop Diagnostics**.
2. Click the **Profiles** tab.
3. Select the profile you want to modify and click **Change**.
4. Enter your changes in the **Profile Details** dialog box.
5. Click **OK**.

To delete a profile:

1. Open **Nanodrop Diagnostics**.
2. Click the **Profiles** tab.
3. Select the profile you want to delete and click **Delete**.

Running an individual operation

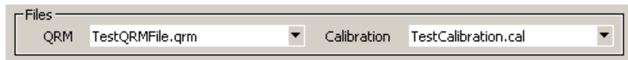
Use Nanodrop Diagnostics to perform the following actions:

Action	Meaning
Prime	Priming clears air bubbles from the syringe path tubing. Prime the Nanodrop if it has been left idle for more than two hours.
Dispense	Delivers liquid into the plate wells.
Wash	Cleans the insides and outsides of the tips. The operation is performed with the tips immersed in the wash station.
Rinse	Cleans the insides of the tips. The operation is performed above the wash station in the park position.
Purge	Purging clears air bubbles from the pressure path and syringe path. Purging is often performed after priming to make sure that the valves contain no air.

The actions use the settings in the selected QRM file, and any calibration corrections from the selected calibration file.

To perform an individual operation:

1. From the **Files** group box of the **Actions** page, select a QRM file and calibration file.



2. Click a button to start an action.
3. To stop the action, click **Abort**.

About calibration files

Calibration files contain the results of gravimetric calibration tests and are used to adjust the volume delivery process to make sure that volumes delivered are accurate.

Calibration files are initially provided by Innovadyne, but you can perform your own calibrations, the results of which are stored in a file. Calibrations are performed through Innovadyne's Nanodrop software and the files can only be selected through the Velocity11 Nanodrop Diagnostics software.

About QRM files

QRM (Quick Run Method) files store collections of Nanodrop settings. QRM files are created through Innovadyne's Nanodrop software and are selected through the Velocity11 Nanodrop Diagnostics software.

Examples of settings stored in QRM files are:

- Dispense volume
- Syringe volume
- Tip clean (enabled or disabled)
- Aspirating air gap rate
- Whether or not to wash

Related information

For information about...	See...
Diagnostics	"About diagnostics" on page 252
Using the Nanodrop	*Nanodrop User Manual*

Using QFill2 Diagnostics

About this topic

The Genetix QFill2 dispenser can be used either in stand-alone mode or as a device within a protocol using the BenchCel.

In stand-alone mode, a QFill2 is operated using a built-in keypad. Velocity11 has developed QFill2 Diagnostics to allow users to operate a QFill2 using BenchWorks software.

This topic explains how to perform simple manipulations of a Genetix QFill2 in real time. To use the QFill2 dispenser within a BenchWorks protocol, set the parameters in the Protocol Task Parameter toolbar, not in the diagnostics software.

Note: In some versions of BenchWorks, you will see “QFill” instead of “QFill2”.

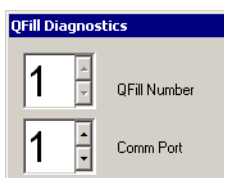
Identifying the QFill2

Before you can manipulate a QFill2, make sure that it is correctly identified.

To make sure that the QFill2 is identified:

1. Check that the **QFill Number** is correct for the device that you want to communicate with.

   This number identifies a specific QFill2 when there is more than one on the BenchCel. You can determine the number of a QFill2 by referring to the label on the device, by trial-and-error, or by looking at the **Device Number** in the device manager.

2. Check that the **COM Port** number matches the serial port number for the device that you want to communicate with.

   You can determine the **COM Port** number from the label on the serial port cable near where it plugs into the QFill2. The number is also the same as the **Input number** in the device manager.

   After the QFill2 has been defined, the **COM Port** number is added automatically when you select the QFill2. You should not need to change it.

About creating and deleting devices

You should not need to use the **Create New QFill** or **Delete QFill** buttons. These are for Velocity11 use only.

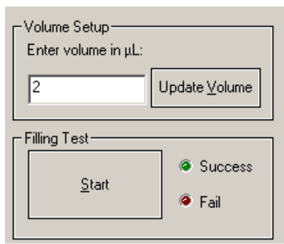
Procedure

Refer to the QFill2 user documentation for information about preparing to fill a plate.

To dispense into a plate:

1. Make sure that there is a plate on the QFill2 stage.
2. Make sure that the reagent bottle has sufficient liquid in it for the dispense.
3. Set the dispense volume:
   a. In the **Volume Setup** group box, type the QFill2 volume into the text box.
   b. Click **Update Volume**.
   c. Click **OK.**

      The volume you last entered is saved until the next time a protocol is run, which will override this value.



4. In the **Filling Test** group box, click **Start**.
5. Monitor the success of the fill by looking at the **Success** and **Fail** indicators.

   The **Success** and **Fail** indicators provide feedback from a sensor that measures pressure in the dispensing bottle. The **Fail** indicator light is illuminated if the:

   - QFill2 is not turned on
   - QFill2 has a disconnected communication cable
   - Bottle fails to pressurize

About the Filtration Port

The Filtration Port defines the COM port used to provide vacuum sensor information for an optional filtration station on the QFill2. You should not need to change this number.

Related information

For information about...	See...
Diagnostics	“About diagnostics” on page 252
Preparing a plate for filling	*QFill2 User Guide*

Using WellMate Diagnostics

About this topic

This topic describes how to use WellMate Diagnostics to:

- ❑ Manage profiles
- ❑ Move the plate stage
- ❑ Execute dispense commands

About profiles

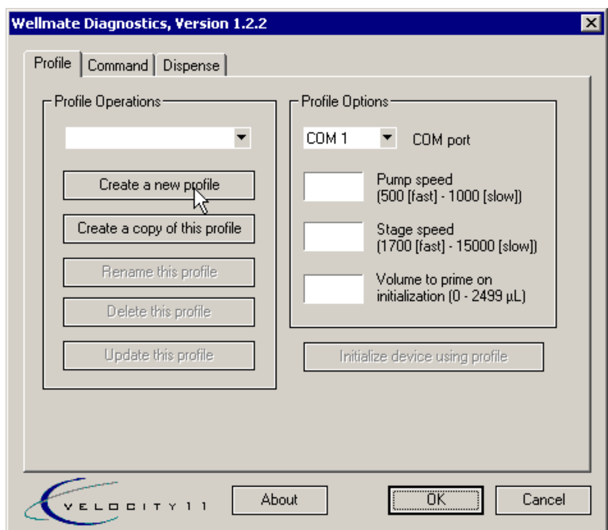
Profiles contain the initialization settings that enable BenchWorks to identify and communicate with the BenchCel. Every device within the automation system must have its own profile. Once created, you will probably not need make changes to the profile.

A profile for your WellMate was created at the factory. However, if you need to create a new profile, you can do so.

Managing profiles

To create a WellMate profile:

1. Open **WellMate Diagnostics**.
2. Click the **Profile** tab.



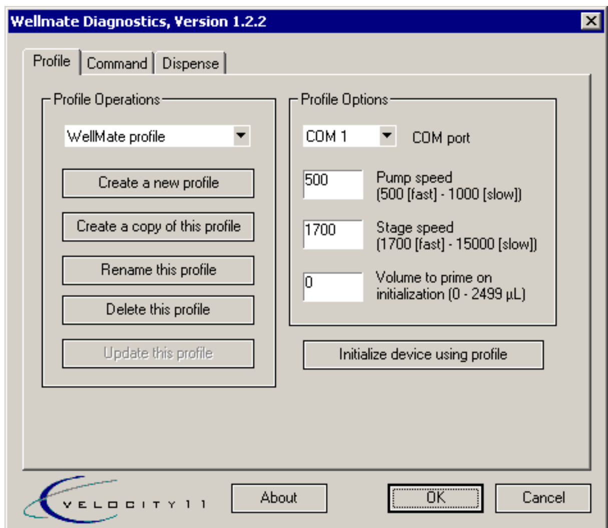
3. Click **Create a new profile**, enter a name for the device in the **New Profile** dialog box and click **OK**.
4. Set the **Profile Options**:
    a. From the **COM port** list box, select the port that the WellMate uses to communicate with the controlling computer.
    b. Enter a value for the **Pump speed**, using the range shown as a guide.
    c. Enter a value for the **Stage speed**, using the range shown as a guide.

d. Enter a volume in the **Volume to prime on initialization** field to prime the lines when the profile initializes.

5. Click **Update this profile**.
6. To open communications to the WellMate and send the profile settings immediately, click **Initialize device using profile**.
7. Click **OK**.

To manage profiles:

1. Open **WellMate Diagnostics.**
2. Click the **Profile** tab.

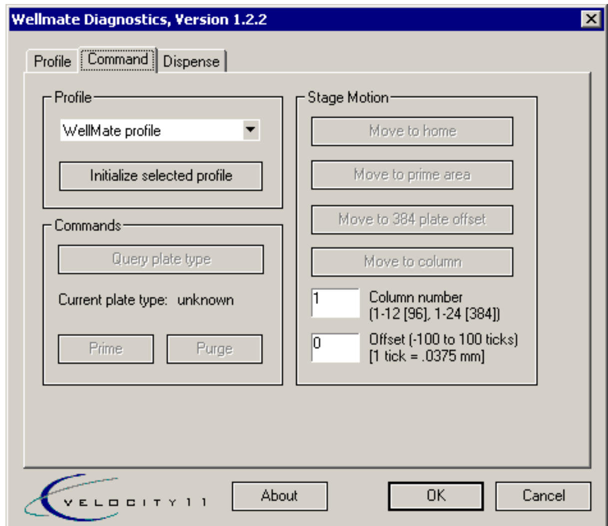


3. Select the profile to manage from the **Profile Operations** list box. Make changes to the profile, if desired.
4. At this point, you may perform any of the operations displayed in the **Profile Management** group box by clicking the appropriate button.

Moving the plate stage

To move the plate stage:

1. Open the **WellMate Diagnostics**.
2. Click the **Command** tab.



3. Perform the desired task using the following table as a guide.

To...	Then...
Home the WellMate	Click **Move to home**.
Move the stage to the priming area	Click **Move to prime area**.
Move the 384-plate offset	Click Move to **384 plate offset**. The stage moves to the end of its travel and then shifts to the next stage for dispensing on a 384-well plate.
Move to a column	Enter the column number you want to move to in the **Column number** field and click **Move to column**.
Change the offset	Enter a value in the **Offset** field and the desired column in the **Column number** field and click **Move to column.** This works in conjunction with the **Move to Column** command. Offset defines how far away from the center of the well to dispense.

Querying the plate type

To query the plate type:

1. Open the **Command** page.
2. To determine the plate type (96- or 384-well) that the WellMate is set for, click **Query plate type**.

   The plate type is displayed next to **Current plate type** text.

Priming and purging

Priming moves the fluid from the reservoir to the tips, and purging moves the fluid from the tips to back to the reservoir.

To prime or purge:

1. Open the **Command** page.
2. Click and hold the mouse on the **Prime** or **Purge** button.

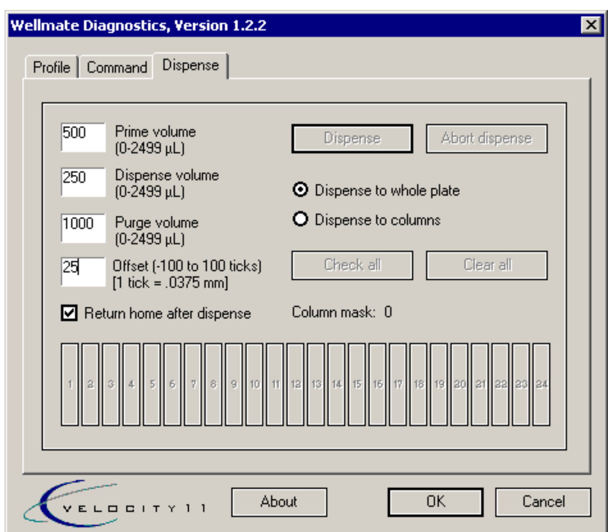
   Fluid is pumped at the **Pump Speed** set in the **Profile** page until you release the mouse.

Executing dispense commands

Use the Dispense page to set the prime, dispense, and purge volumes and to initiate the dispense.

To execute dispense commands:

1. Open the **WellMate Diagnostics** dialog box and click the **Dispense** tab.



2. Enter in the appropriate values using the table below as a guide.

Property	Description
Prime	Volume with which to prime the lines before dispensing.
Dispense	Volume to dispense into the wells.

Property	Description
Offset	The distance in ticks (where 1 tick= 0.0375 millimeters) from the center of the well you want to dispense.
Move to a column	Enter the column number you want to move to in the **Column number** field and then click **Move to Column**.

3. Select the wells of the plate to which you want to dispense:
   a. To dispense to the whole plate, click **Dispense to whole plate**.
   b. To dispense to a specific column or columns, click **Dispense to columns** and then click the button(s) corresponding to the column number.

      When selected, the button changes its appearance.
4. To dispense, click **Dispense**.

   To stop the dispense process before it is finished, click **Abort dispense**.

Related information

For information about...	See...
Diagnostics	"About diagnostics" on page 252
Using the WellMate	*WellMate User Guide*

13 Maintenance and troubleshooting

This chapter tells you how to keep your BenchCel in good working order and what to do when you encounter a problem.

Routine maintenance

About this topic

This topic provides recommendations for maintenance including when and what to perform.

Monthly maintenance

In general, practice good housekeeping by cleaning up spills and routinely cleaning after using.

Every month, check the following:

- ❑ Robot head gripper arm points to ensure they are not too dull to grip the plates.
- ❑ Stacker grippers to ensure they have not deteriorated.
- ❑ Visible moving parts to ensure they are not rubbing against each other.

  Look for rub marks or noises that might indicate rubbing. These are most evident on the white metal cover behind the robot head, and anywhere the robot arms travel.
- ❑ Main air pressure reads between 50 and 90 psi.

After every run

Check the *x*-axis track to ensure it is free of debris, such as pieces of chipped plates and microtubes.

BenchCel instrument maintenance

For information about routine maintenance of the individual instruments integrated with your BenchCel, see the relevant instrument's documentation.

Related information

For information about...	See...
Cleaning	"Cleaning up after a run" on page 80
Gripper pressure	"Adjusting the stacker gripper pressure" on page 320

About error handling

About this topic

The BenchCel is a complex instrument that requires the interaction of hundreds of parameters, configurations, and operator setup tasks for a protocol to run successfully.

Causes of errors

Errors have many causes, including poor protocol writing, incorrect operator setup, variation in plates, hardware failure, and software failure.

Handling errors during normal operation

It is important to understand that error handling is a normal part of operating the BenchCel and that when errors occur it does not necessarily mean that the BenchCel has malfunctioned.

Related information

For information about...	See...
Initialization errors	"Resolving device initialization errors" on page 302
Compiling errors	"Compilation warnings and errors" on page 303
Error messages	"Resolving common problems and error messages" on page 304
Labware problems	"Resolving labware problems" on page 310
Sensors	❑ "Resetting the orientation sensors for a plate" on page 313 ❑ "Checking the stacker sensors" on page 314
Teachpoint problems	"Resolving teachpoint problems" on page 315
Bar code reader errors	❑ "Resolving non-VCode bar code reader errors" on page 317 ❑ "Resolving VCode bar code reader errors" on page 319

Resolving device initialization errors

About this topic

This topic describes the device initialization process and suggests some things to try if you encounter problems with device initialization.

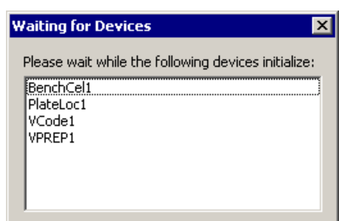
Device initialization process

When you start BenchWorks, the software loads the driver files for all the devices on your system. A record of this process is displayed in the log toolbar.

When you open a protocol file, the device file associated with that protocol opens. The device file tells the software which devices are connected to the system. Then, for some kinds of devices, an initialization step tests the communication between BenchWorks and the device.

Resolving initialization errors

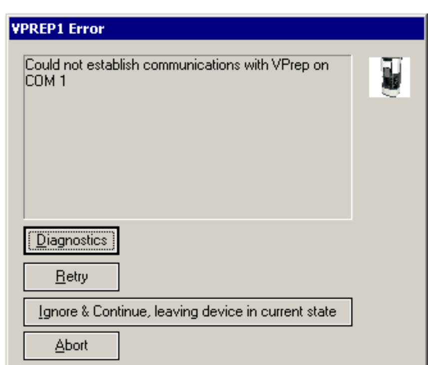
During this process, a message window opens displaying the name of the devices that the software expects to find. Devices are removed from the list as the system computer determines that the devices are ready. In the following example, there are four devices that the software is trying to initialize:



If there is a problem with initializing a device, you receive an error message dialog box for that device.

The problem encountered is stated in the text field of the dialog box.

In this example, BenchWorks could not initialize the VPrep device.



Make sure that the device is turned on and that the communications cable is connected properly.

Click **Diagnostics** and try to resolve the problem in the device profile. If that does not solve the problem, contact the Velocity11 Service Center.

Compilation warnings and errors

About this topic

This topic briefly describes compilation warnings and errors.

Compilation warnings

Warnings alert you to situations that seem to have incorrect intent but that will not cause a collision or a device to fail.

Compilation warnings are displayed in the BenchCel log toolbar, listing the task that caused the warning.

Warnings are generated, for example, when:

- ❑ Pipette tips are on the VPrep head at the start of the protocol so a tips-off task is run before a tips-on task
- ❑ No label is printed during a VCode task

Compilation errors

Errors alert you to situations where a protocol or device will fail.

Compilation errors are listed in the Log toolbar, and a dialog box opens telling you how many errors were found. If you are logged in with technician, operator, or guest privileges, you are unable to continue with the protocol. If you are logged in with administrator privileges, the dialog box allows you to run the protocol despite the errors.

Errors are generated, for example, when:

- ❑ Operating parameters are out of range, denoted by red text in fields of the Task Parameters toolbar
- ❑ A task asks a VPrep to pipette from a plate that does not exist in the system
- ❑ Volumes in pipette steps do not match, such as when a dispense volume is greater than a previous aspirate volume
- ❑ A specific stacker has not been assigned to a stacker task
- ❑ A Signal task has no associated Waitfor task

Resolving common problems and error messages

About this topic

The most common problems that can occur with the BenchCel relate to the following areas:

- ❑ Labware
- ❑ Stacker
- ❑ Robot
- ❑ Arm alignment

This topic describes the possible causes and recommended actions for these types of problems.

Labware and stacker

Problem/Error Message	Possible Cause	Recommended Actions
"Wrong plate type"	The plate you are using is not the same as the plate type selected in the software.	Check the plate you are using with the labware parameters of the plate type selected in the software. Use the correct plate or select the appropriate plate type in the software.
	Incorrect value in the labware definitions for the plate type.	Open the Labware Editor dialog box and check the value in the Number of wells entry field. Make sure that the displayed value matches the number of wells in the plate you are using.
	Incorrect orientation offset.	See "Resolving labware problems" on page 310 for a procedure.
	Orientation sensors need adjusting.	See "Resetting the orientation sensors for a plate" on page 313 for a procedure.

Problem/Error Message	Possible Cause	Recommended Actions
“Plate is rotated”	Plate was placed in the stacker rack in the wrong orientation.	Check the plate orientation in the stacker rack. The notch should be in the back, left corner as you face the BenchCel.
	Notch location is wrong in the labware definition for the plate.	Open the Labware Editor dialog box and click the Stacker tab. Check to ensure that the correct notch is selected in the Notch Locations group box.
	Stacker orientation sensor offset is incorrect.	Open the Labware Editor dialog box and click the Stacker tab. Verify the Orientation sensor offset value. This value is the distance in millimeters, from the bottom of the plate to the middle of the Stacker rack sensor window.
	Orientation sensors need adjusting.	See “Resetting the orientation sensors for a plate” on page 313 for a procedure.
Conflicting messages regarding plate presence and orientation	The plate presence threshold is not set to accommodate the range of plate types you are using.	Open the BenchCel robot diagnostics and click the General Settings tab. Adjust the value in the Plate presence threshold field. Typically, this value is ~100. If you are getting a no plate present message when there is a plate, decrease this value. If you are getting a plate present message when there is none, increase this value.
	Plate orientation sensors need adjusting.	Open the Labware Editor dialog box and click the Stacker tab. Adjust the Orientation sensor threshold (max) and/or Sensor intensity values. See “Resetting the orientation sensors for a plate” on page 313 for a procedure
“No plate in stack”	Plates are not in the correct stack.	Check to ensure plates are located in the correct stack.
“Stack not loaded”	Stacker rack is absent.	Install rack.
	Stacker clamps in the incorrect position to execute the command.	Open the BenchCel Diagnostics dialog box. If there are no plates in the stack, click the retract clamps command.
“Rack not present”	Stacker rack is out of position.	Shift the stacker’s position a little without removing it and then click retry.

Robot and stacker grippers

Problem/Error Message	Possible Cause	Recommended Actions
Robot grippers are not grabbing plates correctly	Gripper offset is incorrect.	Open the Labware Editor dialog box and click the BenchCel tab. Check the value for the Robot gripper offset. This value is the distance from the bottom of the plate to the point where the grippers grab the plate. This should be the middle of the plate.
	Stacking thickness for the plate type is incorrect.	Open the Labware Editor dialog box and click the Plate Properties tab. Check the Stacking Thickness value. See “Defining plate properties” on page 226 for a procedure.
	Plate thickness for the plate type is incorrect.	Open the Labware Editor dialog box and click the Plate Properties tab. Check the Thickness value. See “Defining plate properties” on page 226 for a procedure.
	Gripper limits are incorrect.	Open the Labware Editor dialog box and click the BenchCel tab. Adjust the values for the Gripper open position.
Plates are tilted or dropping from the stacker	Stacker gripper offset is incorrect for the plate type.	Open the Labware Editor dialog box and click the Stacker tab. Check the Stacker gripper offset value. This value is the distance from the bottom of the plate to the center of the stacker gripper. The gripper should grab the plate midway.
	Stacking thickness for the plate type is incorrect.	Open the Labware Editor dialog box and click the Plate Properties tab. Check the Stacking Thickness value. See “Defining plate properties” on page 226 for a procedure.
	Plate thickness for the plate type is incorrect.	Open the Labware Editor dialog box and click the Plate Properties tab. Check the Thickness value. See “Defining plate properties” on page 226 for a procedure.

Problem/Error Message	Possible Cause	Recommended Actions
Plates are not held level by the robot at all times during a plate transfer	Gripper points are out of alignment.	Check the gripper alignment. See "Checking the arm alignment" on page 309 for a procedure. If grippers are out of alignment, contact Velocity11 Service Center.

Robot and scheduler error messages

Error Message	Possible Cause	Recommended Actions
"No plate in grippers"	The robot did not detect a plate when it expected one.	If there is a plate in the grippers and it is incorrectly positioned, use the Robot Diagnostics screen to move to a teachpoint and open the grippers, which will release the plate. After that, use Robot Diagnostics to pick up the plate and retry.
"Plate in grippers"	The robot senses a plate in its grippers and is being asked to perform an action that requires it to not have a plate.	If there is a plate in the grippers, and it is incorrectly positioned, use the Robot Diagnostics dialog box to move to a teachpoint and open the grippers which will release the plate. After that, use the diagnostics to pick up the plate and retry.
"X position out of bounds"	The robot has been commanded to move out of its range in the x direction.	If you encounter this error repeatedly, contact Velocity11 Service Center to upgrade your firmware or obtain workaround information for this problem.
"Z position out of bounds"	The robot has been commanded to move out of its range in the z direction.	
"Theta position out of bounds"	The robot has been commanded to move out of its range.	
"The gripper positions are too close"	The values of the open and holding plate gripper positions of the labware are too close to each other.	Increase the difference between the two gripper positions.

Error Message	Possible Cause	Recommended Actions
“Position error on the Theta axis”	The robot failed to move to the commanded position. For example, something is blocking the robot.	Retry. If that fails, home the robot. If homing fails, contact Velocity11 Service Center.
“Position error on x axis”		
Position error on z axis”		
“Position error on grippers”		
“Flash operation not successful”	The robot encountered difficulty writing to its flash memory.	Click retry. If that doesn’t work, turn the BenchCel off and then on. If you continue to experience this problem, contact Velocity11 Service Center.
“Operation timed out”	The robot encountered difficulty executing a command.	Click retry. If retrying does not work, turn the BenchCel off and then on. If problem continues, contact Velocity11 Service Center.
“Communication timed out”	Communication to the robot or to a peripheral instrument was lost.	Close and restart BenchWorks.
		Turn the BenchCel off and then on.
		Check the communication cables.
		Verify that the Ethernet hub has power. If the problem continues, contact Velocity11 Service Center.
“Deadlock detected”	Every plate position is filled and there is nowhere to move any of the plates.	Re-examine the protocol with the aid of the Log toolbar and run simulator to determine why the deadlock is occurring. Check to be sure that the number of simultaneous plates is not too high. See
“Thermal cutoff active for theta-axis”	The theta motor is overheated. **!! INJURY HAZARD !! Be careful. The robot head might be too hot to touch.**	Wait for the head to cool and retry.

Error Message	Possible Cause	Recommended Actions
“Timeout on theta-axis	The motor’s controller did not respond as expected.	Home the robot and then move the robot in the same axis. If the robot moves, continue on. If the robot does not move, shut down the application and turn the BenchCel off and then on. Contact Velocity11 Service Center if problem continues.
“Timeout on x-axis” “Timeout on z-axis” “Timeoout on grippers”	The motor controller did not respond as expected.	Home the robot and then move the robot in the same axis. If the robot moves, continue on. If the robot does not move, shut down the software and turn the BenchCel off and then on. Contact Velocity11 Service Center if problem continues.

Checking the arm alignment

About this topic

The gripper arms of the BenchCel robot must be in alignment, that is, level with each other in all planes. If the arms become misaligned, the robot will have difficulty placing and picking labware.

Under normal working conditions, the arms should not become unaligned. Usually misalignment is the result of a physical collision of the robot with a peripheral device or other object.

This topic describes when and how to check the arm alignment.

When to check the arm alignment

- ❑ When teachpoints appear to be off and nothing was changed on the BenchCel system
- ❑ When a plate being held by the grippers appears tilted in any direction
- ❑ Any time the robot is not picking up or placing labware properly

Who should adjust the alignment

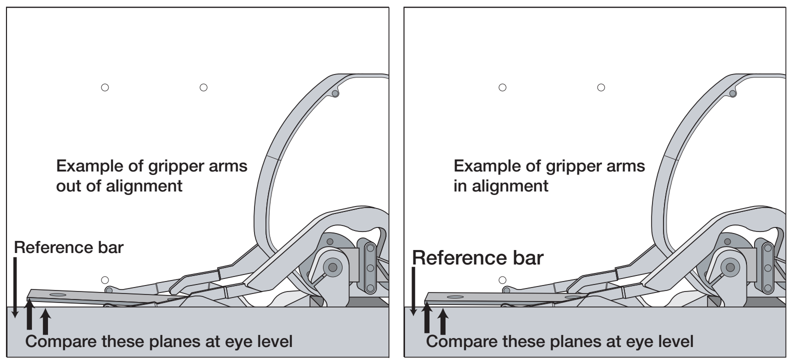
Adjusting the gripper-arm alignment is a tedious process and should be performed by Velocity11 Service Center personnel.

If you suspect the arms are out of alignment, you should follow the procedure in this topic to check their alignment. If they are found to be out of alignment, contact Velocity11 Service Center to set up a service visit.

Procedure

To check the arm alignment:

1. Open the **Control** page in the **BenchCel Diagnostics** dialog box.
2. Place a plate in one of the stackers.
3. Command the robot to pick the plate and then to go home.
4. Using the bar in front of the BenchCel, visually check if the plate is parallel to the bar.
5. If the plate is not parallel, the arms are out of alignment.



Resolving labware problems

About this topic

When using a new labware type, you may encounter some robot handling errors which will require adjustments to offsets and/or sensor thresholds and intensities. This topic identifies some of the typical problems you may encounter and provides some recommended solutions.

Error messages

The following types of errors are generated when there are problems with the labware:

- ❑ Wrong plate type
- ❑ Plate is rotated
- ❑ No plate is present/Plate is rotated, or other conflicting messages

Wrong plate type

This error is generated when the notch sensors, located in the stacker head, detect a plate notch pattern inconsistent with the definition of that plate type in the labware database.

To resolve a wrong plate type error:

1. Check to be sure you are using the correct plate type.
2. If you are using the correct plate type, check the notch setting in the labware parameters:
   a. Open the **Controls** page of the **BenchCel Diagnostics** dialog box.
   b. In the **Labware** group box, select the **Show parameters** check box.
   c. In the **Labware Parameters** dialog box, verify that the correct notch position is selected.
3. Check the **Number of wells** value in the labware entries page of the **Labware Editor** dialog box:
   a. Open the **Labware Editor** dialog box and check the value in the **Number of wells** entry field.
   b. Make sure that the displayed value matches the number of wells in the plate you are using.
4. Check the orientation sensor offset:
   a. Open the **Labware Editor** dialog box and click the **Stacker** tab.
   b. Verify the value in the **Orientation sensor offset**. This is the distance from the bottom of the plate to the middle of the sensor window.
5. Check the orientation (plate notch) sensor readings. See “Checking the stacker sensors” on page 314 for a procedure.

   If readings do not detect the correct orientation, see “Resetting the orientation sensors for a plate” on page 313 for a procedure on how to adjust the sensors.

Plate is rotated

This error is generated when the notch sensors, located in the stacker head, detect a plate notch pattern consistent with the correct plate type placed in the wrong orientation.

To resolve a plate is rotated type error:

1. Check to ensure that the plate is in the correct orientation in the stacker rack.
2. If the plate is in the correct orientation, check the notch setting in the labware parameters:
   a. Open the **Controls** page of the **BenchCel Diagnostics** dialog box.
   b. In the **Labware** group box, select the **Show parameters** check box.
   c. In the **Labware Parameters** dialog box, verify that the correct notch position is selected.

3. Check the orientation sensor offset:
   a. Open the **Labware Editor** dialog box and click the **Stacker** tab.
   b. Verify the value in the **Orientation sensor offset**. This is the distance from the bottom of the plate to the middle of the sensor window.
4. Check the orientation (plate notch) sensor readings. See “Checking the stacker sensors” on page 314 for a procedure.
5. If readings do not detect the correct orientation, see “Resetting the orientation sensors for a plate” on page 313 for a procedure on how to adjust the sensors.

Plate presence and orientation sensors conflict

This error is generated when the plate presence sensor and the orientation sensors disagree as to whether there is a plate present or not. The problem is usually resolved by adjusting the plate presence sensor threshold and/or the orientation sensor offset.

To resolve plate not present error:

1. Check the orientation sensor offset:
   a. Open the **Labware Editor** dialog box and click the **Stacker** tab.
   b. Verify the value in the **Orientation sensor offset**. This is the distance from the bottom of the plate to the middle of the sensor window.
   c. In the **Labware** group box, select the **Show parameters** check box.
2. Check the plate sensor threshold and intensity:
   a. Open the **BenchCel robot** diagnostics and click the **General Settings** tab.
   b. Adjust the value in the **Plate presence threshold** field. Typically, this value is about 100.

      If you are getting a **no plate present** message when there is a plate, decrease this value.

      If you are getting a plate present message when there is none, increase this value.

Resetting the orientation sensors for a plate

About this topic

This topic describes how to reset the orientation sensors for a plate.

When to use

Use this procedure when you are using a new plate type and you are getting wrong plate type or plate is rotated messages in error. The typical situation for this procedure is when the new plate type has different optical properties than other plates with similar physical size.

Before starting

Ensure that all the plate parameters have been properly defined in the labware editor. Check and verify in the **Labware Editor** dialog box:

- ❑ Plate properties parameters
- ❑ BenchCel parameters
- ❑ Stack parameters (except orientation sensor threshold and intensity)

Procedure

To set the orientation sensors for a new plate:

1. Place a plate in one of the stacker racks on the BenchCel.
2. Open the **Controls** page of the **BenchCel Diagnostics** dialog box.
3. Select the correct **Stacker** from the list box.
4. Click **Move to sensor position**.
5. Adjust the orientation sensor intensity:
    a. Start with a intensity setting where the notch-orientation readout values are near saturation (about 200).
    b. Gradually reduce the intensity until the readouts start to drop.
    c. Note the values of the orientation sensor that is detecting the notched side or sides of the plate and the other sensors that are detecting the non-notched sides of the plate.
6. Set the orientation sensor threshold:
    a. Set the threshold to a number approximately halfway between the notch and no-notch readouts noted from the previous step.

       For example, if the orientation sensors for the no-notch sides of a plate read 180, 175, 185 and the notched side read 100, set the threshold to 140.

Typical settings

The following table lists some typical sensor values for plates.

Use it as a guide for the initial settings of the threshold and intensity values.

Plate Type	Orientation Sensor Threshold	Orientation Sensor Intensity
Clear, all types	100	50
Matte white	100	20
Shiny white	100	10
Matte black	60	80
Shiny black	60	50

Checking the stacker sensors

About this topic

Each stacker has the following sensors:

- ❑ Four plate notch sensors—one for each plate corner
- ❑ A rack presence sensor
- ❑ A plate presence sensor
- ❑ An air pressure sensor for the stacker grippers

When to use

Before and after each run, check the stacker sensors to make sure they are correctly sensing plate notches, racks and plates, and that the air pressure is within limits for your BenchCel application.

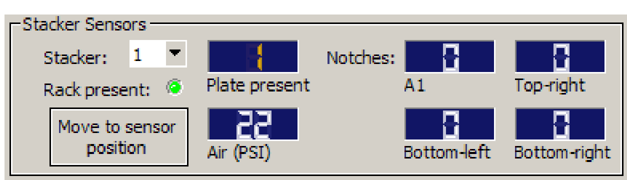
Procedure

To check the stacker sensors:

1. Open BenchCel Diagnostics.

   See "Opening BenchCel Diagnostics" on page 253.
2. From the **Stacker** list box, select the stacker you want to monitor.
3. If the **Notches** read zero, click **Move to sensor position**.

   This commands the robot to move the plate to where the sensors detect it.
4. Check the **Stacker Sensors** group box using the following table.



Setting	Comment
Rack present	This indicator light is green when a rack is present, and red when a rack is not present. To set the **Rack sensor threshold**, see “Changing general settings” on page 278.
Plate present	Any number above the **Plate in stack sensor threshold** indicates a plate is present. Numbers under the threshold are displayed red, while numbers over the threshold are displayed white. To set the **Plate in stack sensor threshold**, see “Changing general settings” on page 278.
Air (PSI)	Displays the air pressure of the selected stacker’s regulator, in pounds per square inch.
Notches	The far-left plate sensor is called A1 because that corner is intended for the A1 well of the microplate. Values that are less than the **Sensor threshold** indicate the presence of a notch. Values that are greater than the **Sensor threshold** indicate the absence of a notch. To set the **Sensor threshold** or **Sensor intensity**, see “Changing general settings” on page 278.

Resolving teachpoint problems

About this topic

Once set properly, you should never need to adjust or redefine a teachpoint unless you do one of the following:

- ❑ Move the BenchCel
- ❑ Change settings on external devices, such as the Z-height on a VCode
- ❑ Move one of the external devices accidently

Symptoms of a teachpoint error

The symptoms of a teachpoint error are the following:

- ❑ Robot is not placing the plate properly at an external teachpoint
- ❑ Robot is not picking the plate properly from an external teachpoint

Note: If the robot is not picking or placing a plate properly from an internal teachpoint (one of the stacks), then you most likely have a gripper alignment problem and not a teachpoint error.

Before changing the teachpoint

Before you adjust or redefine a teachpoint, always check to be sure that the grippers are aligned. See “Checking the arm alignment” on page 309 for a procedure to check the gripper alignment.

When to redefine or adjust the teachpoint

If you are sure the teachpoint is off, then either redefine the teachpoint or make an adjustment in the software as described below.

If...	Then...
You have moved the BenchCel	Redefine the teachpoint
You have changed the stage height on the VCode	Redefine the teachpoint
You have moved an external teachpoint slightly	Readjust the device and recheck the teachpoint. If it is very close, you can make a slight adjustment.

Adjusting the teachpoint

To adjust the teachpoint, you will measure the difference between the marks left on a plate by the grippers after picking the plate from an internal teachpoint and external teachpoint.

To adjust a teachpoint:

1. Open the **Controls** page of **BenchCel Diagnostics** dialog box.
2. Place a new plate in a stacker rack and click on the stacker teachpoint in the animated display.
3. Using the **pick from** command in the animated display, command the robot to pick the plate.

   Note the marks on the plate left by the grippers.
4. Place the same plate on the teachpoint you want to adjust.
5. Select the teachpoint in the animated display and click **pick from** to command the robot to pick up the plate.

   The marks on the plate left by the grippers this time will be in a different place.
6. Measure the distance between the gripper marks in the x and y direction and incorporate those differences in the teachpoint definition.

Resolving non-VCode bar code reader errors

About this topic

The information in this topic refers to errors generated by bar code readers that are not attached to a VCode. These include:

- Robot bar code readers
- Optional VPrep shelf bar code readers
- Optional platepad bar code readers

Types of bar code misread errors

There are two types of bar code misread errors.

Error type	Description
Bar code no-read	A bar code reader is unable to read a bar code when it picks up a plate.
Bar code mismatch	The bar code of the plate that is picked up does not match the bar code that BenchWorks expects for that plate.

Note: BenchWorks does not attempt to read a bar code unless you have specified (in the plate icon parameters) that incoming plates have bar codes on that side.

Bar code error option

The result of a bar code misread error depends on whether “Halt on bar code misreads” error option is selected.

If the option is selected:

- The protocol pauses.
- An error is generated in the log.txt file and VPrep log file.
- A dialog box opens, allowing you to enter the correct bar code.

If the option is not selected:

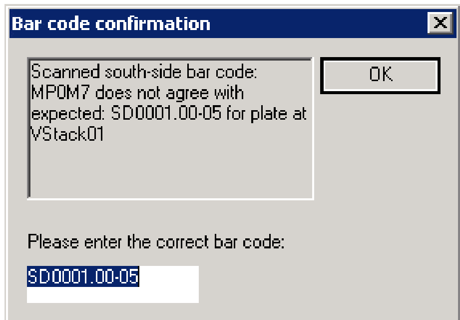
- An error is generated in the log.txt file and VPrep log file.
- The protocol continues without pausing so there are no recovery steps.

Bar code no-read errors

When a bar code cannot be read and the “Halt on bar code misreads” error option is selected, an error message is generated stating “Could not read *side* bar code on plate at *device*.”

Bar code mismatch errors

When the plate that is picked up does not match the bar code that BenchWorks expects in that position, an error message is generated stating “Scanned *side* bar code: *scanned bar code* does not agree with expected: *expected bar code* for plate at *location*.”



Recovering from bar code errors

To recover from a bar code error:

1. Enter the correct bar code in the text box of the **Bar code confirmation** dialog box.
2. Click **OK**.

   The run continues.

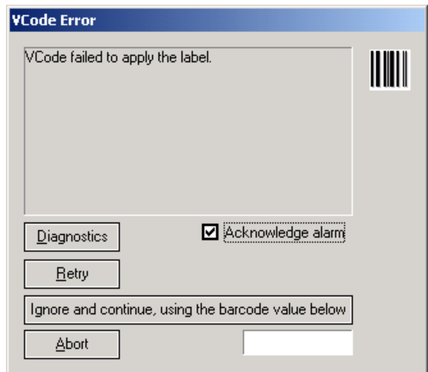
Related information

For information about...	See...
Resolving VCode errors	“Resolving VCode bar code reader errors” on page 319
Setting the error options	“About the Controls page” on page 254
	“About setting error options” on page 43

Resolving VCode bar code reader errors

About VCode errors

VCode errors, which include bar code errors, are generated by VCode devices and displayed in BenchWorks. An example VCode error is shown below.



For more information about resolving VCode errors, see the *VCode User Guide*.

Adjusting the stacker gripper pressure

About this topic

The stacker gripper pressure determines how forcefully a stacker grips a plate. It is set physically for each stacker and not in software. This topic describes how to adjust the BenchCel gripper pressure.

When to adjust the grip pressure

If the plates you are using are flimsy, lower the grip pressure from the default pressure of 50 psi to avoid distorting them.

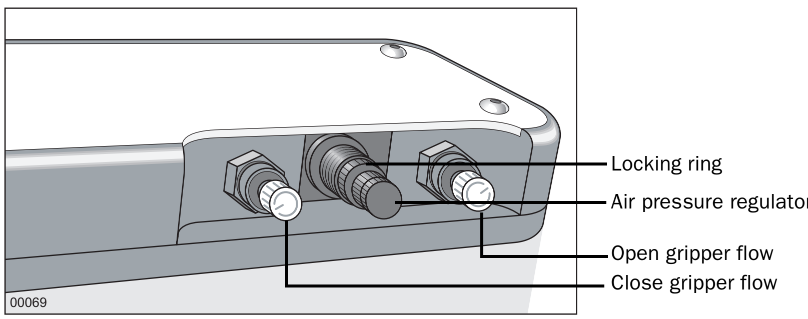
If your plate stack is heavily loaded with plates, the plates are made of a heavy material, or are full of liquid, the bottom plate may fall out of the stacker gripper. In this case, increase the stacker gripper pressure.

!! DAMAGE HAZARD !! Make sure the stacker gripper pressure does not exceed the maximum pressure specified in "Performing pre-run checks" on page 69.

Procedure

To adjust the grip pressure:

1. Locate the air pressure regulator.



   There are three knobs in the back of each stacker on the BenchCel. The central knob controls the air pressure to both stacker grippers.

2. Using your fingers or a pair of pliers, unscrew the locking ring of the central knob by turning it counterclockwise.
3. Turn the regulator clockwise to increase the pressure or counterclockwise to decrease the pressure.

   You can see the sensor reading for the stacker gripper pressure in the **Stacker Sensors** group box of the BenchCel Diagnostics Controls page.

   !! DAMAGE HAZARD !! Do not set the grip pressure lower than 15 psi.
4. Open and close the stacker grippers two to three times to allow the pressure to adjust.
5. Lock the locking ring by turning it clockwise until it is finger-tight.

Related information

For information about...	See...
Stacker sensors group box	"Checking the stacker sensors" on page 314
Stacker clamps	"Extending and retracting stacker clamps" on page 267

Sending a bug report

About this topic

This topic describes how to send a bug report to Velocity11 Service Center from BenchWorks.

Requirements

Before you can send a bug report:

- ❑ The system's computer must be connected to a network with internet access.
- ❑ The outgoing email server must be set up on the system's computer by a BenchCel or network administrator.

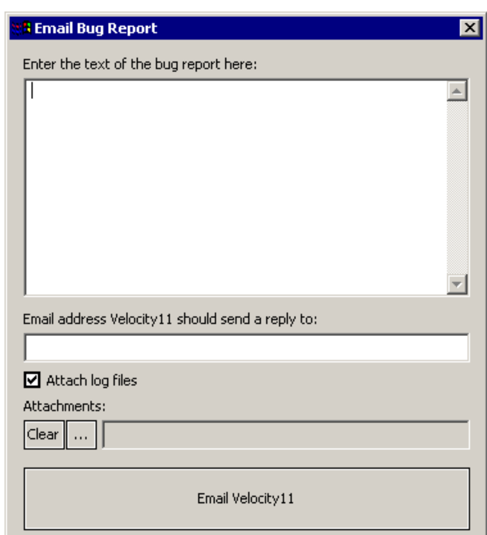
Sending a bug report

A bug report is an email that you create and send from within BenchWorks. The email automatically identifies your BenchCel and is sent directly to Velocity11.

To send a bug report:

1. Select **Help** > **Report a bug**.

   The **Email Bug Report** dialog box opens.



2. Type a description of the error in the text box.

   In your description, provide a summary of the error and, in the case of a software bug, a description of how we can reproduce it.

3. Attach the following files, which can help us to troubleshoot the problem:

File	Find in...
Protocol file you were running when the problem occurred	The location set by your administrator when saving protocols. The default location is C:\BenchWorks Workspace\protocols
Device file	The location set by your administrator when saving device files.
Protocol log file	The location set in log file options.
Velocity11 registry file	The Windows registry. HKEY_LOCAL_MACHINE\SOFTWARE\Velocity11

   a. Select the **Attach log files** check box.
   b. Click the ellipsis button.

   c. Browse to one of the files listed above and click **Open**.
   d. Repeat this step to attach each remaining file.

4. Click **Email Velocity11** and wait until a **Message Sent** message box appears.

Related information

For information about...	See...
Locating the protocol file	"Opening a protocol in BenchWorks" on page 40
Locating the device file	"Working with device files" on page 195
Locating the log files	"Setting log options" on page 54
Exporting a registry file	"Moving or sending a registry file" on page 208

BenchCel ActiveX controls

14

This chapter gives integrators the ActiveX control they need to integrate another company's lab automation device into the BenchCel.

The ActiveX has been verified to work with both Visual Studio 6 and Visual Studio.NET (v 7.1).

Setting up properties

About this topic

Properties are variables whose values can be set or retrieved by the container application.

IPictureDisp* ControlPicture

Description

Read-only property that the client can use to get an icon to represent the ActiveX control.

This example paints a VSpin bitmap over a button.

Visual C++	Visual Basic

/*The CPicture class is imported into your project when the ActiveX is installed*/
CButton button;
//Create a button
CPicture VSpinPic;
VSpinPic = m_CVSpin.GetControlPicture();
//Retrieve the picture
button.SetBitmap((HBITMAP)VspinPic.GetHandle());
/*Paint the bitmap onto the button*/
	'Assume that there is a button
	'named Command1 on the
	'current form. You must set
	'the style property of
	'Command1 to Graphical
	Command1.Picture = VSpin1.ControlPicture

SHORT Speed

Description

Property to specify how fast the BenchCel should move. 0 = slow, 1 = medium, 2 = fast. This property should not be changed during an operation. Setting this property to an invalid value will have no effect (call will be ignored).

Visual C++	Visual Basic

//Set the speed to fast
m_BenchCel.speed = 2;
	'Set the speed to fast
	BenchCel1.Speed = 2

BOOL Blocking

Description

Specifies whether the ActiveX should block during an execution of a command. If true, commands like PickAndPlace will not return until the action completes or an error occurs. The result value return from the call will indicate success (S_OK or 0) or failure. If false, the command will return immediately and fire an event to indicate successful completion of the command. Errors will be indicated through one of two means; 1) the return value might not be S_OK (0), in this case, no event will be fired; 2) an error event is fired. When an error occurs, the

ActiveX expects a call to Abort, Retry or Ignore. ShowDiagsDialog can be called to allow the user to exercise specific diagnostic/corrective functions, but when the main execution resumes, a call to Abort, Retry or Ignore is necessary to continue the operation.

Visual C++	Visual Basic

//Set the BenchCel to block until the command completes
m_BenchCel.Blocking =1;
	'Set the BenchCel to block until the command completes
	m_BenchCel.Blocking = TRUE

Setting up methods

About this topic

Methods are functions that can be called by the container application.

void AboutBox()

Description

Shows a small window that indicates some version information.

Parameters

none

Returns

none

Example

Visual C++	Visual Basic
BenchCel1.AboutBox()	m_BenchCel.AboutBox();

BSTR GetVersion ()

Description

Method to programmatically retrieve the version of the ActiveX.

Parameters

None

Returns

None

Example

Visual C++	Visual Basic
CString strVersion = m_BenchCel.GetVersion();	Version = BenchCel1.GetVersion()

LONG Initialize(BSTR Profile)

Description

Method to connect to the BenchCel device. A BenchCel profile specifies how to connect to the device (serial or Ethernet; if Ethernet, which device on the network and if serial, which port to use) and which teachpoint file to use. If this is called in non-blocking mode, the client application should wait for InitializeComplete before calling other methods. This method should be called before most other methods.

Parameters

Argument Type	Argument Name	Range	Description
BSTR	Profile	Valid profile name	The name of the profile to be used for initialization

Returns

S_OK (0) on success; other value otherwise.

Example

Visual C++	Visual Basic
LONG1Result = m_BenchCel.Initialize("ethernet");	LONG1Result = BenchCel1.Initialize("ethernet")

LONG ShowDiagsDialog(BOOL bModal, SHORT iSecurityLevel)

Description

Method to show the graphical diagnostics menu that allows the user to troubleshoot and correct problems. This method can be called before Initialize to create a profile.

Parameters

Argument Type	Argument Name	Range	Description
BOOL	bModal	TRUE,FALSE	Whether the diagnostics should be shown modally
SHORT	iSecurityLevel	0-3	The security level that the user has to operate the diagnostics 0 = Administrator 1 = Technician 2 = Operator 3 = Guest –1 - No access

Returns

LONG —no meaning.

Example

Visual C++	Visual Basic
m_BenchCel.ShowDiagsDialog(TRUE,0);	BenchCel1.ShowDiagsDialog 1, 0

void Close()

Description

Method to disconnect from the BenchCel device.

Parameters

None

Returns

None

Example

Visual C++	Visual Basic
m_BenchCel.Close();	BenchCel1.Close

BSTR GetLastError()

Description

Method to retrieve a text message explaining the last error. This method can be called in blocking mode, after a command returns with a failure code, or in non-blocking mode, after the Error event has been fired.

Parameters

None

Returns

None

Example

Visual C++	Visual Basic
strError = m_BenchCel.GetLastError();	strError = BenchCel1.GetLastError()

LONG Retry()

Description

Method to retry an action after an error occurred. For example, if there is insufficient air pressure during a LoadStack operation, the application can call Retry after the air pressure has been increased.

Parameters

None

Returns

None

Example

Visual C++	Visual Basic
m_BenchCel.Retry();	BenchCel1.Retry

LONG Abort()

Description

Method to clear an error and state information.

Parameters

None

Returns

None

Example

Visual C++	Visual Basic
m_BenchCel.Abort();	BenchCel1.Abort

LONG Ignore()

Description

Method to ignore the previously issued error. This is not a recommended course of action, as the errors are issued for a reason. However, ignoring some errors, such as “Plate is rotated”, can be appropriate if the operator understands the implications.

Parameters

None

Returns

None

Example

Visual C++	Visual Basic
m_BenchCel.Ignore();	BenchCel1.Ignore

LONG PickAndPlace(BSTR PickFrom, BSTR Place To, VARIANT_BOOL bLidded, LONG nRetractionCode)

Description

Method to transfer a plate. Stacker locations are called “Stacker 1”, “Stacker 2”, etc. Downstacking can be specified by using a stacker location for PickFrom and upstacking can be specified by using a stacker location for PlaceTo. bLidded indicates whether the robot should treat the plate as if it has a lid. nRetractionCode should be 3 (reserved for future options).

Parameters

Argument Type	Argument Name	Range	Description
BSTR	PickFrom	Valid teachpoint name	Destination to pick from
BSTR	PlaceTo	Valid teachpoint name	Destination to place to
VARIANT_ BOOL	bLidded	VARIANT_TRUE, VARIANT_FALSE	Whether the plate is lidded
LONG	nRetractioCode	0–2	0 = Do nothing 1 = Retract arms normally 2 = Retract arms only in Z + 0 clearance that is specified in teachpoint detail 3 = Retract arms so they are vertical

Returns

None

Example

Visual C++	Visual Basic
1Result = m_BenchCel.PickAndPlace("Stacker 1", "PlateLoc", FALSE, 2)	1Result = BenchCel1.PickAndPlace("Stacker 1", "PlateLoc", FALSE, 2)

VARIANT GetTeachpointNames()

Description

Method to retrieve the teachpoints known to the device. This method must be called after initialization is complete and it returns an array of available teachpoints, including the stackers.

Parameters

None

Returns

A safe array of teachpoint names

Example

Visual C++	Visual Basic

VARIANT vTeachpoints = m_BenchCel.GetTeachpointNames();
SAFEARRAY *psa = vTeachpoints.parray;
BSTR* bstrArray;
if (FAILED(SafeArrayAccessData(psa,reinterpret_cast<void**>(&bstrArray))))
{
VariantClear(&vTeachpoints);
return;
}
for (ULONG i = 0; i < psa->rgsabound[0].cElements; i++)
{
MessageBox(CString(bstrAdday[i]));
}
SafeArrayUnaccessData(psa);VariantClear(&vTeachpoints);
	teachpointNames = BenchCel.GetTeachpointNames
	For i= LBound(teachpointNames) To UBound(teachpointNames)
	MsgBox teachpointNames(i)
	Next

LONG ProtocolStart()

Description

Method to be called at the beginning of a run. The device is not expected to move.

Parameters

None

Returns

S_OK on success; other value on failure.

Example

Visual C++	Visual Basic
1Result = m_BenchCel.ProtocolStart();	1Result = BenchCel1.ProtocolStart()

LONG ProtocolFinish()

Description

Method to be called at the end of a run. The device might home during this call.

Parameters

None

Returns

None

Example

Visual C++	Visual Basic
1Result = m_BenchCel.ProtocolFinish();	1Result = BenchCel1.ProtocolFinish()

LONG SetLabware(BSTR bstrLabware)

Description

Method to set the labware to use. The selection will be in effect for all operations until a different labware is set. If diagnostics are shown and the user selects a different labware, the original labware will be restored when the diagnostics window is closed. This method should not be called when any movement is in progress.

Parameters

Argument Type	Argument Name	Range	Description
BSTR	bstrLabware	Valid labware name	Labware to be used for subsequent operations

Returns

S_OK if successful; other value if there was an error.

Example

Visual C++	Visual Basic
1Result = m_BenchCel.SetLabware("MyPlateType");	1Result = BenchCel1.SetLabware("MyPlateType")

LONG GetStackCount(LONG *pCount)

Description

Method to retrieve the number of stacks on the BenchCel device. This method must be called after a successful connection in order for it to indicate the current number.

Parameters

Argument Type	Argument Name	Range	Description
LONG*	pCount	Valid pointer to receive the stack count	If successful, the value pointed to by pCount should indicate the number of stacks the device has

Returns

S_OK if successful; other value otherwise.

Example

Visual C++	Visual Basic
1Result = m_BenchCel.GetStackCount(&numStacks);	1Result = BenchCel1.GetStackCount(numStacks)

LONG IsStackLoaded(SHORT sStack, [in, out] VARIANT_BOOL* pLoaded

Description

Method to test whether a stack has been loaded. The stack number is 0-based. This method should be called after a successful connection.

Parameters

Argument Type	Argument Name	Range	Description
SHORT	sStack	0 to n-1, where n is the number of stacks	Which stack to check
VARIANT_BOOL*	pLoaded	Valid pointer to receive whether or not the stack is loaded	On a successful call, the value pointed to by pLoaded should indicate whether the stack is loaded or not

Returns

S_OK if successful; other value otherwise.

Example

Visual C++	Visual Basic
1Result = m_Benchcel.IsStackLoaded(1,&bStackLoaded);	1Result = BenchCel1.IsStackLoaded(1,bStackLoaded)

LONG IsPlatePresent(SHORT sStack, [in, out] VARIANT_BOOL* pPresent

Description

Method to test whether a stack has a plate and is loaded. If the stack is not loaded, the result returned through pPresent will not be meaningful. The stack number is 0-based. This method should be called after a successful connection.

Parameters

Argument Type	Argument Name	Range	Description
SHORT	sStack	0 to n-1, where n is the number of stacks	Which stack to check

Argument Type	Argument Name	Range	Description
VARIANT_ BOOL*	pLoaded	Valid pointer to receive whether a plate is present	On a successful call, the value pointed to by pPresent should indicate whether the stack is loaded and has a plate available for downstacking

Returns

S_OK if successful, other value otherwise.

Example

Visual C++	Visual Basic
1Result = m_Benchcel.IsPlatePresent(1,&bPlatePresent);	1Result = BenchCel1.IsPlatePresent(1,bPlatePresent)

LONG ReleaseStack(SHORT sStack)

Description

Method to release a stack. A released stack can be freely taken from the device for the loading or unloading of plates. However, the BenchCel cannot downstack from or upstack to a released stack. The stack number is 0-based.

Note: This method can also be used to perform the close clamp function.

Parameters

Argument Type	Argument Name	Range	Description
SHORT	sStack	0 to n-1, where n is the number of stacks	The stack to be released

Returns

S_OK if successful, other value otherwise.

Example

Visual C++	Visual Basic
1Result = m_BenchCel.ReleaseStack(0);	1Result = BenchCel1.ReleaseStack(0)

LONG LoadStack(SHORT sStack)

Description

Method to release a stack. To downstack from or upstack to a stack, the stack must be loaded. A loaded stack is locked into the stacker head and cannot be freely taken from the device. The stack number is 0-based.

Parameters

Argument Type	Argument Name	Range	Description
SHORT	sStack	0 to n-1, where n is the number of stacks	The stack to be loaded

Returns

S_OK if successful; other value otherwise.

Example

Visual C++	Visual Basic
1Result = m_BenchCel.LoadStack(0);	1Result = BenchCel1.LoadStack(0)

LONG MoveToHomePosition()

Description

Method to move the device to the origin. This method is not commonly used.

Parameters

None

Returns

S_OK if successful; other value otherwise.

Example

Visual C++	Visual Basic
1Result = BenchCel.MoveToHomePosition();	1Result = BenchCel1.MoveToHomePosition()

LONG ShowLabwareEditor(BOOL bModal, BSTR bstrLabware)

Description

Method to display the labware editor graphical user interface. Through this interface dialog, the user can specify labware parameters that will be used by the device to handle the plates. Parameters such as plate height and notch information will be associated with a labware name, which can be used by SetLabware to indicate to the device how to handle the next plate.

Parameters

Argument Type	Argument Name	Range	Description
BOOL	bModal	TRUE, FALSE	Whether to show the editor modally or not

Argument Type	Argument Name	Range	Description
BSTR	bstrLabware	Valid labware name	The labware to be selected when the editor is displayed

Returns

S_OK if successful; other value otherwise.

Example

Visual C++	Visual Basic
m_BenchCel.ShowLabwareEditor(1,"MyPlateType");	BenchCel.ShowLabwareEditor 1,"MyPlateType"

LONG Pause()

Description

Method used to pause the BenchCel movement.

Parameters

None

Returns

S_OK if successful; other value if there was an error.

LONG Unpause()

Description

Method used to unpause the BenchCel movement. The BenchCel will continue any remaining movements before the call to pause the device.

Parameters

None

Return

S_OK if successful; other value if there was an error.

LONG OpenClamp()

Description

Method used to open the stacker grippers of a given stack.

Parameters

Argument Type	Argument Name	Range	Description
SHORT	Stack	0 – (number of Stacks – 1)	Specify which stack’s gripper to open

Returns

S_OK if successful; other value if there was an error.

IsConnected()

Description

Method used to check whether a connection to the BenchCel is established. The BenchCel is ready to process commands from the BenchCel Active X driver when a connection has been established (using the Initialize() method).

Parameters

None

Returns

1 if there is a connection and 0 if disconnected.

LONG Delid()

Description

Method used to remove a lid from a plate. You will need to specify where the plate is located and where to place the lid once it is removed from the plate. The DelidTo parameter is optional. The BenchCel robot will hold onto the lid if the DelidTo parameter is empty. The third argument, RetractionCode, specifies how to position the arms after delidding the plate:

1– retract normally

2– retract only in Z to clearance that is specified in the teachpoint details

3– retract so that arms are vertical

Parameters

Argument Type	Argument Name	Range	Description
BSTR	DelidFrom	Available teachpoints	Name of teachpoint where the plate with the lid is located

Returns

S_OK if successful; other value if there was an error.

LONG Relid()

Description

Method used to put a lid on a plate. You will need to specify where the lid is located and where the plate is located. If the first RelidFrom argument is blank, then it is expected that the robot is holding the lid. The third argument, RetractionCode, specifies how to position the arms after relidding the plate:

1– retract normally

2– retract only in Z to clearance that is specified in the teachpoint details

3– retract so that arms are vertical

Parameters

Argument Type	Argument Name	Range	Description
BSTR	DelidFrom	Available teachpoints or blank string	Name of teachpoint where the plate with the lid is located
BSTR	DelidTo	Available teachpoints	Name of teachpoint to place the lid after it has been removed from the plate
Int	nRetraction Code	1–3	Specify the retraction position of the BenchCel robot arms

Returns

S_OK if successful; other value if there was an error.

VARIANT GetLabwareNames()

Description

Method to retrieve a list of defined labware. The strings in this array are the options that should be used for SetLabware.

Parameters

None

Returns

An array of labware names

Example

Visual C++	Visual Basic

VARIANT vLabware =
m_BenchCel.GetLabwareNames(
);
SAFEARRAY *psa =
vLabware.parray;
if
(FAILED(SafeArrayAccessData(ps
a, reinterpret_cast<void**>(
&bstrArray))))
{
VariantClear(&vLabware);
return;
}
for (ULONG i = 0; i < psa-
>rgsabound[0].cElements; i++)
{
MessageBox(CString(bstrArray[i])
));
}
SafeArrayUnaccessData(psa);
VariantCLear(&vLabware);
	LabwareNames =
	BenchCel.GetLabwareNames
	For i = LBound(labwareNames) To
	UBound(labwareNames)
	MsgBox labwareNames(i)
	Next

VARIANT EnumerateProfiles()

Description

Method to retrieve a list of defined profiles. The strings in this array are the options that should be used for Initialize.

Parameters

None

Returns

An array of profile names.

Example

Visual C++	Visual Basic

VARIANT vPRofiles =
m_BenchCel.EnumerateProfiles();
SAFEARRAY *psa =
vProfiles.parray;
BSTR* bstrArray;
if
(FAILED(SafeArrayAccessData(psa,
reinterpret_cast<void**>(
&bstrArray))))
{
VariantClear(&vProfiles);
return;
}
for (ULONG i = 0; i < psa-
>rgsabound[0].cElements; i++)
{
MessageBox(CString(bstrArray[i]))
);
}
SafeArrayUnaccessData(psa);
VaraintClear(&vProfiles);
	profileNames =
	BenchCel.EnumerateProfiles()
	For i = LBound(profileNames) To
	UBound(profileNames)
	MsgBox profileNames(i)
	Next

Index

Note: You can also search our technical documentation on our website at www.velocity11.com/support/support.html.

Symbols

.bar filename extension, *41*
.bwl file format, *30*
.dev file format, *30*
.xml file format, *30*

A

ABgene SEAL-IT 100
  supported instrument, *18*
aborting a run, *74*
acceleration property, *248*, *249*
Acrobat Reader, *4*
Acrobat viewer, *4*
adding
  labware entries, *221*
  new teachpoint, *255*
  user accounts, *205*
administrator privilege, *85*, *204*
Advanced Settings tab, *172*
air
  connecting, *26*
  input on connection panel, *23*
  pressure control stacker head, *24*
  required pressure, *20*, *26*
  stacker gripper flow control, *24*
Aliquot task
  defined, *100*
  setting parameters, *100*
AliQuot, supported instrument, *18*
Apply Label task, *164*
  defined, *101*
  JavaScript properties, *178*
  setting parameters, *102*
aspirate acceleration, *155*, *160*
Aspirate pipette task
  adding, *138*
  defined, *144*
  JavaScript properties, *180*
  setting parameters, *144*
aspirate velocity, *155*, *159*
aspirate, liquid class, *144*
aspiration, quadrant, *145*
automatically release stacker racks after protocol finishes (protocol rule), *50*
axes, BenchCel robot, *256*

B

bar code
  auxiliary reader, *17*
  error, no read, *317*
  errors, *318*
  errors, mismatch, *318*
  field modifiers, *106*
  fields, example, *198*
  format files, *101*
  input file, *165*, *198*
  labelling requirements, *101*
  log file, *53*, *107*
  misread errors, *44*, *317*
  printer, *197*
  readers, about, *197*
  updating input file, *200*
  using data files, *200*
  using input files, *198*
  viewing input file, *199*
barcodelog.txt, *53*
BenchCel
  about connecting, *21*
  about error handling, *301*
  accessories, *14*, *17*
  animated display, description, *255*, *256*
  available configurations, *15*
  changing the robot speed, *262*
  compatible Velocity11 modules, *18*
  compatible with other companies' instruments, *18*
  compressed air requirement, *20*, *26*
  connecting by Ethernet, *25*
  connecting by serial, *25*
  connecting compressed air, *26*
  connection panel, *16*
  creating a protocol, *85*
  creating pre- and post-protocol processes, *87*
  defining plate properties, *229*
  definition, *14*
  description, *14*
  editing teachpoints, *272*
  emergency stop accessory, *18*
  front features, *16*
  hazard warnings, *11*
  integration plate, *18*
  jogging the robot, *255*

jogging the robot gripper position, *261*
jogging the robot theta-axis, *260*
jogging the robot X-axis, *260*
jogging the robot Z-axis, *259*
minimum system requirements, *21*
monitoring a run, *76*
moving between teachpoints, *275*
moving the robot manually, *258*
network cards, *21*
pausing a run, *74*
performing a run, *31*, *67*
performing a run overview, *68*
performing pre-run checks, *69*
preparing for a run workflow, *32*
preparing instruments for a run, *61*
robot axes, *256*
routine maintenance, *300*
setting error options, *43*
setting general options, *41*
setting simultaneous plates, *91*
shutting down, *81*
space setup requirements, *19*
stacker controls, *264*
stacker-gripper controls, *24*
starting, *33*
starting run from command line, *73*
temperature range, *20*
using the robot slider controls, *263*
utility and environment requirements, *20*
BenchWorks
control toolbar, *28*
finding the version number, *5*, *8*
introduction and overview, *27*
logging out, *81*
new features, *7*
protocol definition, *30*
starting, *34*
Bio-Tek ELx405 Washer. *see* Bio-Tek
Bio-Tek, supported instrument, *18*
blowout volume, *152*, *156*, *160*

C

calibration file, Nanodrop, *290*
can be mounted property, *228*
can be sealed? property, *227*
can have lid? property, *227*, *240*
can mount property, *228*
Change Instance pipette task
defined, *146*
setting parameters, *147*
Change Instance process task, *146*
Change Tips pipette task
defined, *148*
requirements for, *148*
using, *148*
check orientation property, *231*
chimney. *see* MicroWash tray
class, labware, *215*
command line BenchWorks launch, *73*
compiling
a protocol, *93*
errors, *303*
warnings, *303*
compressed air requirement, *20*, *26*
computer
connecting by Ethernet, *25*
connecting by serial, *25*
disk space option, *44*
networking. *see* connecting by ethernet
configuration components, *30*
configuration settings for pipette tasks, *192*
connection panel, *22*
indicator lights, *22*
serial interface, *22*
control toolbar buttons, *28*
controls page, overview, *254*
copying a labware entry, *224*
creating a protocol, workflow, *85*
creating process for a tipbox, *149*

D

database
copying, *208*
labware, *152*, *156*, *161*, *183*, *186*, *191*, *208*
liquid library, *208*
deadlock, *91*, *97*, *308*
deleting labware entries, *222*
deleting user accounts, *206*
Delid task
defined, *108*
device
configuring VPrep shelf, *142*
defined, *195*
initialization errors, *302*
device file, *195*, *322*
defined, *195*
definition, *30*
filename extension, *30*
loading, *196*
location, *195*
saving, *195*
device manager, relationship with other configura-
tions, *192*
diagnostics
about, *252*
finding the version number, *8*

identifying a QFill2, *291*
initializing Multidrop, *286*
Multidrop plate filling, *286*
Multidrop priming, *285*, *287*
Multidrop purging, *285*, *287*
Multidrop shaking, *287*
Multidrop using, *285*
Nanodrop, using, *288*
opening, *253*
Qfill2 dispensing, *292*
Qfill2 filtration port, *292*
WellMate executing commands, *296*
WellMate plate stage, *295*
WellMate priming and purging, *296*
WellMate profiles, *293*
WellMate query plate, *296*
WellMate, about, *293*
disable all robot servos, *255*
dispense acceleration, *151*, *155*, *160*
dispense limits, *150*
Dispense pipette task
adding, *139*
defined, *150*
JavaScript properties, *182*
setting parameters, *151*
dispense velocity, *151*, *155*, *160*
dispense volume, *151*
dispensing to waste, *159*, *161*
disposing of waste, *10*
distance from well bottom, *151*
Downstack task
defined, *109*
setting parameters, *109*
Dry Tips pipette task, *153*

E

Echo 550
supported instrument, *18*
email
notification of errors by, *46*, *206*
setting up, *45*, *206*
when errors occur, *44*
emergency stop, description, *18*, *23*
environment requirements, *20*
errors
about, *301*
bar code reader, *317*
causes, about, *301*
handling, *301*
hardware, reporting, *9*
message display. see Log toolbar.
notification of, *46*, *206*
reporting, *9*
setting handling options, *44*
setting handling options for protocols, *47*
setting options, *43*
setting scheduler behavior options, *45*
troubleshooting initialization, *302*
VCode, *319*
Ethernet
connecting to, *25*
interface on connection panel, *22*
parts included, *14*
port, *22*
event messages. see log toolbar

F

file
device, *322*
format for protocol, *84*
protocol, *322*
protocol log, *322*
FileReader plug-in, *164*
text file format, *166*
filtration station, *292*
Find button, *4*
finding the file, *3*
firmware version, *6*
FlexDrop
supported instrument, *18*
fuse, type, *23*

G

general options, setting, *41*
general properties for labware, defining, *225*
general settings
about, *277*
changing, *278*
global changes, *281*
Genetix aliQuot. *see* AliQuot
Genetix QFill2 Dispenser. *see* QFill2
gripper, stacker control, *24*
guest privilege, *204*

H

Handle plates in instance order (protocol rule), *50*
hardware overview, *16*
hazard warnings, *11*
Home the robot, *255*
Homing Offsets, *279*
homing the robot, *257*

I

image root folder, *53*
image, inserting, *232*
Incubate task

defined, *112*
setting parameters, *112*
Innovadyne Nanodrop. *see* Nanodrop
instruments
defined, *14*
preparing for a run, *61*
integration plate, *18*

J

JavaScript
about, *171*
cautions when using, *176*
examples, *177*
global functions, *172*
plate object, *173*
resources, *171*
scripts, *85*
script-writing service, *177*
task object, *174*, *178*
variables, *175*
JavaScript scripts, *86*
job roles for readers of this guide, *2*

L

Labcyte Echo 550. *see* Echo 550
labware
adding, *221*
associating plate with a class, *235*
classes page, *233*
classes, about, *215*
copying, *224*
customizing parameters, *238*
defining BenchCel properties, *229*
defining classes, *233*
defining general properties, *225*
defining plate properties, *226*
defining stacker properties, *230*
deleting, *222*
editor properties, *215*
editor, about, *215*
inserting an image, *232*
looking for definition, *218*
methods for defining, *214*
opening parameters group box, *237*
plate icon, *37*
renaming, *223*
standards for, *218*
using parameters group box, *236*
labware classes sub-page, *234*
labware database, *152*, *156*, *161*, *183*, *186*, *191*, *208*
labware editor
classes page, *217*
entries page, *216*
opening, *220*
relationship with other configurations, *192*
LabwareSelector plug-in, setting up, *164*
length of filter tip/pin tool property, *228*
Liconic STR. *see* StoreX
Liconic STX (StoreX). *see* StoreX
lid departure height property, *228*, *241*
lid gripper offset property, *227*, *241*
lid resting height property, *227*, *241*
lidded stacking thickness property, *227*
lidded thickness property, *227*
lights, connection panel, *22*
linking a pipette process, *141*
liquid class, *151*
liquid library
adding an entry, *248*
calibrating VPrep, *244*
changing database, *208*
classes defined, *244*
database defined, *244*
editor defined, *244*
opening the editor, *246*
liquid library editor
about, *245*, *248*
opening, *246*
relationship with other configurations, *192*
loading a rack, *265*
log file
bar code, *53*, *107*
importing into Excel, *57*
pipette, *52*
protocol, *52*, *58*
searching, *78*
setting options for, *55*
types, *51*
Log toolbar, *29*, *72*, *77*
adding a note, *79*
message display, *55*
options for, *78*
log.txt file
about, *52*
adding a note to, *79*
logging out of BenchWorks, *81*
Loop pipette task
defined, *153*
JavaScript properties, *187*
setting parameters, *154*
lower plate at VCode property, *228*

M

maintenance, routine, *300*
manifold of MicroWash tray, *162*

Materials Safety data Sheet. *see MSDS*
Matrix WellMate. *see* WellMate
maximum robot handling speed property, *228*
MicroWash tray, *161*, *162*
Mix pipette task
  defined, *155*
  JavaScript properties, *184*
  setting parameters, *155*
mixing cycles, *155*
mixing volume, *155*
Molecular Devices instruments with SoftMax Pro. *see* Softmax Pro
monitoring a run, *76*
move-to-position commands, *263*
MSDS, *10*
Multidrop
  diagnostics, using, *285*
  filling a plate, *286*
  initializing, *286*
  priming, *285*, *287*
  purging, *285*, *287*
  shaking, *287*
  supported instrument, *19*
Multidrop task
  defined, *114*
  setting parameters, *115*
Multiskan Ascent
  supported instrument, *19*

N

Nanodrop
  calibration file, *290*
  diagnostics, using, *288*
  profiles, *288*
  supported instrument, *18*
Nanodrop task
  defined, *116*
  setting parameters, *116*
network connections
  obtaining IP addresses, *211*
  viewing, *210*
notch locations property, *231*
Number of Cycles dialog box, *71*

O

online help, *3*
opening the labware editor, *220*
operating system. *see* Windows
operating temperature range, *20*
operator privilege, *204*
options
  log and data file, *54*
  protocol error, *47*
  setting error, *44*
  setting general, *41*
orientation sensor offset property, *231*
orientation sensor threshold (max) property, *231*
orientation sensor threshold (min) property, *231*

P

pager, notification of errors by, *46*, *206*
password, changing, *36*
pausing a run, *74*
PDF guide, *3*, *4*
PerkinElmer FlexDrop. *see* FlexDrop
PerkinElmer ViewLux. *see* ViewLux
Pierce task
  defined, *117*
  setting parameters, *117*
pipette log file, *52*
Pipette plates in instance order (protocol rule), *49*
pipette process
  configuration settings, *192*
  creating for changing pipette tips, *149*
  defined, *40*
  link icon, *39*
  linking a pipette task to, *141*
  setting parameters, *140*
pipette process task
  adding, *136*, *140*
  adding example, *136*
  defined, *39*
pipette task defined, *39*
pipette tips, *148*
  changing, *148*, *149*
  touching, *161*
  washing, *159*
pipetting operation, example, *136*
piplog.txt, *52*
Place Labware task, setting parameters, *118*
Place Plate task
  defined, *120*
  setting parameters, *120*
plate icon
  definition, *37*
  setting parameters, *88*
  versus task, *38*
plate object, JavaScript, *173*
plate properties, defining, *226*
PlateLoc
  see also Seal task
  setting up, *61*
PlatePierce
  see also Pierce task
plate-present indicator, *264*
plates

see also simultaneous plates
plug-in, *85*
FileReader format, *166*
FileReader, about, *164*
setting up LabwareSelector, *164*
post-aspirate delay property, *249*
post-dispense delay property, *249*
post-dispense volume, *152*
post-protocol process
defined, *39*
setting up, *87*
power supply, main, *22*
powering-up, BenchCel, *33*
pre-aspirate volume, *160*
preparing for a run
instruments, *61*
preparing for run
workflow, *32*
pre-protocol process
defined, *38*
setting up, *87*
pre-protocol rules
about, *48*
setting, *48*
pre-run checks, performing, *69*
presentation offset property, *231*
priming, *285*, *287*
printing a protocol, *60*
privileges
defined, *204*
effect of, *204*
types, *204*
process, defined, *38*
profile
creating, *283*
defined, *30*, *282*
editing, *284*
property
acceleration, *248*, *249*
can be mounted, *228*
can be sealed?, *227*
can have lid?, *227*, *240*
can mount, *228*
check orientation, *231*
length of filter tip/pin tool, *228*
lid departure height, *228*, *241*
lid gripper offset, *227*, *241*
lid resting height, *227*, *241*
lidded stacking thickness, *227*
lidded thickness, *227*
lower plate at VCode, *228*
maximum robot handling speed, *228*
notch locations, *231*
orientation sensor offset, *231*
orientation sensor threshold, *231*
post-aspirate delay, *249*
post-dispense delay, *249*
presentation offset, *231*
sealed stacking thickness, *227*
sealed thickness, *227*
sensor intensity, *231*
stacker gripper offset, *230*
stacking thickness, *227*, *238*
thickness, *226*, *238*
use vacuum clamps, *231*
velocity, *248*, *249*
z-axis acceleration into wells, *249*
z-axis acceleration out of wells, *249*
z-axis speed into wells, *248*
z-axis velocity into wells, *249*
z-axis velocity out of wells, *249*
protocol
adding notes about, *94*
compiling, *93*
creating example, *136*
creating pre and post processes, *87*
defined, *38*
file finding, *322*
file format, *84*
filename extension, *30*
log file, *52*, *322*
log file example, *58*
optimizing, *48*
options, setting rules, *48*
overall process of creating, *85*
printing, *60*
run until done, *72*
saving, *94*
setting rules, *49*
simulating, *97*
specifying order of tasks, *132*
understanding before a run, *59*
what you should know, *59*
Protocol Task Parameters toolbar, *166*
pump
inflow, *161*
outflow, *161*
Pump Reagent pipette task
defined, *157*
JavaScript properties, *187*
setting parameters, *158*
purging, *285*, *287*

Q

QFill2
diagnostics, using, *291*

- dispensing, *292*
  - filtration port, *292*
  - identifying, *291*
- QFill2 task
  - defined, *121*
  - setting parameters, *121*
- QFill2, supported instrument, *18*
- QRM file, Nanodrop, *290*
- quadrant, *145*, *152*, *156*, *160*
- quadrant representation in JavaScript, *176*

R

- rack
  - loading and releasing, *265*
  - loading and releasing all racks, *267*
- registry file, *322*
- registry key
  - emailing, *209*
  - exporting, *208*
  - importing, *209*
- releasing a rack, *265*
- Relid task
  - defined, *108*
- renaming a labware entry, *223*
- reporting errors, *9*
- requirements
  - space, *19*
  - system, *21*
  - utility and environment, *20*
- reservoir filling. *see* VPrep
- reset all robot servos, *255*
- Restack task
  - setting parameters, *123*
- retract distance, *151*, *156*, *160*
- robot
  - changing the speed, *262*
  - homing, *257*
  - jogging, *259*
  - jogging the gripper position, *261*
  - jogging the theta-axis, *260*
  - jogging the X-axis, *260*
  - jogging the Z-axis, *259*
  - limits, *280*
  - moving manually, *258*
  - preventing crashes, *41*
  - sending home, *257*
  - slider controls, *263*
- routine maintenance, *300*
- run
  - aborting, *74*
  - defined, *39*
  - monitoring, *76*
  - performing overview, *68*
  - performing pre-run checks, *69*
  - protocol until done, *72*
  - starting, *70*
  - starting from command line, *73*
  - what happens after, *72*
- running a protocol
  - workflow, *32*, *68*

S

- safety
  - interlock circuit, *23*
  - labels, *10*
  - training, *10*
- saving protocols, *94*
- scheduler behavior options, *45*
- screen message options, *78*
- Seal task
  - defined, *124*
  - setting parameters, *124*
- sealed stacking thickness property, *227*
- sealed thickness property, *227*
- SEAL-IT Adhesive Sealer. *see* ABgene SEAL-IT 100
- send the robot home, *255*, *257*
- sensor intensity property, *231*
- serial
  - connecting, *25*
  - interface on connection panel, *22*
- shaking, *287*
- shelf, VPrep configuring, *142*
- shutting down, *81*
- Signal task
  - example, *132*
  - setting parameters, *127*
- simulating a protocol, *97*
- simultaneous plates, *89*, *91*
- SoftMax Pro
  - supported instrument, *18*
- software versions, *5*
- space requirements, *19*
- speed, *255*
- stacker
  - adjusting clamp pressure, *320*
  - adjusting gripper pressure of, *320*
  - configurations, *15*
  - controls, *264*
  - defining properties for labware, *230*
  - extending and retracting clamps, *267*
  - head pressure control, *24*
  - plate-present indicator, *264*
  - settings, *278*
- stacker gripper
  - control, *24*
  - offset property, *230*

stacker sensors, *255*
- checking, *314*
- list of, *314*

Stacker task
- defined, *109*
- setting parameters, *110*

stacking thickness property, *227*, *238*
starting
- a run from a command line, *73*
- BenchWorks, *34*
- runs, *70*
- the BenchCel, *33*

stopping a run, *74*
StoreX
- supported instrument, *18*

system requirements, *21*

T

task object, JavaScript, *174*
- properties, *178*

task parameter setting, *95*
Task Parameters toolbar. *see* Protocol Task Parameters toolbar, *166*
tasks
- adding, *92*
- defined, *37*
- deleting, *92*
- moving, *93*
- specifying order in a protocol, *132*
- types, *95*

teachpoint
- about, *269*
- adding, *270*
- deleting, *273*
- editing, *272*
- file, definition, *30*
- filename extension, *30*
- moving between, *275*
- selecting two, *276*

technician privilege, *85*, *204*
temperature, operating range, *20*
Thermo Electron Multidrop. *see* Multidrop
Thermo Electron Multiskan Ascent. *see* Multiskan Ascent
thickness property, *226*, *238*
tip touch horizontal distance, *152*, *156*
tip touch rise height, *152*, *156*
tip touching, *152*, *156*
tipbox, *149*
tips. *see* pipette tips
toolbars
- closing, *77*
- control, *28*
- Log, *29*, *77*

transfer log, *152*
troubleshooting, *299*
- bar code errors, *317*
- files to send, *322*
- initialization errors, *302*

U

Upstack task
- defined, *109*
- setting parameters, *109*

use vacuum clamp property, *231*
user account
- adding, *205*
- creating, *204*
- deleting, *206*
- privileges, *204*

User Message task, *72*
- defined, *125*
- setting parameters, *126*

utility requirements, *20*

V

vacuum filtration. *see* QFill2
VCode
- about, *197*
- errors, *319*
- setting up, *61*

velocity property, *248*, *249*
Velocity11 privilege, *204*
ViewLux, supported instrument, *18*
volume
- blowout, *152*, *156*, *160*
- dispense, *151*
- mixing, *155*
- post-dispense, *152*
- pre-aspirate, *160*
- wash, *159*

VPrep
- changing tips, *149*
- configuring a shelf, *137*, *142*
- reservoir, *158*
- selecting, *137*
- setting up, *62*

VSpin
- *see also* VSpin with Access2 task

VSpin with Access2 task
- defined, *130*
- setting parameters, *130*

W

Waitfor task
- example, *132*

setting parameters, *127*
warnings, hazard, *11*
warranty, *10*
Wash Tips pipette task
defined, *159*
JavaScript properties, *188*
wash volume, *159*
washing pipette tips, *159*
waste disposal, *10*
WellMate
creating profiles, *293*
diagnostics, using, *293*
execute commands, *296*
moving plate stage, *295*
priming and purging, *296*
querying plate type, *296*
supported instrument, *18*
WellMate task
defined, *128*
setting parameters, *129*
Windows, logging on, *34*
Windows, registry key, copying, *208*

X

X margin settings, *279*
XML schema, *84*

Z

z-axis acceleration into wells property, *249*
z-axis acceleration out of wells property, *249*
z-axis speed into wells property, *248*
z-axis velocity into wells property, *249*
z-axis velocity out of wells property, *249*

Agilent Technologies

User Guide

G5400-90002