

TekExpress®
M-PHY Receiver Automated Test Solution
Printable Online Help



TekExpress®
M-PHY Receiver Automated Test Solution
Printable Online Help

Copyright © Tektronix. All rights reserved. Licensed software products are owned by Tektronix or its subsidiaries or suppliers, and are protected by national copyright laws and international treaty provisions.

Tektronix products are covered by U.S. and foreign patents, issued and pending. Information in this publication supersedes that in all previously published material. Specifications and price change privileges reserved.

TEKTRONIX and TEK are registered trademarks of Tektronix, Inc.

TekExpress is a registered trademark of Tektronix, Inc.

Contacting Tektronix

Tektronix, Inc.
14150 SW Karl Braun Drive
P.O. Box 500
Beaverton, OR 97077
USA

For product information, sales, service, and technical support:

- In North America, call 1-800-833-9200.
- Worldwide, visit www.tektronix.com to find contacts in your area.

Table of Contents

Getting Help and Support

Related Documentation	1
Conventions Used in Help	1
Technical Support	2

Getting Started

Installing the Software	
Compatibility	3
Minimum System Requirements	4
Install the Software	6
Activate the License	7
View Version and License Information	8
Introduction to the Application	
M-PHY Rx Application Overview	9
Application Directories and Usage	10
File Name Extensions	11

Operating Basics

Start the Application	13
Exit the Application	13
Application Controls and Menus	
Application Controls	14
Options Menu	15
Email Settings	16
Instrument Control Settings	17
Panels	
About Panels	18
Setup Panel	18
Status Panel	23
Results Panel	25
Reports Panel	26

Saving and Recalling Test Setups

About Test Setups	27
Save a Test Setup	27
Recall a Saved Test Setup	28
Create a New Test Setup Based on an Existing One	28

Setting Up and Configuring Tests

About Setting Up Tests	29
Equipment Connection Setup	30
View Connected Instruments	31
Test Setup Overview	33
Select Device Parameters	33
Select a Test	35
Configuring Tests	
About Configuring Tests	36
Configure Tests	36
Common Test Parameters	37
Configure Email Notification	37
Select Test Notification Preferences	39
Select Report Options	40

Running Tests and Viewing Results

About Running Tests	43
Before You Click Start	43
Pre-Run Checklist	45
View the Progress of Analysis	46
View Test Results	47
View a Report	47
View Test-Related Files	53

TekExpress Programmatic Interface

About the Programmatic Interface	55
Requirements for Developing TekExpress Client	56
Remote Proxy Object	57
Client Proxy Object	57
Client Programmatic Interface Example	59
Program Example	62
M-PHY Rx Application Commands	
About M-PHY Rx Application Commands	65
Connect Through an IP Address	70
Lock the Server	71
Disable the Popups	72
Set or Get the DUT ID	73
Set the Configuration Parameters for a Suite or Measurement	74
Query the Configuration Parameters for a Suite or Measurement	76
Select a Measurement	77
Select a Suite	78
Select a Channel	79

Configure the Selected Measurement.....	80
Run with Set Configurations or Stop the Run Operation.....	81
Handle Error Codes.....	82
Get or Set the Timeout Value.....	83
Wait for the Test to Complete.....	84
After the Test is Complete.....	86
Save, Recall, or Check if a Session is Saved.....	89
Unlock the Server.....	90
Disconnect from the Server.....	90

HS Tests

Testing Procedures and Examples

Receiver Testing Measurement Procedure.....	91
High-Speed Mode Measurements.....	92
Loopback Validation.....	93
About the Oscilloscope Error Detector.....	94
2.1.1 Differential Input Voltage Amplitude Tolerance (VDIF-RX) Test.....	94
2.1.2 Receiver Eye Opening and Accumulated Diff Input Voltage Test.....	96
2.1.3 Common-Mode Input Voltage Tolerance (VCM-RX) Test.....	98
2.1.7 Receiver Jitter Tolerance (TJRX, SJRX, RJRX, STTJRX, STSJRX) Test.....	99

Reference

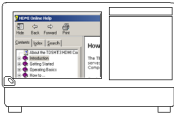
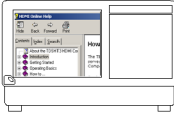

Measurement Parameter Descriptions.....	103
Map the My TekExpress Folder.....	104
Use an AWG7102 in LAN Configuration.....	104
Enable the Oscilloscope Error Detector.....	106

Index

Related Documentation

The following manuals are available as part of the TekExpress M-PHY Rx Automated Solution documentation set.

Table 1: Product documentation

Item	Purpose	Location
Online Help	In-depth operation and UI help	
PDF of the Online Help	In-depth operation and UI help	 +  www.Tektronix.com

See Also

- [Technical Support \(see page 2\)](#)

Conventions Used in Help

Online Help uses the following conventions:

- The term “DUT” is an abbreviation for Device Under Test.
- The term “select” is a generic term that applies to the two mechanical methods of choosing an option: using a mouse or using the touch screen.

Technical Support

Tektronix values your feedback on our products. To help us serve you better, please send us your suggestions, ideas, or comments on your application or oscilloscope. Contact Tektronix through mail, telephone, or the Web site. See [Contacting Tektronix](#) for more information.

When you contact Tektronix Technical Support, please include the following information (be as specific as possible):

General Information

- All instrument model numbers
- Hardware options, if any
- Probes used
- Your name, company, mailing address, phone number, FAX number
- Please indicate if you would like to be contacted by Tektronix about your suggestion or comments.

Application Specific Information

- Software version number
- Description of the problem such that technical support can duplicate the problem
- If possible, save the setup files for all the instruments used and the application.
- If possible, save the TekExpress setup files, log.xml, *.TekX (session files and folders), and status messages text file.

Compatibility

The TekExpress M-PHY Rx application runs on the following Tektronix oscilloscopes:

- DPO/DSA/MSO70604C M-PHYRX (only HS Gear 1)
- DPO/DSA70604B M-PHYRX (only HS Gear 1)
- DPO/DSA/MSO70804C M-PHYRX (only HS Gear 1 and 2)
- DPO/DSA70804B M-PHYRX (only HS Gear 1 and 2)
- DPO/DSA/MSO71254C M-PHYRX
- DPO/DSA71254B M-PHYRX
- DPO/DSA/MSO71604C M-PHYRX
- DPO/DSA71604B M-PHYRX
- DPO/DSA/MSO72004C M-PHYRX
- DPO/DSA72004B M-PHYRX
- DPO/DSA72504D M-PHYRX
- DPO/DSA73304D M-PHYRX
- MSO70604 M-PHYRX (only HS Gear 1)
- MSO70804 M-PHYRX (only HS Gear 1 and 2)
- MSO71254 M-PHYRX
- MSO71604 M-PHYRX
- MSO72004 M-PHYRX

The following are options available for ordering:

- DPOFL-M-PHYRX
- DPOFT-M-PHYRX
- DPO-UP M-PHYRX
- DPO/DSA/MSO70804C M-PHYRX (only HS Gear 1 and 2)
- DPO/DSA/MSO70604C M-PHYRX (only HS Gear 1)
- DPO/DSA/MSO71254C M-PHYRX
- DPO/DSA/MSO71604C M-PHYRX
- DPO/DSA/MSO72004C M-PHYRX

- DPO/DSA72504D M-PHYRX
- DPO/DSA73304D M-PHYRX

The TekExpress M-PHY Rx application can be used with the following probes:

- Tektronix P7380SMA or higher bandwidth
- P73xx or P75xx Differential probes (above 6 GHz bandwidth)

See Also

- [Minimum System Requirements \(see page 4\)](#)

Minimum System Requirements

The following table shows the minimum system requirements for an oscilloscope to run TekExpress.

Table 2: System requirements

Oscilloscope	<ul style="list-style-type: none"> ■ DPO/DSA/70K B or C Series, MSO 70K, MSO 70K C series ■ Oscilloscope with GPIB set as the controller. ■ Error Detector (ERRDT) and Serial Trigger Unit (STU)
Arbitrary Waveform Generator (AWG)	<ul style="list-style-type: none"> ■ AWG7082/AWG7102 or above, for HS-GEAR1 ¹ ■ AWG7122C for up to HS-GEAR2 ■ AWG7122C with Interleave (option 06), for up to HS-GEAR3
Processor	Same as the oscilloscope
Operating System	Same as the oscilloscope: <ul style="list-style-type: none"> ■ Windows XP (32-bit) SP2 and higher ■ Windows 7 64-bit and 32-bit
Memory	Same as the oscilloscope
Hard Disk	Same as the oscilloscope.
Display	Same as the oscilloscope ²
Firmware	TekScope v6.1.1.32 or later

Table 2: System requirements (cont.)

Software	<ul style="list-style-type: none"> ■ National Instruments LabVIEW Runtime 9.0.1 or later ■ National Instruments TestStand engine 4.2.1 ■ Microsoft .NET 4.0 Framework ■ Microsoft Excel 2002 or above ■ Microsoft Internet Explorer 6.0 SP1 or later ■ Adobe Reader 7.0 or equivalent software for viewing portable document format (PDF) files
Probes	Tektronix P7380SMA or higher bandwidth; P73xx or P75xx Differential probes (above 6 GHz bandwidth)
Other Devices	<ul style="list-style-type: none"> ■ Matched pair of SMA cables, two-set minimum for single lane ■ Microsoft compatible mouse or compatible pointing device ■ PCI-GPIB or equivalent interface for instrument connectivity³ ■ GPIB cable/LAN cable for instrument connectivity

¹ M-PHYRX supports non-interleave channels.

² If TekExpress is running on an instrument having a video resolution lower than 800x600 (for example, sampling oscilloscope), it is recommended that you connect a secondary monitor, which must be enabled before launching the application.

³ If TekExpress is installed on a Tektronix oscilloscope, the virtual GPIB port will be used by TekExpress for communicating with oscilloscope applications. If external GPIB communication devices such as USB-GPIB-HS or equivalent are used for instrument connectivity, make sure that the Talker Listener utility is enabled in the DPO/DSA/MSO oscilloscope GPIB menu. For ease of use, connect to an external (secondary) monitor.

See Also

■ [Compatibility \(see page 3\)](#)

■ [Equipment Connection Setup \(see page 30\)](#)

Install the Software

The TekExpress M-PHY Rx web installer page provides links to two software packages, one to be installed on the oscilloscope and one to be installed on the AWG used for testing. For a list of compatible instruments, see [Compatibility \(see page 3\)](#).

- TekExpressM-PHYRXWebInstaller.exe: Installs the TekExpress M-PHY Rx application
- TekExpressM-PHYRxAWGInstaller.exe: Installs the required test patterns

To download and install the files:

1. Close the TekScope application.
2. Go to the www.tek.com Web site and search for M-PHY Rx to locate the installation file. Download the file `TekExpressM-PHYRXWebInstaller.exe`.
3. Double-click the executable file to extract the installation files.

After extraction, the installer launches and the software automatically installs in the following location:

- Windows XP location: `C:\Program Files\Tektronix\TekExpress\TekExpress M-PHY RX`
 - Windows 7 location: `C:\Program Files (x86)\Tektronix\TekExpress\TekExpress M-PHY RX`
4. To open the application, open the TekScope application and then select **TekExpress M-PHY Rx** from the Analyze menu.
 5. Download the file `TekExpressM-PHYRxAWGInstaller.exe` to the AWG.
 6. Double-click the executable file to extract the installation files. After extraction, the installer launches and the test patterns are automatically installed in the following location:
`C:\Tektronix\TekExpress\M-PHY RX`.

See Also

- [Minimum System Requirements \(see page 4\)](#)
- [Compatibility \(see page 3\)](#)

Activate the License

Activate the license using the Option Installation wizard on the oscilloscope. The oscilloscope Online Help has instructions for using the Options Installation window to activate licenses for installed applications. Follow these steps to activate the TekExpress M-PHY Rx license:

1. From the oscilloscope menu, click **Utilities > Option Installation**.

The TekScope Option Installation wizard opens.

2. Press the **F1** key on the oscilloscope keyboard to open the Option Installation help topic. Follow the directions in the topic to activate the license.

See Also

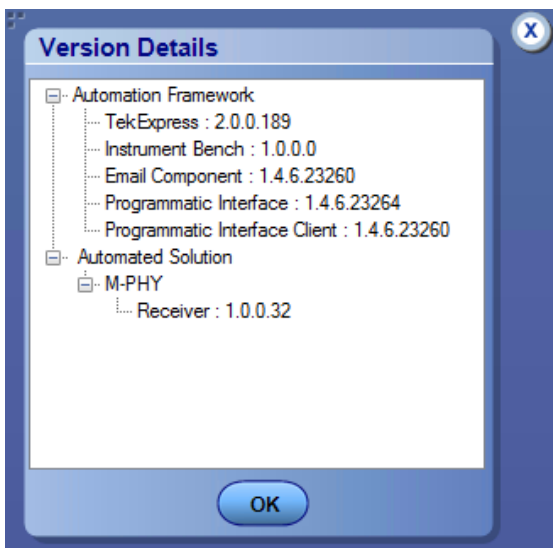
- [View Version and License Information \(see page 8\)](#)

View Version and License Information

Use the following instructions to view version information for the application and for the application modules such as the Programmatic Interface and the Programmatic Interface Client.

To view version information:

1. From the Options menu, select **About TekExpress**.
2. Click the **View Version Details** link to check the version numbers of the installed test suites. Close the dialog box when finished.



To view license information:

1. From the oscilloscope Help menu, select **About TekScope**.

The Options section in the dialog box displays a list of installed options, including M-PHY Rx: MIPI M-PHY receiver solution.

2. To view the Option key, look in the Option Installation Key section.

See Also

- [Activate the License \(see page 7\)](#)
- [Options Menu \(see page 15\)](#)

M-PHY Rx Application Overview

TekExpress is the Tektronix Test Automation Framework, developed to support your current and future test automation needs. TekExpress uses a highly modular architecture that lets you deploy automated test solutions for various standards in a relatively short time.

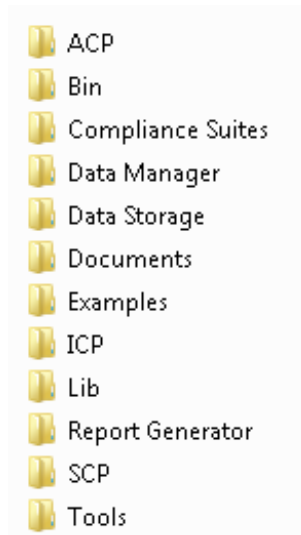
Key Features

M-PHY Rx has the following key features:

- Allows conformance testing to the Base specification version 1.0 and Compliance Test Specification (CTS)
- Automated Testing:
 - Reduces the amount of time required to conduct testing
 - Enables you to test devices faster
- Tests coverage: Automated test setup covers 4 out of 7 total high-speed receiver measurements.
- Selective testing: Allows you to select individual tests or test groups in the tree-structure.
- Reliable Results: Avoids repeated testing through accurate and reliable results from a single run
- Integrated [BER \(see page 94\)](#):
 - Leverage Bit-Error-Rate or Error-Count testing using oscilloscope-integrated ERRDT software in the background.
 - No external or extra hardware is required to perform BER testing.
 - Dialog box to support manual error entry.
- Customize the setup: Provision to configure the test setup according to the DUT-supported configuration, such as the HS Gear A or B, and test time.
- Detailed test reporting:
 - Provides a Pass/Fail summary table. (For details, see [Results Panel \(see page 25\)](#).)
 - Provides a consolidated report for all tests
 - Provides additional information such as test setup hardware and software details, signal type selected, Bit Error, execution time, and user-comments for each measurement

Application Directories and Usage

The application directory and associated files are organized as follows:



The following table lists the default directory names and their usage:

Table 3: Application directories and usage

Directory names	Usage
InstallDir\TekExpress\TekExpress MIPI M-PHY Rx	Contains the application and associated files
TekExpress M-PHY Rx\ACP	Contains instrument and M-PHY Rx application-specific interface libraries
TekExpress M-PHY Rx\Bin	Contains miscellaneous M-PHY Rx application libraries
TekExpress M-PHY Rx\Compliance Suites	Contains compliance-specific files
TekExpress M-PHY Rx\Data Manager	Contains result management-specific libraries of the M-PHY Rx application
TekExpress M-PHY Rx\Data Storage	Contains libraries needed for storing data
TekExpress M-PHY Rx\Documents	Contains the technical documentation for the M-PHY Rx application
TekExpress M-PHY Rx\Examples	Contains various support files
TekExpress M-PHY Rx\ICP	Contains instrument and M-PHY Rx application-specific interface libraries
TekExpress M-PHY Rx\Lib	Contains utility files specific to the M-PHY Rx application
TekExpress M-PHY Rx\Report Generator	Contains Excel Active X interface Library for Report Generation
TekExpress M-PHY Rx\SCP	Contains instrument and M-PHY Rx application-specific interface libraries
TekExpress M-PHY Rx\Tools	Contains instrument and M-PHY Rx application-specific files

See Also

- [View Test-Related Files \(see page 53\)](#)

File Name Extensions

The TekExpress M-PHY Rx application uses the following file name extensions:

File name extension	Description
.TekX	Session files are saved in this format but the extensions may not be displayed.
.seq	The test sequence file
.xml	The encrypted XML file that contains the test-specific configuration information The log file extension is also xml
.wfm	The test waveform file
.mht	Test result reports are saved in this format by default. Test reports can also be saved in HTML format (see page 40) .

See Also

- [View Test-Related Files \(see page 53\)](#)
- [Application Directories and Usage \(see page 10\)](#)

Start the Application

When you open the application after installation, the application checks for a file called `Resources.xml` located in the `My TekExpress` folder. If this file is not found, instrument discovery is performed before launching M-PHY Rx. The `Resources.xml` file contains information regarding instruments that are available on your network. If the application license was not installed using the TekScope menu **Utilities > Option Installation** selection, the application will allow 10 free launches. Each time you open the application without supplying a valid license key, one of the free trials is used.

To run the M-PHY Rx application, do any of the following:

- From the TekScope Analyze menu, select **Analyze > TekExpress M-PHY Rx**.
- Double-click any saved session file.


See Also

- [Activate the License \(see page 7\)](#)

Exit the Application





Use the following method to exit the application:

NOTE. *Using other methods to exit the application results in abnormal termination of the application.*

1. Click  on the application title bar.
2. Do one of the following:
 - If you have an unsaved session or test setup open, you are asked to save it before exiting. To save it, click **Yes**. Otherwise click **No**. The application closes.
 - A message box appears asking if you really want to exit TekExpress. To exit, click **Yes**.

Application Controls

Table 4: Application controls descriptions

Item	Description
Options menu (see page 15)	Opens the Options menu for access to global controls
Panels (see page 18)	Visual frames with sets of related options
Command buttons	Buttons that initiate an immediate action such as the Start, Stop, Pause, Continue, and Clear command buttons
Start button	<p>Start</p>  <p>Use the Start button to continuously acquire and accumulate measurements. If prior acquired measurements have not been cleared, the new measurements are added to the existing set.</p>
Stop button	<p>Stop</p>  <p>Use the Stop button to abort the test.</p>
Pause \ Continue button	Use the Pause button to temporarily interrupt the current acquisition. When a test is paused, the button name changes to Continue.
Clear button	<p>Clear</p>  <p>Use the Clear button to clear all existing measurement results. Adding or deleting a measurement, or changing a configuration parameter of an existing measurement, also clears measurements. This is to prevent the accumulation of measurement statistics or sets of statistics that are not coherent. This button is available only on the Results panel (see page 25).</p>
Application window move icon	 <p>Place the cursor over the three-dot pattern in the upper left corner of the application window. When the cursor changes to a hand, drag the window to the desired location.</p>

Options Menu

The Options menu is located in the upper right corner of the application.

The [Options menu \(see page 15\)](#) has the following selections:

Menu	Function
Default Test Setup	Opens an untitled test setup with defaults selected
Open Test Setup	Opens a saved test setup
Save Test Setup	Saves the current test setup selections
Save Test Setup As	Creates a new test setup based on an existing one
Open Recent	Displays a menu of recently opened test setups to select from
Instrument Control Settings (see page 17)	Shows the list of instruments connected to the test setup and allows you to locate and refresh connections to connected instruments
Keep On Top	Keeps the TekExpress M-PHY Rx utility on top of other open windows on the desktop
Email Settings (see page 16)	Use to configure email options for test run and results notifications
Help	Displays TekExpress Help
About TekExpress	<ul style="list-style-type: none"> ■ Displays application details such as software name, version number, and copyright ■ Provides access to license information (see page 8) for your M-PHY Rx installation ■ Provides a link to the Tektronix Web site



See Also

- [Application Controls \(see page 14\)](#)

Email Settings

Use the Email Settings utility to [configure email notifications \(see page 37\)](#) if you want M-PHY Rx to notify you when a test completes, produces an error, or fails. Select the type of test run information to include in the notification, such as test reports and test logs, the email message format, and the email message size limit.

NOTE. *Recipient email address, sender's address, and SMTP Server are mandatory fields.*

Email Settings

Recipient e-mail Address(es)
Note: Separate Email addresses with a comma

Sender's Address

Email Attachments

Reports
 ScoreCard
 Analysis Screenshot
 Status Log Last 20 Lines Full Log

Server Configuration

SMTP Server SMTP Port
Login
Password
Host Name

Email Configuration

Email Format HTML Plain Text Number of Attempts to Send
Max Email Size (MB) Timeout

Email Test Results When complete or on error

Test Email Apply Close

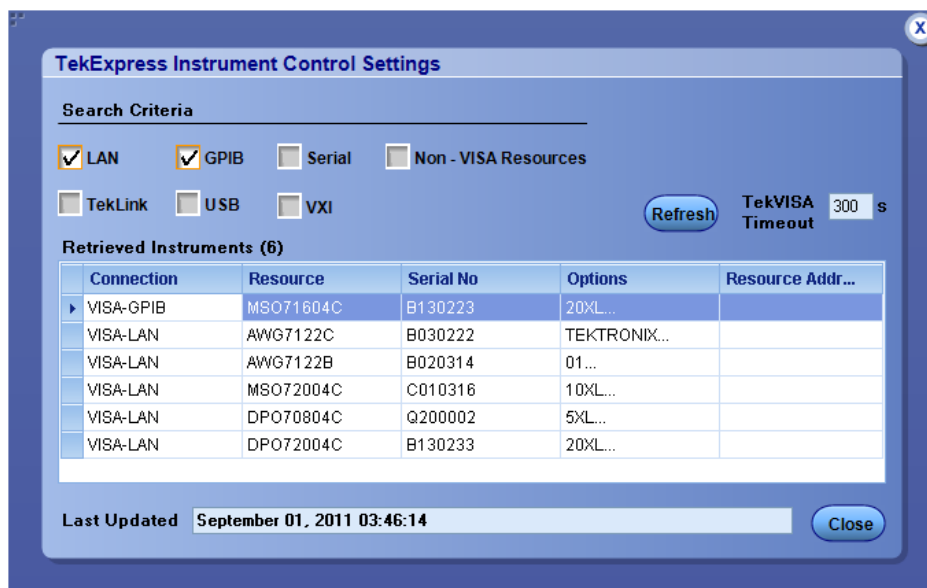
See Also

- [Options Menu \(see page 15\)](#)
- [Select Test Notification Preferences \(see page 39\)](#)

Instrument Control Settings

The Instrument Control Settings dialog box shows the list of resources found on different connections. It serves two purposes:

- Discovers the connected instruments
- Confirms the instrument connection setup



Use the Instrument Control Settings feature to [search for connected instruments \(see page 31\)](#) and view instrument connection details. Connected instruments displayed here can be selected for use in the Global Settings tab in the configuration section. See step 1 of [Configure Tests \(see page 36\)](#) for details.

See Also

- [Options Menu \(see page 15\)](#)

About Panels

TekExpress M-PHY Rx has the following main panels:

Table 5: Application panels

Panel Name	Purpose
Setup (see page 18)	The Setup panel allows you to configure the test setup. Use this panel to: <ul style="list-style-type: none"> ■ Select the device parameters (see page 33). ■ Select the test(s) (see page 35). ■ Configure the selected tests (see page 36). ■ Select test notification preferences (see page 39).
Status (see page 23)	View the progress and analysis status of the selected tests, and view test logs.
Results (see page 25)	View a summary of test results and select results viewing preferences.
Reports (see page 26)	Browse for reports, save reports as specific file types, specify report naming conventions, select report content to include (such as summary information, detailed information, user comments, setup configuration, application configuration), and select report viewing options.

See Also

- [Application Controls \(see page 14\)](#)

Setup Panel

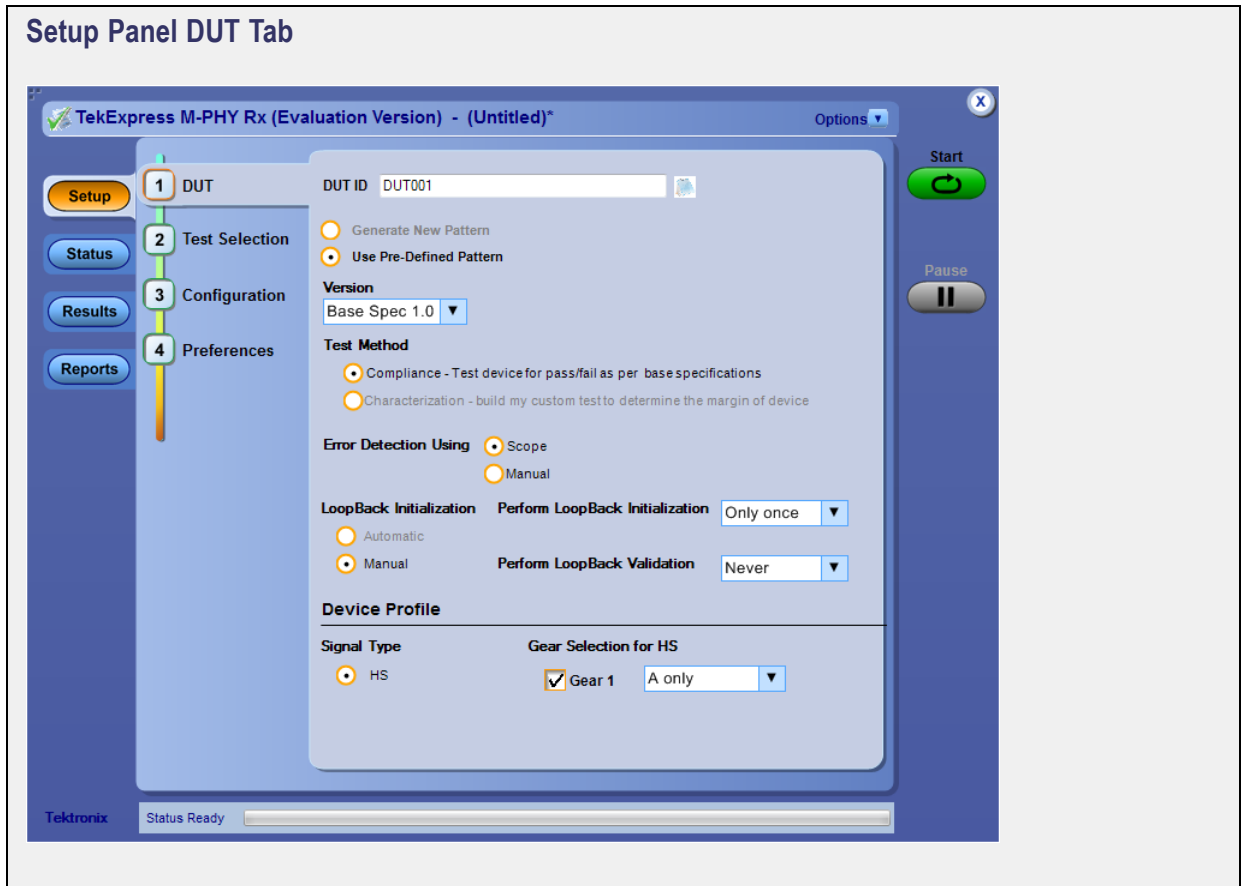
The Setup panel has numbered tabs to guide you through the test setup process:

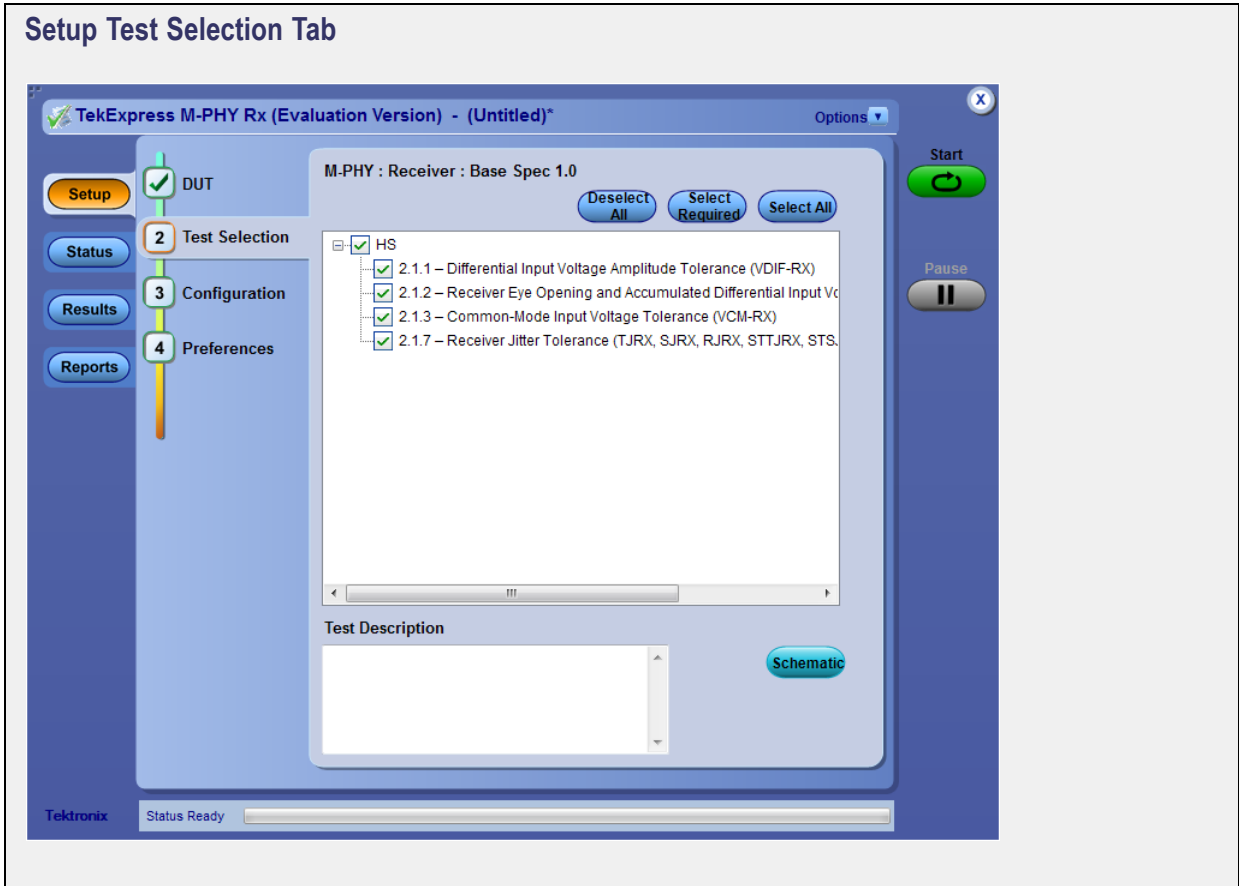
1 – [DUT \(see page 19\)](#), 2 – [Test Selection \(see page 20\)](#) and [Configuration \(see page 21\)](#), and 3 – [Preferences \(see page 22\)](#)

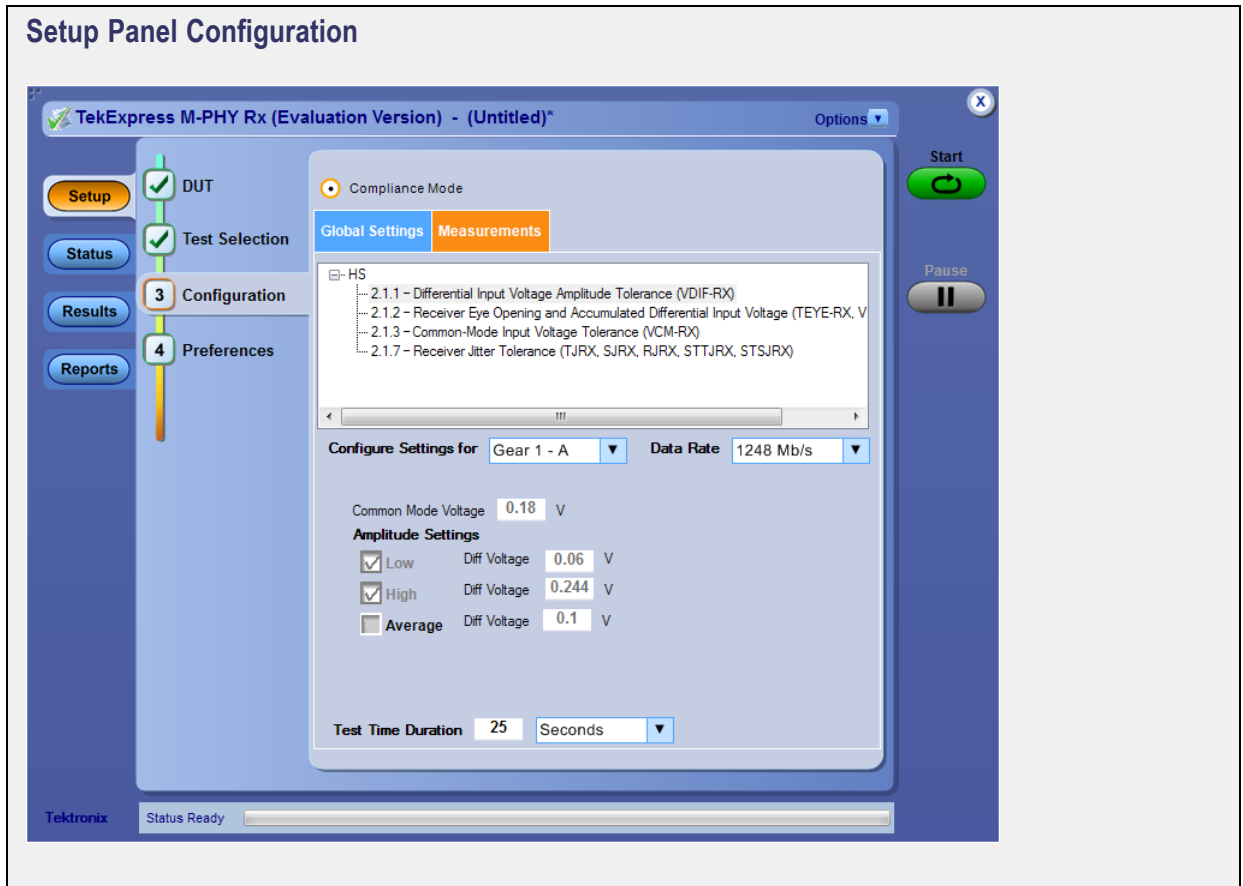
Options selected in a preceding tab affect options available in the next tab down. However, you can switch between the tabs in any order to modify your test parameters. For more information on using the Setup panel, see [About Setting Up Tests \(see page 29\)](#).

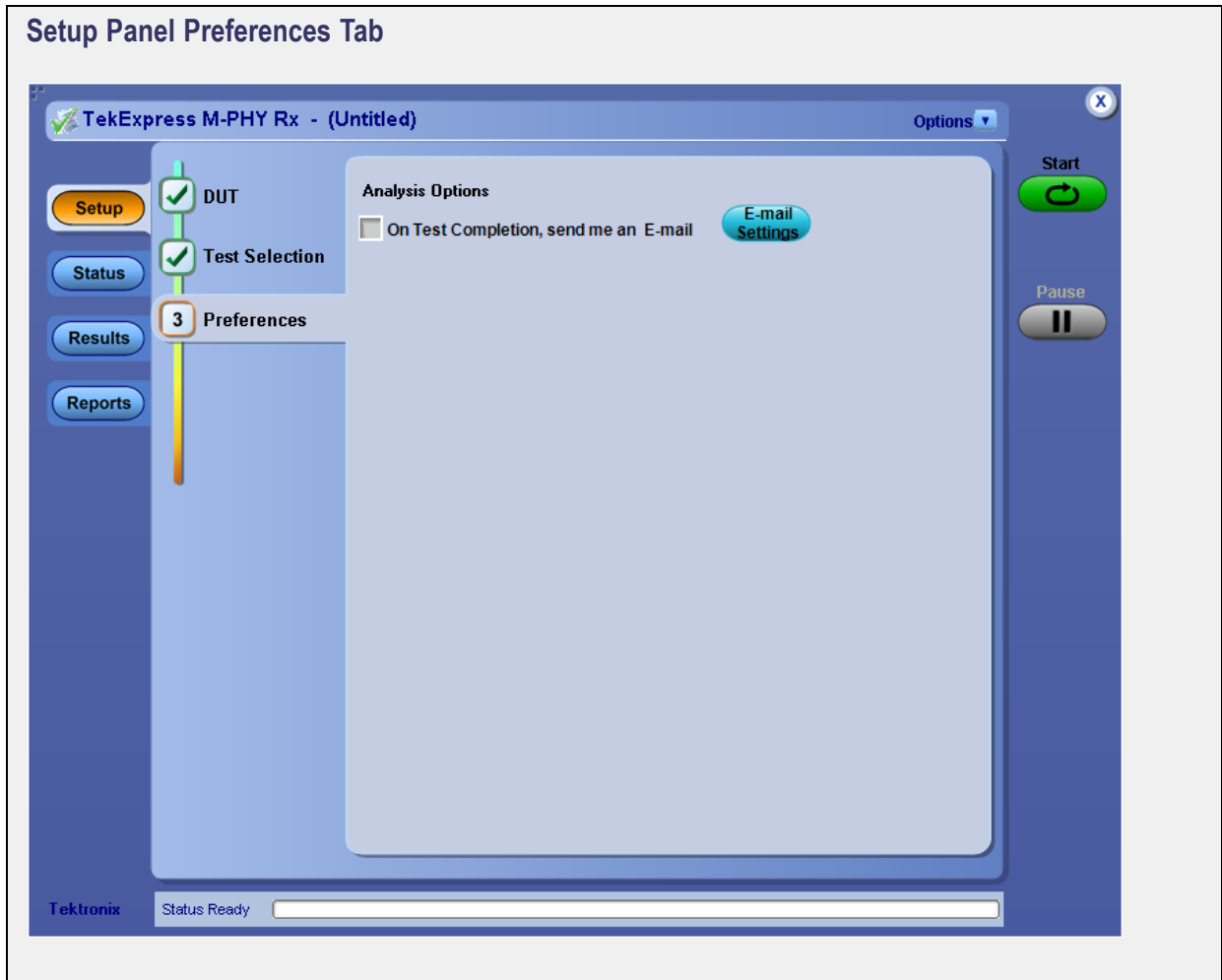
See Also

- [About Panels \(see page 18\)](#)





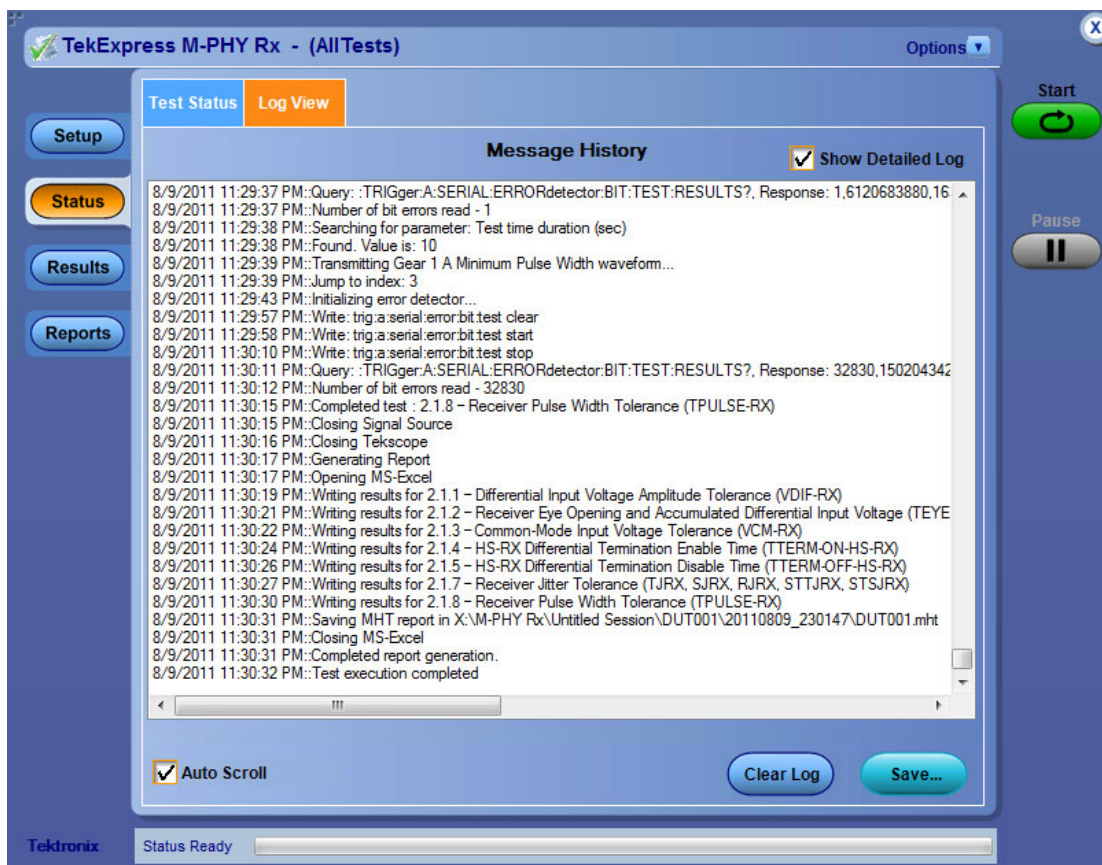




Status Panel

The Status panel has two tabs: the [Test Status \(see page 24\)](#) tab displays the analysis status of the tests as they are being executed; the Log View tab (shown below) displays a view of the test log.

For more information on using the Status panel, see [View the Progress of Analysis \(see page 46\)](#).



See Also

- [About Panels \(see page 18\)](#)

Test Status View

TekExpress M-PHY Rx - (SQE)*

Options

Setup

Status

Results

Reports

Test Name	Analysis Status
2.1.1 – Differential Input Voltage Amplitude Tolerance (V...	Completed
2.1.2 – Receiver Eye Opening and Accumulated Differe...	Completed
2.1.3 – Common-Mode Input Voltage Tolerance (VCM-RX)	Completed
2.1.7 – Receiver Jitter Tolerance (TJRX, SJRX, RJRX, S...	Completed

Start

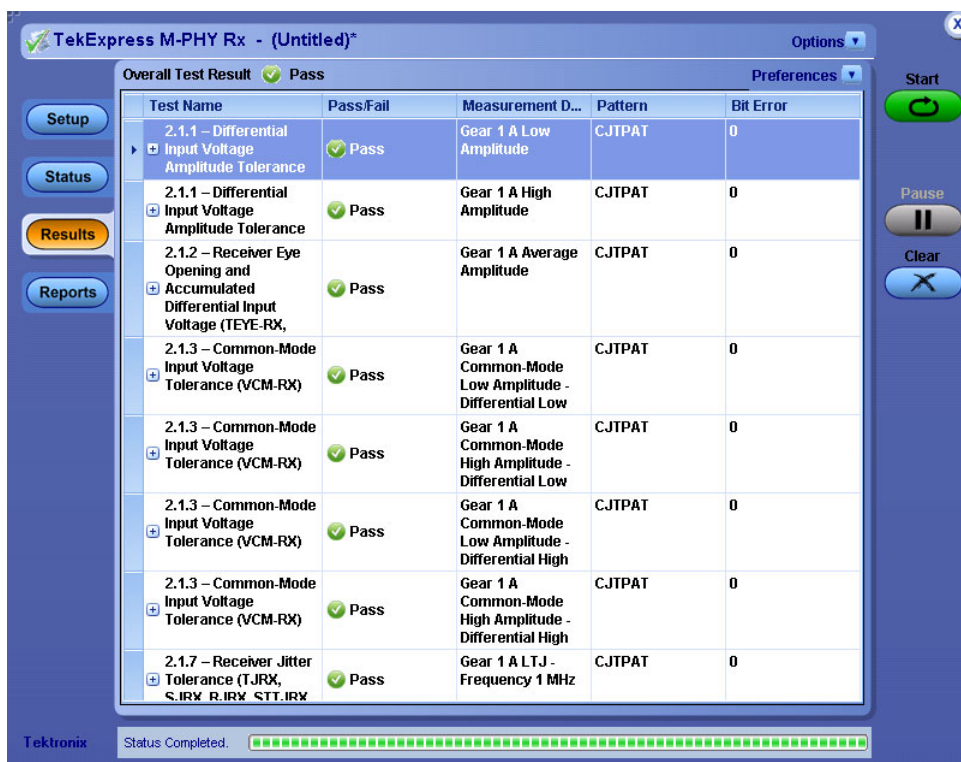
Pause

Tektronix

Status Ready

Results Panel

When a test finishes, the application switches to the Results panel to display a summary of test results. Set viewing preferences for this panel from the Preferences menu in the upper right corner. Viewing preferences include showing whether a test passed or failed, summary results or detailed results, and enabling wordwrap. For information on using this panel, see [View Test Results \(see page 47\)](#).



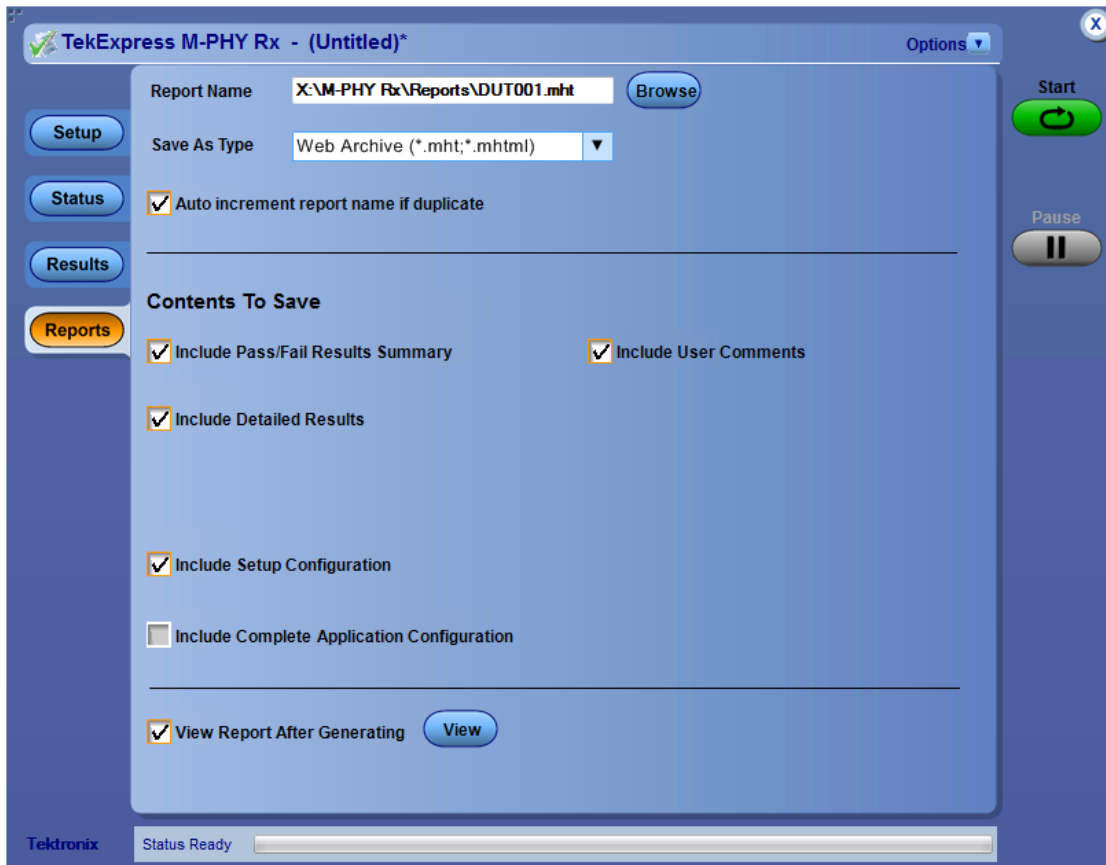
See Also

- [About Panels \(see page 18\)](#)

Reports Panel

Use the Reports panel to browse for reports, name and save reports, select report content to include, and select report viewing options.

For information on setting up reports, see [Select Report Options \(see page 40\)](#). For information on viewing reports, see [View the Report \(see page 47\)](#).



See Also

- [About Panels \(see page 18\)](#)

About Test Setups

TekExpress M-PHY Rx opens with the default setup selected. Run a test before or after saving a setup. When you save a setup, the selected oscilloscope, general parameters, measurement limits, test, and other configuration settings are all saved under the setup name. You can run a saved test without having to do any other setup except ensuring that the oscilloscope is ready. Open the setup and click **Start**.

See Also

- [About Setting Up Tests \(see page 29\)](#)
- [Save a Test Setup \(see page 27\)](#)
- [Recall a Saved Test Setup \(see page 28\)](#)
- [Create a New Test Setup Based on an Existing One \(see page 28\)](#)

Save a Test Setup

Save a test setup before or after running a test using the parameters you want saved. Create a new test setup from any open setup or from the default setup. When you select the default test setup, all parameters are returned to their defaults. The following instructions start from the default test setup:

1. From the Options menu, select **Default Test Setup**.
2. Select the desired options in the Setup panel.
3. Select the desired [report options \(see page 40\)](#).
4. If desired, run the test to be sure that it captures the information you want. If it does not, edit the parameters.
5. From the Options menu, select **Save Test Setup**.

See Also

- [About Setting Up Tests \(see page 29\)](#)
- [About Configuring Tests \(see page 36\)](#)
- [Test Setup Overview \(see page 33\)](#)

Recall a Saved Test Setup

These instructions are for recalling saved test setups:

1. From the Options menu, select **Open Test Setup**.
2. In the File Open dialog box, select the desired setup from the list and then click **Open**.

See Also

- [About Test Setups \(see page 27\)](#)
- [Create a New Test Setup Based on an Existing One \(see page 28\)](#)

Create a New Test Setup Based on an Existing One

Use this method to create a variation on a test setup without having to create the setup from the beginning.

1. From the Options menu, select **Open Test Setup**.
2. In the File Open dialog box, select the desired setup from the list and then click **Open**.
3. Modify the parameters as desired.
4. From the Options menu, select **Save Test Setup As**.
5. In the File Save As dialog box, enter a test setup name and then click **Save**.

See Also

- [About Test Setups \(see page 27\)](#)
- [Test Setup Overview \(see page 33\)](#)

About Setting Up Tests

Set up tests using the tabs in the Setup panel. Settings in the DUT tab use a top-down, left-to-right logic flow, so that any parameter that affects or acts as a filter for other parameters appears either to the top of or to the left of the affected parameters.

Tests are saved when you save a test setup. To avoid overwriting test results, remember to assign a unique name to the test either before running it or immediately after.

The application supports the following tests in Compliance Mode only:

- [2.1.1 Differential Input Voltage Amplitude Tolerance \(VDIF-RX\) \(see page 94\)](#)
- [2.1.2 Receiver Eye Opening and Accumulated Diff Input Voltage \(TEYE-RX, VDIF-ACC-RX\) \(see page 96\)](#)
- [2.1.3 Common-Mode Input Voltage Tolerance \(VCM-RX\) \(see page 98\)](#)
- [2.1.7 Receiver Jitter Tolerance \(TJRX, SJRX, RJRX, STTJRX, STSJRX\) \(see page 99\)](#)

See Also

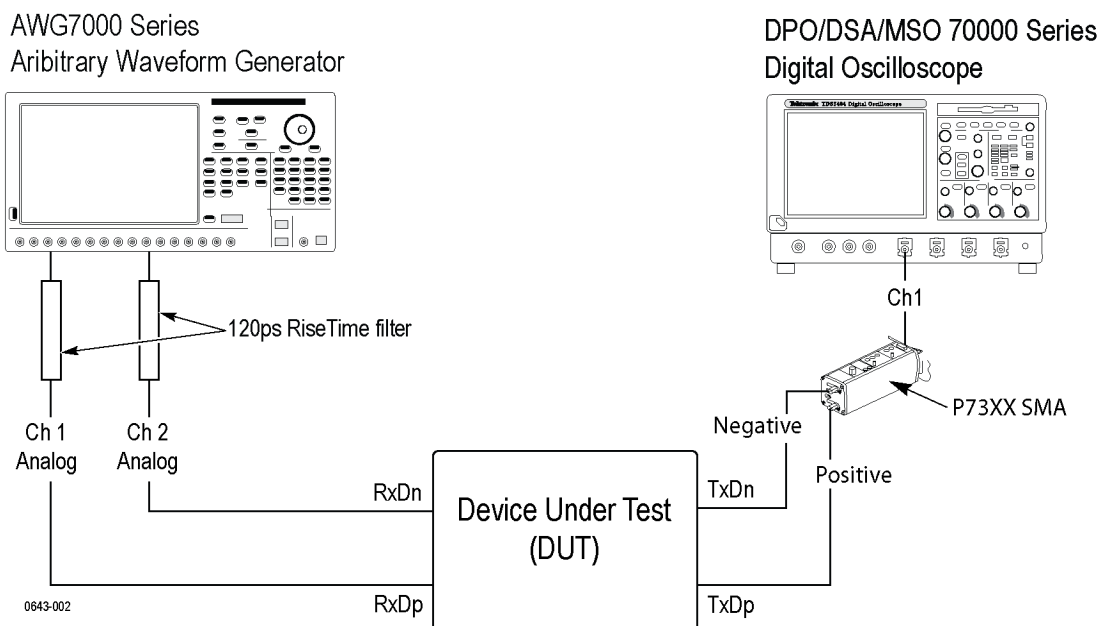
- [About Test Setups \(see page 27\)](#)
- [Before You Click Start \(see page 43\)](#)
- [About Running Tests \(see page 43\)](#)

Equipment Connection Setup

To run tests, you need the following equipment (for details, see [Minimum System Requirements \(see page 4\)](#)):

- A [supported Tektronix oscilloscope \(see page 3\)](#)
- Arbitrary Waveform Generator (AWG)
- Differential probe
- Device under test
- Filters

Connection diagram



See Also

- [View Connected Instruments \(see page 31\)](#)
- [About Setting Up Tests \(see page 29\)](#)
- [Receiver Testing Measurement Procedure \(see page 91\)](#)

View Connected Instruments

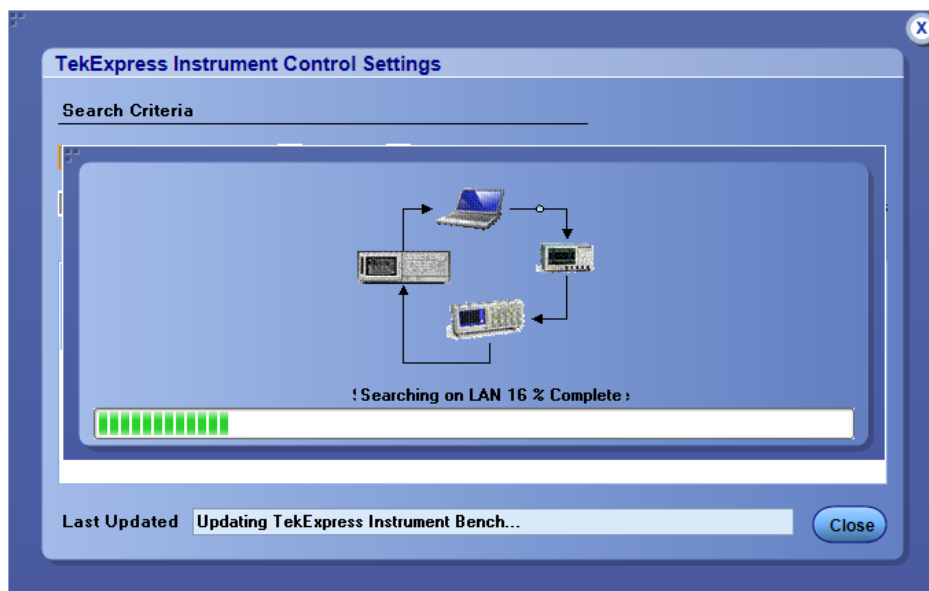
Use the Instrument Control Settings dialog box to view or search for connected instruments required for the tests. The application uses TekVISA to discover the connected instruments.

To refresh the list of connected instruments:

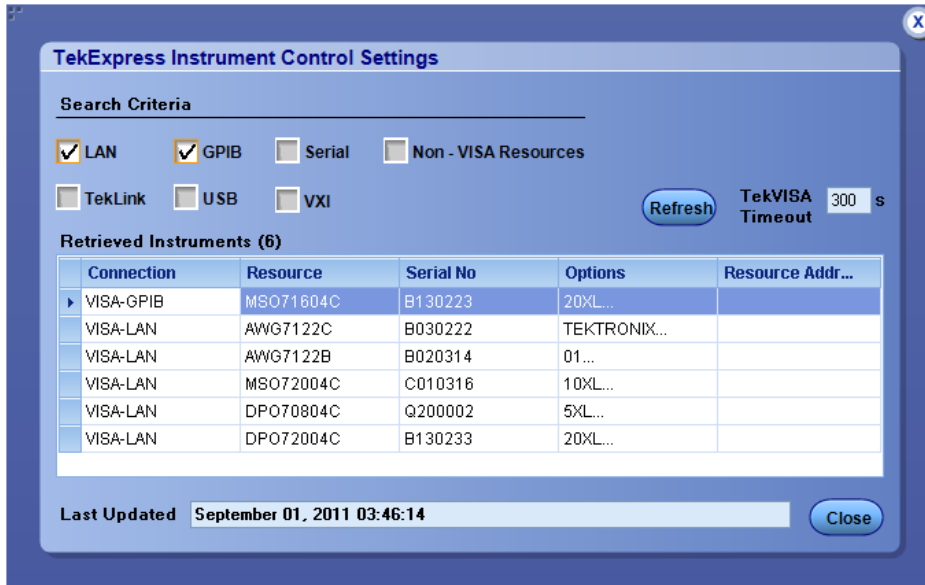
1. From the Options menu, select **Instrument Control Settings**.
2. In the Search Criteria section of the Instrument Control Settings dialog box, select the connection types of the instruments to search for.

Instrument search is based on the VISA layer but different connected cables determine the resource type, such as LAN, GPIB, and USB. For example, if you choose LAN, the search will include all the instruments supported by TekExpress that are communicating over the LAN. If the search does not find any instruments that match a selected resource type, a message appears telling you that no such instruments were found.

3. Click **Refresh**. TekExpress searches for connected instruments.



4. After discovery, the dialog box lists the instrument-related details based on the search criteria you selected. For example, if you selected LAN and GPIB as the search criteria, the application checks for the availability of instruments over LAN, then GPIB.



The details of the instruments are displayed in the Retrieved Instruments table. The time and date of instrument refresh is displayed in the Last Updated field.

See Also

- [Configure Tests \(see page 36\)](#)
- [Equipment Setup \(see page 30\)](#)

Test Setup Overview

Test setup includes configuration parameters and report options. Use the options in the [Setup panel \(see page 18\)](#) and [Reports panel \(see page 26\)](#) to select and configure tests.

1. [Select the device parameters \(see page 33\)](#).
2. [Select one or more tests \(see page 35\)](#).
3. [Configure tests \(see page 36\)](#).
4. [Select test notification preferences \(see page 39\)](#).
5. [Select report options \(see page 40\)](#).

See Also

- [About Test Setups \(see page 27\)](#)
- [Pre-Run Checklist \(see page 45\)](#)
- [Before You Click Start \(see page 43\)](#)
- [About Running Tests \(see page 43\)](#)

Select Device Parameters

Use the Setup panel DUT tab settings to select parameters for the device under test. Options selected here are global settings that apply to all tests for the current session. Options here also affect the list of available tests in the Test Selection tab.


1. In the Setup panel, click the **DUT** tab.
2. If needed, in the DUT ID field, enter the ID for the device. The default value is DUT001. The name you enter here appears on reports.
3. (Optional) To add comments that will appear at the top of the test report for the selected DUT, click the note pad icon () to the right of the DUT ID field and specify a comment up to 256 characters. (To enable or disable comments appearing on the test report, see [Select Report Options \(see page 40\)](#).)
4. Select from the following options if available. (Settings that do not apply to compliance testing cannot be changed and are grayed out.)

Table 6: DUT settings

Setting	Description
Use Pre-Defined Pattern	Uses the built-in pre-defined pattern associated with the selected test. This cannot be changed in Compliance Mode.
Version	Select the M-PHY Rx specification version. The latest version is the default.

Table 6: DUT settings (cont.)

Setting	Description
Test Method	Compliance testing only
Error Detection Using	<ul style="list-style-type: none"> ■ Scope: TekExpress uses the oscilloscope for error testing. This cannot be changed in Compliance Mode. ■ Manual: User enters the error count using a dialog box.
LoopBack Initialization	This is set up manually. The application prompts you to set this value before executing a test.
Perform LoopBack Initialization	<p>Determines how frequently loopback initialization occurs.</p> <ul style="list-style-type: none"> ■ Always: Do loopback initialization before starting every test. ■ Only Once: Do loopback initialization only once before starting the first measurement in the current test session. ■ Never: Do not do loopback initialization.
Perform LoopBack Validation	<p>Determines when loopback validation occurs. This process validates the DUT.</p> <ul style="list-style-type: none"> ■ Only Once: Do loopback validation only once before starting the first measurement of the current session. ■ Always: Do loopback validation at the beginning of every test. <p>NOTE. This may require you to be available to put the DUT in loopback mode several times during testing, depending on the number of tests and the test options selected.</p> <ul style="list-style-type: none"> ■ Never: Do not do loopback validation.
Signal Type	HS: High Speed
Gear Selection for HS	<p>Select the gear and then select the variation of the gear that is supported by the DUT. Each gear can have the following variations:</p> <ul style="list-style-type: none"> ■ A and B: If selected, both options A and B for the selected gear will be tested in the current session. ■ Only A: If selected, only option A for the selected gear will be tested in the current session. ■ Only B: If selected, only option B for the selected gear will be tested in the current session.

See Also

- [About Setting Up Tests \(see page 29\)](#)
- [Loopback Validation \(see page 93\)](#)






Select a Test

Use these instructions to select the tests to run on the connected DUT.

1. In the Setup panel, click the **Test Selection** tab.
2. All tests are selected by default. Click **Deselect All** if you do not want to run all tests.
3. Select the desired test(s):
 - To select all required tests, click the **Select Required** button.
 - To select all tests in the list, click the **Select All** button.
 - To select individual tests, select only the check boxes for the desired tests.

Test Selection Controls

Table 7: Setup panel test controls

Button	Description
	Use to access the configuration settings for a selected test
	Displays the schematic document for the selected test. Use to verify the test setup before running the test
	Deselects all tests in the table
	Selects tests required for compliance and deselects all other tests
	Selects all tests in the table

See Also

- [About Setting Up Tests \(see page 29\)](#)

About Configuring Tests

Use configuration settings to view the measurement parameters for selected tests. Some settings can be changed.

Configuration settings consist of Global parameters and Measurement parameters. Global parameters are common for all tests. Measurement parameters are specific to the test selected in the tree view section of the configuration settings.

See Also

- [Configure Tests](#)
- [Measurement Parameter Descriptions \(see page 103\)](#)
- [About Running Tests \(see page 43\)](#)

Configure Tests

Use these instructions for configuring tests. To return to test selection from the Configuration section, click the **Test Selection** button.

NOTE. *You cannot change test parameters that are grayed out.*

1. Modify [Global settings \(see page 37\)](#) as desired:
 - To select the instruments for testing, click **Global Settings**. In the Instruments Detected section, click in the shaded areas to activate the drop-down lists and then select the desired instruments for Signal Generator and Error Detector. If you do not see the desired instrument in the list, [refresh the list \(see page 31\)](#).
 - To change the Trigger source for the current test session, under Scope Settings, select the desired channel from the drop-down list.
2. To modify any individual [test measurement settings \(see page 103\)](#), click **Measurements**, select the test in the tree view and change the settings.

See Also

- [About Setting Up Tests \(see page 29\)](#)

Common Test Parameters

The following table lists the settings and parameters common to all tests.

Table 8: Common parameters and values

Parameter type	Parameter
Mode	Compliance Mode is selected by default and cannot be deselected. Test parameters that fall outside of compliance values are disabled.
Global Settings	<p>These settings apply to all tests selected for the current session. You can change only some of these settings.</p> <ul style="list-style-type: none"> ■ Signal Generator: Specifies the instrument to use for generating the signals to test. This can be changed. ■ Error Detector: Specifies the instrument to use for detecting errors in the signal. This can be changed. ■ Scope Settings: Specifies the instrument channel to use as the source for the trigger. This can be changed. ■ Signal Source Settings: <ul style="list-style-type: none"> ■ Test Pattern: CJTPAT ■ 8b/10b encoded: When selected, indicates that the waveform pattern selected has to be 8b/10b encoded. ■ Channel Amplitude: The value of the amplitude to be set for the selected channel.
Measurements	These settings apply to the test selected in the tree view of the configuration section. For details, see Measurement Parameter Descriptions (see page 103) .

See Also

- [Configure Tests \(see page 36\)](#)

Compliance Jitter Tolerance Pattern

Configure Email Notification

To be notified by email when a test completes, fails, or produces an error, configure the email settings.

1. From the Options menu, select **Email Settings** to open the [Email Settings \(see page 39\)](#) dialog box.
2. (Required) For Recipient email Address(es), enter your email address. To include multiple addresses, separate the addresses with commas.

3. (Required) For Sender's Address, enter the email address used by the instrument. This address consists of the instrument name, followed by an underscore, followed by the instrument serial number, the @ symbol and then the email server used. For example: DPO72004C_B130099@yourcompany.com.
4. (Required) In the Server Configuration section, type the SMTP Server address of the Mail server configured at the client location, and the SMTP Port number, in the corresponding fields.

If this server requires password authentication, enter a valid login name, password, and host name in the corresponding fields.

NOTE. *If any of the above required fields are left blank, the settings will not be saved and email notifications will not be sent.*

5. In the Email Attachments section, select from the following options:
 - **Reports:** Select to receive the test report with the notification email.
 - **Status Log:** Select to receive the test status log with the notification email. If you select this option, then also select whether you want to receive the full log or just the last 20 lines.

NOTE. *The ScoreCard and Analysis Screenshot options are not available in M-PHY Rx.*

6. In the Email Configuration section, select as desired:
 - Select the message file format to send: HTML (the default) or plain text.
 - Enter a maximum file size for the email message. Messages with attachments larger than this limit will not be sent. The default is 5 MB.
 - To limit the number of attempts that the system makes to send a notification, enter the number in the Number of Attempts to Send field. The default is 1. You can also specify a timeout.
7. Select the **Email Test Results When complete or on error** check box. Use this check box to quickly enable or disable email notifications.
8. To test your email settings, click **Test Email**.
9. To apply your settings, click **Apply**.
10. Click **Close** when finished.

Email Settings

Recipient e-mail Address(es)
Note: Separate Email addresses with a comma

Sender's Address

Email Attachments

Reports
 ScoreCard
 Analysis Screenshot
 Status Log Last 20 Lines Full Log

Server Configuration

SMTP Server SMTP Port
Login
Password
Host Name

Email Configuration

Email Format HTML Plain Text Number of Attempts to Send
Max Email Size (MB) Timeout

Email Test Results When complete or on error

Test Email Apply Close

Select Test Notification Preferences

Select how you want to be notified by the system when a test measurement fails:

1. In the Setup panel, click the **Preferences** tab.
2. To receive an email when testing completes, select the **On Test Completion, send me an email** check box.

The application will send you an email when the current test completes. If you select this option, select the **Email Test Results when complete or on error** check box in the Email Settings. Click **Email Settings** to [configure the email settings \(see page 37\)](#).

See Also

- [About Setting Up Tests \(see page 29\)](#)
- [Select Report Options \(see page 40\)](#)

Select Report Options

Use the [Reports panel \(see page 26\)](#) to select which test information to include in the report, and the naming conventions to use for the report. For example, always give the report a unique name or select to have the same name increment each time you run a particular test. Generally, you would select report options before running a test or when creating and saving test setups. Report settings are included in saved test setups.

In the Reports panel, select from the following options:

Table 9: Report options

Setting	Description
Report Name	<p>Displays the name and location where the report will be saved when generated. The report and related files are stored in the Untitled Session folder located by default at \My TekExpress\ M-PHY Rx \Untitled Session. The report file in this folder gets overwritten each time you run a test unless you specify a unique name or select to auto increment the report name.</p> <p>Change the report name or location.</p> <p>Do one of the following:</p> <ul style="list-style-type: none"> ■ In the Report Name field, type over the current folder path and name. ■ Double-click in the Report Name field. Make selections from the popup keyboard and then click the Enter button. <p>Be sure to include the entire folder path, the file name, and the file type. For example: C:\Documents and Settings\your user name\My Documents\My TekExpress\M-PHY Rx\DUT001_Test_211.mht.</p> <p>NOTE. <i>You cannot change the file location using the Browse button.</i></p> <p>Open an existing report.</p> <p>Click Browse, locate and select the report file and then click View at the bottom of the panel.</p>
Save As Type	<p>Saves a report in a file type different from the default. Lists supported file types to choose from.</p> <p>NOTE. <i>If you select a file type different from the default, be sure to change the report file name extension in the Report Name field to match.</i></p>
Auto increment report name if duplicate	<p>If the application finds a report with the same name as the one being generated, the application automatically increments the name of the report. For example: DUT001, DUT002, DUT003. This option is enabled by default.</p>
Include Pass/Fail Results Summary	<p>The color block labeled Test Result indicating whether the test passed or failed will be included in the report. For details, see Report Contents in View a Report (see page 47).</p>
Include Detailed Results	<p>The report will include parameter limits, execution time, and any test-specific comments generated during the test.</p>
Include Setup Configuration	<p>Information about hardware and software used in the test will be included in the summary box at the top of the report. Information includes: the oscilloscope model and serial number, probe model and serial number, AWG model and serial number, firmware version for the oscilloscope and AWG, SPC and factory calibration status, and software versions for applications used in the measurements.</p>

Table 9: Report options (cont.)

Setting	Description
Include Complete Application Configuration	A table listing general, common, and acquired parameters used in the test will be included in the report. This option is disabled by default.
Include User Comments	Select to include any comments about the test that you or another user added in the DUT tab of the Setup panel. Comments appear in the Comments section under the summary box at the beginning of each report.
View Report After Generating	Automatically opens the report in your Web browser when the test completes. This option is selected by default.

See Also

- [View a Report \(see page 47\)](#)
- [About Setting Up Tests \(see page 29\)](#)

About Running Tests

After selecting and configuring the test, [review the pre-run checklist \(see page 45\)](#) and then click **Start** to run the tests. While tests are running, you cannot access the Setup or Reports panels. To monitor the test progress, switch back and forth between the Status panel and the Results panel.

The application displays a report when the tests are complete. While the tests are running, other applications may display windows in the background. The TekScope application takes precedence over other applications, but you can switch to other applications by using the **Alt + Tab** key combination. To keep the TekExpress M-PHY Rx application on top, select **Keep On Top** from the M-PHY Rx Options menu.

See Also

- [About Configuring Tests \(see page 36\)](#)
- [About Setting Up Tests \(see page 29\)](#)
- [Before You Click Start \(see page 43\)](#)

Before You Click Start

Before you run tests for the first time, do the following:

1. Understand where your test files are stored on the instrument.

After you install and launch TekExpress M-PHY Rx, it creates the following folders on the oscilloscope:

- \Program Files (x86)\Tektronix\TekExpress\TekExpress M-PHY Rx

NOTE. The Program Files folder for Windows 7 is *Program Files (x86)* for 64-bit. Windows XP uses *Program Files*.

- \My Documents\My TekExpress\M-PHY Rx
- \My Documents\My TekExpress\M-PHY Rx\Untitled Session

Every time you launch TekExpress M-PHY Rx, an **Untitled Session** folder is created in the M-PHY Rx folder. The **Untitled Session** folder is automatically deleted when you exit the M-PHY Rx application.



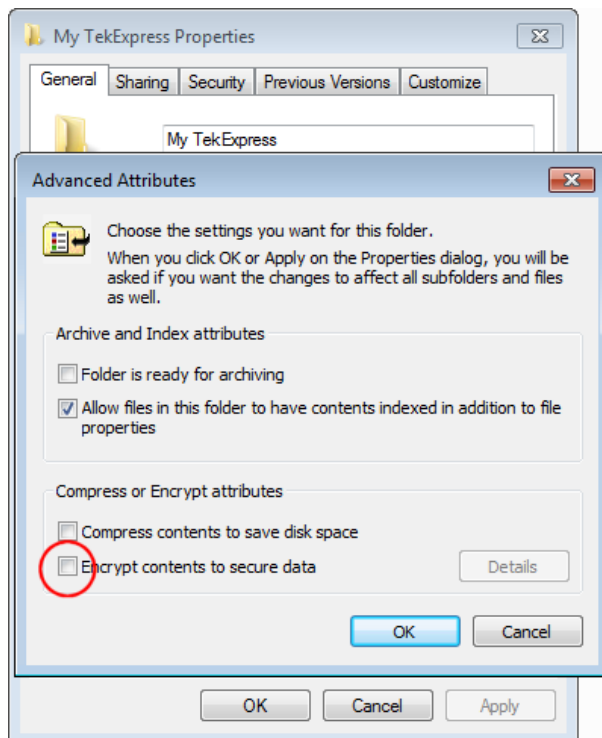
CAUTION. Do not modify any of the session files or folders because this may result in loss of data or corrupted session files. Each session has multiple files associated with it. When you save a session, a .TekX file, and a folder named for the session that contains associated files, is created on the oscilloscope X: drive.

2. [Map the shared My TekExpress folder \(see page 104\)](#) as X: (X drive) on all instruments used in test setup running Microsoft Windows Operating System.

The My TekExpress folder has the shared name format <domain><user ID>My TekExpress. Or, if the instrument is not connected to a domain, then the shared name format is <instrument name><user ID>My TekExpress. This shared folder is used to save the test session files and is used during any other file transfer operations.

NOTE. If the X: drive is mapped to any other shared folder, the application will display a warning message asking you to disconnect the X: drive manually.

3. Make sure that the My TekExpress folder has read and write access and that the contents are not set to be encrypted:
 - a. Right-click the folder and select **Properties**.
 - b. Select the **General** tab and then click **Advanced**.
 - c. In the Advanced Attributes dialog box, make sure that the option **Encrypt contents to secure data** is NOT selected. Example



4. Review the [pre-run checklist \(see page 45\)](#) before you run a test.

See Also

- [View Test-Related Files \(see page 53\)](#)
- [Application Directories and Usage \(see page 10\)](#)
- [File Name Extensions \(see page 11\)](#)

Pre-Run Checklist

Do the following before you click Start to run a test. If this is the first time you are running a test for a setup, refer to the information in [Before You Click Start \(see page 43\)](#).

1. Make sure that all the required instruments are properly warmed up (approximately 20 minutes).
2. Perform Signal Path Compensation (SPC).
 - a. On the oscilloscope main menu, select the **Utilities** menu.
 - b. Select **Instrument Calibration**.
3. Perform deskew on any cables.
4. Verify that the application is able to find the DUT. If it cannot, [perform a search for connected instruments \(see page 31\)](#).
 - a. Launch the M-PHY Rx application.
 - b. Select the **Setup** panel, click the **Test Selection** tab. Select any test and then click **Configure**.
 - c. In the Configuration section, click **Global Settings**.
 - d. In the Instruments Detected section, click the drop-down arrow to the right of the listed instruments and make sure that the DUT is listed.

See Also

- [Equipment Connection Setup \(see page 30\)](#)

View the Progress of Analysis

The [Status panel \(see page 23\)](#) displays a record of the test as it is executed. By default, the application switches to this panel after you click the Start button to run a test. Choose from the following two views by selecting the named tab, even while a test is in progress.

Test Status

The Test Status tab presents a collapsible table with information about each test as it is running.

Table 10: Status tab table

Column	Description
Test Name	Name of the test
Analysis Status	Progress state of the analysis: <ul style="list-style-type: none"> ■ To be started ■ In progress ■ Completed ■ Aborted

Log View

The Log View tab provides a list of actions that happen as the test executes. Use this tab to review or troubleshoot tests.

Table 11: Log View options

Item	Description
Message History	This window timestamps and displays all run messages.
Show Detailed Log	Select this check box to record a detailed history of test execution.
Auto Scroll	Select this check box to have the program automatically scroll down as information is added to the log during the test.
Clear Log	Click this button to clear all messages in the Message History window.
Save	Use to save the log file as a text file for examination. Displays a standard Save File window and saves the status messages in the file that you specify.

See Also

- [View Test Results \(see page 47\)](#)

View Test Results

When a test finishes, the application switches to the [Results panel \(see page 25\)](#), which displays a summary of test results. The Overall Test Result is displayed at the top left of the Results table. If all of the tests for the session pass, the overall test result will be Pass. If even one test out of multiple tests fails, the overall test result will show Fail.

Each test result occupies a row in the Results table. By default, results are displayed in summary format with the measurement details collapsed and with the Pass/Fail column visible. Change the view in the following ways:

- To expand all tests listed, select **View Results Details** from the Preferences menu in the upper right corner.
- To expand and collapse tests, click the plus and minus buttons.
- To collapse all expanded tests, select **Preferences > View Results Summary**.
- To remove or restore the Pass/Fail column, select **Preferences > Show Pass/Fail**.
- To enable or disable the wordwrap feature, select **Preferences > Enable Wordwrap**.
- To expand the width of a column, place the cursor over the vertical line that separates the column from the column to the right. When the cursor changes to a double-ended arrow, hold down the mouse button and drag the column to the desired width.
- To sort the test information by column, click the column head. When sorted in ascending order, a small up arrow is displayed. When sorted in descending order, a small down arrow is displayed.
- To clear all test results displayed, click **Clear**.

See Also

- [View a Report \(see page 47\)](#)
- [View the Progress of Analysis \(see page 46\)](#)

View a Report

After the analysis, a report is automatically generated. By default, the report is displayed in your Web browser unless you cleared the **View Report After Generating** check box in the Reports panel before running the test. If you cleared this check box, or if you want to view a different test report, do the following:

1. In the Reports panel, click the **Browse** button and locate the desired report.

The path to the report displays in the Report Name field.

NOTE. If you did not save the test setup after running the report and you either closed the application or you ran another report, the report file was not saved.

2. At the bottom of the Reports panel, click **View**.

For information on changing the file type, file name, and other report options, see [Select Report Options](#) (see page 40).

Report Contents

In the report, a table shows the test name, measurement details, test result (pass/fail), mode, and analysis time.

Setup Configuration Information

Setup configuration information is listed in the summary box at the beginning of the report. This information includes the oscilloscope model and serial number, probe model and serial number, and software versions. To exclude this information from the report, clear the **Include Setup Configuration** check box in the Reports panel before running the test.



TekExpress M-PHY-RX Receiver Test Report

DUT ID : <u>DUT001</u>	Device Type : <u>M-PHY</u>
Date/Time : <u>Sept. 13, 2011 / 04:22:03</u>	Execution Time : <u>8 Min</u>
MOI/CTS/UTD/Spec Version : <u>Base Spec 1.0</u>	
Overall Compliance Mode : <u>Yes</u>	
Overall Test Result : <u>Pass</u>	
Scope Model : <u>DSA71254B</u>	Scope FW Version : <u>5.3.4 DEVBUILD</u>
Scope Serial Number : <u>Q100003</u>	SPC, Factory Calibration : <u>PASS,PASS</u>
Probe Model : <u>N/A</u>	TekExpress Version : <u>1.0.0.28</u>
Probe Serial Number : <u>N/A</u>	DPOJET Version : <u>NA</u>
AWG Model : <u>AWG7102</u>	AWG Firmware Version : <u>3.3.181.755</u>
AWG Serial Number : <u>B010150</u>	

Comments

Type Your Comments

User Comments

If you selected to include comments in the test report, any comments you added in the DUT tab of the Setup panel appear in the Comments section directly below the summary box.

AWG Model: <u>AWG7102</u>		AWG Firmware
AWG Serial Number: <u>B010150</u>		
Comments		
Loopback Initialization - Only Once		
Loopback Validation - Always		
Gear1 A only		
Report all options selected		
AWG in GPIB network		
Test Name		Measurement Details

Pass/Fail Summary

The Pass/Fail Summary column indicates whether a test passed or failed. If the test passed, the column cell is green. If the test failed, it is red. To exclude this information from the report, clear the **Include Pass/Fail Results Summary** check box in the Reports panel before running the test.

Test Name	Measurement Details	Pattern	Bit Error	Test Result	Limit
2.1.1 – Differential Input Voltage Amplitude Tolerance (VDIF-RX)	Gear 1 A Low Amplitude	CJTPAT	0	Pass	=0
	Average Amplitude - Informative	CJTPAT	0	N.A	=0
	High Amplitude	CJTPAT	0	Pass	=0
2.1.2 – Receiver Eye Opening and Accumulated Differential Input Voltage (TEYE-RX, VDIF-ACC-RX)	Gear 1 A Low Amplitude	CJTPAT	0	Pass	=0
	Average Amplitude - Informative	CJTPAT	0	N.A	=0
	High Amplitude	CJTPAT	0	Pass	=0

Detailed Results

Includes detailed measurement information, as shown in the example. The Comments column is for test-specific information; for example, if the test had to be aborted, the reason is listed in the Comments column. To exclude this information from the report, clear the **Include Detailed Results** check box in the Reports panel before running the test.

Test Result	Limit	Execution Time	Comments
	=0		Test Aborted (Loopback Validation Failed)
	=0		Test Aborted (Loopback Validation Failed)
	=0	<1 Min	Test Aborted (Loopback Validation Failed)

Test Result	Limit	Execution Time	Comments
Pass	=0	4 Min	
N.A	=0		
Pass	=0		
Pass	=0	4 Min	
N.A	=0		
Pass	=0		

Application Configuration

This table lists general, common, and acquired parameters used in the test. General parameters are those selected to be included in all tests (where applicable). General parameters are set in the DUT tab. By default, this is not selected for inclusion in the test. To include this information in the report, select the **Include Complete Application Configuration** check box before running the

General Parameters	
Common Parameters	
Parameter Name	Parameter Value
TestPattern	CJTPAT
8B10B Pattern Encoding	Yes
Gear Selection for HS	Gear 1
HS Gear 1	Include
HS Gear 2	Exclude
HS Gear 3	Exclude
Data Rate - Gear 1 A (Mbps)	1248
Data Rate - Gear 1 B (Mbps)	1457.6
Loopback Initialization by	Manual
Loopback initialization required	Only once
Loopback validation required	Never
Scope Channel for Trigger	CH1
Signal Generator Channel Amplitude (Vpp)	0.8
Waveform Library Location	C:\Tektronix\TekExpress\MIP1 M-PHY Rx
Waveform Creation Option	Use Pre-created Waveforms
Save Created Waveform	Never
Compliance Test	Yes
DUT Mode	HS
Gear Type	A only
Number of retries for instrument IO errors	3
Time between retries (seconds)	20
DUTID Comment	Type Your Comments
Auto increment report name if duplicate	TRUE
Include Pass/Fail Results Summary	TRUE
Include Detailed Results	TRUE
Include Setup Configuration	TRUE
Include Complete Application Configuration	TRUE
Include User Comments	TRUE

test.


No Acquire Instruments	
No Acquire Utilities	
Analyze Instrumentation	
Parameter Name	Parameter Value
Signal Generator	GPIB0::4::INSTR
Error Detector	GPIB8::1::INSTR
No Analyze Utilities	
2.1.3 – Common-Mode Input Voltage Tolerance (VCM-RX)	
No Acquire Parameters	
Analyze Parameters	
Parameter Name	Parameter Value
Common mode amplitude - Low (V)	0.025
Select Common mode amplitude - Low	Include
Common mode amplitude - Avg (V)	0.18
Select Common mode amplitude - Avg	Include
Common mode amplitude - High (V)	0.325
Select Common mode amplitude - High	Include
Differential amplitude - High (V)	0.244
Differential amplitude - Low (V)	0.08
Test time duration (sec)	10
HS_PREPARE_length	30
STALL Length	30
AWG DC Output Common mode amplitude - Low (V)	0.025
AWG DC Output Common mode amplitude - Avg (V)	0.18
AWG DC Output Common mode amplitude - High (V)	0.325

See Also

- [View Test Results \(see page 47\)](#)
- [View Test-Related Files \(see page 53\)](#)

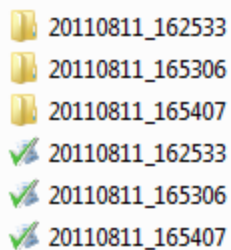
View Test-Related Files

Files related to tests are stored in the M-PHY RX folder under the My TekExpress shared folder. In the M-PHY RX folder, each test setup has a test setup file and a test setup folder, both with the test setup name. The test setup file is preceded by the M-PHY Rx icon and usually has no visible file extension.

For example,  Diff_InputVoltAmpTol.

Inside the test setup folder is another folder named for the DUT ID used in the test sessions. The default is DUT001.

Inside the DUT001 folder are the session folders and files. Each session has a folder and file pair, both named for the test session using the naming convention (date)_(time). Each session file is stored outside its matching session folder.



The first time you run a new, unsaved session, the session files are stored in the Untitled Session folder located at . . \My TekExpress\M-PHY RX. When you name and save the session, the files are placed in a folder with the name that you specify.

See Also

- [File Name Extensions \(see page 11\)](#)

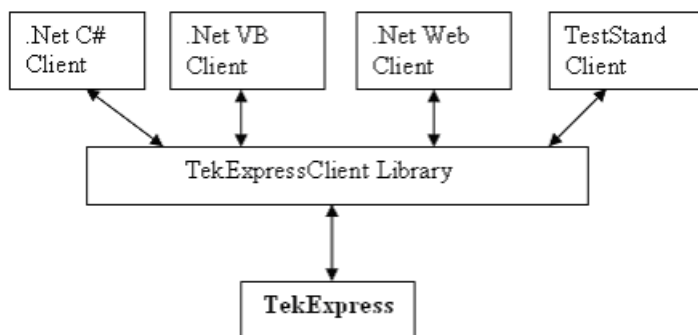
About the Programmatic Interface

The Programmatic interface allows you to seamlessly integrate the TekExpress test automation application with the high-level automation layer. This also allows you to control the state of the TekExpress application running on a local or a remote computer.

For simplifying the descriptions, the following terminologies are used in this section:

- **TekExpress Client:** A high-level automation application that communicates with TekExpress using TekExpress Programmatic Interface.
- **TekExpress Server:** The TekExpress application when being controlled by TekExpress Client.

TekExpress leverages .Net Marshalling to enable the Programmatic Interface for TekExpress Client. TekExpress provides a client library for TekExpress clients to use the programmatic interface. The TekExpress client library is inherited from .Net MarshalByRef class to provide the proxy object for the clients. The TekExpress client library maintains a reference to the TekExpress Server and this reference allows the client to control the server state.



See Also

- [Requirements for Developing TekExpress Client \(see page 56\)](#)
- [Remote Proxy Object \(see page 57\)](#)
- [Client Proxy Object \(see page 57\)](#)

Requirements for Developing TekExpress Client

While developing the TekExpress Client, use the TekExpressClient.dll. The client can be a VB .Net, C# .Net, TestStand or Web application. The examples for interfaces in each of these applications are in the samples folder.

References Required

- *TekExpressClient.dll* has an internal reference to *Iidlglib.dll* and *IRemoteInterface.dll*.
- *Iidlglib.dll* has a reference to *TekDotNetLib.dll*.
- *IRemoteInterface.dll* provides the interfaces required to perform the remote automations. It is an interface that forms the communication line between the server and the client.
- *Iidlglib.dll* provides the methods to generate and direct the secondary dialog messages at the client-end.

NOTE. *The end-user client application does not need any reference to the above mentioned DLL files. It is essential to have these DLLs (IRemoteInterface.dll, Iidlglib.dll and TekDotNetLib.dll) in the same folder as that of TekExpressClient.dll.*

Required Steps for a Client

The following steps are used by the client to programmatically control the server using TekExpressClient.dll:

Develop a client UI to access the interfaces exposed through the server. This client loads TekExpressClient.dll to access the interfaces. After TekExpressClient.dll is loaded, the client UI can call the specific functions to run the operations requested by the client. When the client is up and running, it does the following to run a remote operation:

1. To connect to the server, the client provides the IP address of the PC where the server is running.
2. The client locks the server application to avoid conflict with any other Client that may try to control the server simultaneously. “Lock” would also disable all user controls on the server so that server state cannot be changed by manual operation.

If any other client tries to access a server that is locked, it will receive a notification that the server is locked by another client.

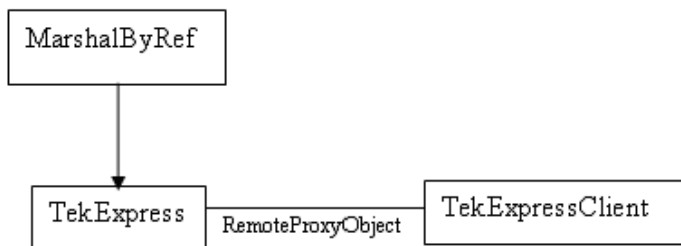
3. When the client has connected to and locked the server, the client can access any of the programmatic controls needed to run the remote automations.
4. After the client operations finish, the client unlocks the server.

See Also

- [About M-PHY Rx Application Commands \(see page 65\)](#)

Remote Proxy Object

The server exposes a remote object to let the remote client access and perform the server-side operations remotely. The proxy object is instantiated and exposed at the server-end through marshalling.



The following is an example:

```
RemotingConfiguration.RegisterWellKnownServiceType (typeof (TekExpressRemoteInterface), "TekExpress Remote interface", WellKnownObjectMode.Singleton);
```

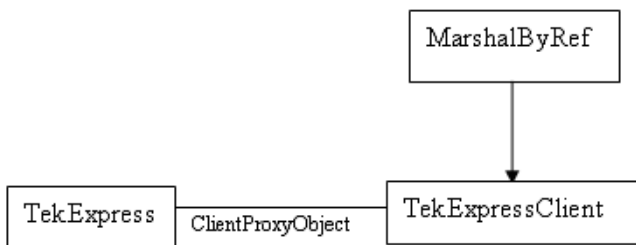
This object lets the remote client access the interfaces exposed at the server side. The client gets the reference to this object when the client gets connected to the server.

For example,

```
//Get a reference to the remote object  
remoteObject = (IRemoteInterface)Activator.GetObject(typeof(IRemoteInterface), URL.ToString());
```

Client Proxy Object

Client exposes a proxy object to receive certain information.



For example,

```
//Register the client proxy object
wellKnownServiceTypeEntry[] e = RemotingConfiguration.GetRegisteredWellKnownServiceTypes();

clientInterface = new ClientInterface();

RemotingConfiguration.RegisterWellKnownServiceType(typeof(ClientInterface),
"Remote Client Interface", wellKnownObjectMode.Singleton);

//Expose the client proxy object through marshalling
RemotingServices.Marshal(clientInterface, "Remote Client Inteface");
```

The client proxy object is used for the following:

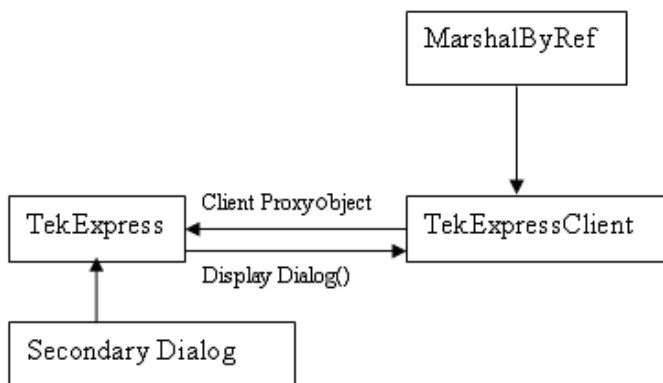
- To get the secondary dialog messages from the server.
- To get the file transfer commands from the server while transferring the report.

Examples

```
clientObject.clientIntf.DisplayDialog(caption, msg, iconType, btnType);
clientObject.clientIntf.TransferBytes(buffer, read, fileLength);
```

For more information, click the following links:

[Secondary Dialog Message Handling](#)



The secondary dialog messages from the Secondary Dialog library are redirected to the client-end when a client is performing the automations at the remote end.

In the secondary dialog library, the assembly that is calling for the dialog box to be displayed is checked and if a remote connection is detected, the messages are directed to the remote end.

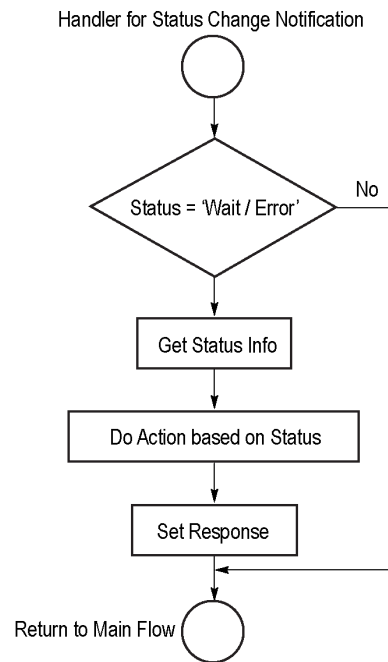
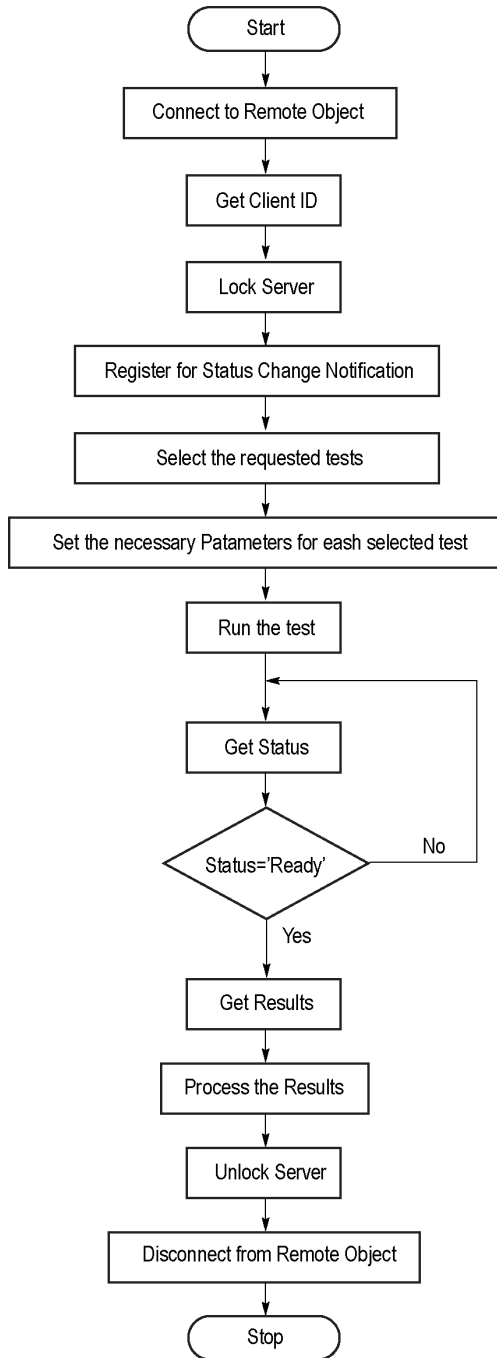
File Transfer Events

When the client requests the transfer of the report, the server reads the report and transfers the file by calling the file transfer methods at the client-end.

Client Programmatic Interface Example

An example of the client programmatic interface is described and shown as follows:

Process flowchart



0043-001

1. Connect to a server or remote object using the programmatic interface provided.
2. Get the client ID that is created when connecting to the remote object. This client ID is one of the required parameters to communicate with the server.

NOTE. The server identifies the client with this ID only and rejects any request if the ID is invalid.

3. Lock the server for further operations. This disables the application interface.

NOTE. *You can get values from the server or set values from the server to the client only if the application is locked.*

4. Register for receiving notifications on status change events on the server. To register you need to give a handler as a parameter. For details, see [Handler of Status Change Notification \(see page 61\)](#).

NOTE. *Whenever there is a change in the status of the server, all the clients registered with the server receive a notification from the server.*

5. Select the tests to run through the programmatic interface.
6. Set the necessary parameters for each test.
7. Run the tests.
8. Poll for the status of the application.

NOTE. *Skip this step if you are registered for the status change notification and the status is Ready.*

9. After completing the tests, get the results.
10. Create a report or display the results and verify or process the results.
11. Unlock the server after completing all the tasks.
12. Disconnect from the remote object.

Handler of Status Change Notification

1. Get the status. If the status is Wait or Error, get the information that contains the title, message description, and the expected responses for the status.
2. Perform the actions based on the status information.
3. Set the response as expected.

See Also

- [About M-PHY Rx Application Commands \(see page 65\)](#)
- [Program Example \(see page 62\)](#)

Program Example

This program example shows how to communicate between a PC and TekExpress M-PHY Rx remotely.

A typical application does the following:

1. Start the application.

2. Connect through an IP address.

```
m_Client.Connect("localhost") 'True or False
```

```
clientID = m_Client.getClientID
```

3. Lock the server.

```
m_Client.LockServer(clientID)
```

4. Disable the Popups.

```
m_Client.SetVerboseMode(clientID, false)
```

5. Set the Dut ID.

```
m_Client.SetDutId(clientID, "DUT_Name")
```

6. Select a measurement.

```
mClient.SelectTest(clientID, "M-PHY", "Receiver", "2.1.1 - Differential  
Input Voltage Amplitude Tolerance (VDIF-RX)", true)
```

7. Select a channel.

```
mClient.SetGeneralParameter(clientID, "M-PHY", "Receiver", "2.1.1 -  
Differential Input Voltage Amplitude Tolerance (VDIF-RX)", "Scope Channel  
for Trigger$CH1")
```

8. Configure the selected measurement.

```
mClient.SetGeneralParameter(clientID, "M-PHY", "Receiver", "2.1.1 -  
Differential Input Voltage Amplitude Tolerance (VDIF-RX)", Loopback  
validation required$Always)'Set loopback validation to Always
```

```
mClient.SetGeneralParameter(clientID, "M-PHY", "Receiver", "2.1.1 -  
Differential Input Voltage Amplitude Tolerance (VDIF-RX)", Gear Type$Both A  
and B)'Set Gear type to both A and B
```

9. Run with set configurations.

```
m_Client.Run(clientID)
```

10. Wait for the test to complete.

```
Do
```

```
Thread.Sleep(500)
```

```
m_Client.Application_Status(clientID)
Select Case status
Case "wait"
'Get the Current State Information
mClient.GetCurrentStateInfo(clientID, waitingMsBxCaption, waitingMsBxMes-
sage, waitingMsBxButtontexts)
'Send the Response
mClient.SendResponse(clientID, waitingMsBxCaption, waitingMsBxMessage,
waitingMsBxResponse)
End Select
Loop Until status = "Ready"
```

11. After the Test is Complete.

```
'Save all results values from folder for current run
m_Client.TransferResult(clientID, logDirname)
```

12. Unlock the server.

```
m_Client.UnlockServer(clientID)
```

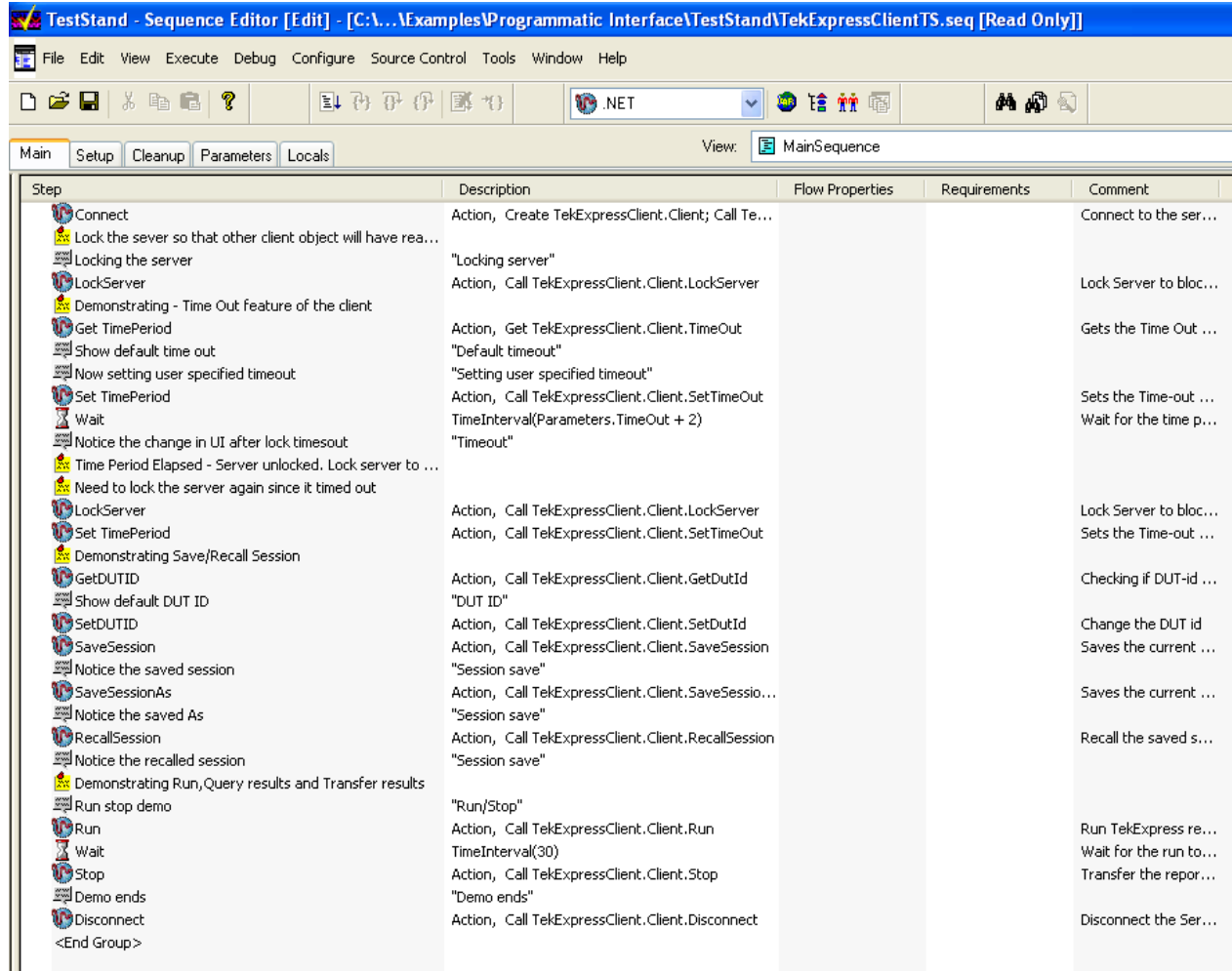
13. Disconnect from server.

```
m_Client.Disconnect()
```

14. Exit the application.

NI TestStand Client Example

The following is an example for NI TestStand Client available in the path: C:\Program Files\Tektronix\TekExpress\TekExpress M-PHY RX\Examples\Programmatic Interface\TestStand



About M-PHY Rx Application Commands

Click a client action below to see the command name, description, parameters, return value, and an example, associated with the action.

[Connect through an IP address \(see page 70\)](#)

[Lock the server \(see page 71\)](#)

[Disable the popups \(see page 72\)](#)

[Set or get the DUT ID \(see page 73\)](#)

[Set the configuration parameters for a suite or measurement \(see page 74\)](#)

[Query the configuration parameters for a suite or measurement \(see page 76\)](#)

[Select a measurement \(see page 77\)](#)

[Select a suite \(see page 78\)](#)

[Select a channel \(see page 79\)](#)

[Configure the selected measurement \(see page 80\)](#)

[Run with set configurations or stop the run operation \(see page 81\)](#)

[Handle Error Codes \(see page 82\)](#)

[Get or set the timeout value \(see page 83\)](#)

[Wait for the test to complete \(see page 84\)](#)

[After the test is complete \(see page 86\)](#)

[Save, recall, or check if a session is saved \(see page 89\)](#)

[Unlock the server \(see page 90\)](#)

[Disconnect from server \(see page 90\)](#)

string id			
Name	Type	Direction	Description
id	string	IN	Identifier of the client performing the remote function
Ready: Test configured and ready to start			
Running: Test running			
Paused: Test paused			
Wait: A popup that needs your inputs			
Error: An error is occurred			

string dutName			
Name	Type	Direction	Description
dutName	string	IN	The new DUT ID of the setup

out bool saved			
Name	Type	Direction	Description
saved	bool	OUT	Boolean representing whether the current session is saved

This parameter is used as a check in SaveSession() and SaveSessionAs() functions.

string ipAddress			
Name	Type	Direction	Description
ipAddress	string	IN	The ip address of the server to which the client is trying to connect. This is required to establish the connection between the server and the client.

out string clientID			
Name	Type	Direction	Description
clientid	string	OUT	Identifier of the client that is connected to the server clientid = unique number + ipaddress of the client. For example, 1065-192.157.98.70

NOTE. If the dutName parameter is null, the client is prompted to provide a valid DUT ID.

NOTE. The server must be active and running for the client to connect to the server. Any number of clients can be connected to the server at a time.

NOTE. When the client is disconnected, it is unlocked from the server and then disconnected. The id is reused.

string dutId			
Name	Type	Direction	Description
dutId	string	OUT	The DUT ID of the setup

The dutId parameter is set after the server processes the request.

string device			
Name	Type	Direction	Description
device	string	IN	Specifies the name of the device

string suite			
Name	Type	Direction	Description
suite	string	IN	Specifies the name of the suite

string test			
Name	Type	Direction	Description
test	string	IN	Specifies the name of the test to obtain the pass or fail status

string parameterString			
Name	Type	Direction	Description
parameterString	string	IN	Selects or deselects a test

int rowNr			
Name	Type	Direction	Description
rowNr	int	IN	Specifies the zero based row index of the sub-measurement for obtaining the result value

NOTE. When the client tries to lock a server that is locked by another client, the client gets a notification that the server is already locked and it must wait until the server is unlocked. If the client locks the server and is idle for a certain amount of time then the server is unlocked automatically from that client.

out string[] status

Name	Type	Direction	Description
status	string array	OUT	The list of status messages generated during the run

string name

Name	Type	Direction	Description
name	string	IN	The name of the session being recalled

The name parameter cannot be empty. If it is empty, the client is prompted to provide a valid name.

NOTE. *When the run is performed, the status of the run is updated periodically using a timer.*

string name

Name	Type	Direction	Description
name	string	IN	The name of the session being saved

The name parameter cannot be empty. If it is empty, the client is prompted to provide a valid name.

Once the session is saved under ‘name’ you cannot use this method to save the session in a different name. Use SaveSessionAs instead.

string name

Name	Type	Direction	Description
name	string	IN	The name of the session being recalled

The same session is saved under different names using this method. The name parameter cannot be empty. If it is empty, the client is prompted to provide a valid name.

bool isSelected

Name	Type	Direction	Description
isSelected	bool	IN	Selects or deselects a test

string time

Name	Type	Direction	Description
time	string	IN	The time in seconds that refers to the timeout period

The time parameter gives the timeout period, which is the time the client is allowed to be locked and idle. After the timeout period if the client is still idle, it gets unlocked.

The time parameter should be a positive integer; otherwise, the client is prompted to provide a valid timeout period.

bool_verbose

Name	Type	Direction	Description
_verbose	bool	IN	Specifies whether the verbose mode should be turned ON or OFF

NOTE. *When the session is stopped, the client is prompted to stop the session and is stopped at the consent.*

string filePath

Name	Type	Direction	Description
filePath	string	IN	The location where the report must be saved in the client

NOTE. *If the client does not provide the location to save the report, the report is saved at C:\ProgramFiles.*

NOTE. *When the client is disconnected, the client is unlocked automatically.*

out string caption

Name	Type	Direction	Description
caption	string	OUT	The wait state or error state message sent to you

out string message			
Name	Type	Direction	Description
message	string	OUT	The wait state/error state message sent to you

out string[] buttonTexts			
Name	Type	Direction	Description
buttonTexts	string array	OUT	An array of strings containing the possible response types that you can send

string response			
Name	Type	Direction	Description
response	string	IN	A string containing the response type that you can select (it must be one of the strings in the string array buttonTexts)

out string clientID			
Name	Type	Direction	Description
clientID	string	OUT	Identifier of the client that is connected to the server clientID = unique number + ipaddress of the client. For example, 1065-192.157.98.70

Connect Through an IP Address

Command name	Parameters	Description	Return value	Example
Connect()	string ipAddress (see page 66) out string clientID (see page 66)	This method connects the client to the server. Note (see page 66) The client provides the IP address to connect to the server. The server provides a unique client identification number when connected to it.	Return value is either True or False	<pre>m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as boolean returnval = m_Client.Connect(ipaddress,m_clientID)</pre>

NOTE. *The Fail condition for PI commands occurs in any of the following cases:*

The server is LOCKED and the message displayed is "Server is locked by another client".

The session is UNLOCKED and the message displayed is "Lock Session to execute the command".

The server is NOTFOUND and the message displayed is "Server not found...Disconnect!".

When none of these fail conditions occur, then the message displayed is "Failed...".

Lock the Server

Command name	Parameters	Description	Return value	Example
LockSession()	string clientID (see page 70)	This method locks the server. Note (see page 67) The client must call this method before running any of the remote automations. The server can be locked by only one client.	String value that gives the status of the operation after it has been performed The return value is "Session Locked..." on success.	m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string returnval = m_Client.LockServer(clientID)

NOTE. *The Fail condition for PI commands occurs in any of the following cases:*

The server is LOCKED and the message displayed is "Server is locked by another client".

The session is UNLOCKED and the message displayed is "Lock Session to execute the command".

The server is NOTFOUND and the message displayed is "Server not found...Disconnect!".

When none of these fail conditions occur, then the message displayed is "Failed...".

Disable the Popups

Command name	Parameters	Description	Return value	Example
SetVerboseMode()	string clientID (see page 70) bool verbose (see page 69)	This method sets the verbose mode to either true or false. When the value is set to true, any message boxes that appear during the application will be routed to the client machine that is controlling TekExpress. When the value is set to false, all the message boxes are shown on the server machine.	String that gives the status of the operation after it has been performed When Verbose mode is set to true, the return value is "Verbose mode turned on. All dialog boxes will be shown to client". When Verbose mode is set to false, the return value is "Verbose mode turned off. All dialog boxes will be shown to server".	<code>m_Client = new Client()</code> <code>//m_Client is a reference to the Client class in the Client DLL.</code> returnval as string Verbose mode is turned on <code>return=m_Client.SetVerboseMode(clientID, true)</code> Verbose mode is turned off <code>returnval=m_Client.SetVerboseMode(clientID, false)</code>

NOTE. *The Fail condition for PI commands occurs in any of the following cases:*

The server is LOCKED and the message displayed is "Server is locked by another client".

The session is UNLOCKED and the message displayed is "Lock Session to execute the command".

The server is NOTFOUND and the message displayed is "Server not found...Disconnect!".

When none of these fail conditions occur, then the message displayed is "Failed...".

Set or Get the DUT ID

Command name	Parameters	Description	Return value	Example
SetDutId()	string clientID (see page 70) string dutName (see page 66)	This method changes the DUT ID of the setup. The client must provide a valid DUT ID.	String that gives the status of the operation after it has been performed Return value is "DUT Id Changed" on success	<pre>m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string return=m_Client.SetDutId(clientID,desiredDutId) Note (see page 66)</pre>
GetDutId()	string clientID (see page 70) string dutId (see page 67)	This method gets the DUT ID of the current setup.	String that gives the status of the operation after it has been performed	<pre>m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string return=m_Client.GetDutid(clientID, out DutId)</pre>

NOTE. *The Fail condition for PI commands occurs in any of the following cases:*

The server is LOCKED and the message displayed is "Server is locked by another client".

The session is UNLOCKED and the message displayed is "Lock Session to execute the command".

The server is NOTFOUND and the message displayed is "Server not found...Disconnect!".

When none of these fail conditions occur, then the message displayed is "Failed...".

Set the Configuration Parameters for a Suite or Measurement

Command name	Parameters	Description	Return value	Example
SetGeneralParameter	string clientID (see page 70) string device (see page 67) string suite (see page 67) string test (see page 67) string parameterString (see page 67)	This method sets the number of video lanes for the selected measurement. NOTE. Use this command to select a lane, channel, or source type.	String that gives the status of the operation after it has been performed The return value is "" (an empty String) on success.	<pre>m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string Select Channel (see page 75) Select Measurement Method (see page 75)</pre>
SetAnalyzeParameter()	string clientID (see page 70) string device (see page 67) string suite (see page 67) string test (see page 67) string parameterString (see page 67)	This method sets the configuration parameters in the Analyze panel of the Configuration Panel dialog box for a given suite or measurement.	The return value is "" (an empty String) on success.	<pre>m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL . returnval as string</pre>
SetAcquireParameter()	string clientID (see page 70) string device (see page 67) string suite (see page 67) string test (see page 67) string parameterString (see page 67)	This method sets the configuration parameters in the Acquire panel of the Configuration Panel dialog box for a given suite or measurement.	<pre>returnVal = remoteObject.SetAcquireParameter(id, device, suite, test, parameterString) if ((OP_STATUS) returnVal != OP_STATUS.SUCCESS) return CommandFailed(returnVal)</pre>	

NOTE. *The Fail condition for PI commands occurs in any of the following cases:*

The server is LOCKED and the message displayed is "Server is locked by another client".

The session is UNLOCKED and the message displayed is "Lock Session to execute the command".

The server is NOTFOUND and the message displayed is "Server not found...Disconnect!".

When none of these fail conditions occur, then the message displayed is "Failed...".

Select Channel Example

```
returnval=mClient.SetGeneralParameter(clientID, "M-PHY", "Receiver", "2.1.1 – Differential Input  
Voltage Amplitude Tolerance (VDIF-RX)", "Scope Channel for Trigger$CH1")
```

Select Loopback Validation Method Example

```
returnval=mClient.SetGeneralParameter(clientID, "M-PHY", "Receiver", "2.1.1 – Differential Input  
Voltage Amplitude Tolerance (VDIF-RX)", Loopback validation required$Always)
```

Query the Configuration Parameters for a Suite or Measurement

Command name	Parameters	Description	Return value	Example
GetGeneralParameter()	string clientID (see page 70) string device (see page 67) string suite (see page 67) string test (see page 67) string parameterString (see page 67)	This method gets the general configuration parameters for a given suite or measurement.	The return value is the general configuration parameter for a given suite or measurement that is set.	<pre>m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string</pre> Query Channel (see page 77) Query Loopback Validation Method (see page 77)
GetAnalyzeParameter()	string clientID (see page 70) string device (see page 67) string suite (see page 67) string test (see page 67) string parameterString (see page 67)	This method gets the configuration parameters set in the Analyze panel of the Configuration Panel dialog box for a given suite or measurement.	The return value is the configuration parameter set in the Analyze panel of the Configuration Panel dialog box for a given suite or measurement.	<pre>m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL . returnval as string</pre>
GetAcquireParameter()	string clientID (see page 70) string device (see page 67) string suite (see page 67) string test (see page 67) string parameterString (see page 67)	This method gets the configuration parameters set in the Acquire panel for a given suite or measurement.	The return value is the configuration parameter set in the Acquire panel for a given suite or measurement.	

NOTE. The Fail condition for PI commands occurs in any of the following cases:

The server is LOCKED and the message displayed is "Server is locked by another client".

The session is UNLOCKED and the message displayed is "Lock Session to execute the command".

The server is NOTFOUND and the message displayed is "Server not found...Disconnect!".

When none of these fail conditions occur, then the message displayed is "Failed...".

Query Channel for Trigger Example

```
returnval = mClient.GetGeneralParameter(clientID, "M-PHY", "Receiver", "2.1.1 – Differential Input Voltage Amplitude Tolerance (VDIF-RX)", "Scope Channel for Trigger")
```

Query Loopback Validation Method Example

```
returnval=mClient.GetGeneralParameter(clientID, "M-PHY", "Receiver", "2.1.1 – Differential Input Voltage Amplitude Tolerance (VDIF-RX)", Loopback validation required)
```

Select a Measurement

Command name	Parameters	Description	Return value	Example
SelectTest()	string clientID (see page 70) string device (see page 67) string suite (see page 67) string test (see page 67) bool isSelected (see page 68)	<p>This method selects or deselects a given test.</p> <p>If this Setting parameter is set to true, you can select a measurement.</p> <p>If this Setting parameter is set to false, you can deselect a measurement.</p>	<p>String that displays the status of the operation after it has been performed</p> <p>The return value is "" (an empty String) on success.</p>	<pre>m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL returnval as string Select Measurement 2.1.1 – Differential Input Voltage Amplitude Tolerance (VDIF-RX): returnval = mClient.SelectTest(clientID, "M-PHY", "Receiver", "2.1.1 – Differential Input Voltage Amplitude Tolerance (VDIF-RX)", true)</pre>

NOTE. *The Fail condition for PI commands occurs in any of the following cases:*

The server is LOCKED and the message displayed is "Server is locked by another client".

The session is UNLOCKED and the message displayed is "Lock Session to execute the command".

The server is NOTFOUND and the message displayed is "Server not found...Disconnect!".

When none of these fail conditions occur, then the message displayed is "Failed...".

Select a Suite

Command name	Parameters	Description	Return value	Example
SelectSuite()	string clientID (see page 70) string device (see page 67) string suite (see page 67) bool isSelected (see page 68)	<p>This method selects or deselects a given suite.</p> <p>When this parameter is set to true, you can select a suite.</p> <p>When this parameter is set to false, you can deselect a suite.</p>	<p>String that gives the status of the operation after it has been performed</p> <p>The return value is "" (an empty String) on success.</p>	<pre>m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string Select Suite (Default): returnval=mClient.SelectSuite(clientID, "M-PHY", "Receiver", true)</pre>

NOTE. The Fail condition for PI commands occurs in any of the following cases:

The server is LOCKED and the message displayed is "Server is locked by another client".

The session is UNLOCKED and the message displayed is "Lock Session to execute the command".

The server is NOTFOUND and the message displayed is "Server not found...Disconnect!".

When none of these fail conditions occur, then the message displayed is "Failed...".

Select a Channel

Command name	Parameters	Description	Return value	Example
SetGeneralParameter()	string clientID (see page 70) string device (see page 67) string suite (see page 67) string test (see page 67) string parameterString (see page 67)	This method sets the parameters that are not specific to any given test. NOTE. Using this command we can select a lane, channel, or source type.	String that gives the status of the operation after it has been performed The return value is "" (an empty String) on success.	<pre>m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string</pre> Select Channel for Trigger (see page 80) Select Loopback Validation Method (see page 80)
SetAnalyzeParameter()	string clientID (see page 70) string device (see page 67) string suite (see page 67) string test (see page 67) string parameterString (see page 67)	This method sets the configuration parameters in the Analyze panel of the Configuration Panel dialog box for a given suite or measurement.	The return value is "" (an empty String) on success.	<pre>m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string</pre>
SetAcquireParameter()	string clientID (see page 70) string device (see page 67) string suite (see page 67) string test (see page 67) string parameterString (see page 67)	This method sets the configuration parameters in the Acquire panel of the Configuration Panel dialog box for a given suite or measurement.	<pre>returnVal = remoteObject.SetAcquireParameter(id, device, suite, test, parameterString) if ((OP_STATUS) returnVal != OP_STATUS.SUCCESS) return CommandFailed(returnVal)</pre>	

NOTE. *The Fail condition for PI commands occurs in any of the following cases:*
The server is LOCKED and the message displayed is "Server is locked by another client".
The session is UNLOCKED and the message displayed is "Lock Session to execute the command".
The server is NOTFOUND and the message displayed is "Server not found...Disconnect!".
When none of these fail conditions occur, then the message displayed is "Failed...".

Select Channel for Trigger Example

```
returnval = mClient.SetGeneralParameter(clientID, "M-PHY", "Receiver", "2.1.1 – Differential Input Voltage Amplitude Tolerance (VDIF-RX)", "Scope Channel for Trigger$CH1")
```

Select Loopback Validation Method Example

```
returnval=mClient.SetGeneralParameter(clientID, "M-PHY", "Receiver", "2.1.1 – Differential Input Voltage Amplitude Tolerance (VDIF-RX)", Loopback validation required$Always)
```

Configure the Selected Measurement

Command name	Parameters	Description	Return value	Example
SetAnalyzeParameter()	string clientID (see page 70) string device (see page 67) string suite (see page 67) string test (see page 67) string parameterString (see page 67)	This method sets the Analyze parameters (Configuration parameters) for a given test.	The return value is "" (an empty String) on success.	<pre>m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string</pre>

NOTE. *The Fail condition for PI commands occurs in any of the following cases:*
The server is LOCKED and the message displayed is "Server is locked by another client".
The session is UNLOCKED and the message displayed is "Lock Session to execute the command".
The server is NOTFOUND and the message displayed is "Server not found...Disconnect!".
When none of these fail conditions occur, then the message displayed is "Failed...".

Run with Set Configurations or Stop the Run Operation

Command name	Parameters	Description	Return value	Example
Run()	string clientID (see page 70)	Runs the selected tests Note (see page 68) After the server is set up and configured, run it remotely using this function.	String that gives the status of the operation after it has been performed. The return value is "Run started..." on success.	m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string returnval=m_Client.Run(clientID)
Stop()	string clientID (see page 70)	Stops the currently running tests Note (see page 69)	String that gives the status of the operation after it has been performed The return value is "Stopped..." on success.	m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string returnval=m_Client.Stop(clientID)

NOTE. The Fail condition for PI commands occurs in any of the following cases:

The server is LOCKED and the message displayed is "Server is locked by another client".

The session is UNLOCKED and the message displayed is "Lock Session to execute the command".

The server is NOTFOUND and the message displayed is "Server not found...Disconnect!".

When none of these fail conditions occur, then the message displayed is "Failed...".

Handle Error Codes

The return value of the remote automations at the server-end is OP_STATUS, which changes to a string value depending on its code, and returned to the client. The values of OP_STATUS are as follows:

Value	Code	Description
FAIL	-1	The operation failed.
SUCCESS	1	The operation succeeded.
NOTFOUND	2	Server not found
LOCKED	3	The server is locked by another client, so the operation cannot be performed.
UNLOCK	4	The server is not locked. Lock the server before performing the operation.
NULL	0	Nothing

NOTE. The Fail condition for PI commands occurs in any of the following cases:

The server is LOCKED and the message displayed is "Server is locked by another client".

The session is UNLOCKED and the message displayed is "Lock Session to execute the command".

The server is NOTFOUND and the message displayed is "Server not found...Disconnect!".

When none of these fail conditions occur, then the message displayed is "Failed...".

Get or Set the Timeout Value

Command name	Parameters	Description	Return value	Example
GetTimeOut()	string clientID (see page 70)	Returns the current timeout period set by the client	String that gives the status of the operation after it has been performed The default return value is 1800000.	m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string returnval=m_Client.GetTimeOut()
SetTimeOut()	string clientID (see page 70) string time (see page 69)	Sets a timeout period specified by the client. After this timeout period expires, the server is unlocked automatically.	String that gives the status of the operation after it has been performed On success the return value is "TimeOut Period Changed".	m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string returnval=m_Client.SetTimeOut(clientID, desiredTimeOut)

NOTE. *The Fail condition for PI commands occurs in any of the following cases:*

The server is LOCKED and the message displayed is "Server is locked by another client".

The session is UNLOCKED and the message displayed is "Lock Session to execute the command".

The server is NOTFOUND and the message displayed is "Server not found...Disconnect!".

When none of these fail conditions occur, then the message displayed is "Failed...".

Wait for the Test to Complete

The commands in this group execute while tests are running. The `GetCurrentStateInfo()` and `SendResponse()` commands are executed when the application is running and in the wait state.

Command name	Parameters	Description	Return value	Example
<code>ApplicationStatus()</code>	string clientID (see page 70)	This method gets the status of the server application. The states at a given time are Ready , Running , Paused , Wait , or Error . (see page 65)	String value that gives the status of the server application	<pre>m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string returnval=m_Client.ApplicationStatus(clientID)</pre>
<code>QueryStatus()</code>	string clientID (see page 70) out string[] status (see page 68)	An interface for the user to transfer Analyze panel status messages from the server to the client	String that gives the status of the operation after it has been performed On success the return value is "Transferred...".	<pre>m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string returnVal=m_Client.QueryStatus(clientID, out statusMessages) if ((OP_STATUS)returnVal == OP_STATUS.SUCCESS) return "Status updated..." else return CommandFailed(returnVal)</pre>

Command name	Parameters	Description	Return value	Example
GetCurrentState-Info() NOTE. This command is used when the application is running and is in the wait or error state.	string clientID (see page 70) out string caption (see page 69) out string message (see page 70) out string[] buttonTexts (see page 70)	This method gets the additional information of the states when the application is in Wait or Error state. Except client ID, all the others are Out parameters.	This command does not return any value. This function populates the Out parameters that are passed when invoking this function.	<pre>m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL m_Client.GetCurrentState-Info(clientID, caption,message, buttonTexts)</pre>
SendResponse() NOTE. This command is used when the application is running and is in the wait or error state.	string clientID (see page 70) out string caption (see page 69) out string message (see page 70) string response (see page 70)	After receiving the additional information using the method GetCurrentStateInfo(), the client can decide which response to send and then send the response to the application using this function. The response should be one of the strings that was received earlier as a string array in the GetCurrentStateInfo function. The _caption and _message should match the information received earlier in the GetCurrentStateInfo function.	This command does not return any value.	<pre>m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL m_Client.SendResponse(clientID, caption,message, response)</pre>

NOTE. The Fail condition for PI commands occurs in any of the following cases:

The server is LOCKED and the message displayed is "Server is locked by another client".

The session is UNLOCKED and the message displayed is "Lock Session to execute the command".

The server is NOTFOUND and the message displayed is "Server not found...Disconnect!".

When none of these fail conditions occur, then the message displayed is "Failed...".

After the Test is Complete

Command name	Parameters	Description	Return value	Example
GetPassFailStatus()	string clientID (see page 70) string device (see page 67) string suite (see page 67) string test (see page 67)	This method gets the pass or fail status of the measurement after test completion. NOTE. <i>Execute this command after completing the measurement.</i>	String that gives the status of the operation after it has been performed Returns the pass or fail status in the form of a string	<pre>m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string returnval=m_Client.GetPassFailStatus(clientID, device, suite, "2.1.1 – Differential Input Voltage Amplitude Tolerance (VDIF-RX)") //Pass or Fail</pre>
GetResultsValue()	string clientID (see page 70) string device (see page 67) string suite (see page 67) string test (see page 67) string parameterString (see page 67)	This method gets the result values of the measurement after the run.	String that gives the status of the operation after it has been performed Returns the result value in the form of a string	<pre>m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string returnval=m_Client.GetResultsValue(clientID,"M-PHY", "Receiver", "2.1.1 – Differential Input Voltage Amplitude Tolerance (VDIF-RX)", "Bit Error")</pre>
GetResultsValueForSubMeasurements()	string clientID (see page 70) string device (see page 67) string suite (see page 67) string test (see page 67) string parameterString (see page 67) int rowNr (see page 67)	This method gets the result values for individual sub-measurements after the run.	String that gives the status of the operation after it has been performed Returns the result value in the form of a string	<pre>m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string 2.1.1 – Differential Input Voltage Amplitude Tolerance (VDIF-RX) returnval=m_Client.GetResultsValueForSubMeasurements(clientID, "M-PHY", "Receiver", "2.1.1 – Differential Input Voltage Amplitude Tolerance (VDIF-RX)", "Bit Error", 0 returnval=m_Client.GetResultsValueForSubMeasurements(clientID, "M-PHY", "Receiver", "2.1.1 – Differential Input Voltage Amplitude Tolerance (VDIF-RX)", "Bit Error", 1</pre>

Command name	Parameters	Description	Return value	Example
GetReportParameter()	string clientID (see page 70) string device (see page 67) string suite (see page 67) string test (see page 67) string parameterString (see page 88)	<p>This method gets the general report details such as oscilloscope model, TekExpress version, and M-PHY Rx version.</p>	<p>The return value is the oscilloscope model, TekExpress version, and M-PHY Rx version.</p>	<pre>m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string</pre> <p>Oscilloscope Model</p> <pre>returnval=m_Client.GetReportParameter(clientID,"Scope Model")</pre> <p>TekExpress Version</p> <pre>returnval=m_Client.GetReportParameter(clientID,"TekExpress Version")</pre> <p>M-PHY Rx Version</p> <pre>returnval=m_Client.GetReportParameter(clientID,"Application Version")</pre>
TransferReport()	string clientID (see page 70) string filePath (see page 69)	<p>This method transfers the report generated after the run.</p> <p>The report contains the summary of the run.</p> <p>The client must provide the location where the report is to be saved at the client-end.</p>	<p>String that gives the status of the operation after it has been performed</p> <p>Transfers all the result values in the form of a string</p>	<pre>m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string</pre> <pre>returnval=m_Client.TransferReport(clientID,"C:\Report")</pre>
TransferImages()	string clientID (see page 70) od string filePath (see page 69)	<p>This method transfers all the images (screen shots) from the folder for the current run (for a given suite or measurement).</p> <p>NOTE. Every time you click Start, a folder is created in the X: drive. Transfer the waveforms before clicking Start .</p>	<p>String that gives the status of the operation after it has been performed</p> <p>Transfers all the images in the form of a string</p>	<pre>m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string</pre> <pre>returnval=m_Client.TransferImages(clientID, "C:\Waveforms")</pre>

NOTE. *The Fail condition for PI commands occurs in any of the following cases:*

The server is LOCKED and the message displayed is "Server is locked by another client".

The session is UNLOCKED and the message displayed is "Lock Session to execute the command".

The server is NOTFOUND and the message displayed is "Server not found...Disconnect!".

When none of these fail conditions occur, then the message displayed is "Failed...".

string parameterString			
Name	Type	Direction	Description
parameterString	string	IN	Specifies the oscilloscope model, TekExpress version, and M-PHY Rx version

Save, Recall, or Check if a Session is Saved

Command name	Parameters	Description	Return value	Example
CheckSession-Saved()	string clientID (see page 70) out bool saved (see page 66)	This method checks whether the current session is saved.	Return value is either True or False	<pre>m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string returnval=m_Client.CheckSessionSaved(m_clientID, out savedStatus)</pre>
RecallSession()	string clientID (see page 70) string name (see page 68)	Recalls a saved session. The client provides the session name.	String that gives the status of the operation after it has been performed The return value is "Session Recalled..."	<pre>m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string returnval=m_Client.RecallSession(clientID, savedSessionName)</pre>
SaveSession()	string clientID (see page 70) string name (see page 68)	Saves the current session. The client provides the session name.	String that gives the status of the operation after it has been performed The return value is "Session Saved..."/"Failed..."	<pre>m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string returnval=m_Client.SaveSession(clientID, desiredSessionName)</pre>
SaveSessionAs()	string clientID (see page 70) string name (see page 68)	Saves the current session under a different name every time this method is called. The client provides the session name.	String that gives the status of the operation after it has been performed The return value is "Session Saved..."	<pre>m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string returnval=m_Client.SaveSessionAs(clientID, desiredSessionName)</pre>

NOTE. The Fail condition for PI commands occurs in any of the following cases:

The server is LOCKED and the message displayed is "Server is locked by another client".

The session is UNLOCKED and the message displayed is "Lock Session to execute the command".

The server is NOTFOUND and the message displayed is "Server not found...Disconnect!".

When none of these fail conditions occur, then the message displayed is "Failed...".

Unlock the Server

Command name	Parameters	Description	Return value	Example
UnlockSession()	string clientID (see page 70)	This method unlocks the server from the client. The ID of the client to be unlocked must be provided. Note (see page 69)	String that gives the status of the operation after it has been performed The return value is "Session Un-Locked..."	m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string returnval=m_Client.Unlock-Server(clientID)

NOTE. The Fail condition for PI commands occurs in any of the following cases:

The server is LOCKED and the message displayed is "Server is locked by another client".

The session is UNLOCKED and the message displayed is "Lock Session to execute the command".

The server is NOTFOUND and the message displayed is "Server not found...Disconnect!".

When none of these fail conditions occur, then the message displayed is "Failed...".

Disconnect from the Server

Command name	Parameters	Description	Return value	Example
Disconnect()	string clientID (see page 70)	This method disconnects the client from the server. Note (see page 66)	Integer value that gives the status of the operation after it has been performed 1 for Success -1 for Failure	m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string returnval=m_Client.Disconnect(m_clientID)

NOTE. The Fail condition for PI commands occurs in any of the following cases:

The server is LOCKED and the message displayed is "Server is locked by another client".

The session is UNLOCKED and the message displayed is "Lock Session to execute the command".

The server is NOTFOUND and the message displayed is "Server not found...Disconnect!".

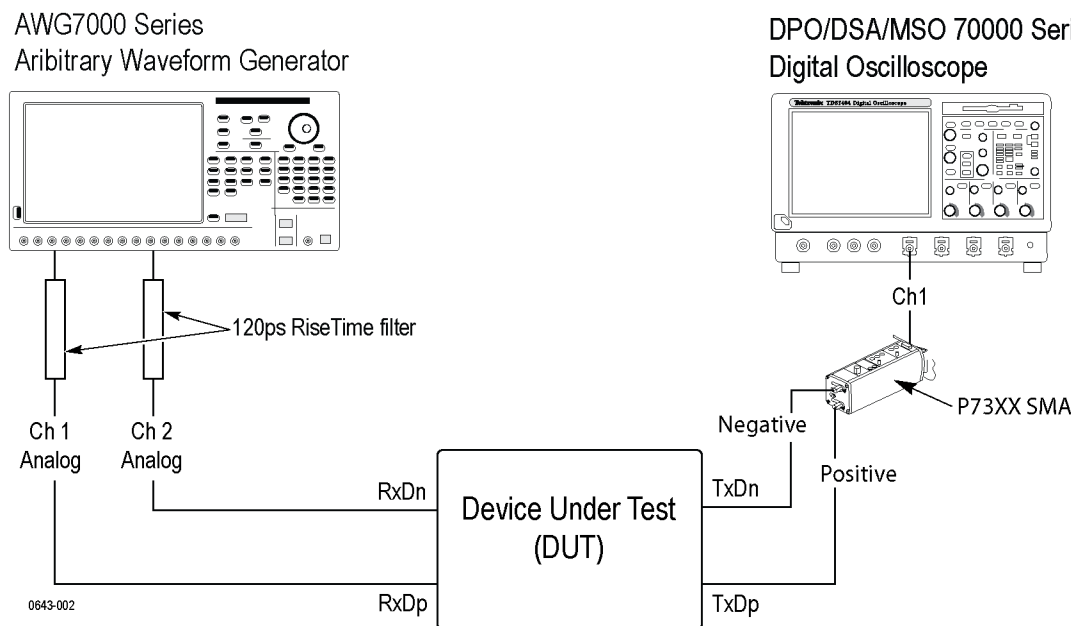
When none of these fail conditions occur, then the message displayed is "Failed...".

Receiver Testing Measurement Procedure

Tests are conducted with 8b/10b encoded continuous [CJTPAT](#).

The general procedure for executing receiver tests is as follows:

1. Connect CH1 and CH2 of the AWG to the RxDp and RxDn pins of the DUT using rise time filters and a matched pair of SMA cables. For guidance, see the connection diagram.



2. Connect the TxDp and TxDn pins of the DUT to the oscilloscope using a matched pair of SMA cables and a differential probe.
3. Set the DUT in loopback mode: Configure the DUT so that its Rx port and Tx port have the same data transfer mode, signaling type, gear, and drive strength settings.
4. For generating a signal using the AWG, generate the MIPI M-PHY single-ended signals with the characteristics given in the Signal Characteristics tables for each test.
5. After the oscilloscope synchronizes to the signal and loopback is validated, the TekScope error detector script runs. The script validates the received bits and reports the number of bit errors and the [BER](#).

See Also

- [Loopback Validation \(see page 93\)](#)

High-Speed Mode Measurements

The following are the parameters for high-speed mode measurements:

- Pattern Differential Amplitude: Low, Average, High
- Common-Mode Amplitude: Low, Average, High

NOTE. For tests that include the Average Common Mode Voltage or the Average Differential Voltage, you can omit the Average parameter by clearing the **Average** check box in the test Measurement configuration settings.

- 8b/10b encoded: YES or NO
- Sinusoidal Jitter:
 - Frequency: Absolute or range
 - Amplitude: Absolute value in terms of [UI](#)
- PREPARE length: Absolute value in terms of [SI](#)
- STALL length: Absolute value in terms of SI

For each test, these parameters are set to the default measurement values for Compliance Mode. These parameters cannot be changed in Compliance Mode. For specific parameters for each test, refer to the individual test topic.

See Also

- [Receiver Testing Measurement Procedure \(see page 91\)](#)

Unit Interval

Sinusoidal Jitter

Loopback Validation

Use this test to verify that DUT loopback has occurred successfully and that the DUT and the oscilloscope error detector are synchronized. Clean signals and with no impairments are used for this validation. If the DUT loopback passes, then the test solution executes tests.

Table 12: Characteristics of signal as input to the DUT from an AWG

Pattern	CJTPAT	CJTPAT
Data rate (Mb/s)	1248 (GEAR1-A)	1456 (GEAR1-B)
Differential voltage (mV)	Low Amp	60
	High Amp	244
Common-mode voltage (mV)	100	100
8b/10b encoded	Yes	Yes
Modulation	NRZ	NRZ

Table 13: Parameters

Parameter name	Value	Min	Max
Differential voltage – Low (V)	0.06	0.06	0.245
Differential voltage – Avg (V)	0.1	0.06	0.245
Differential voltage – High (V)	0.244	0.06	0.245
Common-mode voltage (V)	0.18	0.025	0.33
Test time duration (sec)	10	1	120
HS_PREPARE_length	30	0	15
STALL length	30	0	250

Table 14: Oscilloscope error detector settings

Parameter	Value
Error Detector Type	Bit
Error Detector Standard	ANY8B10B
Synchronization Pattern	K28.5

Table 14: Oscilloscope error detector settings (cont.)

Parameter	Value
Data Rate	Gear 1A/B
Error Limit	1e-10

See Also

- [Receiver Testing Measurement Procedure \(see page 91\)](#)

About the Oscilloscope Error Detector

Performing error detection using the oscilloscope requires the following:

- A multi-channel, programmable lab-grade signal source capable of generating appropriate High Speed and LP signaling.
- A [compatible oscilloscope \(see page 3\)](#) with ERRDET and STU options enabled.
- A DUT that supports loopback mode.
- Error Detector (ERRDT) and Serial Trigger Unit (STU) options enabled on the oscilloscope. If needed, [enable or verify that these options are enabled \(see page 106\)](#).

TekExpress M-PHY Rx uses the oscilloscope-based bit error detector. This bit error detector requires two input parameters: the bit sequence (SOF), which it uses for synchronization, and the bit length. Bit length specifies the number of bits that need to be stored in memory and later used for comparison and error detection. The synchronization pattern does not have to be at the beginning of the signal test pattern, just somewhere in the signal test pattern, because the fixed-length signal test pattern is sent repeatedly by the signal generator.

2.1.1 Differential Input Voltage Amplitude Tolerance (VDIF-RX) Test

This test verifies that the high-speed receiver is able to successfully receive high-speed signaling that meets the maximum and minimum conformance requirements for Differential Input Voltage Amplitude (V_{DIF_RX}). This measurement is performed for both large and small amplitude drive strengths.

Signal Characteristics

The following table outlines the characteristics of the signal feed as it is input to the DUT from the AWG.

Table 15: Signal characteristics for VDIF-RX

Pattern		CJTPAT	CJTPAT
Data rate (Mb/s)		1248 (GEAR1-A)	1456 (GEAR1-B)
Differential Voltage (mV)	Low Amp	60	60
	Ave. Amp	100	100
	High Amp	244	244
8b/10b encoded		Yes	Yes
Modulation		NRZ	NRZ

Measurement Parameters

The following table lists parameters for the test. Unless specified, these parameters cannot be configured in Compliance Mode.

Table 16: Parameters for VDIF-RX

Parameter name	Value	Min	Max
Differential voltage – Low (V)	0.06	0.06	0.245
Differential voltage – Avg (V) ¹	0.1	0.06	0.245
Differential voltage – High (V)	0.244	0.06	0.245
Common-mode voltage (V)	0.18	0.025	0.33
Test time duration (sec) ²	10	1	120
HS_PREPARE_length	30	0	15
STALL length	30	0	250

¹ To omit the Average Differential Voltage measurement, clear the **Average** check box.

² You can change the duration of the test time and the duration units.

Limits

The M-PHY specification ¹ defines minimum and maximum limits for V_{DIF_RX} for both terminated and unterminated cases as shown in the following table:

Table 17: Differential Input Voltage requirements

Parameter	Termination	Conformance Min	Conformance Max
$V_{DIF_RT_RX}$	Terminated	60 mV	245 mV
$V_{DIF_NT_RX}$	Unterminated	120 mV	490 mV

¹ [MIPI Alliance Specification for M-PHY V 1.00.00, 02/08/2011](#)

Oscilloscope Error Detector Settings

Table 18: Oscilloscope error detector settings

Parameter	Value
Error Detector Type	Bit
Error Detector Standard	ANY8B10B
Synchronization Pattern	Marker 0 – K28.5
Data Rate	Gear 1A/B
Error Limit	1e-10

See Also

- [Measurement Parameter Descriptions \(see page 103\)](#)
- [Common Test Parameters \(see page 37\)](#)
- [Receiver Testing Measurement Procedure \(see page 91\)](#)
- [About the Oscilloscope Error Detector \(see page 94\)](#)

Sinusoidal Jitter

Random Jitter

Mobile Industry Processor Interface

2.1.2 Receiver Eye Opening and Accumulated Diff Input Voltage Test

This test verifies that the eye opening and accumulated Differential Input Voltage ($T_{\text{EYE-RX}}$ and $V_{\text{DIF-ACC-RX}}$) of the high-speed receiver (HS-RX) are within conformance limits.

Signal Characteristics

Table 19: Signal characteristics for TEYE-RX, VDIF-ACC-RX

Pattern	CJTPAT	CJTPAT
Data rate (Mb/s)	1248 (GEAR1–A)	1456 (GEAR1–B)
Differential voltage (mV)	100	100
Common-mode voltage (mV)	150	150

Pattern	CJTPAT	CJTPAT
8b/10b encoded	Yes	Yes
Modulation	NRZ	NRZ

Measurement Parameters

The following table lists parameters for the test. Unless specified, only Gear 1 type, Data Rate, and Test Time Duration parameters can be configured in Compliance Mode.

Table 20: Parameters for TEYE-RX, VDIF-ACC-RX

Parameter name	Value	Min	Max
Differential voltage (V)	0.1	0.06	0.245
Common-mode voltage (V)	0.18	0.025	0.33
Test time duration (sec)	10	1	120
HS_PREPARE_length	30	0	15
STALL length	30	0	250

Limits

Verify that T_{EYE_TX} is greater than $0.2 U_{IHS}$ for $V_{DIF_ACC_RX} = 40$ mV.

Oscilloscope Error Detector Settings

Table 21: Oscilloscope error detector settings

Parameter	Value
Error Detector Type	Bit
Error Detector Standard	ANY8B10B
Synchronization Pattern	Marker 0 – K28.5
Data Rate	Gear 1A/B
Error Limit	1e-10

See Also

- [Measurement Parameter Descriptions \(see page 103\)](#)
- [Common Test Parameters \(see page 37\)](#)
- [Receiver Testing Measurement Procedure \(see page 91\)](#)
- [About the Oscilloscope Error Detector \(see page 94\)](#)

Non Return to Zero

2.1.3 Common-Mode Input Voltage Tolerance (VCM-RX) Test

This test verifies that the high-speed receiver (HS-RX) is able to successfully receive high-speed signaling that meets the maximum and minimum conformance requirements for common-mode voltage amplitude.

Signal Characteristics

Table 22: Signal characteristics for VCM-RX

Pattern		CJTPAT	CJTPAT
Data Rate (Mb/s)		GEAR1-A: 1248	GEAR1-B: 1456 Mb/s
Differential voltage (mV) ¹	Low amp	60	60
	High amp	244	244
Common-mode voltage		■ 25	■ 25
		■ 180	■ 180
		■ 325	■ 325
8b/10b encoded		Yes	Yes
Modulation		NRZ	NRZ

¹ Both the Low and High amplitude Differential voltage measurements are carried out using three different common-mode voltages shown in the table under Common-Mode voltage.

Measurement Parameters

The following table lists parameters for the test. Unless specified, only the Gear 1 type, Data Rate, Average amplitude setting, and Test Time Duration parameters can be configured in Compliance Mode.

Table 23: Parameters for VCM-RX

Parameter name	Value	Min	Max
Common-mode voltage – Low (V)	0.025	0.025	0.33
Common-mode voltage – Avg (V)	0.18	0.025	0.33
Common-mode voltage – High (V)	0.325	0.025	0.33
Differential voltage – High (V)	0.244	0.06	0.245
Differential voltage – Low (V)	0.06	0.06	0.245
Test time duration (sec)	10	1	120
HS_PREPARE_length	30	0	15
STALL length	30	0	250

Limits

V_{CM_RX} is between 25 mV and 330 mV.

Oscilloscope Error Detector Settings

Table 24: Oscilloscope error detector settings

Parameter	Value
Error Detector Type	Bit
Error Detector Standard	ANY8B10B
Synchronization Pattern	Marker 0 – K28.5
Data Rate	Gear 1A/B
Error Limit	1e-10

See Also

- [Measurement Parameter Descriptions \(see page 103\)](#)
- [Common Test Parameters \(see page 37\)](#)
- [Receiver Testing Measurement Procedure \(see page 91\)](#)
- [About the Oscilloscope Error Detector \(see page 94\)](#)

2.1.7 Receiver Jitter Tolerance (TJRX, SJRX, RJRX, STTJRX, STSJRX) Test

This test verifies that the high-speed receiver is able to successfully receive high-speed signaling that meets the conformance requirements of both low-frequency and short-term jitter.

Signal Characteristics

Table 25: Signal characteristics for TJRX, SJRX, RJRX, STTJRX, STSJRX

Pattern	CJTPAT	CJTPAT
Data rate (Mb/s)	1248 (GEAR1–A)	1456 (GEAR1–B)
LTJ	NA	1 MHz to (datarate/2) with 0.016 UIhs
STJ	NA	1 MHz to (datarate/2) with 0.0079 UIhs
Differential voltage (mV)	100	100
Common-mode voltage (mV)	150	150

Pattern	CJTPAT	CJTPAT
8b/10b encoded	Yes	Yes
Modulation	NRZ	NRZ

As specified in the CTS, long-term and short-term jitter impairments are added to the signal.

Measurement Parameters

The following table lists parameters for the test. Unless specified, only the Gear 1 type, Data Rate, and Test Time Duration parameters can be configured in Compliance Mode.

Table 26: Parameters for TJRX, SJRX, RJRX, STTJRX, STSJRX

Parameter name	Value	Min	Max
Differential voltage (V)	0.1	0.06	0.245
Common-mode voltage (V)	0.18	0.025	0.33
LTJ – Frequency 1 (MHz)	1	1	100
LTJ – Frequency 2 (MHz)	10	1	100
LTJ – Frequency 3 (MHz)	1/(30 × UI)	1	100
LTJ – Amp (UI)	0.35	1	100
STJ – Frequency 1 (Hz)	1/(30 × UI)	1	100
STJ – Frequency 3 (Hz)	1/(2 × UI)	1	100
STJ Amp (UI)	0.2	1	100
Test time duration (sec)	10	1	120
HS_PREPARE_length	30	0	15
STALL Length	30	0	250

Limits

For all Lanes, verify that:

- SJRX is approximately $0.35 \times \text{UIhs}$
- TJRX is approximately $0.52 \times \text{UIhs}$
- STSJRX is approximately $0.20 \times \text{UIhs}$
- STTJSA_NT_TX is approximately $0.30 \times \text{UIhs}$

Oscilloscope Error Detector Settings

Table 27: Oscilloscope error detector settings

Parameter	Value
Error Detector Type	Bit
Error Detector Standard	ANY8B10B

Parameter	Value
Synchronization Pattern	Marker 0 – K28.5
Data Rate	Gear 1A/B
Error Limit	1e-10

See Also

- [Measurement Parameter Descriptions \(see page 103\)](#)
- [Common Test Parameters \(see page 37\)](#)
- [Receiver Testing Measurement Procedure \(see page 91\)](#)
- [About the Oscilloscope Error Detector \(see page 94\)](#)

Long-Term Jitter

Short-Term Jitter

UIhs = 1/Datarate

Measurement Parameter Descriptions

Measurement parameters are displayed for the test selected in the tree view section. Not all of the parameters listed apply to all tests. You cannot change parameters that fall outside of compliance measurements. Individual test parameters are listed for each test in the HS Tests section of this online Help.

Table 28: Test measurement parameters

Parameter	Description	Applies to ...
Configure Settings for Gear1	Select the desired gear option from the drop-down list to see the parameter values for that gear option.	All tests
Data Rate	Data rate for the gear option displayed in the Configure Settings for field. If more than one data rate is displayed in the drop-down list, you can select the data rate to match the DUT configuration.	All tests
Common Mode Voltage	Common Mode voltage of the selected gear. Cannot be edited in Compliance Mode	2.1.1, 2.1.2, 2.1.7
Diff Voltage	Differential voltage of the selected gear. Cannot be edited in Compliance Mode.	2.1.2, 2.1.7
Diff Amp Low (V)	Differential amplitude used to create the test pattern	2.1.1, 2.1.3
Diff Amp High (V)	Differential amplitude used to create the test pattern	2.1.1, 2.1.3
Diff Amp Ave (V)	Differential amplitude used to create the test pattern	2.1.1
Time Test Duration	The time interval to receive the signal from the DUT after transmitting it. Units can be either in Seconds or e-10 BER . This can be edited in Compliance mode.	All tests

See Also

- [Common Test Parameters \(see page 37\)](#)
- [Configure Tests \(see page 36\)](#)
- [High-Speed Mode Measurements \(see page 92\)](#)

Bit Error Rate

Map the My TekExpress Folder

To map the My TekExpress folder on the instruments, follow these steps:

1. Open Windows Explorer.
2. From the Windows Explorer menu, click **Computer**.
3. In the menu bar, click **Map network drive**.
4. Select the Drive letter as **X:** (if there is any previous connection on X:, disconnect it first through **Tools > Disconnect Network drive** menu of Windows Explorer. Windows 7 users: if you do not see the Tools menu, press the **Alt** key).
5. In the Folder field, enter the remote My TekExpress folder path (for example, \\192.158.97.65\ My TekExpress).
6. Click **Finish**.

To determine the IP address of the instrument where the My TekExpress folder exists, do the following:

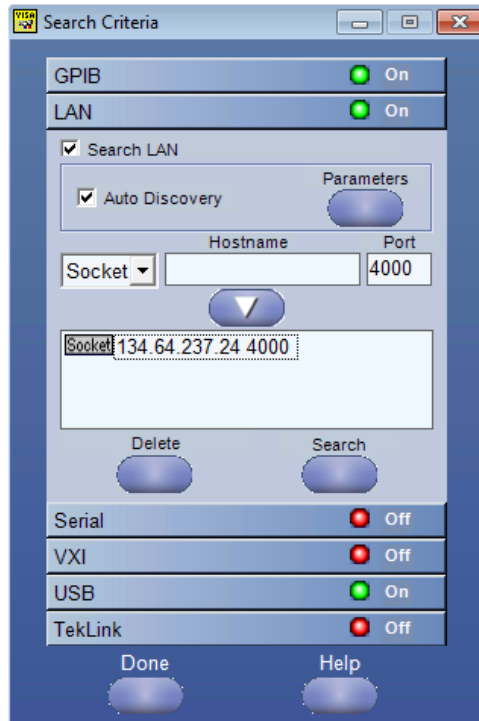
1. On the instrument where the My TekExpress folder exists, click **Start** and select **Run**.
2. Type “cmd” and then press **Enter**.
3. At the command prompt, type “ipconfig” and then press **Enter**.

Use an AWG7102 in LAN Configuration

Use these instructions to set up an AWG7102 to be used over a LAN.

NOTE. GPIB is recommended for use with an AWG7102 model because LAN communications is not reliable.

1. At the AWG7102, start the Socket Server from the TekVISA LAN Server Control.
2. At the oscilloscope, open TekVISA OpenChoice Instrument Manager.
3. In the Instruments section, click **Search Criteria**.
4. In the Search Criteria dialog box, click **LAN**. Check that **Search LAN** and **Auto Discovery** are both selected.
5. Select **Socket** from the drop-down list to the left of the Host Name field.
6. In the Host Name field, type the AWG7102 IP address and then click the down arrow to move the address to the search field. Example.



7. Click **Search**.
8. When you see the AWG7102 IP appear in the TekVISA Instruments list, click **Done** in the Search Criteria dialog box.
9. From the TekExpress Options menu, select **Instrument Control Settings** and then click **Refresh** in the dialog box.
10. When you see the AWG7102 listed in the Resource column, click **Close**.

The AWG7102 should now be available as a selection in the Signal Generator list in the test configuration Global Settings tab.

Enable the Oscilloscope Error Detector

The ERRDT and STU options are installed on the oscilloscope as part of the TekScope software but need to be enabled with a license before use.

To enable the error detector options:

1. After purchasing a license, from the TekScope menu select **Utilities > Option Installation**.
2. After the TekScope Options Installation wizard opens, press then **F1** key to view step-by-step instructions on using the wizard to install options.

To verify that the ERRDT and STU options are enabled:

1. From the TekScope Help menu, select **About TekScope**.
2. In the Options window, “ERRDET: Error Detector” and “STU” should appear in the list. If they do not, they are not enabled.

See Also

- [About the Oscilloscope Error Detector \(see page 94\)](#)

Index

A

- About TekExpress, 9
- Acquire Status, 46
- Activating the license, 7
- Analysis Status, 46
- Application controls, 14
- Application features, 9
- Application version, 8
- Application window, 14
- ApplicationStatus(), 84
- Auto Scroll, 46
- AWG software, 6
- AWG7102, 104

C

- Channel selection commands, 79
- CheckSessionSaved(), 89
- Clear test log, 46
- Client Programmatic Interface
 - flowchart, 59
- Client Proxy Object, 57
- Command buttons, 14
- Commands
 - Connect(), 70
 - GetAcquireParameter(), 76
 - GetAnalyzeParameter(), 76
 - GetDutId(), 73
 - GetGeneralParameter(), 76
 - LockSession(), 71
 - SelectTest(), 77
 - SetAcquireParameter(), 74
 - SetAnalyzeParameter(), 74
 - SetDutId(), 73
 - SetGeneralParameter(), 74
 - SetVerboseMode(), 72
- Commands list, 65
- Common measurements, 37
- Common-Mode Input Voltage
 - Tolerance test, 98
- Configuration Parameter
 - query, 76
- Configure Selected Measurement
 - commands, 80

- Configuring tests, 36
- Connect(), 70
- Connected instruments, 31
- Connection diagram, 91
- Connection requirements, 30

D

- Default directory, 10
- Diff Input Voltage Amp Tolerance
 - Test, 94
- Directories, 10
- Disable Popups command, 72
- Disconnect from Server
 - command, 90
- Disconnect(), 90
- DUT ID commands, 73
- DUT settings, 33
- DUT term, 1

E

- Email notifications, 37
- Email settings, 16
- Equipment setup, 30
- ERRDT, 106
- Error code commands, 82
- Error detector, 94
 - enabling, 106
- Evaluation mode, 13
- Exiting the application, 13

F

- File name extensions, 11

G

- Gear selection, 33
- GetAcquireParameter(), 76
- GetAnalyzeParameter(), 76
- GetCurrentStateInfo(), 85
- GetDutId(), 73
- GetGeneralParameter(), 76
- GetPassFailStatus(), 86

- GetReportParameter(), 87
- GetResultsValue(), 86
- GetResultsValueForSubMeasurements(), 86
- GetTimeOut(), 83
- Global settings, 37

H

- HS Tests
 - Common-Mode Input Voltage
 - Tolerance, 98
 - Diff Input Voltage Amp
 - Tolerance, 94
 - Receiver Eye Opening and
 - VDIF-ACC-RX, 96
 - Receiver Jitter Tolerance, 99

I

- Installing the software, 6
- Instruments
 - selecting, 36
 - viewing connected, 17
- Instruments Detected, 36
- Interface error codes, 82
- IP address, 104
- IP address connection
 - commands, 70

K

- Keep On Top, 13

L

- LAN, 104
- License activation, 7
- License Agreement, 8
- LockSession(), 71
- Log files, 46
- Log View, 46
- Loopback initialization, 33
- Loopback validation, 93

- M**
- M-PHY Rx Application
 - Commands, 65
- M-PHY Rx features, 9
- Measurement parameters, 103
- Measurement procedure, 91
- Measurement selection
 - commands, 77
- Menus, 14
 - Options, 15
 - Preferences, 47
- Message History, 46
- Mode, 37
- My TekExpress folder, 53
 - locating, 43
 - mapping, 104
- O**
- Options Menu, 15
 - Instrument Control Settings, 17
 - Keep On Top, 13
- Oscilloscopes supported, 3
- P**
- Panels, 18
 - Reports, 26
 - Results, 25
 - Setup, 18
 - Status, 23
- Pass/Fail Summary, 47
- Pattern, 37
- Pre-Run Checklist, 43
- PREPARE length, 103
- Probes, 4
- Program example, 62
- Programmatic Interface, 55
- Q**
- QueryStatus(), 84
- R**
- RecallSession(), 89
- Receiver Eye Opening test, 96
- Receiver Jitter Tolerance tests, 99
- Receiver testing, 91
- Related Documentation, 1
- Remote Proxy Object, 57
- Report contents, 47
 - selecting, 40
- Report file type, 40
- Report names, 40
- Report Options, 40
- Reports, 47
- Reports Panel, 26
- Results Panel, 25
- Run commands, 81
- Run(), 81
- S**
- SaveSession(), 89
- SaveSessionAs(), 89
- Saving test setups, 27
- Schematic button, 35
- Select Required button, 35
- Selecting DUT parameters, 33
- SelectSuite(), 78
- SelectTest(), 77
- SendResponse(), 85
- Serial Trigger Unit, 106
- Server control, 56
- Server locking commands, 71
- Session files, 53
- Session folders, 53
- Session Status commands, 89
- Set Configuration Parameters
 - commands, 74
- SetAcquireParameter(), 74
- SetAnalyzeParameter(), 80
- SetDutId(), 73
- SetGeneralParameter(), 74
- SetTimeout(), 83
- Setting up tests, 33
- Setup Panel, 18
- SetVerboseMode(), 72
- Show Detailed Log, 46
- Signal generator, 37
- Signal Path Compensation (SPC), 43
- Software installation, 6
- Specification version, 33
- STALL length, 103
- Status Panel, 46
- Stop commands, 81
- Stop(), 81
- STU, 106
- Suite selection commands, 78
- System requirements, 4
- T**
- Technical support, 2
- TekExpress Client, 55
- TekExpress Client requirements, 56
- TekExpress Server, 55
- Test configuration, 36
- Test groups, 29
- Test notifications, 39
- Test parameters, 37
- Test Reports, 47
- Test Results, 47
- Test Results commands, 86
- Test selection controls, 35
- Test setup files, 53
- Test setup overview, 29
- Test setup steps, 33
- Test setups, 27
 - creating based on existing, 28
 - recalling, 28
 - saving, 27
- Test Status, 46
- Test Status commands, 84
- Tests, 36
 - running, 43
 - selecting, 35
 - setting up, 29
- TestStand Client example, 64
- Timeout Value commands, 83
- TransferImages(), 87
- TransferReport(), 87
- U**
- Unlock Server command, 90
- UnlockSession(), 90
- Untitled Session folder, 43
- User Comments, 33
 - including in test reports, 40
 - location in reports, 47