TekExpress®
HDM Advanced Analysis and Compliance Solution

Online Help

Tektronix

**TekExpress®
HDM Advanced Analysis and Compliance Solution**

Online Help

Tektronix

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

TekExpress programmatic interface

Algorithms

HDM Transmitter Group 1: Clock Tests

HDM Receiver Group 1: Sink Tests

Reference

Index

TekExpress HDM Advanced Analysis and Compliance Solution

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, www.tektronix.com.

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.

- If possible, save the waveform on which you are performing the measurement as a .wfm file.

Overview and Key Specifications

The TekExpress HDM Advanced Analysis and Compliance Solution gives you the tools to easily run high definition multimedia interface (HDMI) tests under the HDMI Compliance Test Specifications 2.0. It provides a complete and reliable solution for quick testing.

The application functionality is generally divided into three parts: the Source Suite, the Sink Electrical Suite, and the Sink Protocol Suite. You can select which suite and which version of the CTS you would like to use by clicking on the Setup Panel and clicking the DUT tab . See Application panel overview (see page 30) for more information.

Supported Tests:

TekExpress HDM Source supports these automated Single ended and Differential Tests. (see page 3)

TekExpress HDM Sink supports these automated Sink Tests. (see page 4)

Supported Resolutions:

HDM Source supports these resolutions. (see page 5)

HDM Sink supports these resolutions. (see page 5)

See also:

Supported Tests: HDM Source

TekExpress HDM Source supports the following automated single-ended and differential tests:

Differential Tests

■ HF1-2: Source TMDS Electrical – 6G – TRISE, TFALL

This test confirms that the rise times and fall times on the TMDS differential signals fall within the limits of the specification.

■ HF1-3: Source TMDS Electrical – 6G – Inter-Pair Skew

This test confirms that the skew between two data lanes on the TMDS differential signals fall within the limits of the specification.

■ HF1-5: Source TMDS Electrical – 6G – Differential Voltage

This test confirms that the voltage level of the TMDS differential signals complies with the limits mentioned in specification.

■ HF1-6: Source TMDS Electrical – 6G – Clock Duty Cycle

This test confirms that the duty cycle and clock rate of the TMDS differential clock complies with the limits mentioned in specification.

■ HF1-7: Source TMDS Electrical – 6G – Clock Jitter

This test confirms that the clock jitter of the TMDS differential clock complies with the limits mentioned in specification.

Single ended tests

■ HF1-8: Source TMDS Electrical – 6G – Data Eye Diagram

This test confirms that the HDMI Data output has signal quality that meets the eye opening required by the specification

■ HF1-1: Source TMDS Electrical – 6G – VL

This test confirms that the low voltage level of the TMDS single ended signals complies with the limits mentioned in specification.

■ HF1-4: Source TMDS Electrical – 6G – Intra-Pair Skew

This test confirms that the skew between the positive and negative lane of the TMDS single ended signals complies with the limits mentioned in specification.

Supported Tests: HDM Sink

TekExpress HDM Sink supports the following automated HDM Sink Electrical and Sink Protocol tests.

These tests are supported for CTS Version 2.0:

HDM Sink Electrical Tests

■ HF 2-1-Min/Max Differential Swing Tolerance

■ HF 2-2-Intra-Pair Skew

■ HF 2-3-Jitter Tolerance

HDM Sink Protocol Tests

■   HF2-23 Sink Pixel Decoding YCBCR 4:2:0

Supported Resolutions: HDM Source

CTS 2.0: All of the TekExpress HDM tests can be performed for DUTs operating in resolutions between 3.4 GHz to 6 GHz. Other resolutions that fall within this range can be entered in the Low Resolution and High Resolution fields on the DUT tab of the Setup panel.

When Low and High resolutions are the same, the software performs the test at that one resolution.

Table 1: Supported resolutions (Source)

Measurement name	DUT Configuration	
	Highest supported TMDS Character rate	**Lowest supported TMDS Character rate**
1-8: TMDS – Data Eye Diagram	Yes	N/A
1-7: TMDS Clock Jitter	Yes	N/A
Vswing	Yes	N/A
1-2: TMDS Rise time	Yes	N/A
1-2: TMDS Fall time	Yes	N/A
1-6 TMDS Clock Duty Cycle		
Duty Cycle (Maximum)	Yes	Yes
Duty Cycle (Minimum)	Yes	Yes
Clock rate = 1/Mean Period	Yes	Yes
1-3: TMDS – Inter-Pair Skew	Yes	N/A
1-5: Differential Voltage	N/A	Yes
Eye	N/A	Yes
Maximum Differential Voltage Vhigh	N/A	Yes
Maximum Differential Voltage Vlow	N/A	Yes
Single ended tests		
1-4: TMDS – Intra-Pair Skew	Yes	N/A
1-4: TMDS – VL	N/A	Yes

Supported Resolutions: HDM Sink

CTS 2.0: All of the TekExpress HDM Sink tests can be performed at data rates 5.94 G and 2.97 G.

Equipment Connection Setup (HDM Source)

You need the following equipment to run HDM Transmitter tests. (For details, see Minimum System Requirements (see page 17)):

- A supported Tektronix oscilloscope (see page 17)

- Four Differential Probes – P7313SMA, for Single Ended and Differential tests

- Device under test (DUT)

- Yazaki Fixtures

- PWS4205, PWS2185 or 2220-30-1/ 2220J-30-1 Dual Power Supply (if using external Power Supply option)

- Any approved EDID analyzer

Connection diagrams and the tests they apply to are listed below.

Single Ended and Differential Tests

Table 2: Source differential tests

Test number and characteristic tested	Setup diagram
HF1-2: Source TMDS Electrical – 6G – TRISE, TFALL	Source differential diagram (see page 8)
HF1-3: Source TMDS Electrical – 6G – Inter-Pair Skew	Source differential diagram (see page 8)
HF1-5: Source TMDS Electrical – 6G – Differential Voltage	Source differential diagram (see page 8)
HF1-6: Source TMDS Electrical – 6G – Clock Duty Cycle	Source differential diagram (see page 8)
HF1-7: Source TMDS Electrical – 6G – Clock Jitter	Source differential diagram (see page 8)

Source differential diagram
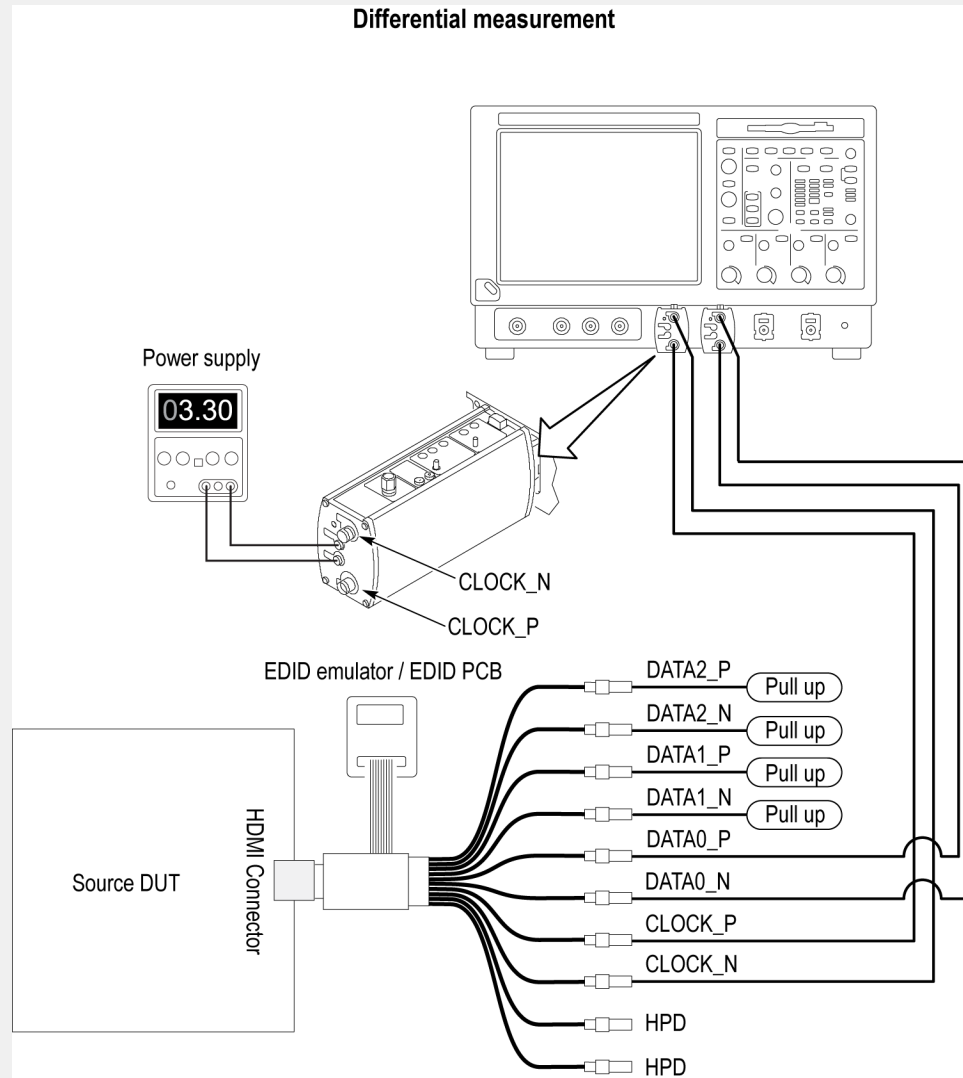
Differential measurement

Power supply

03.30

CLOCK_N

CLOCK_P

EDID emulator / EDID PCB

Source DUT

HDMI Connector

DATA2_P   (Pull up)
DATA2_N   (Pull up)
DATA1_P   (Pull up)
DATA1_N   (Pull up)
DATA0_P
DATA0_N
CLOCK_P
CLOCK_N
HPD
HPD

Notes: ──(Pull up)
Connect to biased 50Ω termination such as input port of unoccupied differential SMA probes.

0926-001

Table 3: Source single ended tests

Test number and characteristic tested	Setup diagram
HF1-8: Source TMDS Electrical – 6G – Data Eye Diagram	Source data eye diagram (see page 9)
HF1-1: Source TMDS Electrical – 6G – VL and VSwing	Source single ended diagram (see page 10)
HF1-4: Source TMDS Electrical – 6G – Intra-Pair Skew	Source single ended diagram (see page 10)

Source data eye diagram

Data eye diagram measurement

Source single ended diagram

Single ended measurement

Notes: ──(Pull up)

Connect to biased 50Ω termination such as input port of unoccupied differential SMA probes.   0926-002
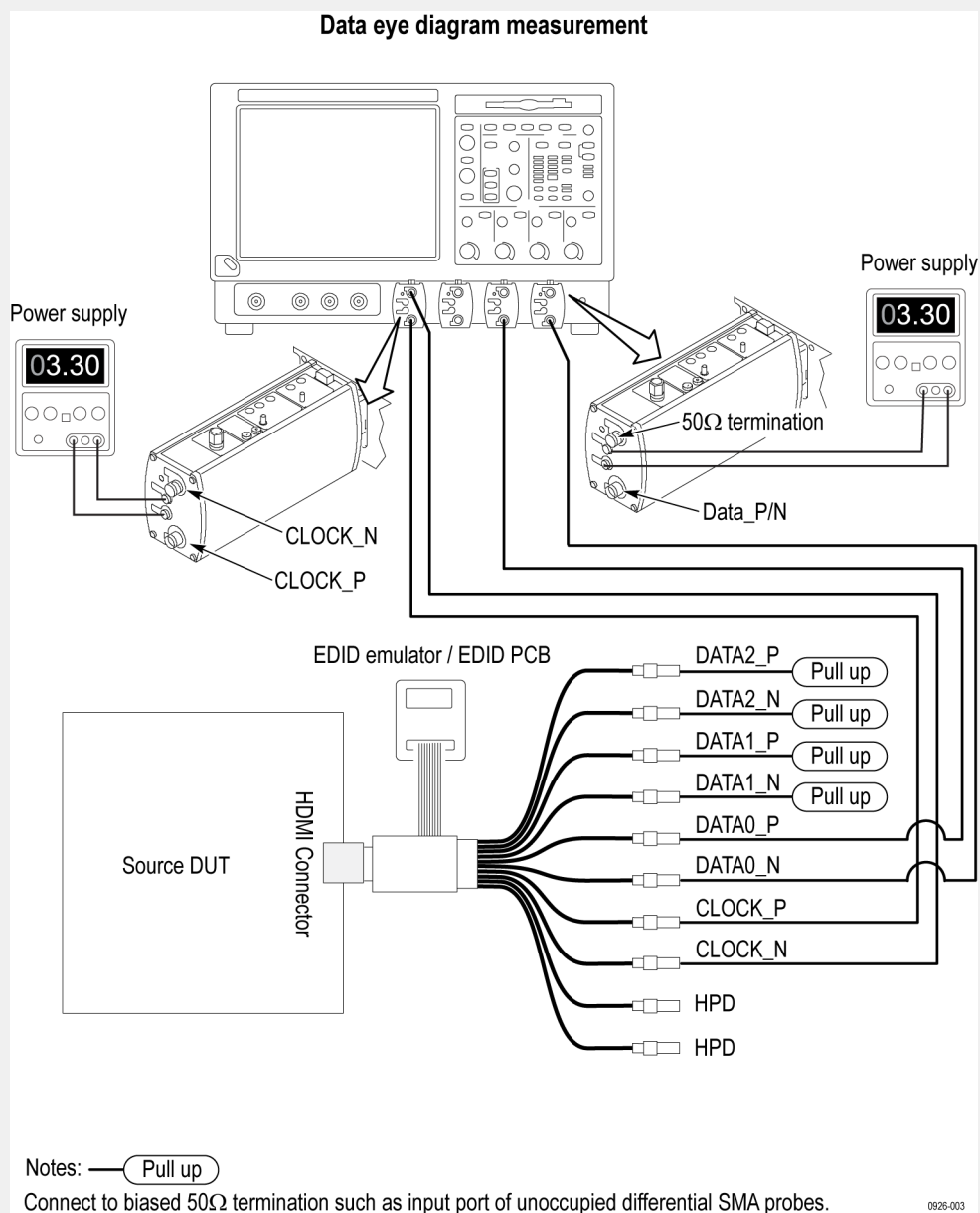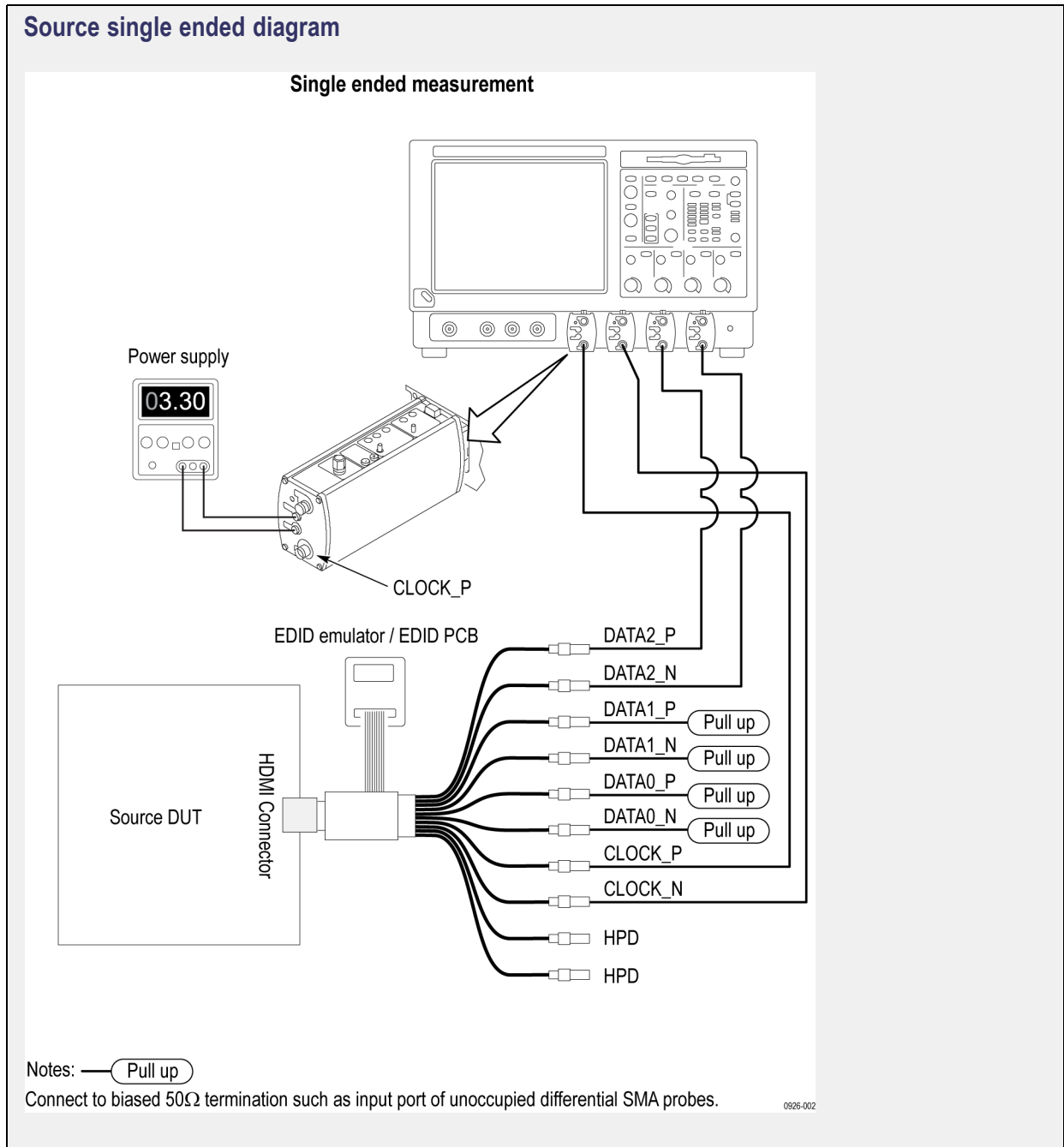
See Also

-
-

Equipment Connection Setup (HDM Sink)

You need the following equipment (for details, see Minimum System Requirements (see page 17)):

- A supported Tektronix oscilloscope (see page 17)

- Four Differential Probes – P7313SMA

- Device under test (DUT)

- Yazaki Fixtures

- AWG70002A with option 01, 03 and 225 (quantity 2), OR AWG7122C with option 1, 2/6, 8 (quantity 2)

Connection diagrams and the tests they apply to are listed below.

Sink Tests

Table 4: Sink electrical tests

Test number and characteristic tested	Setup diagram
*HF 2-1: Min/Max Differential Swing Tolerance	Sink min diagram (see page 12)
	Sink max diagram (see page 13)
*HF 2-2: Intra-Pair Skew	Sink intra-pair skew diagram (see page 12)
*HF 2-3: Jitter Tolerance	Sink jitter tolerance diagram (see page 12)

Diagram for sink min, intra-pair skew, jitter tolerance and protocol tests
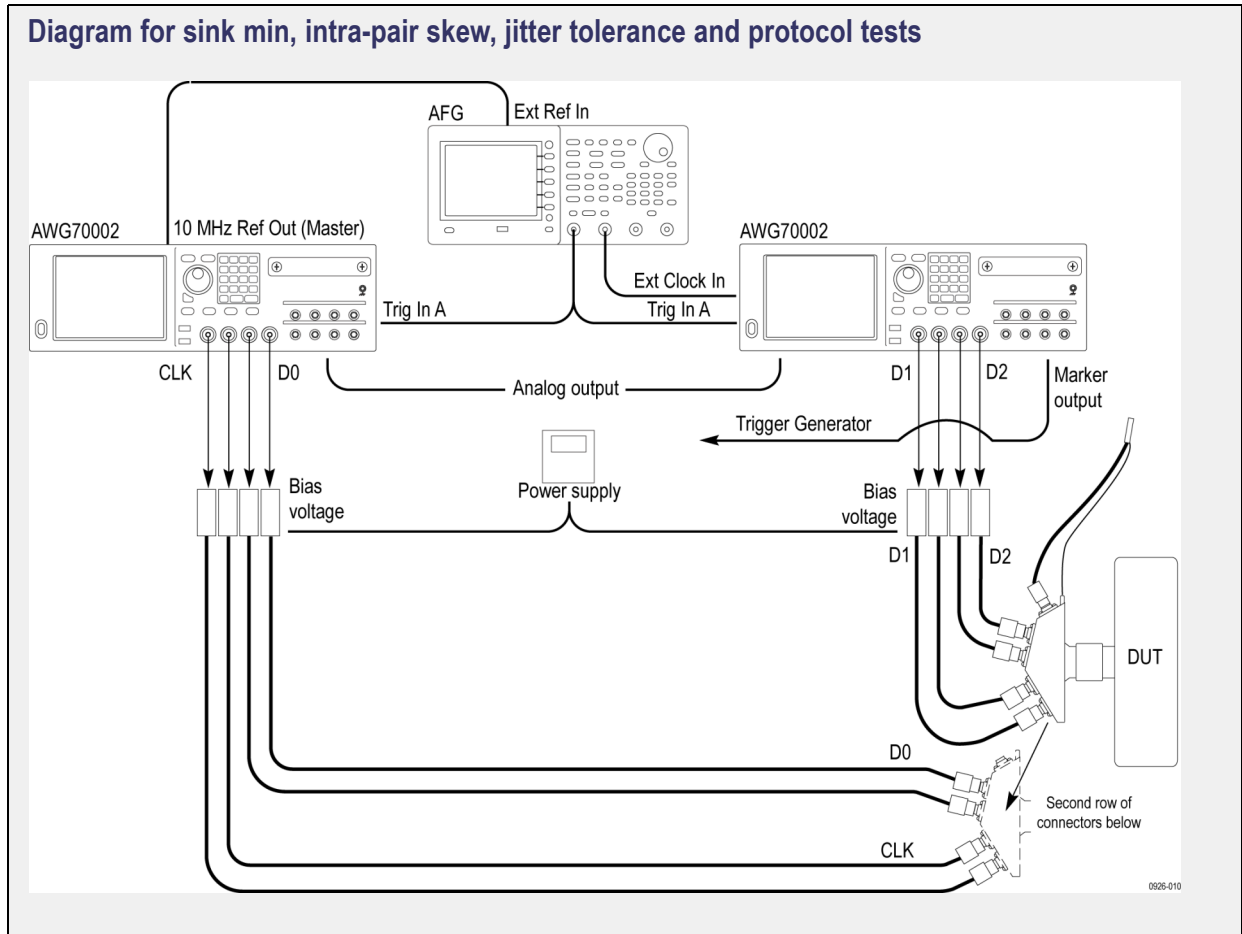
Sink max test diagram
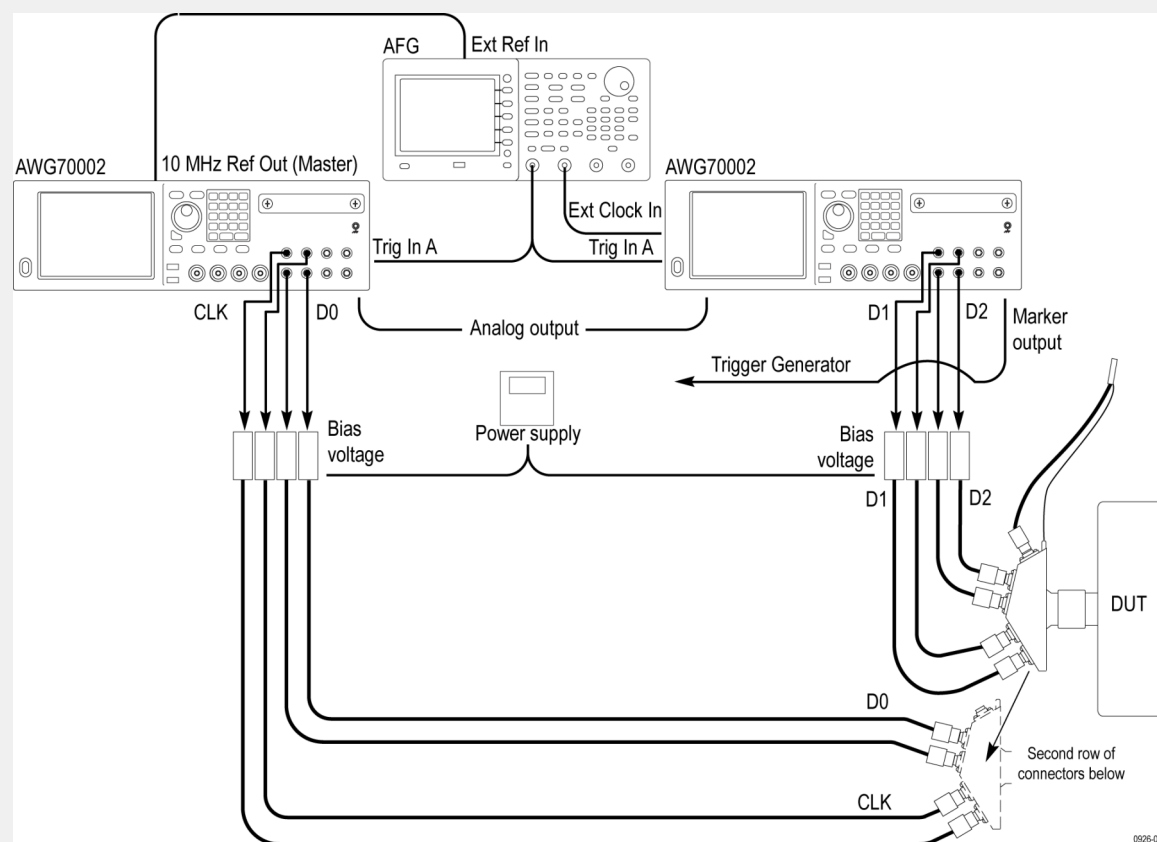
NOTE. For AWG7122C, the connection remains the same as the nominal pattern.

Table 5: Sink protocol tests

Test number and characteristic tested	Setup diagram
*HF2-23: Sink Pixel Decoding YCBCR 4:2:0	Sink protocol diagram (see page 12)

Application Directories and Usage

The application directory and associated files are organized as follows:
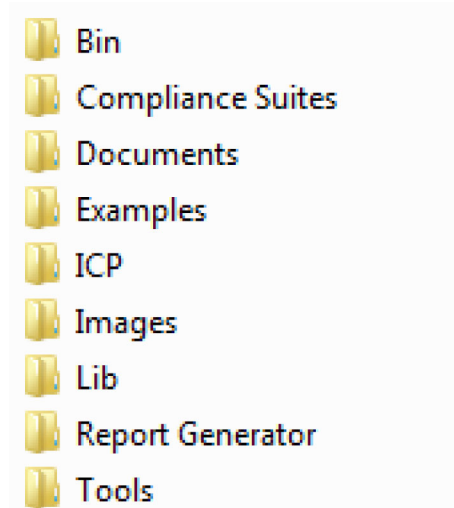
The following table lists the default directory names and their usage.

Table 6: Application directories and usage

Directory names	Usage
InstallDir\TekExpress\TekExpress HDM	Contains the application and associated files
TekExpress HDM\Bin	Contains miscellaneous HDM application libraries
TekExpress HDM\Compliance Suites	Contains compliance-specific files and filter files
TekExpress HDM\Documents	Contains the technical documentation for the HDM application
TekExpress HDM\Examples	Contains various support files
TekExpress HDM\ICP	Contains instrument and HDM application-specific interface libraries
TekExpress HDM\Lib	Contains utility files specific to the HDM application
TekExpress HDM\Report Generator	Contains style sheets for report generation
TekExpress HDM\Tools	Contains instrument and HDM application-specific files

See Also

- View Test-Related Files

View Test-Related Files

Files related to HDM tests are stored in the `HDM` folder under the `My TekExpress` shared folder. In the `HDM` 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 HDM icon and usually has no file extension displayed.

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 of its matching session folder.
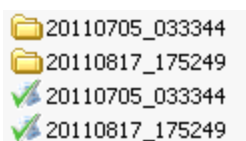
Each session folder contains image files of any plots generated by the test session, and any waveform files if prerecorded waveform files were used during the session.

The first time you run a new, unsaved session, the session files are stored in the `Untitled Session` folder located at `..\My TekExpress\HDM`. Once you name and save the session, the Untitled Session folder name is changed to the one you specified.

> **NOTE.** *By default, test report files are saved in the session folder. You can <u>change the report file location</u> for a specific test.*

See Also

- [File Name Extensions (see page 16)](#)

File Name Extensions

The TekExpress HDM software 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
.py	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 MHTML or PDF formats .
.pdf	Test report can also be saved in pdf format.
.flt	Filter file used with transmitter tests

See Also

- Application Directories and Usage (see page 14)

Compatibility

The TekExpress HDM application runs on the following Tektronix oscilloscopes:

- DPO/DSA/MSO72004/B/C, DPO/DSA/MSO71604/B/C Digital Oscilloscopes with Option DJA

- DPO/DSA73304D and DPO/DSA72504D Digital Oscilloscopes with Option DJA

- DPO/MSO72304/DX72504/DX73304 Series Digital Oscilloscopes

NOTE. *You can also use a 12.5 GHz bandwidth oscilloscope, but there may be a 10% variation in the measurement result.*

See Also

- Minimum System Requirements (see page 17)

Minimum System Requirements

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

Table 7: System requirements

	BW ≥ 16 GHz, 16 M Record Length/Ch - Opt. 4M/2XL or more (for Eye Diagram and Jitter tests), and 20XL for Inter Pair Skew tests
Oscilloscope	For a list of compatible oscilloscopes, see Compatibility (see page 17).
Processor	Same as the oscilloscope
Operating System	Same as the oscilloscope
Memory	Same as the oscilloscope
Hard Disk	Same as the oscilloscope
Display	Same as the oscilloscope [1]
Software	■ TekExpress HDM Advanced Analysis and Compliance software ■ DPOJET, Jitter and Eye Diagram Analysis Tool Ver 6.1.0 for Win 7 oscilloscopes ■ Microsoft Internet Explorer 6.0 SP1 or later ■ Microsoft Photo Editor 3.0 or equivalent software for viewing image files ■ Adobe Reader 8.0 or equivalent software for viewing portable document format (PDF) files
Probes	■ Four Differential Probes – P7313SMA for Single-Ended and Differential tests

Table 7: System requirements (cont.)

HDM Sources	■ HDM Tx DUTs
	■ HDM Ref waveforms
TDR Tests	■ Oscilloscope – DSA8200 or equivalent with 80E03 and 80E04 modules and I-Connect software
HDM Generator for Sink and Dongle and Cable tests	■ AWG70002A with option 01, 03 and 225 (Quantity 2) OR AWG7122C with option 1, 2/6, 8 (Quantity 2)
	■ AFG3102/AFG3252 :1No
HDM Fixtures	HDM fixture kits from Tektronix
	■ Yazaki HDMI 2.0 Fixture
HDM Accessory Kit (to be used with AWG)	TF-HDM-DS-Acc kit from Tektronix contains the following:
	■ (8) Bias Tees from Mini-Circuits – ZX85-12G-S+
	■ (2) 120 ps TTC filter from Pico Second Pulse Labs – 5115-110-120PS
Other Devices	■ Microsoft compatible mouse or compatible pointing device
	■ Four USB ports (two USB ports minimum)
	■ PCI-GPIB or equivalent interface for instrument connectivity [2]
	■ (10) SMA cables, 174-1428-xx
	■ (3) BNC cables
	■ BNC T connector
	■ (3) BNC-to-SMA adapter
	■ External power supply PWS4721/PWS4602/PWS4323/PWS4305/PWS4205
	■ Any approved DDC Master
	■ Any approved EDID analyzer

[1] If TekExpress is running on an instrument having a video resolution lower than 800 x 600 (for example, a sampling oscilloscope), it is recommended that you connect a secondary monitor, which must be enabled before launching the application.

[2] 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, ensure that the Talker Listener utility is enabled in the DPO/DSA oscilloscope's GPIB menu. For ease of use, connect to an external (secondary) monitor.

See Also

■

Install the Software

The software can be installed on any compatible instrument running Windows 7.

1. Close all applications (including the TekScope application).

2. Go to the www.tek.com Web site and search for HDMI to locate the installation file. Download the file `HDM WebInstaller.exe`.

3. Double-click the executable file to extract the installation files. After extraction, the installer launches and displays the InstallShield Wizard.

4. The software automatically installs in the following location:

   - `C:\Program Files\Tektronix\TekExpress\TekExpress HDM`

5. The installer updates the TekScope Analyze menu to include TekExpress HDM:
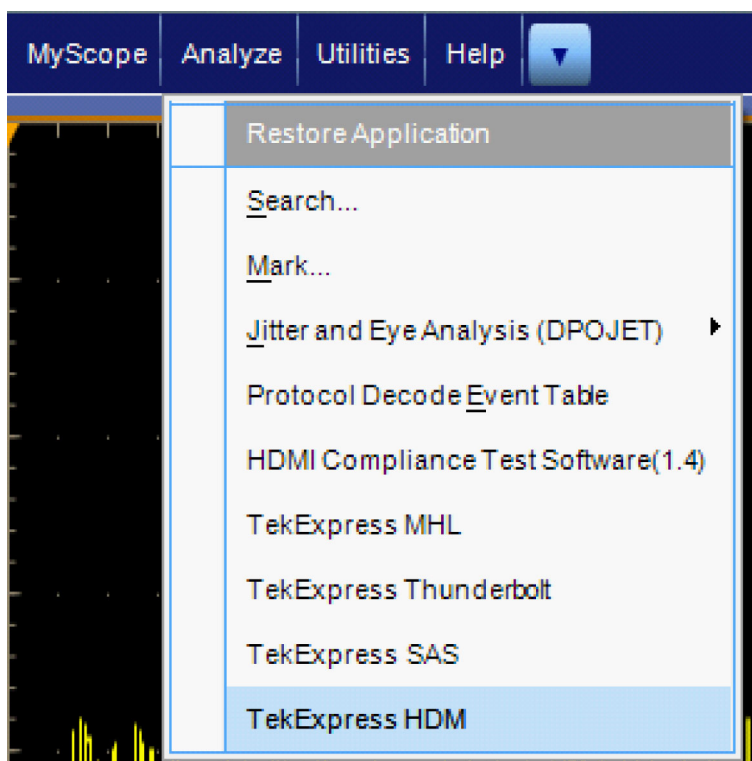
See Also

Activate the License

Activate the license using the option installation wizard on the oscilloscope. Follow these steps to activate the TekExpress HDM license:

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

    The TekScope Option Installation wizard opens.
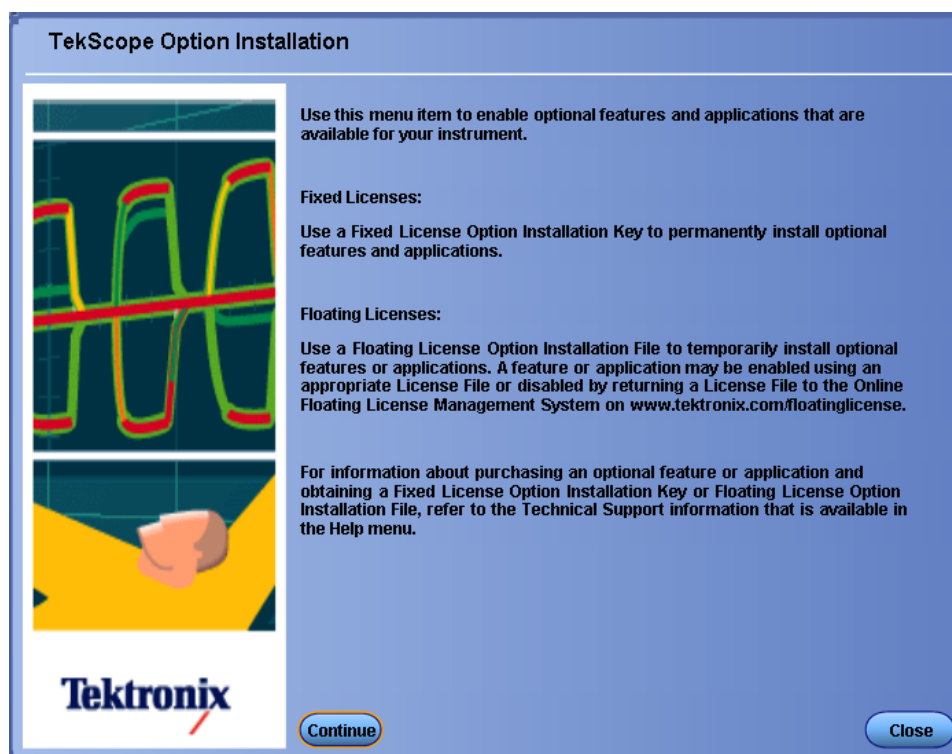
2. Instructions for using the Options Installation window to activate licenses for installed applications is provided in the oscilloscope online help. Press the **F1** key on the oscilloscope keyboard to open the Option Installation help topic. Follow the directions in the topic to activate the license.

See Also

■ View Version and License Information (see page 21)

View Version and License Information

Use the following instructions to view application version information and version information 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.
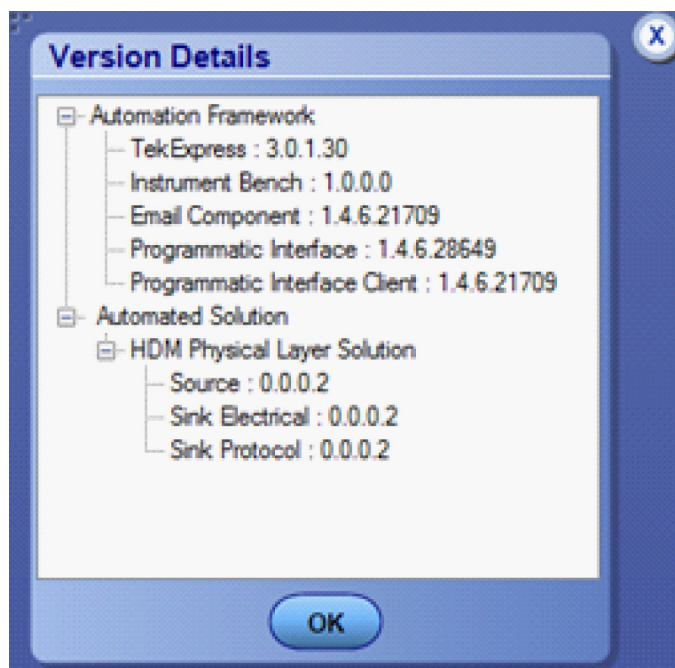
To view license information:

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

    The Options section in the dialog box displays a list of installed options, including TekExpress HDM.

2. To view the Option key, look in the Option Installation Key section. When finished, click **OK** to close the dialog box.

See Also

- Activate the License (see page 20)
- Options Menu (see page 25)

Run the application

To run the HDMI application, do either of the following:

- Select **Analyze > TekExpress HDM** from the TekScope menu.

- Double-click any saved HDMI session file.

When you open the application after installation, the application checks for a file called `Resources.xml` located in the `C:\Users\<username>\My Documents` folder. The `Resources.xml` file gets mapped to the X: drive when the application launches. Session files are then stored inside the `X:\HDM` folder. The `Resources.xml` file contains information about available network-connected instruments. If this file is not found, the application runs an instrument discovery program before launching HDM to locate available instruments.

If the application license was not installed using the TekScope menu **Utilities > Option Installation** selection, you can open the application up to 10 times in evaluation mode. Each time you open the application without supplying a valid license key, one of the free trials is used.

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 8: Application control descriptions

Item	Description
Options menu (see page 25)	Opens the Options menu for access to global controls
Panels (see page 30)	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	**Start** Starts continuous measurement acquisition and analysis. If prior acquired measurements have not been cleared, the new measurements are added to the existing set.
Stop button	**Stop** Stops (aborts) the current measurement acquisition.
Pause \ Continue button	**Pause** **Continue** Use the Pause button to temporarily interrupt the current acquisition. When a test is paused, the button name changes to Continue.
Clear button	**Clear** Clears all existing measurement results. Adding or deleting a measurement, or changing a configuration parameter of an existing measurement, also clears measurements. This prevents the accumulation of measurement statistics or sets of statistics that are not coherent. This button is available only on the Results panel (see page 41).
Clear Log	**Clear Log** This button is available only on the Status panel (see page 39).
Save	**Save...** This button is available only on the Status panel (see page 39).
Application window move icon	**Tek** 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.

Options menu overview

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

The Options menu (see page 26) 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 from which to select
Instrument Control Settings (see page 26)	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 HDM utility on top of other open windows on the desktop
Email settings dialog box (see page 27)	Use to configure email options for test run and results notifications
Help	Displays TekExpress Help
About TekExpress	■ Displays application details such as software name, version number, and copyright ■ Provides access to software version and license information (see page 21) for your HDM installation ■ Provides a link to the Tektronix Web site

Options menu
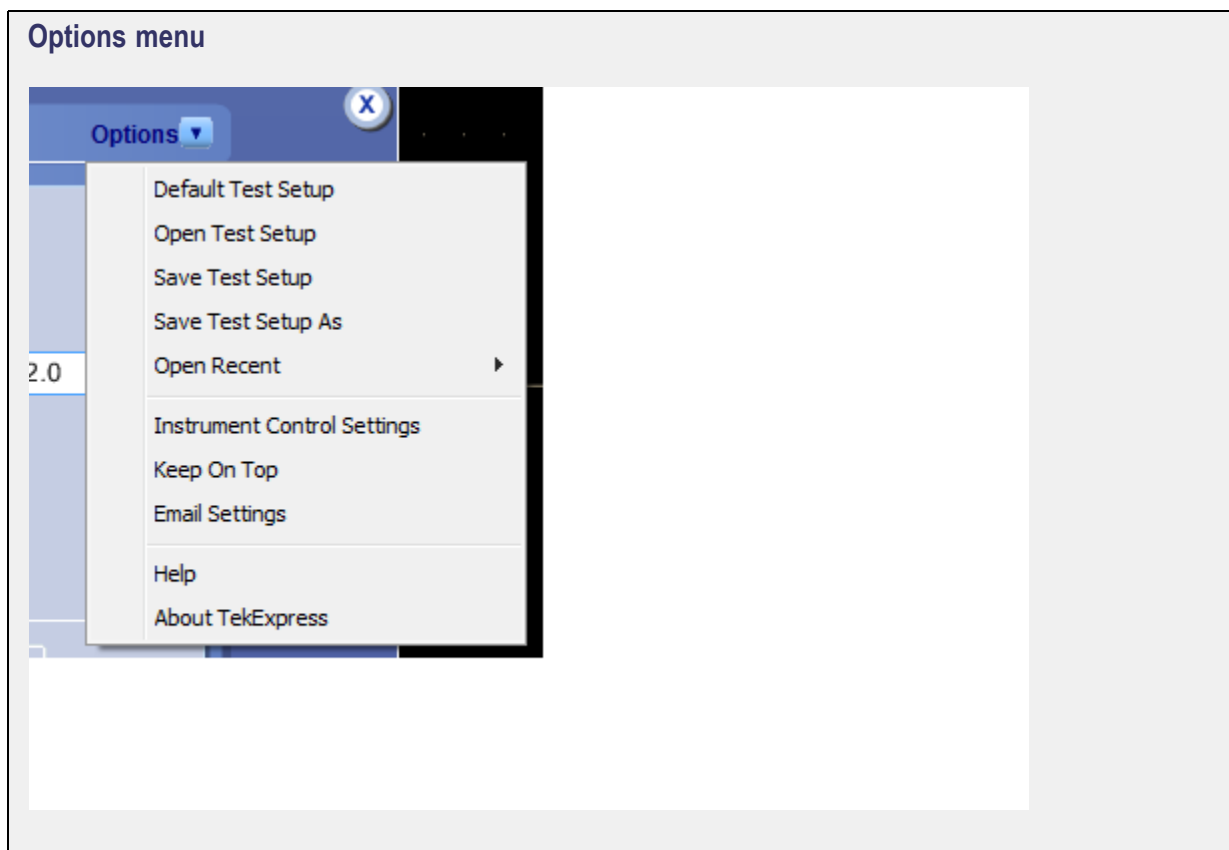
See also

Instrument control settings

Use the TekExpress Instrument Control Settings dialog box to search for and list the connected resources (instruments) found on specified connections (LAN, GPIB, USB, and so on) and each instruments connection information. You access this dialog box from the Options menu.
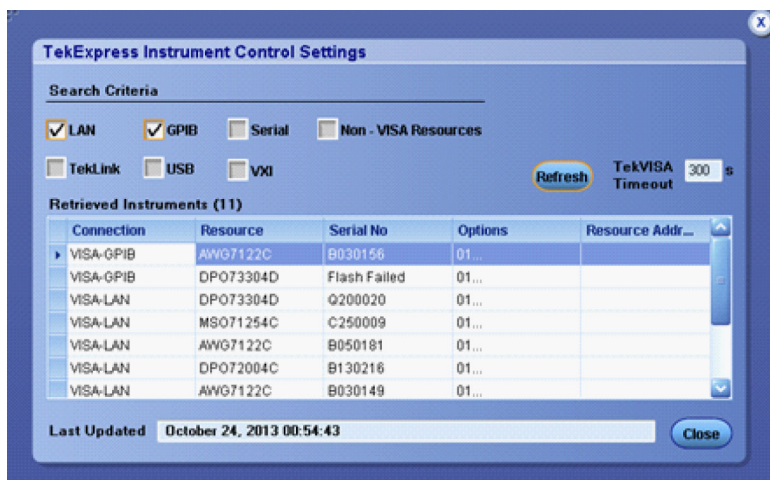
Use the Instrument Control Settings feature to and view instrument connection details. Connected instruments displayed here can be selected for use in the Global Settings tab in the configuration section. See step 1 of Configure Tests (see page 37) for details.

See also

Options menu overview (see page 25)

Email Settings dialog box overview

Use the Email Settings dialog box to enable HDMI to send an email message when a test completes, produces an error, or fails. Select the type of test run information to attach to the email (such as test reports and test logs), the email message format, and the email message size limit.

Open the Email Settings dialog box from the **Options** menu.

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

See also

Configure email settings

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

1. Select **Options > Email Settings** to open the Email settings dialog box.

2. (Required) For Recipient email Address(es), enter one or more email addresses to which to send the test notification. 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**: Attach the test report to the notification email.

    – **Status Log**: Attach the test status log to 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.

6. In the Email Configuration section:

    – Select the message file format to send: HTML (the default) or plain text.

    – Enter a maximum file size for the email message (message plus attachment files). Messages with attachments larger than this limit will not be sent. The default is 5 MB.

    – Enter the number in the Number of Attempts to Send field, to limit the number of attempts that the system makes to send a notification. The default is 1. You can also specify a timeout period.

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.

Application panel overview

Panels group related configuration, test and results settings.

Source and Sink

The TekExpress HDM panels are:

Table 9: Application panels

Panel name	Description
Setup (see page 30)	Set the DUT, test, acquisition and report parameters.
Status (see page 39)	View the progress and analysis status of the selected tests, and view test logs.
Results (see page 41)	View a summary of test results and select results viewing preferences.
Reports (see page 44)	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 24)

Setup panel overview

Source

The Setup panel contains sequentially ordered tabs that help guide you through a typical test setup process.

Set the DUT parameters (see page 31).

Select test(s) (see page 34).

Set lane acquisition source (see page 35).

Configure the selected tests (see page 37).

Select test notification preferences (see page 36).

Items selected in a preceding Setup tab may change options available in the following tabs. You can switch between the tabs in any order to modify your test parameters.

Sink

The Setup panel contains sequentially ordered tabs that help guide you through a typical test setup process.

Items selected in a preceding Setup tab may change options available in the following tabs. You can switch between the tabs in any order to modify your test parameters.

Set DUT parameters

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

Source

1. Click **Setup > DUT**.

2. (Optional) Enter the ID for the device. The default value is DUT001. The DUT ID parameter is added to reports.

3. (Optional) To add comments to the test report, click the note pad icon ( ) to the right of the DUT ID field. Enter comment text up to 256 characters. To enable or disable displaying comments in the test report, see Select report options (see page 45).

4. Select from the following parameters. Settings that do not apply to compliance testing cannot be changed and are grayed out.

Table 10: DUT tab settings

Setting	Description
Acquire live waveforms	Acquire active signals from the oscilloscope for testing.
Use pre-recorded waveform files	Run tests on a saved run session file. Select a run session file from the list.
View	Determines where to access the test configuration settings:
	■ **Compliance**: View configuration settings by clicking **Setup > Test Selection > Configure**
	■ **Advanced**: Enables the **Setup > Configuration** tab in which to view configuration settings.
Termination Source	Internal\External Other\Tektronix Power Supply
Data Rates	Highest: Select the Highest data rate to include in the tests.
	Lowest: Select the Lowest data rate to include in the tests

Table 10: DUT tab settings (cont.)

Setting	Description
TBit For High Data Rate (ps)	Sets the TBit Calculated For Highest Data Rate
TBit For Low Data Rate (ps)	Set the TBit Calculated For Lowest Data Rate
Probe Attenuations (X)	Differential: 12.5/2.5
	Single-Ended: 12.5/2.5
Recalc TBit	Check or Uncheck Recalc TBit. TBit Should be Calculated for at least 1 Time.
Number of Lanes to Test	Select the number of lanes to be tested: 1 Lane / 2 Lanes / 3 Lanes / 4 Lanes

See also
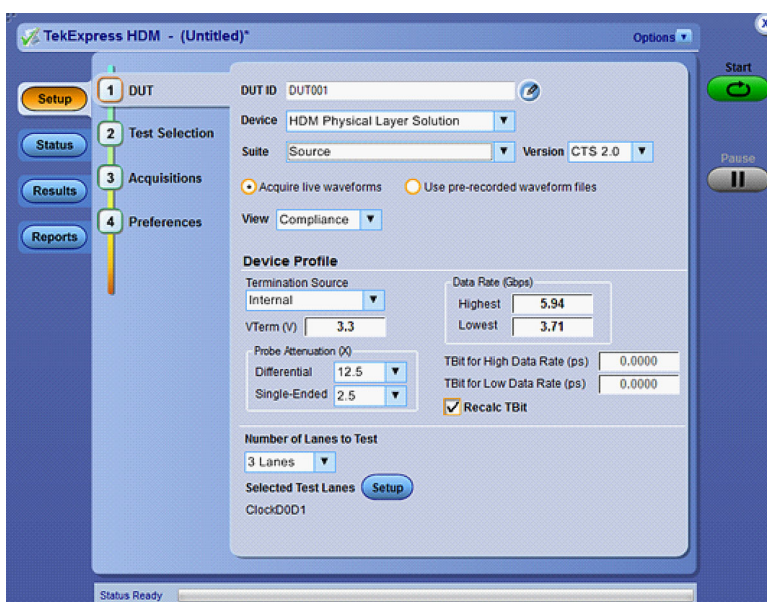
Sink
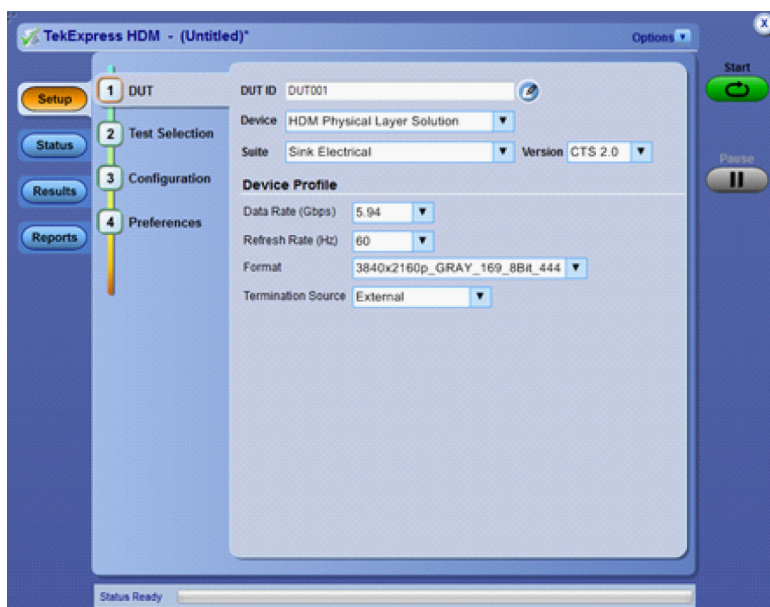
Table 11: DUT tab settings (sink)

Setting	Description
Termination Source	If Tek Power Supply is connected, select Termination source as "Tek Power Supply' or "External". Or It will be "External" only.
Data Rate (Gps)	Select Data Rate from Device Profile
Refresh rate	Select Refresh Rate (Hz) from Device Profile
Format	Select Format from Device Profile

Select tests

Use the Test Selection tab to select the tests to run on the connected DUT.

1. Click **Setup > Test Selection**.

2. Select the test(s) to run:

    ▬   Click one or more check boxes adjacent to each test.

    ▬   Click **Deselect All** to deselect all tests. All tests are selected by default.

    ▬   Click **Select Required** to select all tests that are required for compliance.

    ▬   Click **Select All** to select all tests.

3. Configure the test(s):

    ▬   Click **Configure** to open the configuration settings for a selected test.

NOTE. *The Configure button is not displayed if the View in the DUT tab is set to Advanced.*

    ▬   Click **Schematic** to display a schematic diagram that shows the DUT test setup. Use the diagram
        to verify the test setup before running the test.

See also
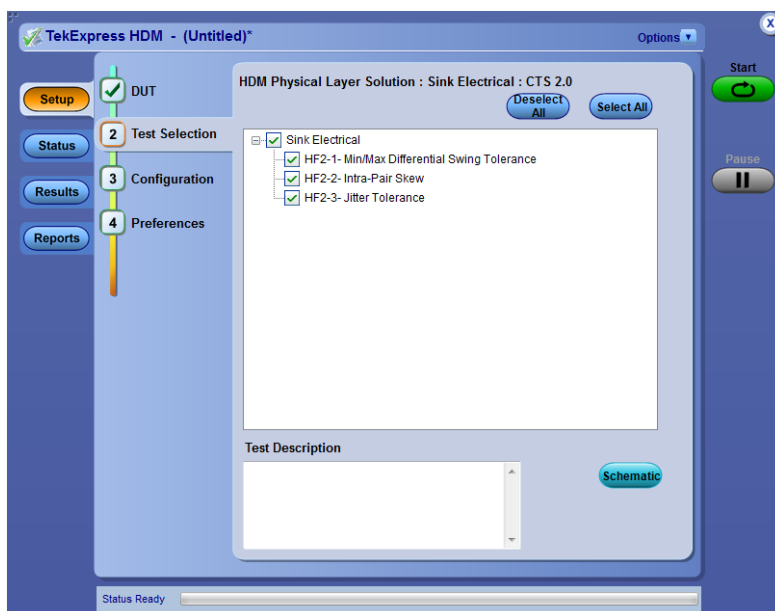
Set acquisition parameters (source only)

Use the Acquisitions tab to set the signal source (channel) used to acquire data. Acquisition options are available only when acquiring a live waveform.

1. Click **Setup > Acquisitions**.

2. Select **Source** field to select the oscilloscope channels to use for those lanes.

3. Select Probe Type from **Probe Type** Field as Differential, Single-Ended, or Eye-Diagram.

4. Check or uncheck the **Enable Filter** check box and configure the Filter Configuration by clicking the **Filter Config** button and choose from the displayed field.

5. Select **Acquire Step By Step** to have the software prompt you to continue after each phase of the test completes.
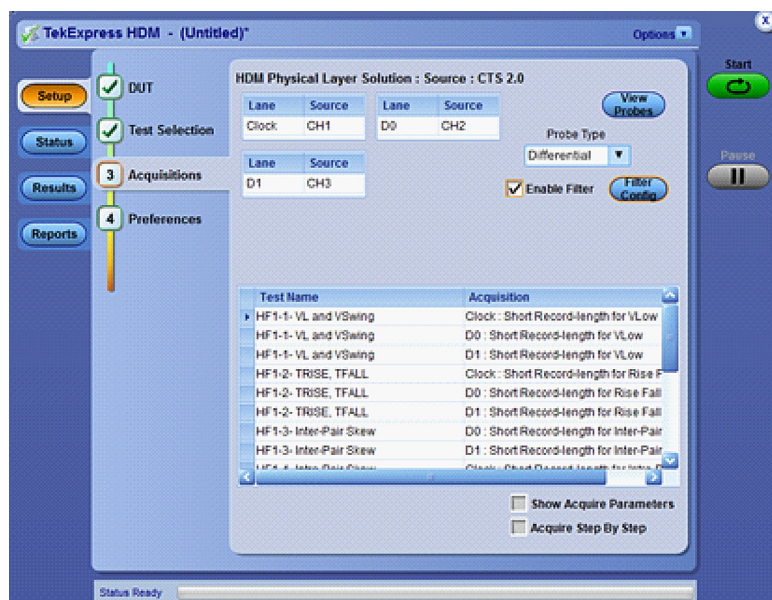
Set test notification preferences

Use the Preferences tab to set the application to send an email when a test measurement completes:

1. Click **Setup > Preferences**.

2. Select the **Email Test Results when complete** check box in the Email Settings dialog box. Click **Email Settings** to <u>configure the email settings (see page 28)</u>.

See also

<u>Select report options (see page 45)</u>

About configuring test parameters (source only)

Use the configuration settings to view the measurement parameters for selected tests. How the test configurations are accessed depend on the View selected in the DUT tab.

■ If you selected Compliance View in the DUT tab, then in the Test Selection tab, select the desired test in the list and then click the Configure button.

■ If you selected Advanced View in the DUT tab, click the Configuration tab in the Setup panel.

See also

Configure tests parameters

Configure test parameters

The Configuration parameters let you set global and individual test parameters. To return to test selection from the Configuration tab or panel, click the **Test Selection** button.

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

Source and Sink

1. Modify Global settings (see page 38) 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 select an instrument. If you do not see the desired instrument in the list, see Instrument Control Settings (see page 26).

2. To modify any individual test measurement settings, click **Measurements**, select the test in the tree view, and change the settings.

Common test parameters

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

Table 12: Common parameters and values

Parameter type	Parameter
Mode	Determines whether test parameters are in compliance mode or can be edited (User Defined Mode). ■ Compliance: Most test parameter values cannot be edited. ■ User Defined: Most test parameter values can be edited.
Global Settings	These settings apply to all tests selected for the current session. You can change only some of these settings. ■ Scope Settings: Specifies the instrument channel to use as the source for the trigger. This can be changed. 　■ 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.
Limits Editor (source only)	Shows the upper and lower limits for the applicable measurement using different types of comparisons. In Compliance Mode, you are able to view the measurement high and low limits used for the test selected in the tree view of the Measurements tab. When running tests in User Defined Mode, you can edit the limit settings (see page 38) in the Limits Editor. When running tests in User Defined Mode, the cells in the Limits Editor table are active for editing so you can change parameters. When running tests in User Defined Mode, the cells in the Limits Editor table are active for editing so you can change parameters.
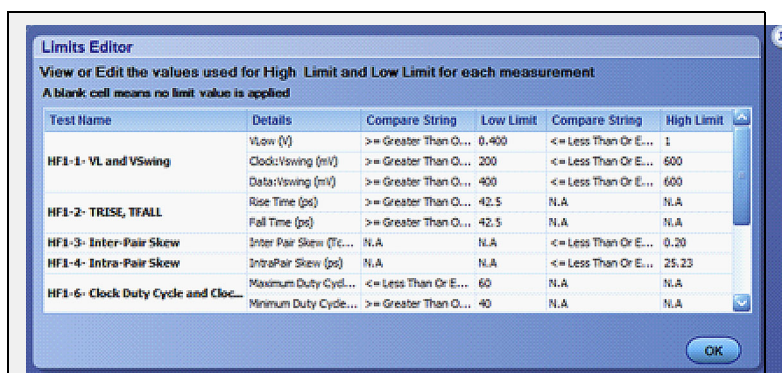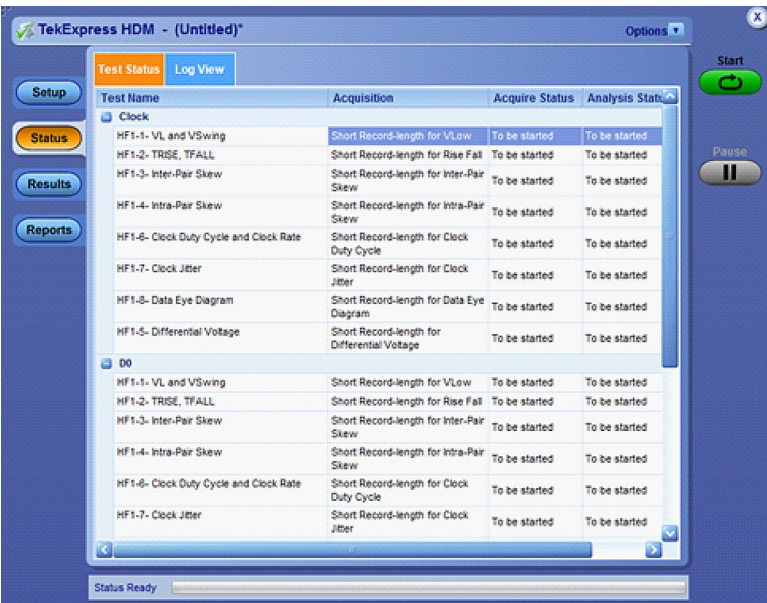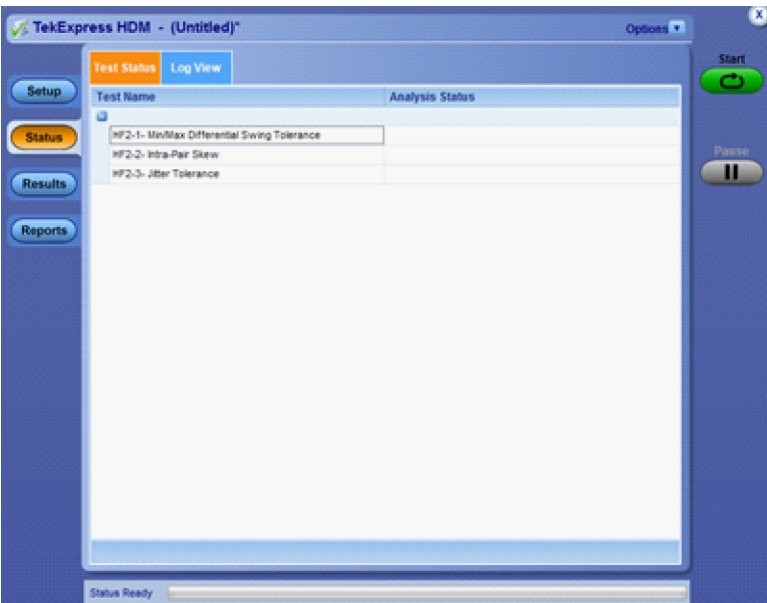
Status panel overview

The Status panel provides status on test acquisition and analysis (Test Status tab) and a listing of test tasks performed (Log View tab). The application opens the Test Status tab when you start a test run. You can select the Test Status or the Log View tab to view these items while tests are running.

The Test Status tab lists a high level status for each test.

Table 13: Test Status tab

Column	Description
Test Name	Name of the test
Acquire Status, Analysis Status	Status of the signal acquisition or test analysis ■ To be started ■ In progress ■ Completed ■ Aborted **NOTE.** *Sink tests have a minimal set of status indicators.*

Column   **Description**

The Log View tab provides a list of the actions executed during the test. Use this information to review or troubleshoot tests.

Table 14: Log View fields

Item	Description
Message History	Displays all run messages with timestamp information.
Show Detailed Log	Records a detailed history of test execution.

Item	Description
Auto Scroll	Sets the program to automatically scroll down the Message History window as information is added to the log during the test.
Clear Log	Clears all messages in the Message History window.
Save	Saves 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

Results panel overview

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 43).

Sink results

Source results

View test results

When a test finishes, the application switches to the Results panel (see page 41), 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 one or more tests fail, 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 test-related files

Files related to tests are stored in the `My TekExpress\HDM` folder. In the `HDM` 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 HDM icon and usually has no visible file extension.

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.
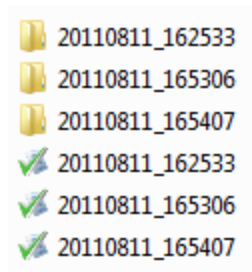
20110811_162533
20110811_165306
20110811_165407
20110811_162533
20110811_165306
20110811_165407

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

See also

Reports panel overview

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

For information on setting up reports, see . For information on viewing reports, see .

Select report options

Use the [Reports panel (see page 44)](#) 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 15: Report options

Setting	Description
Generate New Report	Creates a new report.
Replace current test values with the new results	Replaces the previous test results with the latest test results. Newly added tests results are appended to the end of the report.
Append to Previous Report	Appends the latest test results to the end of the current test results report.

Table 15: Report options (cont.)

Setting	Description
Report Name	Displays the name and location from which to open a report. The default location is at \My TekExpress\SAS\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.
	Change the report name or location.
	Do one of the following:
	■ 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.
	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\HDM Tx\DUT001_Test_211.mht.
	NOTE. You cannot change the file location using the Browse button.
	Open an existing report.
	Click **Browse**, locate and select the report file and then click **View** at the bottom of the panel.
Save As Type	Saves a report in the specified file type. Lists supported file types to choose from.
	NOTE. 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.
Auto increment report name if duplicate	Sets the application to automatically increment the name of the report file if the application finds a file with the same name as the one being generated. For example: DUT001, DUT002, DUT003. This option is enabled by default.
Include Pass/Fail Results Summary	Sets the application to include the color block labeled Test Result (indicating whether the test passed or failed) in the report. For details, see Report Contents in View a Report (see page 47).
Include Detailed Results	Sets the application to include parameter limits, execution time, and test-specific comments generated during the test.
Include Plot Images	Sets the application to include plotted diagrams.
Include Setup Configuration	Sets the application to include information about hardware and software used in the test in the summary box at the top of the report. Information includes: the oscilloscope model and serial number, probe model and serial number, the oscilloscope firmware version, SPC and factory calibration status, and software versions for applications used in the measurements.
Include Complete Application Configuration	Sets the application to include a table listing general, common, and acquired parameters used in the test. 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)

View a report

The application automatically generates a report when test analysis is completed and displays the report in your default 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. Click the **Reports > Browse** button and locate and select the report file to view.

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

Report contents

A report shows specified test details, as defined in the Reports panel.

Setup Configuration Information

Setup configuration information is listed in the summary box at the top 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.

Source

Sink

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.

Test results

This table lists the tests results. The contents of this table depend on the selections made in the Reports panel before running the test (Include Pass/Fail Results Summary, Include User Comments, Include Detailed Results, Include Complete Application Configuration, and so on).

If Include Plots Images is selected, then measurement waveform plots are shown below the test results table.
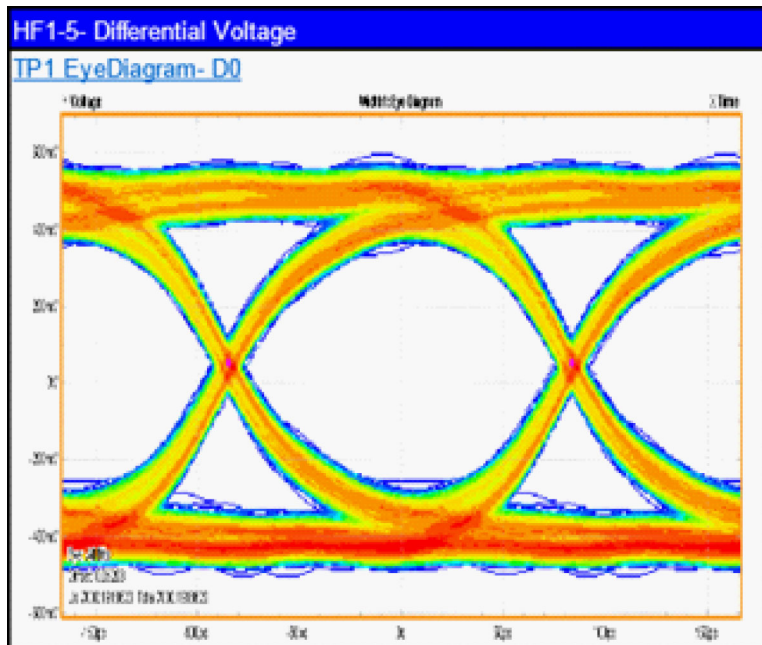
See also

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.
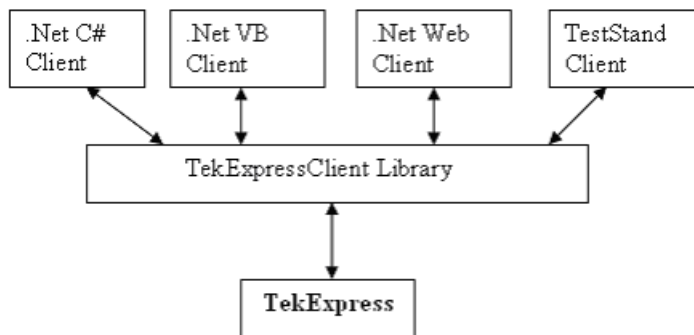
See also

Requirements for developing TekExpress client

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

HDM application commands flow (see page 60)

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.
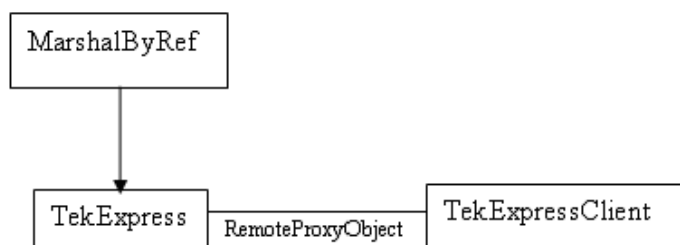
The following is an example:

```
RemotingConfiguration.RegisterWellKnownServiceType (typeof (TekExpressRe-
moteInterface), "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.
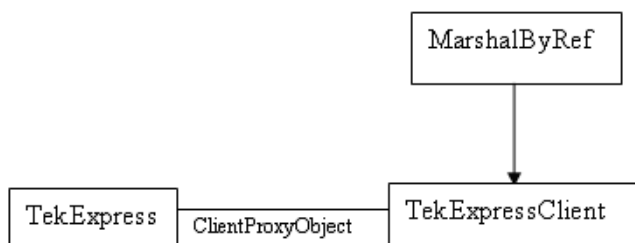
For example,

//Register the client proxy object

```
WellKnownServiceTypeEntry[] e = RemotingConfiguration.GetRegisteredWell-
KnownServiceTypes();
```

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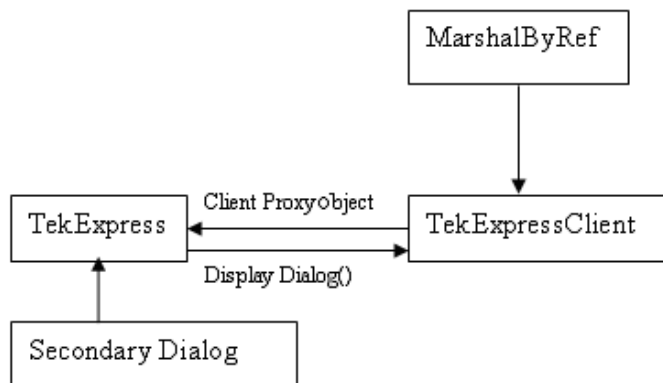
The secondary dialog messages from the Secondary Dialog library are redirected to the client-end when a client is performing the automations at the remote end.

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 overview

The following is an overview of the client programmatic interface:

Process flowchart
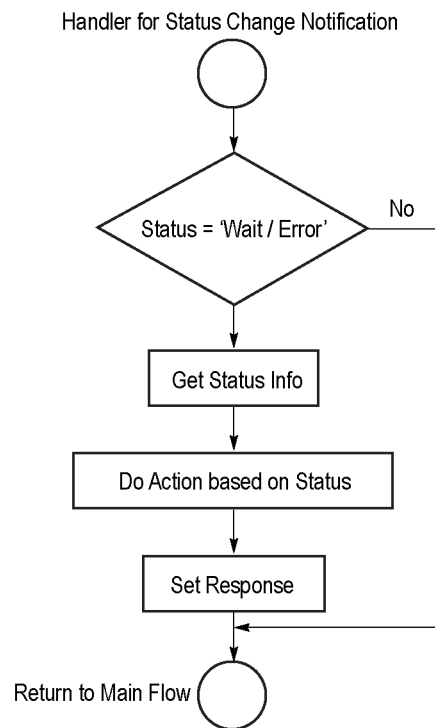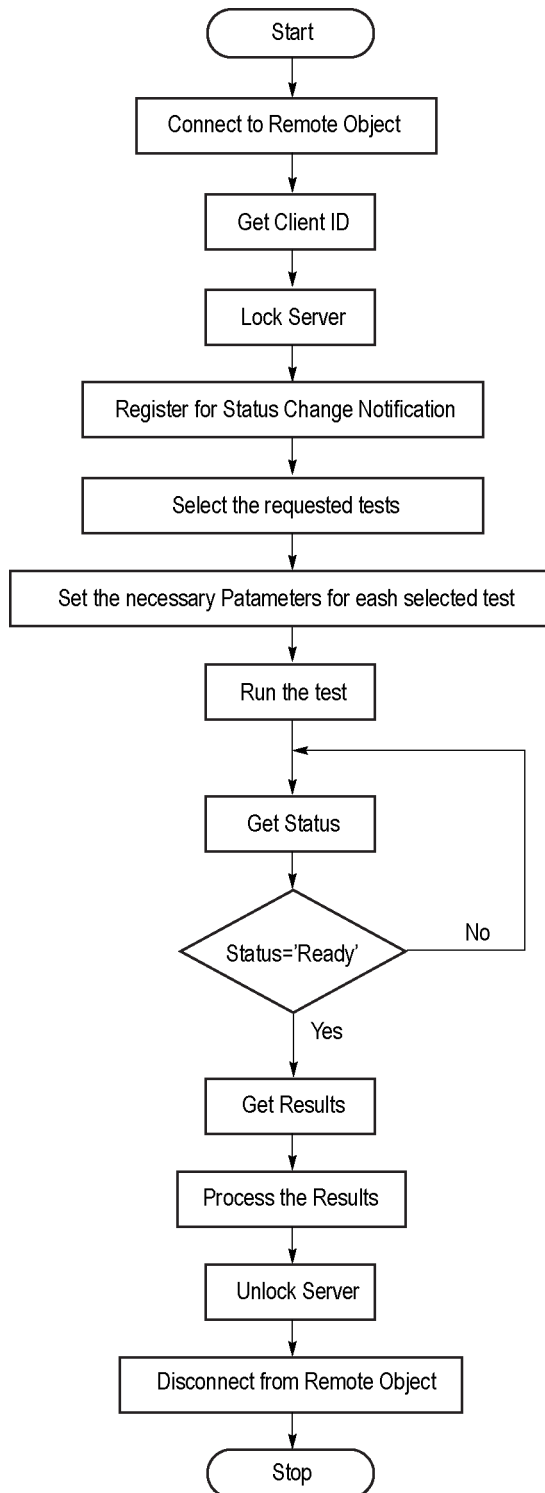
1. Connect to a server or remote object using a programmatic interface.

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 57).

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

Program remote access code example

This code example shows how to communicate between a remote PC and TekExpress SAS.

Table 16: Remote access code example

Task	Code
Start the application	
Connect through an IP address.	`m_Client.Connect("localhost") 'True or False` `clientID = m_Client.getClientID`
Lock the server	`m_Client.LockServer(clientID)`
Disable the Popups	`m_Client.SetVerboseMode(clientID, false)`
Set the DUT ID	`m_Client.SetDutId(clientID, "DUT_Name")`

Select a test	For Source:
	`SelectTest(clientId, device, Suite, test name, true)`
	`m_Client.SelectTest(clientID, "HDM Physical Layer Solution",`
	`"Source", "HF1-2- TRISE, TFALL", true);`

Run with set configurations	`m_Client.Run(clientID)`

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, WaitingMsbBxCaption,` `WaitingMsbBxMessage, WaitingMsbBxButtontexts)`

Send the response	`mClient.SendResponse(clientID, WaitingMsbBxCaption, WaitingMsb-`
	`BxMessage, WaitingMsbBxResponse)`
	`End Select`
	`Loop Until status = "Ready"`

Save results	`'Save all results values from folder for current run` `m_Client.TransferResult(clientID, logDirname)`
Unlock the server	`m_Client.UnlockServer(clientID)`
Disconnect from server	`m_Client.Disconnect()`
Exit the application	

HDM application commands flow

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

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 For example "HDM Physical Layer Solution"

string suite

Name	Type	Direction	Description
suite	string	IN	Specifies the name of the suite For Example "Source"; "Sink Electrical"; "Sink Protocol"

string test

Name	Type	Direction	Description
test	string	IN	Specifies the name of the test to obtain the pass or fail status. Test Names for Source: "HF1-1- VL and VSwing"; "HF1-2- TRISE, TFALL"; "HF1-3- Inter-Pair Skew"; "HF1-4- Intra-Pair Skew"; "HF1-5- Differential Voltage"; "HF1-6- Clock Duty Cycle and Clock Rate"; "HF1-7- Clock Jitter"; "HF1-8- Data Eye Diagram"

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 WaitingMsbBxCaption

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

out string WaitingMsbBxMessage

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

out string[] WaitingMsbBxButtontexts

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

string WaitingMsbBxResponse

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 61)	This method connects the client to the server. Note (see page 61)	Return value is either True or False	m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL.
	out string clientID (see page 61)	The client provides the IP address to connect to the server.		returnval as boolean
		The server provides a unique client identification number when connected to it.		returnval = m_Client.Con-nect(ipaddress,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...".

Lock the server

Command name	Parameters	Description	Return value	Example
LockSession()	string clientID (see page 66)	This method locks the server. Note (see page 63) 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

Use these commands to disable popup messages that require user intervention.

Command name	Parameters	Description	Return value	Example
SetVerboseMode()	string clientID (see page 66) bool _verbose (see page 64)	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".	m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string **Verbose mode is turned on** return=m_Client.SetVerbose-Mode(clientID, true) **Verbose mode is turned off** returnval=m_Client.SetVer-boseMode(clientID, false)

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 66)
	 string dutName (see page 61)
		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
				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 61)

GetDutId()	string clientID (see page 66)
	 string dutId (see page 62)
		This method gets the DUT ID of the current setup.	String that gives the status of the operation after it has been performed	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)

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 66) string device (see page 62) string suite (see page 62) string test (see page 62) string parameterString (see page 62)	This method sets the general parameter values. *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.	Sets the value for the general Parameter. mClient.SetGeneralParameter(clientID, "HDM Physical Layer Solution", "Source", "HF1-2- TRISE, TFALL", "Voltage termination$Internal"
SetAnalyzeParameter()	string clientID (see page 66) string device (see page 62) string suite (see page 62) string test (see page 62) string parameterString (see page 62)	This method sets the configuration parameters in the Analyze panel of the Configuration Panel dialog box for a suite or measurement.	String that gives the status of the operation after it has been performed The return value is "" (an empty String) on success.	Sets the value for the Analyze Parameter. mClient.SetAnalyzeParameter(clientID, "HDM Physical Layer Solution", "Source", "HF1-2- TRISE, TFALL", "Signal Type$DATA")
SetAcquireParameter()	string clientID (see page 66) string device (see page 62) string suite (see page 62) string test (see page 62) string parameterString (see page 62)	This method sets the configuration parameters in the Acquire panel of the Configuration Panel dialog box for a suite or measurement.	String that gives the status of the operation after it has been performed The return value is "" (an empty String) on success.	Sets the value for the Acquire Parameter. mClient.SetAcquireParameter(clientID, "HDM Physical Layer Solution", "Source", "HF1-2- TRISE, TFALL", "Short Record-length for VLow$Record Length (M)$4"

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 the configuration parameters for a suite or measurement

Command name	Parameters	Description	Return value	Example
GetGeneralParameter()	string clientID (see page 66) string device (see page 62) string suite (see page 62) string test (see page 62) string parameterString (see page 62)	This method gets the general configuration parameters for a suite or measurement.	The return value is the general configuration parameter for a specified suite or measurement that is set.	Gets the value for the general Parameter. mClient.GetGeneralParameter(clientID, "HDM Physical Layer Solution", "Source", "HF1-2- TRISE, TFALL", "Voltage termination")
GetAnalyzeParameter()	string clientID (see page 66) string device (see page 62) string suite (see page 62) string test (see page 62) string parameterString (see page 62)	This method gets the configuration parameters set in the Analyze panel of the Configuration Panel dialog box for a specified suite or measurement.	The return value is the configuration parameter set in the Analyze panel of the Configuration Panel dialog box for a specified suite or measurement.	Gets the value for the Analyze Parameter. mClient.GetAnalyzeParameter(clientID, "HDM Physical Layer Solution", "Source", "HF1-2- TRISE, TFALL", "Signal Type")
GetAcquireParameter()	string clientID (see page 66) string device (see page 62) string suite (see page 62) string test (see page 62) string parameterString (see page 62)	This method gets the configuration parameters set in the Acquire panel for a specified suite or measurement.	The return value is the configuration parameter set in the Acquire panel for a specified suite or measurement.	Gets the value for the Acquire Parameter. mClient.GetAcquireParameter(clientID, "HDM Physical Layer Solution", "Source", "HF1-2- TRISE, TFALL", "Short Record-length for VLow$Record Length (M)")

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 parameter example**
>
> returnval=mClient.GetAnalyzeParameter(clientID, "Transmitter", "Drive", "5.3.9 - Random Jitter (RJ)", "Clock Edge")

Select a test

Command name	Parameters	Description	Return value	Example

SelectTest()	string clientID (see page 66)
	
	string device (see page 62)
	
	string suite (see page 62)
	
	string test (see page 62)
	
	bool isSelected (see page 64)
		This method selects or deselects a specified test.
		
		If this Setting parameter is set to true, you can select a measurement.
		
		If this Setting parameter is set to false, you can deselect a measurement.
			String that displays the status of the operation after it has been performed
			
			The return value is "" (an empty String) on success.
				m_Client = new Client()
				//m_Client is a reference to the Client class in the Client DLL
				returnval as string
				
				For Source:
				
				SelectTest(clientId, device, Suite, test name, true)
				
				m_Client.SelectTest(clientID, "HDM Physical Layer Solution", "Source", "HF1-2- TRISE, TFALL", true);

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 66) string device (see page 62) string suite (see page 62) bool isSelected (see page 64)	This method selects or deselects a specified suite. When this parameter is set to true, you can select a suite. When this parameter is set to false, you can deselect a suite.	String that gives the status of the operation after it has been performed The return value is "" (an empty String) on success.	m_Client.SelectSuite(clientID, "HDM Physical Layer Solution", "Source", true)

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 66)	Runs the selected tests Note (see page 63) 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 66)	Stops the currently running tests Note (see page 64)	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:

Code	Value	Description
-1	FAIL	The operation failed
1	SUCCESS	The operation succeeded
2	NOT FOUND	Server not found
3	LOCKED	The server is locked by another client, so the operation cannot be performed
4	UNLOCK	The server is not locked; lock the server before performing the operation
0	NULL	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 66)	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.GetTime-Out()
SetTimeOut()	string clientID (see page 66) string time (see page 64)	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.SetTime-Out(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
ApplicationStatus()	string clientID (see page 66)	This method gets the status of the server application. The states are Ready, Running, Paused, Wait, or Error. (see page 60)	String value that gives the status of the server application	m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string returnval=m_Client.Applica-tionStatus(clientID)
QueryStatus()	string clientID (see page 66) out string[] status (see page 63)	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...".	m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string Query status example (see page 79)

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 66) out string Wait-ingMsbBx-Caption (see page 65) out string Wait-ingMsbBxMes-sage (see page 65) out string[] WaitingMsb-BxButtontexts (see page 65)	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.	m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL mClient.GetCurrentState-Info(clientID, WaitingMsbBx-Caption, WaitingMsbBxMes-sage, WaitingMsbBxButton-texts)
SendResponse() **NOTE.** *This command is used when the application is running and is in the wait or error state.*	string clientID (see page 66) out string Wait-ingMsbBx-Caption (see page 65) out string Wait-ingMsbBxMes-sage (see page 65) string Wait-ingMsbBxRe-sponse (see page 65)	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.	m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL mClient.SendResponse(cli-entID, WaitingMsbBxCaption, WaitingMsbBxMessage, WaitingMsbBxResponse)

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 status example**
>
> returnVal=m_Client.QueryStatus(clientID, out statusMessages)
>
> if ((OP_STATUS)returnVal == OP_STATUS.SUCCESS)
>
>   return "Status updated..."
>
> else
>
>   return CommandFailed(returnVal)

After the test is complete

Command name	Parameters	Description	Return value	Example
GetPassFailStatus()	string clientID (see page 66) string device (see page 62) string suite (see page 62) string test (see page 62)	This method gets the pass or fail status of the measurement after test completion. **NOTE.** *Execute this command after completing the measurement.*	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	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, "HF1-1- VL and VSwing")
GetResultsValue()	string clientID (see page 66) string device (see page 62) string suite (see page 62) string test (see page 62) string parameterString (see page 62)	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	m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as srting returnval=m_Client.GetResultsValue(clientID,device,suite, "HF1-1- VL and VSwing")
GetResultsValueForSubMeasurements()	string clientID (see page 66) string device (see page 62) string suite (see page 62) string test (see page 62) Results Panel Column Name int rowNr (see page 63)	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	m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string Get results for a submeasurement example (see page 81)

Command name	Parameters	Description	Return value	Example
GetReportParameter()	string clientID (see page 66) string device (see page 62) string suite (see page 62) string test (see page 62) string parameterString (see page 81)	This method gets the general report details such as oscilloscope model, TekExpress version, and HDM version.	The return value is the oscilloscope model, TekExpress version, and HDM version.	m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string **Oscilloscope Model** returnval=m_Client.GetReportParameter(clientID,"Scope Model") **TekExpress Version** returnval=m_Client.GetReportParameter(clientID,"TekExpress Version") **HDM Version** returnval=m_Client.GetReportParameter(clientID,"Application Version")
TransferReport()	string clientID (see page 66) string filePath (see page 65)	This method transfers the report generated after the run. The report contains the summary of the run. The client must provide the location where the report is to be saved at the client-end.	String that gives the status of the operation after it has been performed Transfers all the result values in the form of a string	m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string returnval=m_Client.TransferReport(clientID,"C:\Report")
TransferImages()	string clientID (see page 66) string filePath (see page 65)	This method transfers all the images (screen shots) from the specified client and folder for the current run (for a suite or measurement). *NOTE. Every time you click Start, a folder is created in the X: drive. Transfer the waveforms before clicking Start.*	String that gives the status of the operation after it has been performed Transfers all the images in the form of a string	m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string returnval=m_Client.TransferImages(clientID, "C:\Waveforms")

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 HDM version

Get results for a submeasurement example

This example returns the specified submeasurement results for test HF1-2- TRISE, TFALL.

returnval=m_Client.GetResultsValue(clientID," "HDM Physical Layer Solution", "Source", "HF1-2-TRISE, TFALL","Value",0)

returnval=m_Client.GetResultsValue(clientID," "HDM Physical Layer Solution", "Source", "HF1-2-TRISE, TFALL","Value",1)

Save, recall, or query a saved session

Command name	Parameters	Description	Return value	Example
CheckSession-Saved()	string clientID (see page 66) out bool saved (see page 61)	This method checks whether the current session is saved.	Return value is either True or False	m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string returnval=m_Client.Check-SessionSaved(m_clientID, out savedStatus)
RecallSession()	string clientID (see page 66) string name (see page 63)	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..."	m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string returnval=m_Client.RecallSes-sion(clientID, savedSession-Name)
SaveSession()	string clientID (see page 66) string name (see page 63)	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 Sav-ed..."/"Failed..."	m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string returnval=m_Client.SaveSes-sion(clientID, desiredSession-Name)
SaveSessionAs()	string clientID (see page 66) string name (see page 64)	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..."	m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string returnval=m_Client.SaveSes-sionAs(clientID, desiredSes-sionName)

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 66)	This method unlocks the server from the client. The ID of the client to be unlocked must be provided. Note (see page 65)	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 66)	This method disconnects the client from the server. Note (see page 61)	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.Discon-nect(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...".

About Algorithms

For all measurements, use the following guidelines to set up the oscilloscope.

Oscilloscope Setup Guidelines

The following guidelines are generalized.

- The signal is any channel, reference, or math waveform.

- The vertical scale for the waveform must be set so that the waveform does not exceed the vertical range of the oscilloscope.

- The sample rate must be set to capture sufficient waveform detail and avoid aliasing.

- Longer record lengths increase measurement accuracy but the oscilloscope takes longer to measure each waveform.

HF1-2: Source TMDS Electrical – 6G – T_{RISE}, T_{FALL}

This test confirms that the rise times and fall times on the TMDS differential signals fall within the limits of the specification.

Required Test Equipment

In addition to the DUT, you will need the following:

- Digital Storage Oscilloscope: DPO/DSA/MSO Series Oscilloscope with BW greater than or equal to 16 GHz

- Four Differential Probes – P7313SMA, for Single Ended and Differential tests

- Test Fixture: Yazaki fixture TPA-P Source to be used for Source testing

- Power supply: PWS4205 or PWS2185 (if using external Power Supply option)

Connect the equipment as shown in .

Set the DUT to operate in normal mode.

Measurement Algorithm

The algorithm calculates the rise and fall time of the waveform as per the steps below.

1. Given a waveform, find the V_{High} and V_{Low}:

2. Measure V_{Low} –

    a. Capture 1000 or more repetitions, triggered at the vertical mid-point of the High-to-Low transition of a H-L-L-L bit sequence. Each capture must be of duration 3*TBIT.

    b. Display the voltage (vertical) histogram on the scope, with the histogram data accumulated only from the last 2-bits of the H-L-L-L sequence.

    c. Read the V_{Low} value as the most common low-level (mode) voltage shown on the histogram.

3. Measure V_{High}

    a. Capture 1000 or more repetitions, triggered at the vertical mid-point of the High-to-Low transition of a L-H-H-H bit sequence. Each capture must be of duration 3*TBIT.

    b. Display the voltage (vertical) histogram on the scope, with the histogram data accumulated only from the last 2-bits of the L-H-H-H sequence.

    c. Read the V_{High} value as the most common low-level (mode) voltage shown on the histogram.

4. Measure T_{RISE} as the mode of the sampled edge times from 20% to 80% of the differential swing voltage (V_{High}) rising edge.

5. Measure T_{FALL} as the mode of the sampled edge times from 80% to 20% of the differential swing voltage (V_{Low}) on the falling edge.

HF1-3: Source TMDS Electrical – 6G – Inter-Pair Skew

This test confirms that the skew between two data lanes on the TMDS differential signals fall within the limits of the specification.

Required Test Equipment

In addition to the DUT, you will need the following:

- Digital Storage Oscilloscope: DPO/DSA/MSO Series Oscilloscope with BW greater than or equal to 16 GHz

- Four Differential Probes – P7313SMA, for Single Ended and Differential tests

- Test Fixture: Yazaki fixture TPA-P Source to be used for Source testing

- Power supply: PWS4205 or PWS2185 (if using external Power Supply option)

Connect the equipment as shown in .

Measurement Algorithm

The algorithm calculates the skew between the differential pairs in the TMDS portion as per the below steps. Inter pair skew test is a differential test and is only for data signals. Clock signal is excluded for this test, but is used for calculating the Tbit.

1. Connect and acquire the Data 0 and Data 1 signals.

2. Set the Digital Oscilloscope horizontal setting: sampling rate >= 25 Gs/sec and Record Length set to 100M.

3. Decode the bit stream of Data 0 and Data 1.

4. Search for the 20 bit unscrambled control code (SSCP) on Data 0 and Data 1. SSCP might have one of the following 10 bit values:

    - 1101010100

    - 0010101011

    - 0101010100

    - 1010101011

5. Stop acquisition and measure TSKEW between channels.

6. Repeat the test for the remaining combinations of TMDS_DATAx pairs.

HF1-6: Source TMDS Electrical – 6G – Clock Duty Cycle and Clock Rate

This test confirms that the duty cycle and clock rate of the TMDS differential clock complies with the limits mentioned in specification.

Required Test Equipment

In addition to the DUT, you will need the following:

■ Digital Storage Oscilloscope: DPO/DSA/MSO Series Oscilloscope with BW greater than or equal to 16 GHz

■ Four Differential Probes – P7313SMA, for Single Ended and Differential tests

■ Test Fixture: Yazaki fixture TPA-P Source to be used for Source testing

■ Power supply: PWS4205 or PWS2185 (if using external Power Supply option)

Connect the equipment as shown in .

Set the DUT to operate in normal mode.

Measurement Algorithm

The algorithm calculates the Clock Duty Cycle of the waveform as per the below steps.

1. DPOJET measurement is used to perform this test.

2. The Duty Cycle (Max) and Duty Cycle (Min) measurements calculate the difference in positive (or negative) pulse widths from one cycle to the next.

3. The application calculates these measurements using the following equations:

$$\Delta W_n^+ = \Delta W_n^+ - \Delta W_{n-1}^+$$

$$\Delta W_n^- = \Delta W_n^- - \Delta W_{n-1}^-$$

Where:

  ▬ $\Delta W+$ is the difference between positive pulse widths of adjacent clock cycles.

  ▬ $\Delta W-$ is the difference between negative pulse widths of adjacent clock cycles.

  ▬ $W+$ is the positive pulse width measurement.

  ▬ $W-$ is the negative pulse width measurement.

HF1-7: Source TMDS Electrical – 6G – Clock Jitter

This test confirms that the clock jitter of the TMDS differential clock complies with the limits mentioned in specification.

Required Test Equipment

In addition to the DUT, you will need the following:

- Digital Storage Oscilloscope: DPO/DSA/MSO Series Oscilloscope with BW greater than or equal to 16 GHz

- Four Differential Probes – P7313SMA, for Single Ended and Differential tests

- Test Fixture: Yazaki fixture TPA-P Source to be used for Source testing

- Power supply: PWS4205 or PWS2185 (if using external Power Supply option)

Connect the equipment as shown in <u>Source differential diagram (see page 8)</u>.

Set the DUT to operate in normal mode.

Measurement Algorithm

TekExpress HDM automatically executes the calculations described below.

1. Acquire differential clock signal with same settings as before: Compliance Mode: 50 GS/sec Compliance Mode: 18M.

2. Apply 12.5x BW enhancement filter and save the wfm: - C:\TekScope\Math Arbitrary Filters\Probe-Filters\P7313_SMA_12.5x_16GHz_50GSs.flt and save this wfm as Clk_BWE.wfm.

3. Pass Clk_BWE.wfm from step 2 to algorithm for swing calculation.

4. Swing limits: If (Vs > 1200 mV) OR (Vs < 400 mV) then FAIL.

5. Apply Cable effect on Clk_BWE.wfm from step 2, that is apply sdlaTp2clk.flt filter. Save this wfm as Clock_CableEffect.wfm.

   - Recall Clk_BWE.wfm on Ref1

   - Load Arbfilt1 = sdlaTp2clk.flt

   - Math1 = Arbfilt1(Ref1)

   - Save Math1 as Clock_CableEffect.wfm

6. Recall Clock_CableEffect.wfm on Ref1

7. Load load arbfilt1 = HDMI2_EQ_50G.flt

8. Math1 = arbfilt1(Ref1)

9. Pass Math1 from step 8 to DPOJET for jitter calculation. DPOJET settings are same as before:

   ‒ Edge->Rising->50% TIE CH1

   ‒ Signal Type: Clock

   ‒ Clock Edge: Rise

   ‒ Clock Recovery => Method: PLL Custom BW

   ‒ PLL Model: Type I

   ‒ Loop BW :4 MHz

   ‒ Filter Spec : I st Order

   ‒ Freq (F1) : 4 MHz

10. If Clock jitter exceeds 0.3*TBIT then FAIL.

HF1-8: Source TMDS Electrical – 6G – Data Eye Diagram

This test confirms that the HDMI Data output has signal quality that meets the eye opening required by the specification.

Required Test Equipment

In addition to the DUT, you will need the following:

■ Digital Storage Oscilloscope: DPO/DSA/MSO Series Oscilloscope with BW greater than or equal to 16 GHz

■ Four Differential Probes – P7313SMA, for Single Ended and Differential tests

■ Test Fixture: Yazaki fixture TPA-P Source to be used for Source testing

■ Power supply: PWS4205 or PWS2185 (if using external Power Supply option)

Connect the equipment as shown in .

Set the DUT to operate in normal mode.

Measurement Algorithm

TekExpress HDM automatically executes the calculations described below.

This is a DPOJET based measurement.

CTS Procedure

■ Source Eye Diagram test is measured at TP2_EQ.

■ TP2 is the signal after passing along a worst cable.

■ Worst cable has worst attenuation and skew of 112 ps.

Measurement procedure

1. Connect Clock of the HDMI 2.0 source DUT to Ch1 of the oscilloscope. (Adding Type 4 Cable Emulator and inserting 112 ps delay on all TMDS data positive lanes.)

2. Connect Data0 (0/1/2/) +ve of the HDMI 2.0 Source to Ch3 of the oscilloscope.

3. Connect Data0 (0/1/2/) -ve of the HDMI 2.0 Source to Ch4 of the oscilloscope.

4. On Clock signal :

   ▬ Load Arbfilt1 = sdlaTp3DIFFSrc1.flt – sdlaTp2clk.flt

   ▬ Math1 = Arbfilt1 (Ch1)

   ▬ Save Math1 as 1-7 TMDS Data Eye Diagram_Clock.wfm

5. On Data signal, load below filter files:

   ▬ ArbFilt1 = sdlaTp2ADp.flt

   ▬ ArbFilt2 = sdlaTp2ADn.flt

   ▬ ArbFilt3 = sdlaTp2BDp.flt

   ▬ Arbfilt4 = sdlaTp2BDn.flt

6. Define Math waveforms

   ▬ Math2 = Arbfilt1 (Ch3) + Arbfilt2 (Ch4)

   ▬ Math3 = ArbFilt3 (Ch3) + Arbfilt4 (Ch4)

   ▬ Save Math4 = Math 2 – Math3 as 1-7 TMDS Data Eye Diagram_Data.wfm

7. Clear all Maths and filter file definitions.

8. Recall 1-7 TMDS Data Eye Diagram_Clock.wfm on Ref1.

9. Recall 1-7 TMDS Data Eye Diagram_Data.wfm on Ref2.

10. Load arbfilt1 = HDMI2_EQ_50G.flt.

    Math1 = arbfilt1 (Ref1)

    Math2 = arbfilt1 (Ref2)

11. Perform DPOJET ->Mask Hits measurements with below settings:

- Clock Recovery => Method: Explicit Clock – PLL,
- Clock Source: Math1
- Clock Edge: Rise
- Clock Multiplier: 40
- PLL Model: Type I
- Damping: 700 m
- Loop BW: 4 MHz
- Mask File = Data Rate_5.92 Gbps.msk
- Clock offset = Auto
- BitType = All
- Plot = Eye Diagram

12. Perform DPOJET ->Eye width measurements with below settings:

- Clock Recovery => Method: Explicit Clock – PLL,
- Clock Source: Math1
- Clock Edge: Rise
- Clock Multiplier: 40
- PLL Model: Type I
- Damping: 700 m
- Loop BW: 4 MHz
- Clock offset = Auto
- Bit Type = All
- Measurement setting ->Cursor gating ON

13. Enable the measurement with vertical cursors on at 5% and 95% of the oscilloscope acquisition.

14. Perform the above measurement on Data1 and Data2 by repeating the steps from (2) to (13).

(Adding Type4 Cable Emulator and inserting 112 psec delay on all TMDS data negative lanes.)

1. Connect Clock of the HDMI 2.0 source DUT to Ch1 of the oscilloscope.

2. Connect Data0 (0/1/2/) +ve of the HDMI 2.0 Source to Ch3 of the oscilloscope.

3. Connect Data0 (0/1/2/) -ve of the HDMI 2.0 Source to Ch4 of the oscilloscope.

4. On Clock signal:

    -   Load Arbfilt1 = sdlaTp3DIFFSrc1.flt – sdlaTp2clk.flt

    -   Math1 = Arbfilt1 (Ch1)

    -   Save Math1 as 1-7 TMDS Data Eye Diagram_Clock.wfm

5. On Data signal, load these filter files:

    -   ArbFIlt1 = sdlaTp2ADp.flt

    -   ArbFilt2 = sdlaTp2ADn.flt

    -   Arbfilt3 = sdlaTp2BDp.flt

    -   Arbfilt4 = sdlaTp2BDn.flt

6. Define Math waveforms:

    -   Math2 = Arbfilt1 (Ch4) + Arbfilt2 (Ch3)

    -   Math3 = ArbFilt3 (Ch4) + Arbfilt4 (Ch3)

    -   Save Math4 = Math 3 – Math2 as 1-7 TMDS Data Eye Diagram_Data.wfm

7. Clear all Maths and filter file definitions.

8. Recall 1-7 TMDS Data Eye Diagram_Clock.wfm on Ref1

9. Recall 1-7 TMDS Data Eye Diagram_Data.wfm on Ref2

10. Load arbfilt1 = HDMI2_EQ_50G.flt

    Math1 = arbfilt1(Ref1)

    Math2 = arbfilt1(Ref2).

11. Perform DPOJET ->Auto Fit Mask Hits with below settings:

    -   Clock Recovery => Method: Explicit Clock – PLL,

    -   Clock Source: Math1

    -   Clock Edge: Rise

    -   Clock Multiplier: 40

    -   PLL Model: Type I

    -   Damping: 700 m

    -   Loop BW: 4 MHz

    -   Mask File = Data Rate_5.92 Gbps.msk

    -   Clock offset = Auto

    -   BitType = All

    -   Plot = Eye Diagram

12. Perform DPOJET ->Eye width measurements with below settings:

- Clock Recovery => Method: Explicit Clock – PLL,

- Clock Source: Math1

- Clock Edge: Rise

- Clock Multiplier: 40

- PLL Model: Type I

- Damping: 700 m

- Loop BW: 4 MHz

- Clock offset = Auto

- Bit Type = All

- Measurement setting ->Cursor gating ON

13. Enable the measurement with vertical cursors on at 5% and 95% of the scope acquisition. Perform the above measurement on Data 1 and Data 2 by repeating the steps from (2) to (13). Repeat the test for remaining supported test character rates. Only one video format/pixel-size combination is required per TMDS character rate.

TP2 mask

Table 17: Mask coordinates at different data rates

TMDS Bit Rate (Gbps)	H (Tbit)	V (mV)
3.71	0.6	335
4.46	0.56	295
5.94	0.4	150

HDM Data Eye Diagram Test - Mask Movement

When performing the HDM Data Eye Diagram test (Source), if the test fails due to mask hits, you are presented with a dialog box to enter a timing value that will be used to shift the Mask horizontally, thus allowing the test to pass. You will have up to three opportunities to adjust the mask position. If, after three attempts, the test still fails due to any mask hits, the test will stop and report the test as failed. The first message you see when the Eye Diagram test fails due to mask hits provides a short description of what you need to adjust the mask position.
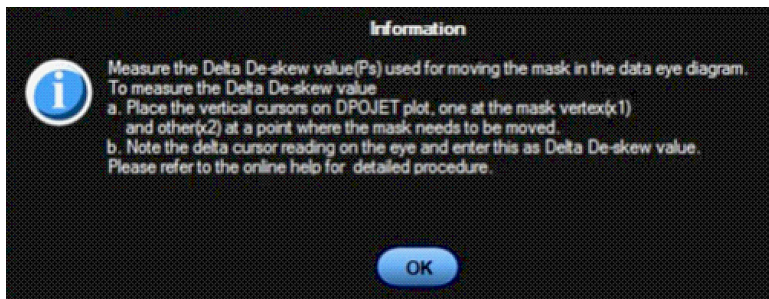
After you select OK, the Measure the Deskew Value dialog box displays. Once you determine the amount of time (in picoseconds) to shift the mask horizontally, enter the value here.

Measure the Deskew Value

Enter the measured Delta Deskew value(Ps) for moving the mask in the data eye diagram.

Delta Deskew Value(Ps)   -65

OK   Cancel
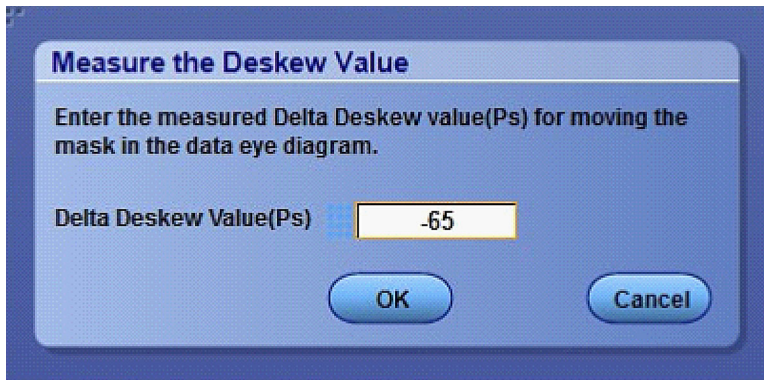
HF1-1: Source TMDS Electrical – 6G – V_L and V_{Swing}

This test confirms that the low voltage level of the TMDS single-ended signals and the Swing of the single-ended signals complies with the limits listed in the specification.

Required Test Equipment

In addition to the DUT, you will need the following:

- Digital Storage Oscilloscope: DPO/DSA/MSO Series Oscilloscope with BW greater than or equal to 16 GHz

- Four Differential Probes – P7313SMA, for Single Ended and Differential tests

- Test Fixture: Yazaki fixture TPA-P Source to be used for Source testing

- Power supply: PWS4205 or PWS2185 (if using external Power Supply option)

Connect the equipment as shown in .

Set the DUT to operate in normal mode.

Measurement Algorithm

The algorithm calculates the rise and fall time of the waveform as per the below steps.

Given a waveform, find the V_{High} and V_{Low}

1. Measure V_{Low}:

   a. Capture 1000 or more repetitions, triggered at the vertical mid-point of the High-to-Low transition of a H-L-L-L bit sequence. Each capture must be of duration 3*TBIT.

   b. Display the voltage (vertical) histogram on the scope, with the histogram data accumulated only from the last 2-bits of the H-L-L-L sequence.

   c. Read the V_{Low} value as the most common low-level (mode) voltage shown on the histogram

   d. Capture 1,000 or more repetitions, triggered at the vertical mid-point of the Low-to-High transition of a L-H-H-H bit sequence. Each capture shall be of duration 3*Tbit.

   e. Display the voltage (vertical) histogram on the oscilloscope, with the histogram data accumulated only from the last 2-bits of the L-H-H-H sequence.

   f. Read the V_{High} value as the most common high-level voltage shown on the histogram.

   g. Calculate $V_{Swing} = V_{High} - V_{Low}$.

2. In the case of Data Signals:

   - If (V_{Low} < 2.30 V) OR (2.90 V < V_{Low}), then FAIL.

   - If (V_{Swing} < 400 mV) or (600 mV < V_{Swing}), then FAIL.

3. In the case of Clock Signals:

   - If (VL_{Low} < 2.30 V) OR (3.10 V < V_{Low}), then FAIL.

   - If (V_{Swing} < 200 mV) or (600 mV < V_{Swing}), then FAIL.

4. Repeat the test for all eight TMDS signals.

HF1-4: Source TMDS Electrical – 6G – Intra-Pair Skew

This test confirms that the skew between the positive and negative lane of the TMDS single ended signals complies with the limits mentioned in specification.

Required Test Equipment

In addition to the DUT, you will need the following:

- Digital Storage Oscilloscope: DPO/DSA/MSO Series Oscilloscope with BW greater than or equal to 16 GHz

- Four Differential Probes – P7313SMA, for Single Ended and Differential tests)

- Test Fixture: Yazaki fixture TPA-P Source to be used for Source testing

- Power supply: PWS4205 or PWS2185 (if using external Power Supply option)

Connect the equipment as shown in .

Set the DUT to operate in normal mode.

Measurement Algorithm

TekExpress HDM automatically executes the calculations described below.

The algorithm calculates the skew within any Differential pair in the TMDS portion as per below steps. Intra-pair skew test is a single-ended measurement.

1. Capture D0+, D0- single-ended signals.

2. Search for at least 1000 occurrences of HLLL pattern on D0+ and LLLH on D0-. Capture these as qualified region for the measurement.

3. Measure the average skew between the corresponding rising edges of D0+ and falling edges of D0- of the qualified region.

4. Repeat the test for all remaining TMDS differential pairs.

HF1-5: Source TMDS Electrical – 6G – Differential Voltage

This test confirms that the voltage level of the TMDS differential signals complies with the limits in the specification.

Required Test Equipment

In addition to the DUT, you will need the following:

■ Digital Storage Oscilloscope: DPO/DSA/MSO Series Oscilloscope with BW greater than or equal to 16 GHz

■ Four Differential Probes – P7313SMA, for Single Ended and Differential tests

■ Test Fixture: Yazaki fixture TPA-P Source to be used for Source testing

■ Power supply: PWS4205 or PWS2185 (if using external Power Supply option)

Connect the equipment as shown in [Source differential diagram (see page 8)](#).

Set the DUT to operate in normal mode.

Measurement Algorithm

The algorithm calculates the Differential voltage on Differential TMDS lanes clock, D0, D1, D2 as per below steps.

Given a waveform, find the Maximum Differential Voltage and Minimum Differential Voltage.

Measure Minimum Differential Voltage:

1. Capture 1000 or more repetitions, triggered at the vertical mid-point of the High-to-Low transition of a H-L-L-L bit sequence. Each capture must be of duration 3*TBIT.

2. Display the voltage (vertical) histogram on the scope, with the histogram data accumulated only from the last 2-bits of the H-L-L-L sequence.

3. Read the Minimum Differential Voltage value as the most common low-level (mode) voltage shown on the histogram.

Measure Maximum Differential Voltage:

1. Capture 1000 or more repetitions, triggered at the vertical mid-point of the High-to-Low transition of a L-H-H-H bit sequence. Each capture must be of duration 3*TBIT.

2. Display the voltage (vertical) histogram on the scope, with the histogram data accumulated only from the last 2-bits of the L-H-H-H sequence.

3. Read the Maximum Differential Voltage value as the most common low-level (mode) voltage shown on the histogram.

HF2-1: Min/Max Differential Swing Tolerance

This HDMI Receiver Test confirms that the Sink properly supports TMDS differential voltages at minimum and maximum levels.

Required Test Equipment

In addition to the DUT and high-bandwidth digital oscilloscope, you will need the following:

- HDMI Signal Generator: AWG70002A with option 01, 03 and 225 (quantity 2) OR AWG7122C with option 1, 2/6, 8 (quantity 2) and Option SDX with Direct Synthesis capability, to simulate Cable emulator effect and Rise time filter effect.

- Rise time Filter: 2 Picosecond Pulse labs 120 ps

- Test Fixture: 1 set Yazaki TPA-P (Part Number)

- DC Power Supply: Tek Power Supply or any other

- Bias Tees: 8 Mini Circuits Bias Tees ZX85–12G-S+

- A/V Display

Measurement Algorithm

The below procedure is fully automated by Tektronix HDM Compliance Software (Option HDM).

1. Connect the equipment as shown in .

2. Calibrate the HDM Signal Generator AWG7K/70K output for the following settings.

   a. Frequency: Highest frequency supported by the Sink DUT

   b. Pattern: HDMI Gray Ramp data streams

   c. Rise/fall times:

      Set the rise/fall time of the TMDS CLOCK to between 75 ps and 110 ps.

      Set the rise/fall time of the TMDS DATA between 42.5 ps and 65 ps.

   d. Set the common mode voltage (VICM) to 3.1 V.

   e. Set the differential swing voltage to 150 mV. Confirm the Sink DUT support. If the Sink DUT does not support the transmitted resolution, then FAIL: "Min diff swing unsupported at VICM1 range".

   f. Change VICM to 3.3 V. Confirm that the Sink DUT continues to support the transmitted image without errors. If the Sink DUT does not support the transmitted resolution, then FAIL: "Min diff swing unsupported at VICM2 range".

   g. Connect the equipment as shown in .

Note: For AWG7122C, the connection remains the same as the nominal pattern.

h. Set the common mode voltage (VICM) to 2.6 V.

i. Change the differential voltage to 1200 mV (600 mV/single-ended signal = max swing).

j. Verify that the Sink DUT continues to support the transmitted image without errors.

k. If the Sink DUT does not support the transmitted image, then FAIL: "Max diff swing unsupported at VICM1 range".

l. Change VICM to 3.3 V. Verify that the Sink DUT continues to support the transmitted image without errors.

m. If the Sink DUT fails to support the transmitted image, then FAIL: "Max diff swing unsupported at VICM2 range".

HF2-2: Intra-Pair Skew Test

This HDMI Receiver test confirms that the maximum allowed timing skew within each TMDS pair is supported by the Sink DUT.

Required Test Equipment

In addition to the DUT and high-bandwidth digital oscilloscope, you will need the following:

- HDMI Signal Generator: AWG70002A with option 01, 03 and 225 (quantity 2) OR AWG7122C with option 1, 2/6, 8 (quantity 2) and Option SDX with Direct Synthesis capability, to simulate Cable emulator effect and Rise time filter effect.

- Rise time Filter: 2 Picosecond Pulse labs 120 ps

- Test Fixture: 1 set Yazaki TPA-P (Part Number)

- DC Power Supply: Tektronix Power Supply or any other

- Bias Tees: 8 Mini Circuits Bias Tees ZX85–12G-S+

- A/V Display

Measurement Algorithm

The below procedure is fully automated by Tektronix HDM Compliance Software (Option HDM).

1. Connect the equipment as shown in .

2. Calibrate the HDM Signal Generator AWG7K/70K output for the following settings.

a. Frequency: Highest frequency supported by the Sink DUT

b. Pattern: HDMI Gray Ramp data streams

c. Rise/fall times:

Set the rise/fall time of the TMDS CLOCK to between 75 ps and 110 ps.

Set the rise/fall time of the TMDS DATA between 42.5 ps and 65 ps.

   d. Set VICM to 2.8 V.

   e. Set Single Ended Swing to 500 mV.

3. Do the following for each TMDS differential pair Clock and Data:

   a. Apply worst skew at TP2 to the pair being tested:

Set skew of .15*Tbit on one line against the other line.

Apply the Worst Cable Emulator after the TMDS Signal Generator to cause 112 ps signal delay on the line of the TMDS differential pair against the other line.

   b. Enable Scrambling mode on DUT.

   c. Operate the TMDS Signal Generator to output the tested Video Format.

   d. If the Sink DUT fails to adequately support the transmitted image, then FAIL.

   e. Switch the setting of the signal delay on the other line for the tested TMDS differential pair.

   f. If the Sink DUT fails to adequately support the transmitted image, then FAIL

   g. Repeat the test for the remaining untested pairs.

   h. For all TMDS signals, set (VICM) to 3.3 V and repeat the procedure.

HF2-3: Jitter Tolerance Test

This HDMI Receiver test confirms that the maximum allowed clock jitter on TMDS signal is supported by the Sink DUT.

Required Test Equipment

In addition to the DUT and high-bandwidth digital oscilloscope, you will need the following:

- HDMI Signal Generator: AWG70002A with option 01, 03 and 225 (quantity 2) OR AWG7122C with option 1, 2/6, 8 (quantity 2) and Option SDX with Direct Synthesis capability, to simulate Cable emulator effect and Rise time filter effect.

- Rise time Filter: 2 Picosecond Pulse labs 120 ps

- Test Fixture: 1 set Yazaki TPA-P (Part Number)

- DC Power Supply: Tektronix Power Supply or any other

- Bias Tees: 8 Mini Circuits Bias Tees ZX85–12G-S+

- A/V Display

Measurement Algorithm

The below procedure is fully automated by Tektronix HDM Compliance Software (Option HDM).

1. Connect the equipment as shown in .

2. Calibrate the HDM Signal Generator AWG7K/70K output for the following settings.

    a. Frequency: Highest frequency supported by the Sink DUT

    b. Pattern: HDMI Gray Ramp data streams

    c. Rise/fall times:

        Set the rise/fall time of the TMDS CLOCK to between 75 ps and 110 ps.

        Set the rise/fall time of the TMDS DATA between 42.5 ps and 65 ps.

    d. Set VICM to 2.9 V.

    e. Set Single Ended Swing to 400 mV.

    f. Below are the jitter frequencies inserted on the signal:

        D_JITTER = 500 kHz (on TMDS_CLOCK), C_JITTER = 10 MHz.

        D_JITTER = 1 MHz (on TMDS_CLOCK), C_JITTER = 7 MHz.

NOTE. *All jitter amounts described below are relative to a recovered clock as measured with a Clock Recovery Unit).*

All four jitter frequencies define sinusoidal jitter). [Make TP2 worst condition]

---

    g. Apply the Worst Cable Emulator under the condition that the Worst Cable Emulator causes delay on all TMDS DATA positive (+) lines against negative (−) lines.

    h. Generate TP2 signal and adjust the jitter amplitude as follows:

        C_JITTER to be 0.3*Tbit at TP2 after processing with the CRU and applying the Reference Cable Equalizer for 3.4 Gbps–6 Gbps.

        Data waveform (eye diagram) to nearly touch the 2 corners (right and left) points of the TP2_EQ eye mask, but without causing a TP2_EQ eye mask violation after processing with the CRU and applying the Reference Cable Equalizer for 3.4 Gbps–6 Gbps.

    i. Connect the TMDS Signal Generator to the Sink DUT.

        a.   Use the I2C Analyzer to write 1 to the Scrambling_Enable bit of the Sink DUT.

        b. Operate the TMDS Signal Generator to output the tested Video Format.

        c. If the CDF field Sink_Error_Counter is "Y" and the DUT passed all tests in Section 8.2.3 (Test ID HF2-18, HF2-19, HF2-20), then measure the error rate using the Error Counter. If the Error rate is more than 10-9, then FAIL.

HF2-23: Sink Pixel Decoding YCBCR 4:2:0

This test confirms that the Sink DUT supports YCbCr 4:2:0 Pixel decoding and signaling.

Required Test Equipment

In addition to the DUT and high-bandwidth digital oscilloscope, you will need the following:

- HDMI Signal Generator: AWG70002A with option 01, 03 and 225 (quantity 2) OR AWG7122C with option 1, 2/6, 8 (quantity 2)

- AFG: 1 AFG3102/AFG3252

- Rise time Filter: 2 Picosecond Pulse labs 120 ps (5115110-120)

- Test Fixture: HDMI 2-0 compatible Fixture and cables: 1 TF-HDM-TPA-S, 1 BNC T Connector, 3 BNC cables

- External power supply: 1– PWS4721 / PWS4602 / PWS4323 / PWS4305 / PWS4205

- Bias Tees: 8 Mini Circuits Bias Tees ZX85–12G-S+

- DDC Master: Any approved DDC Master

- EDID Analyzer: Any approved EDID analyzer

Measurement Algorithm

The below procedure is fully automated by TekExpress HDM Software.

1. If the CDF field Sink_HDMI_YCBCR_420 is "N", then SKIP this test.

2. Connect the Sink DUT to any approved DDC master and EDID analyzer.

3. Turn on the Sink DUT, have the DDC Master output +5 V power and read the sink DUT's EDID in response to a hotplug.

4. If Hotplug Detect signal is not received or EDID is unreadable, then FAIL.

5. Connect the Sink DUT to the 297 MHz Video Generator.

6. Make the connection as shown in .

7. Recall nominal pattern YCBCR 420 8 Bit Pattern with pixel clock frequency 297 MHz -0.5 % (or -0.6% when the frame rate is 60 Hz) with test image as shown in below Figure, AVI infoframe with Y2, Y1, Y0 = 011 and appropriate VIC, synchronize AWG's.

8. If the Image /Video appear to be distorted or disturbed, then Fail.

9. If color bars in the upper half of the active video are not in the order shown in the Figure, then Fail.

10. If black and white bars in the lower half of the active video are not evenly spaced, then Fail.

11. If black and white bars in the lower half of the active video contain a strong blue or red tint when examined at close range, then Fail.

12. Change pixel clock frequency to 297 MHz +0.5%, synchronize AWG's and repeat steps 5 to 8.

13. If steps 5 to 8 pass then the test results are Pass; for all other conditions it is Fail.
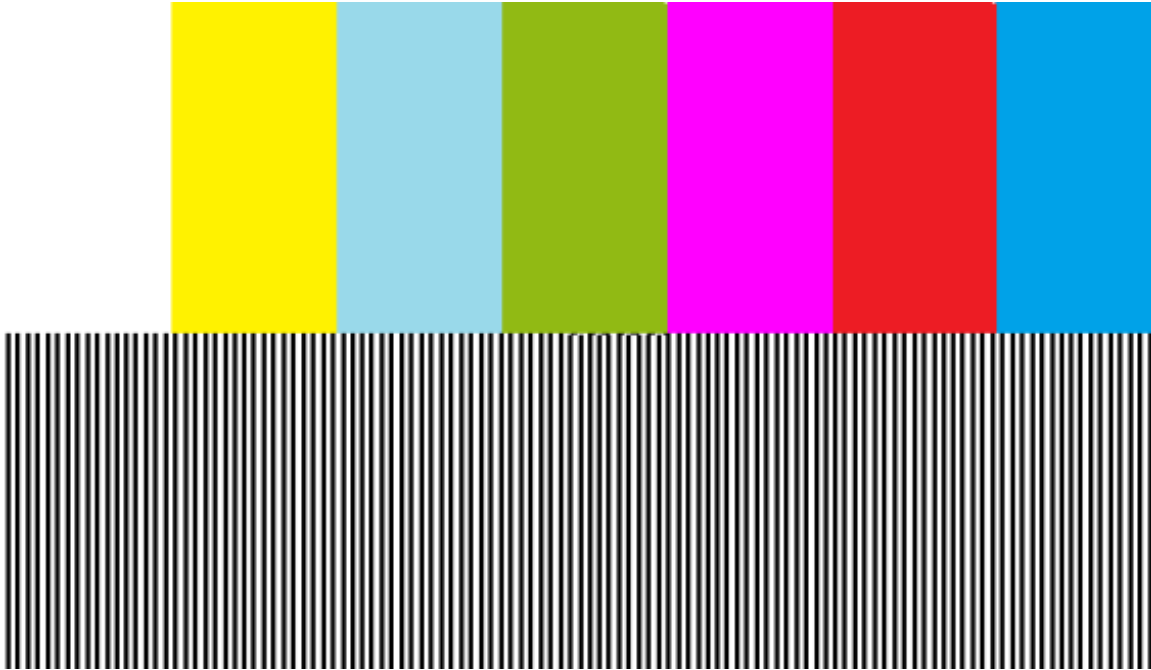
Map the My TekExpress Folder

Map the shared `My TekExpress` folder 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 share name format is `<instrument name><user ID> My TekExpress`. This shared folder is used to save the waveform files and is used during any other file transfer operations.

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**.

Index