

Agilent Automated Centrifuge Loader ActiveX

Version 9.0.0

User Guide

Original Instructions

Notices

© Agilent Technologies, Inc. 2012

No part of this manual may be reproduced in any form or by any means (including electronic storage and retrieval or translation into a foreign language) without prior agreement and written consent from Agilent Technologies, Inc. as governed by United States and international copyright laws.

User Guide Part Number

G5405-90004

Edition

Revision 00, April 2012

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)1763208826
euroservice.automation@agilent.com

Documentation feedback:
documentation.automation@agilent.com

Web:
www.agilent.com/lifesciences/automation

Acknowledgements

Microsoft® and Windows® are either registered trademarks or 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 sub-contract, 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

(June 1987) or DFAR 252.227-7015 (b)(2) (November 1995), as applicable in any technical data.

Safety Notices

 **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.

Contents

Preface	iii
About this guide	iv
Reporting problems	iv
Centrifuge Loader ActiveX control	1
About ActiveX controls	2
Properties	3
Methods	6
Events	21

Contents



Preface

This preface contains the following topics:

- “About this guide” on page iv
- “Reporting problems” on page iv

About this guide

What this guide covers

This guide describes the ActiveX controls for the Agilent Automated Centrifuge Loader.

This guide does not provide instructions for setting up and using the Centrifuge Loader. For these details, see the *Automated Centrifuge Loader User Guide*.

Accessing Agilent Technologies Automation Solutions user guides

You can search the online knowledge base or download the latest version of any PDF file from the Agilent Technologies website at www.agilent.com/lifesciences/automation. Safety information for the devices appears in the corresponding device user guide. You can also search the knowledge base or the PDF files for safety information.

Related topics

For information about...	See...
How to set up and use the Centrifuge Loader	<i>Automated Centrifuge Loader User Guide</i>
Reporting problems	"Reporting problems" on page iv

Reporting problems

Contacting Automation Solutions Technical Support

If you find a problem with the Centrifuge Loader, contact Automation Solutions Technical Support. For contact information, see Notices on the back of the title page.

Reporting hardware problems

When contacting Agilent Technologies, make sure you have the serial number of the device ready.

Reporting software problems

When you contact Automation Solutions Technical Support, make sure you provide the following:

- Short description of the problem
- Relevant software version number (for example, automation control software, diagnostics software, ActiveX control software, and firmware)
- Error message text (or screen capture of the error message dialog box)
- Relevant files, such as log files

Reporting user guide problems

If you find a problem with this user guide or have suggestions for improvement, send your comments in an email to documentation.automation@agilent.com.

Related topics

For information about..	See...
How to set up and use the Centrifuge Loader	<i>Automated Centrifuge Loader User Guide</i>
Accessing user information	“Accessing Agilent Technologies Automation Solutions user guides” on page iv

Preface

Reporting problems



Centrifuge Loader ActiveX control

This chapter gives integrators the ActiveX control information required to integrate the Agilent Automated Centrifuge Loader into another company's lab automation system. The ActiveX has been verified to work with both Visual C++ and Visual Basic .NET.

This chapter contains the following topics:

- “About ActiveX controls” on page 2
- “Properties” on page 3
- “Methods” on page 6
- “Events” on page 21

About ActiveX controls

What is the Centrifuge Loader ActiveX control

The Centrifuge Loader ActiveX control is the software component that allows third-party lab automation systems to interact with the Centrifuge Loader.

How the Centrifuge Loader ActiveX control is used

In an Agilent Technologies automation system that is running the VWorks software, ActiveX interfaces are not used to communicate with devices. However, some integrations, such as those with LIMS, require that a third-party application control the Centrifuge Loader. The Centrifuge Loader ActiveX control enables third-party applications to interface with the Centrifuge Loader.

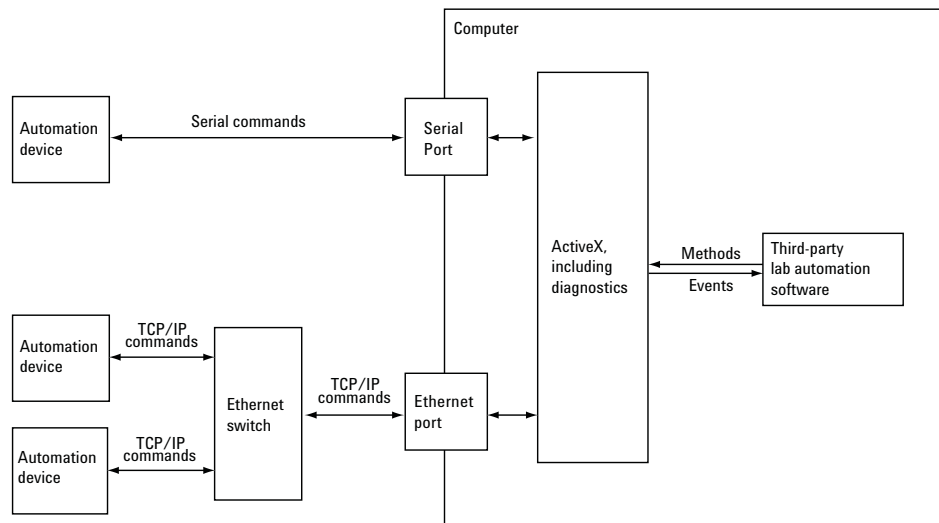
Each ActiveX control consists of a collection of the following:

- *Methods*. Functions that can be called to invoke individual operations
- *Properties*. Attributes or features of the ActiveX control
- *Events*. Notifications that methods have completed or resulted in errors

To ensure proper integration, you must know the available methods and properties for the ActiveX control.

The following diagram illustrates the use of the Centrifuge Loader ActiveX control in a lab automation system environment. Actions you perform are conducted through ActiveX methods. System responses are relayed back through ActiveX events or through return values and variables passed to methods.

Note: Although the Centrifuge Loader ActiveX control generates events, the third-party application must implement handlers for them.



Properties

Blocking

VARIANT_BOOL Blocking

Description

Determines whether methods should block until completion or return immediately for asynchronous operation.

Acceptable values

- VARIANT_TRUE (C++) or True (Visual Basic .NET). The ActiveX is forced to block or wait until a method completes before it returns control to the caller.
- VARIANT_FALSE (C++) or False (Visual Basic .NET). Returns control to the application immediately, and the caller should handle the events accordingly.

Default value

VARIANT_FALSE or False

Blocking affects some methods differently. See each method's description for the effect. Unless otherwise noted:

- In non-blocking mode (Block = False), a method:
 - Starts another thread of execution to perform the given method, returning control to the application immediately.
 - Returns 0 on launching new thread successfully; otherwise returns nonzero.
 - If the method is successful, an event indicating completion is fired; if unsuccessful, an Error event is initiated.
- In blocking mode (Block = True), a method:
 - Is executed.
 - Returns 0 if it completes successfully; returns nonzero otherwise.
- Error message can be reviewed by calling GetLastError().

Visual C++ example

```
//set the Centrifuge Loader in blocking mode
VARIANT_BOOL blocking=VARIANT_TRUE;
m_CentrifugeLoader.PutBlocking(blocking);
//set the Centrifuge Loader in non-blocking mode
blocking=VARIANT_FALSE;
m_CentrifugeLoader.PutBlocking(blocking);
//returns the blocking value
blocking=m_CentrifugeLoader.GetBlocking();
//user should handle events if non-blocking mode is
selected!
```

Visual Basic .NET example

```
'set Centrifuge Loader in blocking mode
CentrifugeLoader1.Blocking=True
'set Centrifuge Loader in non-blocking mode
CentrifugeLoader1.Blocking=False
'returns the blocking value
Dim bMode as Boolean
bMode= CentrifugeLoader1.Blocking
'user should handle events if non-blocking mode is
selected!
```

ControlPicture

IPictureDisp*ControlPicture

Description

A read-only picture of the Centrifuge Loader that can be used in the container's application.

Parameters

None

Visual C++ example

```

/*the CPicture class will be imported in to your project
When the ActiveX is installed*/
CButton button;
//create button
button.Create("Button", WS_CHILD | WS_VISIBLE | BS_BITMAP,
CRect(10, 10, 60, 60), pParentWnd/*pointer of parent
window*/, 1);
CPicture CentrifugeLoaderPic;
//retrieve the picture
CentrifugeLoaderPic=m_CentrifugeLoader.GetControlPicture(
);
//paint the bitmap on to the button
button.SetBitmap((HBITMAP)
CentrifugeLoaderPic.GetHandle());

```

Visual Basic .NET example

```

Dim iPicture As System.Drawing.Image=
CentrifugeLoader1.ControlPicture()
button.BackgroundImage=iPicture

```

Related topics

For information about...	See...
Centrifuge Loader ActiveX methods	“Methods” on page 6
Centrifuge Loader ActiveX events	“Events” on page 21
Overview of ActiveX controls	“About ActiveX controls” on page 2
Reporting problems	“Reporting problems” on page iv

Methods

Abort

```
LONG Abort( )
```

Description

Aborts a current task that is in the error state and clears the error.

Parameters

None

Returns

0 if successful; Other value if there was an error.

Visual C++ example

```
LONG lres=m_CentrifugeLoader.Abort();
```

Visual Basic .NET example

```
Dim ires as Integer  
ires=CentrifugeLoader1.Abort()
```

AboutBox

```
void AboutBox( )
```

Description

Displays the Centrifuge Loader About dialog box that contains the ActiveX, firmware, and hardware version numbers.

Parameters

None

Returns

None

Visual C++ example

```
m_CentrifugeLoader.AboutBox();
```

Visual Basic .NET example

```
CentrifugeLoader1.AboutBox()
```

Close

```
LONG Close( )
```

Description

Disconnects from the Centrifuge Loader. After successful closing, the CloseComplete event is sent.

Parameters

None

Returns

0 if successful; Other value if there was an error

Visual C++ example

```
LONG lres=m_CentrifugeLoader1.Close();
```

Visual Basic .NET example

```
Dim ires as Integer  
ires=CentrifugeLoader1.Close()
```

CloseDoor

```
LONG CloseDoor( )
```

Description

Closes the Centrifuge Loader's door. After successful closing of the door, the CloseDoorComplete event is sent.

Parameters

None

Returns

0 if successful; Other value if there was an error

Visual C++ example

```
LONG lres=m_CentrifugeLoader.CloseDoor();
```

Visual Basic .NET example

```
Dim ires as Integer  
ires=CentrifugeLoader1.CloseDoor()
```

EnumerateProfiles

```
VARIANT EnumerateProfiles( )
```

Description

Retrieves a list of defined profiles. The strings in this array are the profile names that should be used for the Initialize method.

Parameters

None

Returns

An array of profile names

Visual C++ example

```
VARIANT  
vProfiles=m_CentrifugeLoader1.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);  
VariantClear(&vProfiles);
```

Visual Basic .NET example

```
Dim i as Integer  
Dim sProfiles() As String  
sProfiles=CentrifugeLoader1.EnumerateProfiles()  
For i=0 To sProfiles.GetLength(0)-1  
    MsgBox sProfiles(i)  
Next
```

GetActiveXVersion

```
BSTR GetActiveXVersion( )
```

Description

Retrieves the Centrifuge Loader's ActiveX version number.

Parameters

None

Returns

ActiveX version number (string)

Visual C++ example

```
CString ActiveXVer=  
m_CentrifugeLoader1.GetActiveXVersion();
```

Visual Basic .NET example

```
Dim sVersion As String  
sVersion=CentrifugeLoader1.GetActiveXVersion()
```

GetCentrifugeActiveXVersion

```
BSTR GetCentrifugeActiveXVersion( )
```

Description

Retrieves the Microplate Centrifuge ActiveX version number

Parameters

None

Returns

Centrifuge ActiveX version number (string)

Visual C++ example

```
CString CentrifugeActiveXVer=  
m_CentrifugeLoader1.GetCentrifugeActiveXVersion();
```

Visual Basic .NET example

```
Dim sCentrifugeActiveXVer As String  
sCentrifugeActiveXVer=CentrifugeLoader1.GetCentrifugeActi  
veXVersion()
```

GetCentrifugeHardwareVersion

```
BSTR GetCentrifugeHardwareVersion( )
```

Description

Retrieves the Microplate Centrifuge hardware version number. The Microplate Centrifuge hardware version is only available after profile is initialized.

Parameters

None

Returns

Centrifuge hardware version number (string)

Visual C++ example

```
CString CentrifugeHardwareVer=  
m_Centrifuge1Loader.GetCentrifugeHardwareVersion();
```

Visual Basic .NET example

```
Dim sCentrifugeHardwareVersion As String  
sCentrifugeHardwareVersion=Centrifuge1Loader.GetCentrifug  
eHardwareVersion()
```

GetFirmwareVersion

```
BSTR GetFirmwareVersion( )
```

Description

Retrieves the Centrifuge Loader firmware version number. The firmware version is only available after profile is initialized.

Parameters

None

Returns

Firmware version number (string)

Visual C++ example

```
CString FirmwareVer=  
m_CentrifugeLoader1.GetFirmwareVersion();
```

Visual Basic .NET example

```
Dim sFirmwareVersion As String  
sFirmwareVersion= CentrifugeLoader1.GetFirmwareVersion ( )
```

GetHardwareVersion

```
BSTR GetHardwareVersion( )
```

Description

Retrieves the Centrifuge Loader's hardware version number. The hardware version is only available after profile is initialized.

Parameters

None

Returns

Hardware version number (string)

Visual C++ example

```
CString HardwareVer=
m_CentrifugeLoader1.GetHardwareXVersion();
```

Visual Basic .NET example

```
Dim sHardwareVersion As String
sHardwareVersion= CentrifugeLoader1.GetHardwareVersion()
```

GetLastError

```
BSTR GetLastError( )
```

Description

Retrieves the last known error condition.

Parameters

None

Returns

An error string.

Visual C++ example

```
CString str=m_CentrifugeLoader1.GetLastError();
```

Visual Basic .NET example

```
Dim sError As String=""
sError=CentrifugeLoader1.GetLastError()
```

Home

```
LONG Home( )
```

Description

Homes the Centrifuge Loader. Although Initialize() homes the Centrifuge Loader, this method can be used to rehome the Centrifuge Loader when necessary. After Home operation is successful, the HomeComplete event is sent.

Parameters

None

Return

0 if successful; Other value if there was an error

Visual C++ example

```
LONG lres=m_CentrifugeLoader.Home();
```

Visual Basic .NET example

```
Dim ires as Integer
ires=CentrifugeLoader1.Home()
```

Ignore

```
LONG Ignore( )
```

Description

Ignores the previously issued error and moves to the next step in the task. This is not a recommended course of action, as the errors are issued for a reason. However, ignoring some errors can be appropriate if the operator understands the implications.

Parameters

None

Returns

0 if successful; Other value if there was an error

Visual C++ example

```
LONG lres=m_CentrifugeLoader1.Ignore();
```

Visual Basic .NET example

```
Dim ires as Integer  
ires=CentrifugeLoader1.Ignore()
```

Initialize

```
LONG Initialize(BSTR Profile)
```

Description

Initializes the profile and starts communication with the Centrifuge Loader using the parameters set in the profile. The profile specifies the serial or Ethernet connection used to communicate with the Centrifuge Loader. The parameters for each profile can be adjusted in the Diagnostics dialog box (by a call to the ShowDiagsDialog method) on the Profiles page.

Parameters

Name	Type	Description
Profile	BSTR	The name of the profile to be used for initialization.

Returns

0 if successful, and initiates the InitializeComplete event; Other value if there was an error

Visual C++ example

```
//connect via serial connection specified in the profile
LONG lres=
m_CentrifugeLoader.Initialize(_bstr_t("CentrifugeLoader
Serial profile"));
```

Visual Basic .NET example

```
'connect via serial connection specified in the profile
Dim ires as Integer
ires=CentrifugeLoader1.Initialize("CentrifugeLoader
Serial profile")
```

LoadPlate

```
LONG LoadPlate(LONG bucket_num, DOUBLE gripper_offset,
DOUBLE plate_height, LONG speed, LONG options);
```

Description

Loads a plate into the Microplate Centrifuge bucket specified by bucket_number. The path that the gripper head takes is calculated and optimized with the plate_height. The plate is picked up at a distance above its base by the gripper_offset value in millimeters.

The Centrifuge Loader moves at the speed specified by speed and uses the options specified in options.

Parameters

Name	Type	Range	Description
bucket_number	LONG	1-2	The bucket number of the Microplate Centrifuge bucket to load with a plate.
gripper_offset	DOUBLE	2.6-21.1	The distance from the bottom of the plate (in mm) where the center of the gripper should grab the plate. The default gripper offset is 8 mm.
plate_height	DOUBLE	2.6-48.3	The plate height. The default plate height is 15 mm.
speed	LONG	0-2	The speed at which the Centrifuge Loader should move. 0 = slow, 1 = medium, 2 = fast

Name	Type	Range	Description
options	LONG	0-7	<p>Bitmask of the following form:</p> <p>0x00000000 = No options set</p> <p>0x00000001 = Ignore plate sensor</p> <p>0x00000002 = Grip plates gently</p> <p>0x00000003 = Ignore plate sensor and grip plates gently</p> <p>0x00000004 = Assume maximum plate height</p> <p>0x00000005 = Assume maximum plate height and Ignore plate sensor</p> <p>0x00000006 = Assume maximum plate height and Grip plates gently</p> <p>0x00000007 = Use all options</p> <p>The options that should be applied to the LoadPlate() task. If the assume maximum plate height option is used, the plate_height parameter is ignored.</p>

Returns

0 if successful; Other value if there was an error

Visual C++ example

```
LONG lres=m_CentrifugeLoader.LoadPlate(1,8.0,15.0,2,1);
```

Visual Basic .NET example

```
Dim ires as Integer
ires=CentrifugeLoader1.LoadPlate(1,8.0,15.0,2,1);
```

OpenDoor

```
LONG OpenDoor(SHORT bucket_num)
```

Description

Opens the door to the bucket_num. After successful opening of the door, the OpenDoorComplete event is sent.

Parameters

Name	Type	Range	Description
bucket_num	SHORT	1-2	Presented bucket

Returns

0 if successful; Other value if there was an error

Visual C++ example

```
LONG lres=m_CentrifugeLoader.OpenDoor(1);
```

Visual Basic .NET example

```
Dim ires as Integer
ires=CentrifugeLoader1.OpenDoor(1)
```

Park

```
LONG Park( )
```

Description

Parks the Centrifuge Loader. Parking moves the gripper head under and behind the Centrifuge Loader plate stage, out of the way of the primary plate handling robot. After Park operation is successful, the ParkComplete event is sent.

Parameters

None

Returns

0 if successful; Other value if there was an error

Visual C++ example

```
LONG lres=m_CentrifugeLoader.Park();
```

Visual Basic .NET example

```
Dim ires as Integer
ires=CentrifugeLoader1.Park()
```

Retry

```
LONG Retry( )
```

Description

Retries the last action after an error occurred.

Parameters

None

Returns

0 if successful; Other value if there was an error

Visual C++ example

```
LONG lres=m_CentrifugeLoader1.Retry();
```

Visual Basic .NET example

```
Dim ires as Integer
ires=CentrifugeLoader1.Retry()
```

ShowDiagsDialog

```
void ShowDiagsDialog(VARIANT_BOOL modal, SHORT securityLevel)
```

Description

Displays the Diagnostics dialog box that allows the operator to troubleshoot and correct problems. This method can be called before the Initialize method to create a profile. Contents displayed are based on the operator's access level.

Parameters

Name	Type	Range	Description
modal	VARIANT_BOOL	VARIANT_TRUE/ VARIANT_FALSE	VARIANT_TRUE = display dialog box as modal (does not permit users to access the parent window) VARIANT_FALSE = display dialog box as modeless (permits users to access the parent window)
securityLevel	SHORT	-1, 0, 1, 2, 3,	The security level the operator has in the dialog box: 0 = Administrator 1 = Technician 2 = Operator 3 = Guest -1 = No access

Returns

None

Visual C++ example

```
m_CentrifugeLoader.ShowDiagsDialog(VARIANT_TRUE,0);
```

Visual Basic .NET example

```
CentrifugeLoader1.ShowDiagsDialog(True,0)
```

SpinCycle

```
LONG SpinCycle(DOUBLE vel_percent, DOUBLE accel_percent,
DOUBLE decel_percent, LONG timer_mode, LONG time, LONG
bucket_num_load, LONG bucket_num_unload, DOUBLE
gripper_offset_load, DOUBLE gripper_offset_unload, DOUBLE
plate_height_load, DOUBLE plate_height_unload, LONG
speed_load, LONG speed_unload, LONG load_options, LONG
unload_options);
```

Description

Runs a spin cycle.

The cycle can either:

- Spin the Microplate Centrifuge without using the Centrifuge Loader.
- Perform a spin using the Centrifuge Loader. The Centrifuge Loader loads a plate, hovers until the spin completes, then unloads a plate.

To spin the Microplate Centrifuge without the Centrifuge Loader:

1 Specify 0 for bucket_num_load.

- 2 Specify the bucket to present (1 or 2) after the spin is complete in bucket_num_unload. In this case, the last eight arguments are ignored.

To perform a spin with the Centrifuge Loader:

- 1 Specify bucket to load (1 or 2) with the bucket_num_load parameter.
- 2 Specify bucket to unload (1 or 2) with the bucket_num_unload parameter.

Microplate Centrifuge parameters

The first five parameters determine how the Microplate Centrifuge should spin.

Name	Type	Range	Description
vel_percent	DOUBLE	1-100	Percentage of maximum velocity at which to spin the Microplate Centrifuge.
accel_percent	DOUBLE	1-100	Percentage of maximum acceleration at which to accelerate the Microplate Centrifuge.
decel_percent	DOUBLE	1-100	Percentage of maximum deceleration at which to decelerate the Microplate Centrifuge.
timer_mode	LONG	0-1	The timer mode to use. 0 = total time, 1 = time at speed
time	LONG	1-86400	Number of seconds to spin.
bucket_num_load	LONG	0-2	The bucket number of the Microplate Centrifuge bucket to load with a plate. 0 = do not load/unload, just spin the Microplate Centrifuge.
bucket_num_unload	LONG	0-2	The bucket number of the Microplate Centrifuge bucket from which to unload a plate. If bucket_num_load was 0, the bucket_num_unload specifies which Microplate Centrifuge bucket to present after the spin.

Centrifuge Loader parameters

All eight parameters determine how the Centrifuge Loader handles plates.

Name	Type	Range	Description
gripper_offset_load	DOUBLE	2.6-21.1	The distance in mm, from the bottom of the plate where the center of the gripper should hold the plate during the loading process. The default gripper offset is 8 mm.
gripper_offset_unload	DOUBLE	2.6-21.1	The distance in mm, from the bottom of the plate where the center of the gripper should hold the plate during the unloading process. The default gripper offset is 8 mm.
plate_height_load	DOUBLE	2.6-48.3	The height of the plate to load. The default plate height is 15 mm.

Name	Type	Range	Description
plate_height_unload	DOUBLE	2.6-48.3	The height of the plate to unload. The default plate height is 15 mm.
speed_load	LONG	0-2	The speed at which the Centrifuge Loader should move during the loading process. 0 = slow, 1 = medium, 2 = fast
speed_unload	LONG	0-2	The speed at which the Centrifuge Loader should move during the unloading process. 0 = slow, 1 = medium, 2 = fast
options_load	LONG	0-7	Bitmask of the following form: 0x00000000 = No options set 0x00000001 = Ignore plate sensor 0x00000002 = Grip plates gently 0x00000003 = Ignore plate sensor and grip plates gently 0x00000004 = Assume maximum plate height 0x00000005 = Assume maximum plate height and Ignore plate sensor 0x00000006 = Assume maximum plate height and Grip plates gently 0x00000007 = Use all options The options that should be applied during the loading process. If the assume maximum plate height option is used, the plate_height_load parameter is ignored.
options_unload	LONG	0-7	Bitmask of the following form: 0x00000000 = No options set 0x00000001 = Ignore plate sensor 0x00000002 = Grip plates gently 0x00000003 = Ignore plate sensor and grip plates gently 0x00000004 = Assume maximum plate height 0x00000005 = Assume maximum plate height and Ignore plate sensor 0x00000006 = Assume maximum plate height and Grip plates gently 0x00000007 = Use all options The options that should be applied during the unloading process. If the assume maximum plate height option is used, the plate_height_unload parameter is ignored.

Returns

0 if successful; Other value if there was an error

Visual C++ example

```
LONG lres=
m_CentrifugeLoader1.SpinCycle(55.0,60.0,90.0,1,10,1,1,8.0
,8.0,15.0,15.0,2,2, 1,1);
```

Visual Basic .NET example

```
Dim ires as Integer
ires=CentrifugeLoader1.SpinCycle(55.0,60.0,90.0,1,10,1,1,
8.0,8.0,15.0,15.0,2,2,1,1);
```

StopSpinCycle

```
LONG StopSpinCycle(SHORT bucket_num);
```

Description

Stops currently running spin cycle and opens the Microplate Centrifuge door to the bucket_num. After the spin cycle has been successfully stopped and the door is opened, the StopSpinCycleComplete event is sent.

Parameters

Name	Type	Range	Description
bucket_num	SHORT	1-2	Presented bucket

Returns

0 if successful; Other value if there was an error

Visual C++ example

```
LONG lres=m_CentrifugeLoader.StopSpinCycle(1);
```

Visual Basic .NET example

```
Dim ires as Integer
ires=CentrifugeLoader1.StopSpinCycle (1)
```

UnloadPlate

```
LONG UnloadPlate(LONG bucket_num, DOUBLE gripper_offset,
DOUBLE plate_height, LONG speed, LONG options);
```

Description

Unloads a plate into the Microplate Centrifuge bucket specified by bucket_number. The path that the gripper head takes is calculated and optimized with the plate_height. The plate is picked up at a distance above its base by the gripper_offset value in millimeters.

The Centrifuge Loader moves at the speed specified by speed and uses the options specified in options.

Parameters

Name	Type	Range	Description
bucket_number	LONG	1-2	The bucket number of the Microplate Centrifuge bucket from which to unload a plate.

Name	Type	Range	Description
gripper_offset	DOUBLE	2.6-21.1	The distance from the bottom of the plate (in mm) where the center of the gripper should grab the plate. The default gripper offset is 8 mm.
plate_height	DOUBLE	2.6-48.3	The plate height. The default plate height is 15 mm.
speed	LONG	0-2	The speed at which the Centrifuge Loader should move. 0 = slow, 1 = medium, 2 = fast
options	LONG	0-7	Bitmask of the following form: 0x00000000 = No options set 0x00000001 = Ignore plate sensor 0x00000002 = Grip plates gently 0x00000003 = Ignore plate sensor and grip plates gently 0x00000004 = Assume maximum plate height 0x00000005 = Assume maximum plate height and Ignore plate sensor 0x00000006 = Assume maximum plate height and Grip plates gently 0x00000007 = Use all options The options that should be applied to the UnloadPlate() task. If the assume maximum plate height option is used, the plate_height parameter is ignored.

Returns

0 if successful; Other value if there was an error

Visual C++ example

```
LONG lres=m_CentrifugeLoader.UnloadPlate(1,8.0,15.0,2,1);
```

Visual Basic .NET example

```
Dim ires as Integer
ires=CentrifugeLoader1.UnloadPlate(1,8.0,15.0,2,1);
```

Related topics

Centrifuge Loader ActiveX properties	“Properties” on page 3
Centrifuge Loader ActiveX events	“Events” on page 21
Overview of ActiveX controls	“About ActiveX controls” on page 2
Reporting problems	“Reporting problems” on page iv

Events

Error

```
void Error(SHORT Number, BSTR* Description, LONG Scode, BSTR
Source, BSTR HelpFile, LONG HelpContext, VARIANT_BOOL*
CancelDisplay)
```

Description

This event is sent when an error occurs during any non-blocking method execution.

Parameters

Name	Type	Range	Description
Description	BSTR*		The description of the error.
CancelDisplay	VARIANT_BOOL*	VARIANT_TRUE/ VARIANT_FALSE	The option to hide the error message. Use VARIANT_TRUE for C++ Use True for Visual Basic .NET

Note: SHORT Number, LONG Scode, BSTR Source, BSTR HelpFile, and LONG HelpContext are not used.

Returns

None

InitializeComplete

```
void InitializeComplete( )
```

Description

This event occurs when the Initialize method is successful.

Parameters

None

Returns

None

CloseComplete

```
void CloseComplete()
```

Description

This event occurs when the Close method is successful.

Parameters

None

Returns

None

CloseDoorComplete

```
void CloseDoorComplete()
```

Description

This event occurs when the CloseDoor method is successful.

Parameters

None

Returns

None

ControllerMessage

```
void ControllerMessage(LONGLONG timestamp_ft, LONG  
timestamp_us, LONG message_type, BSTR message, LONG  
info_type, LONG info_value1, LONG info_value2, LONG  
info_value3);
```

Description

ControllerMessage is a conduit through which the ActiveX posts messages. This advanced feature should not be used for most applications. If you need further assistance, please contact Automation Solutions Technical Support.

Parameters

Name	Type	Description
message_time_coarse	LONGLONG	The message generation time (in seconds) as a FILETIME.
message_time_fine	LONG	Additional message generation time information (milliseconds after message_time_coarse).
message_type	LONG	The message type.
message_text	BSTR	The message text.
info_type	LONG	The information type.
info_value_1	LONG	The information value (datum 1).
info_value_2	LONG	The information value (datum 2).
info_value_3	LONG	The information value (datum 3).

Returns

None

HomeComplete

```
void HomeComplete()
```

Description

This event occurs when the Home method is successful.

Parameters

None

Returns

None

LoadPlateComplete

```
void LoadPlateComplete(SHORT bucket_num)
```

Description

This event occurs when the LoadPlate method is successful.

Parameters

Name	Type	Range	Description
bucket_num	SHORT	1-2	Loaded bucket

Returns

None

OpenDoorComplete

```
void OpenDoorComplete(SHORT bucket_num)
```

Description

This event occurs when the OpenDoor method is successful.

Parameters

Name	Type	Range	Description
bucket_num	SHORT	1-2	Presented bucket

Returns

None

ParkComplete

```
void ParkComplete()
```

Description

This event occurs when the Park method is successful.

Parameters

None

Returns

None

SpinCycleComplete

```
void SpinCycleComplete(SHORT bucket_num)
```

Description

This event occurs when the SpinCycle method is successful.

Parameters

Name	Type	Range	Description
bucket_num	SHORT	1-2	Presented bucket

Returns

None

StopSpinCycleComplete

```
void StopSpinCycleComplete(SHORT bucket_num)
```

Description

This event occurs when the StopSpinCycle method is successful.

Parameters

Name	Type	Range	Description
bucket_num	SHORT	1-2	Presented bucket

Returns

None

UnloadPlateComplete

```
void UnloadPlateComplete(SHORT bucket_num)
```

Description

This event occurs when the UnloadPlate method is successful.

Parameters

Name	Type	Range	Description
bucket_num	SHORT	1-2	Unloaded bucket

Returns

None

Related topics

Centrifuge Loader ActiveX methods	“Methods” on page 6
Centrifuge Loader ActiveX properties	“Properties” on page 3
Overview of ActiveX controls	“About ActiveX controls” on page 2
Reporting problems	“Reporting problems” on page iv



User Guide

G5405-90004

Revision 00, April 2012