TekExpress®
MHL Advanced Analysis and Compliance Solution

Online Help

077-0620-03

Tektronix

TekExpress®
MHL Advanced Analysis and Compliance Solution

Online Help

www.tektronix.com
077-0620-03

Tektronix

TekExpress MHL Advanced Analysis and Compliance Solution Online Help, 076-0256-03.

Contacting Tektronix

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

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

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

Table of Contents

Getting Help and Support

Technical Support 1

Overview and Key Specifications

Overview and Key Specifications 3
Supported Tests: MHL Transmitter 5
Supported Tests: MHL Receiver 8
Supported Resolutions: MHL Transmitter 10
Supported Resolutions: MHL Receiver 11

Operating Basics

Equipment Connection Setup (MHL Transmitter) 14
    Clock and Data Tests 14
Equipment Connection Setup (MHL Receiver) 18
    Sink Tests 18
    Dongle Tests 23
Application Directories and Usage 27
View Test-Related Files 28
File Name Extensions 29

Getting Started

Installing the Software
    Compatibility 31
    Minimum System Requirements 31
    Install the Software 32
    Activate the License 34
    View Version and License Information 35
Starting the Software
    Start the Application 35
    Options Menu 36
    View Connected Instruments: the Instrument Control Settings 37
    Application Controls 39
    Exit the Application 43

Application Basics

Application Basics 45

Setting Up Tests

Setting Up Tests: the Setup Panel 47
Selecting Device Parameters: the DUT tab 52
Choosing Tests: the Test Selection tab 62
Selecting Acquisitions
Acquiring Waveforms: the Acquisitions tab (MHL Transmitter only) 67
Use Prerecorded Waveforms for Analysis (MHL Transmitter only) 72
Acquire Live Waveforms for Analysis (MHL Transmitter only) 73
Configuring Tests
Configuring Tests: the Configuration Screen 74
Compliance Mode or User Defined Mode 78
Measurement Parameter Descriptions (MHL Transmitter) 79
Measurement Parameter Descriptions (MHL Receiver) 83

Running Tests

Before You Click Start 85
Pre-Run Check List 87
Configure Email Notification 87
Running the Tests and Viewing their Progress: the Status Panel 89

Viewing Results

Viewing Test Results: the Results Panel 95

Configuring and Viewing Reports

Configuring and Viewing Reports: the Reports Panel 97

Saving and Recalling Test Setups

About Saving and Recalling Test Setups 101
Saving a Test Setup 101
Recalling a Saved Test Setup 102
Creating a New Test Setup Based on an Existing One 102
Deleting a Test Setup 103

TekExpress Programmatic Interface

About the Programmatic Interface 105
Requirements for Developing TekExpress Client 105
Remote Proxy Object 107
Client Proxy Object 107
Client Programmatic Interface Example 109
Program Example 112
MHL Application Commands
About TekExpress MHL Application Commands 116

Connect Through an IP Address 122
Lock the Server 123
Disable the Popups 124
Set or Get the DUT ID 125
Set the Configuration Parameters for a Suite or Measurement 126
Query the Configuration Parameters for a Suite or Measurement 129
Select a Measurement 131
Select a Suite 138
Set Pixel Mode 138
Set Resolution 139
Set Termination Source Method 141
Set Termination Voltage 142
Run with Set Configurations or Stop the Run Operation 143
Handle Error Codes 144
Get or Set the Timeout Value 145
Wait for the Test to Complete 145
After the Test is Complete 148
Save, Recall, or Check if a Session is Saved 151
Unlock the Server 152
Disconnect from the Server 152

Algorithms

About Algorithms 153
MHL Transmitter Group 1: Clock Tests
3.1.1.1: Standby Output Voltage Test - V_{OFF} 153
3.1.1.5: Common-Mode Output Swing Voltage Test - $V_{CMSWING}$ 155
3.1.1.7: Common-mode Rise and Fall Times Test - T_{R_CM}, T_{F_CM} 156
3.1.1.10: MHL Clock Duty Cycle Test – 24 Bit or Packed Pixel Mode (CTS 1.X); Normal Mode (CTS 2.0) 158
3.1.1.11: MHL Clock Jitter Test (CTS 1.X) – in Normal Mode (CTS 2.0) 159
3.1.1.14: MHL Clock Duty Cycle Test - Packed Pixel Mode 160
3.1.1.15: MHL Clock Jitter Test in Packed Pixel Mode 160
MHL Transmitter Group 2: Data Tests
3.1.1.2: Single-Ended High Level Voltage Test - V_{SE_HIGH} 160
3.1.1.3: Single-Ended Low Level Voltages Test - V_{SE_LOW} 161
3.1.1.4: Differential Output Swing Voltage Test - V_{DF_SWING} 163
3.1.1.6: Differential Rise and Fall Times Test - T_{R_DF}, T_{F_DF} 164
3.1.1.8: Differential Intra-Pair Skew Test - T_{SKEW_DF} 165
3.1.1.12: MHL Data Eye Diagram Test (CTS 1.X) – in Normal Mode (CTS 2.0) 166
3.1.1.16: MHL Data Eye Diagram Test in Packed Pixel Mode 167
MHL Receiver Group 1: Sink Tests
4.1.1.1: Input Signal DC Voltage Level Tolerance Test 167

4.1.1.2: Input Signal Minimum and Maximum Swing Voltages Tolerance Test..... 170
4.1.1.3: Intra-Pair Skew Tolerance Test..... 174
4.1.1.4: Jitter Tolerance Test (CTS 1.X) – in Normal Mode (CTS 2.0) 175
4.1.1.8: Jitter Tolerance Test in Packed Pixel Mode..... 177
MHL Receiver Group 2: Dongle Tests
5.1.1.1: Input Signal Single-Ended Voltage Level Tolerance..... 177
5.1.1.2: Input Signal Minimum and Maximum Swing Voltage Tolerance Test..... 180
5.1.1.3: Intra-Pair Skew Tolerance Test..... 184
5.1.1.4: Jitter Tolerance Test (CTS 1.X) – in Normal Mode (CTS 2.0) 186
5.1.1.9: Jitter Tolerance Test in Packed Pixel Mode..... 188

Reference

MHL Data Eye Diagram Test - Mask Movement 189
Map the My TekExpress Folder..... 191

Index

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 MHL Advanced Analysis and Compliance Solution gives you the tools to easily run Mobile High-definition Link (MHL) tests under the MHL Compliance Test Specifications 1.X and 2.0. It provides a complete and reliable solution for quick testing.

The application functionality is generally divided into two parts: the MHL Transmitter Suite, and the MHL Receiver Suite. You can select which suite and which version of the CTS you would like to use by clicking on the Setup Panel (see page 47) and clicking the DUT tab (see page 52). See Application Basics (see page 45) for more information.

Supported Tests:

TekExpress MHL Transmitter supports these automated Clock and Data Tests. (see page 5)

TekExpress MHL Receiver supports these automated Sink and Dongle Tests. (see page 8)

Additional tests may be performed manually by loading the test patterns on the AWG.

Supported Resolutions:

MHL Transmitter supports these resolutions. (see page 10)

MHL Receiver supports these resolutions. (see page 11)

MHL Sink Protocol Tests based on AWG direct synthesis patterns

NOTE. *Contact your local Tektronix representative for these manual test patterns.*

- 4.2.1.1: Character Synchronization

  This test confirms that the Sink DUT synchronizes if the data stream provides only minimum length Control Periods.

- 4.2.1.2: Packet Types

  This test confirms that the Sink DUT accepts all valid packet types.

- 4.2.2.1: Video Formats

  This test verifies that the sink DUT supports the video formats with no distortion.

- 4.2.2.2: Pixel Encoding

  This test verifies that the sink supports RGB pixel encoding.

- 4.2.2.3: Video Quantization Range

This test verifies that the Sink has the correct QY and QS bits in the Video Capability Data Block, and the Sink displays with correct Video Quantization Range.

- 4.2.3.1: Audio Test IEC 60958 / IEC61937

  This test verifies that the Sink supports the required Audio formats and reproduces audio properly.

- 4.2.3.2: Audio Clock Regeneration

  This test verifies that the Sink properly regenerates audio when different Audio Clock Regeneration parameters are used.

MHL Dongle Protocol Tests based on AWG direct synthesis patterns

NOTE. *Contact your local Tektronix representative for these manual test patterns.*

- 5.2.1.1: Character Synchronization

  This test confirms that the Dongle DUT synchronizes if the data stream provides only minimum length Control Periods.

- 5.2.1.2: Packet Types

  This test confirms that the Dongle DUT accepts all valid packet types.

- 5.2.2.1: Video Formats

  This test verifies that the Dongle DUT supports the video formats with no distortion.

- 5.2.2.2: Pixel Encoding

  This test verifies that the Dongle supports various pixel encodings.

- 5.2.2.3: Video Quantization Range

  This test verifies that the Dongle has correct QY and QS bits in the Video Capability Data Block, and the Dongle with A/V Display displays with correct Video Quantization Range.

- 5.2.3.1: Audio Test IEC 60958 / IEC61937

  This test verifies that the Dongle converts audio properly.

- 5.2.3.2: Audio Clock Regeneration

  This test verifies that the Dongle properly regenerates audio when different Audio Clock Regeneration parameters are used.

MHL Cable Assembly Electrical Tests

NOTE. *Contact your local Tektronix representative for the MOI of cable tests.*

- Impedance
- Intra-Pair Skew

- Delay
- Insertion Loss
- Differential and Common Mode Conversion

See also:

- Application Basics (see page 45)
- Equipment Connection Setup MHL Transmitter (see page 14)
- Equipment Connection Setup MHL Receiver (see page 18)
- Install the Software (see page 32)
- Application Directories and Usage (see page 27)
- File Name Extensions (see page 29)

Supported Tests: MHL Transmitter

TekExpress MHL Transmitter supports the following automated Clock and Data tests:

These tests are supported for CTS Version 1.X:

Clock Tests

- 3.1.1.1: Standby (Off) Output Voltage Test - V_{OFF}

  This test measures that the MHL source output voltage is within the specified level limits when the source device is in Standby State or power off mode as specified in the CDF.

- 3.1.1.5: Common-mode Output Swing Voltage Test - $V_{CMSWING}$

  This test confirms that common-mode output voltage swing amplitude is within the specified limits when the DUT operates in normal mode.

- 3.1.1.7: Common-mode Rise and Fall Times Test - T_{R_CM}, T_{F_CM}

  This test confirms that the rise time and fall time of the common-mode output signal are within the specified limits.

- 3.1.1.10: MHL Clock Duty Cycle Test - 24 Bit or Packed Pixel Mode

  This test confirms that the MHL clock duty cycle in 24-bit or packed pixel mode does not exceed the limits allowed by the specification.

- 3.1.1.11: MHL Clock Jitter Test

  This test confirms that the MHL Clock output does not contain excessive jitter larger than the limit allowed by the specification.

Data Tests

- 3.1.1.2: Single-ended High Level Voltage Test - V_{SE_HIGH}

  This test confirms that the single-ended high output voltage level is within the specified limits when the DUT is in normal mode.

- 3.1.1.3: Single-ended Low Level Voltage Test - V_{SE_LOW}

  This test confirms that the single-ended low output voltage level is within the specified limits when the DUT is in normal mode.

- 3.1.1.4: Differential Output Swing Voltage Test - V_{DF_SWING}

  This test confirms that the differential output voltage swing amplitude is within the specified limits when the DUT is in normal mode.

- 3.1.1.6: Differential Rise and Fall Times Test - T_{R_DF}, T_{F_DF}

  This test confirms that the rise and fall times of the differential output signal are equal to or larger than the minimum limit.

- 3.1.1.8: Differential Intra-Pair Skew Test - T_{SKEW_DF}

  This test confirms that the timing skew in the differential signal pair is below the specified limits.

- 3.1.1.12: MHL Data Eye Diagram Test

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

NOTE. *If the MHL Data Eye diagram test fails due to mask hits, TekExpress MHL Solution provides the option to the user to move the mask horizontally to achieve zero or minimal hits. Refer to the* *Mask Movement (see page 189)* *procedure.*

These tests are supported for CTS Version 2.0:

Clock Tests

- 3.1.1.1: Standby (Off) Output Voltage Test - V_{OFF}

  This test measures that the MHL source output voltage is within the specified level limits when the source device is in Standby State or power off mode as specified in the CDF.

- 3.1.1.5: Common-mode Output Swing Voltage Test - $V_{CMSWING}$

  This test confirms that common-mode output voltage swing amplitude is within the specified limits when the DUT operates in normal mode.

- 3.1.1.7: Common-mode Rise and Fall Times Test - T_{R_CM}, T_{F_CM}

  This test confirms that the rise time and fall time of the common-mode output signal are within the specified limits.

- 3.1.1.10: MHL Clock Duty Cycle Test - Normal Mode

  This test confirms that the MHL clock duty cycle does not exceed the limits allowed by the specification in Normal Mode.

- 3.1.1.11: MHL Clock Jitter Test - Normal Mode

  This test confirms that the MHL Clock output does not contain excessive jitter greater than the limit allowed by the specification in Normal Mode.

- 3.1.1.14: MHL Clock Duty Cycle Test - Packed Pixel Mode

  This test confirms that the MHL clock duty cycle in packed pixel mode does not exceed the limits allowed by the specification.

- 3.1.1.15: MHL Clock Jitter Test - Packed Pixel Mode

  This test confirms that the MHL Clock output does not contain excessive jitter larger than the limit allowed by the specification in PackedPixel Mode.

Data Tests

- 3.1.1.2: Single-ended High Level Voltage Test - V_{SE_HIGH}

  This test confirms that the single-ended high output voltage level is within the specified limits when the DUT is in normal mode.

- 3.1.1.3: Single-ended Low Level Voltage Test - V_{SE_LOW}

  This test confirms that the single-ended low output voltage level is within the specified limits when the DUT is in normal mode.

- 3.1.1.4: Differential Output Swing Voltage Test - V_{DF_SWING}

  This test confirms that the differential output voltage swing amplitude is within the specified limits when the DUT is in normal mode.

- 3.1.1.6: Differential Rise and Fall Times Test - T_{R_DF}, T_{F_DF}

  This test confirms that the rise and fall times of the differential output signal are equal to or larger than the minimum limit.

- 3.1.1.8: Differential Intra-Pair Skew Test - T_{SKEW_DF}

  This test confirms that the timing skew in the differential signal pair is below the specified limits.

- 3.1.1.12: MHL Data Eye Diagram Test - Normal Mode

  This test confirms that the MHL Data output has signal quality that meets the eye opening required by the specification in Normal Mode.

- 3.1.1.16: MHL Data Eye Diagram Test - Packed Pixel Mode

  This test confirms that the MHL Data output has signal quality that meets the eye opening required by the specification in Packed Pixel Mode.

NOTE. *If the MHL Data Eye diagram test fails due to mask hits, TekExpress MHL Solution provides the option to the user to move the mask horizontally to achieve zero or minimal hits. Refer to the* *Mask Movement (see page 189)* *procedure.*

Supported Tests: MHL Receiver

TekExpress MHL Receiver supports the following automated Sink and Dongle tests.

These tests are supported for CTS Version 1.X:

MHL Sink Tests

- 4.1.1.1: Input Signal DC Voltage Level Tolerance Test

  This test confirms that the sink device supports input signal DC voltage level allowed by the specification.

- 4.1.1.2: Input Signal Minimum and Maximum Swing Voltages Tolerance Test

  This test confirms that the sink device supports input signal DC voltage level and swing voltage allowed by the specification.

- 4.1.1.3: Intra Pair Skew Tolerance Test

  This test confirms that the sink device can tolerate the maximum intra-pair skew allowed by the specification.

- 4.1.1.4: Jitter Tolerance Test

  This test confirms that the sink device can tolerate the maximum clock and data jitter amounts allowed by the specification.

MHL Dongle Tests

- 5.1.1.1: Input Signal Single-Ended Voltage Level Tolerance Test

  This test confirms that the dongle device supports input signal single-ended voltage level allowed by the specification.

- 5.1.1.2: Input Signal Minimum and Maximum Swing Voltages Tolerance Test

  This test confirms that the dongle device supports input signal minimum and maximum swing voltages allowed by the specification.

- 5.1.1.3: Intra-Pair Skew Tolerance Test

  This test confirms that the dongle device can tolerate the maximum intra-pair skew allowed by the specification.

- 5.1.1.4: Jitter Tolerance Test

  This test confirms that the dongle device can tolerate the maximum clock and data jitter amounts allowed by the specification.

These tests are supported for CTS Version 2.0:

MHL Sink Tests

- 4.1.1.1: Input Signal DC Voltage Level Tolerance Test

This test confirms that the sink device supports input signal DC voltage level allowed by the specification.

- 4.1.1.2: Input Signal Minimum and Maximum Swing Voltages Tolerance Test

  This test confirms that the sink device supports input signal DC voltage level and swing voltage allowed by the specification.

- 4.1.1.3: Intra Pair Skew Tolerance Test

  This test confirms that the sink device can tolerate the maximum intra-pair skew allowed by the specification.

- 4.1.1.4: Jitter Tolerance Test in Normal Mode

  This test confirms that the sink device can tolerate the maximum clock and data jitter amounts allowed by the specification in Normal Mode.

- 4.1.1.8: Jitter Tolerance Test – Packed Pixel Mode

  This test confirms that the sink device can tolerate the maximum clock and data jitter amounts allowed by the specification in Packed Pixel Mode.

MHL Dongle Tests

- 5.1.1.1: Input Signal Single-Ended Voltage Level Tolerance Test

  This test confirms that the dongle device supports input signal single-ended voltage level allowed by the specification.

- 5.1.1.2: Input Signal Minimum and Maximum Swing Voltages Tolerance Test

  This test confirms that the dongle device supports input signal minimum and maximum swing voltages allowed by the specification.

- 5.1.1.3: Intra-Pair Skew Tolerance Test

  This test confirms that the dongle device can tolerate the maximum intra-pair skew allowed by the specification.

- 5.1.1.4: Jitter Tolerance Test – Normal Mode

  This test confirms that the dongle device can tolerate the maximum clock and data jitter amounts allowed by the specification in Normal Mode.

- 5.1.1.9: Jitter Tolerance Test – Packed Pixel Mode

  This test confirms that the dongle device can tolerate the maximum clock and data jitter amounts allowed by the specification in Packed Pixel Mode.

Supported Resolutions: MHL Transmitter

CTS 1.X: All of the TekExpress MHL tests can be performed for DUTs operating in resolutions between 25 MHz and 74.25 MHz. 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. For instance, a DUT could have any of the following:

Low : 25 High : 25
Low : 25 High : 27
Low : 25 High : 74.25

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

CTS 2.0: All of the TekExpress MHL tests can be performed for DUTs operating at these data rates (Gbps):

24 Bit mode:
Low : 0.75 High : 2.22
Low : 0.81 High : 2.22

Packed Pixel mode:
Low : 2.97 High : 2.97

Other data rates that fall within this range can be entered in the Low Data Rate and High Data Rate fields on the DUT tab of the Setup panel.

Table 1: Supported resolutions (Transmitter)

Test group	Test name	Supported resolutions
MHL Clock	3.1.1.1: Standby Output Voltage – V_{OFF}	—
	3.1.1.5: Common-mode Output Swing Voltage $V_{CMSWING}$	Low
	3.1.1.7: Common-mode Rise and Fall Times-T_{R_CM}, T_{F_CM}	High
	CTS 1.X 3.1.1.10: MHL Clock Duty Cycle – 24 Bit or Packed Pixel Mode	High
	CTS 2.0 3.1.1.10: MHL Clock Duty Cycle – Normal Mode	High
	CTS 1.X 3.1.1.11: MHL Clock Jitter	Low and High resolutions
	CTS 2.0 3.1.1.11: MHL Clock Jitter – Normal Mode	Low and High resolutions
	3.1.1.14: MHL Clock Duty Cycle – Packed Pixel Mode	High (PP)
	3.1.1.15: MHL Clock Jitter – Packed Pixel Mode	High (PP)

Table 1: Supported resolutions (Transmitter) (cont.)

Test group	Test name	Supported resolutions
MHL Data	3.1.1.2: Single-ended High Level Voltage – V_{SE_HIGH}	Low
	3.1.1.3: Single-ended Low Level Voltages – V_{SE_LOW}	Low
	3.1.1.4: Differential Output Swing Voltage – V_{DF_SWING}	Low
	3.1.1.6: Differential Rise and Fall Times – T_{R_DF}, T_{F_DF}	High
	3.1.1.8: Differential Intra-Pair Skew – T_{SKEW_DF}	Low
	CTS 1.X 3.1.1.12: MHL Data Eye Diagram	Low and High resolutions
	CTS 2.0 3.1.1.12: MHL Data Eye Diagram – Normal Mode	Low and High resolutions
	3.1.1.16: MHL Data Eye Diagram – Packed Pixel Mode	High (PP)

Supported Resolutions: MHL Receiver

CTS 1.X: All of the TekExpress MHL Receiver tests can be performed for DUTs operating in resolutions of 25, 27 and 74.25 MHz according to the standard. 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. Either of the Resolutions can be **High** according to the Manufacturer. For instance, a DUT could have any of the following:

Low : 25 High : 25
Low : 25 High : 27
Low : 25 High : 74.25

CTS 2.0: All of the TekExpress MHL tests can be performed for DUTs operating at these data rates (Gpbs):

24 Bit mode:
Low : 0.75 High : 2.22
Low : 0.81 High : 2.22

Packed Pixel mode:
Low : 2.97 High : 2.97

Other data rates that fall within this range can be entered in the Low Data Rate and High Data Rate fields on the DUT tab of the Setup panel.

Table 2: Supported resolutions (MHL Receiver)

Test group	Test name	Supported resolutions
MHL Sink	4.1.1.1: Input Signal DC Voltage Level Tolerance	Highest
	4.1.1.2: Input Signal Minimum and Maximum Swing Voltages Tolerance	Highest
	4.1.1.3: Intra-Pair Skew Tolerance	Highest
	CTS 1.X 4.1.1.4: Jitter Tolerance CTS 2.0 4.1.1.4: Jitter Tolerance in Normal Mode	Lowest and Highest
	CTS 2.0 4.1.1.8: Jitter Tolerance in Packed Pixell Mode	Highest
MHL Dongle	5.1.1.1: Input Signal Single-Ended Voltage Level Tolerance	Highest
	5.1.1.2: Input Signal Minimum and Maximum Swing Voltage Tolerance	Highest
	5.1.1.3: Intra-Pair Skew Tolerance	Highest
	CTS 1.X 5.1.1.4: Jitter Tolerance CTS 2.0 5.1.1.4: Jitter Tolerance in Normal Mode	Lowest and Highest
	CTS 2.0 5.1.1.9: Jitter Tolerance in Packed Pixel Mode	Highest

Equipment Connection Setup (MHL Transmitter)

You need the following equipment to run MHL Transmitter tests. (For details, see Minimum System Requirements (see page 31)):

- A supported Tektronix oscilloscope (see page 31)
- Two Differential Probes – P7313SMA, for Single Ended and Differential tests
- One Differential – Probe P7240 for Common Mode tests
- Device under test (DUT)
- Wilder/Tektronix Fixtures
- C-Bus Sink boards

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

Clock and Data Tests

Table 3: Connection diagram 1 (Transmitter)

3.1.1.4: Differential Output Swing Voltage Test - V_{DF_SWING}
3.1.1.5: Common-mode Output Swing Voltage Test - $V_{CMSWING}$
3.1.1.6: Differential Rise and Fall Times Test -T_{R_DF}, T_{F_DF}
3.1.1.7: Common-mode Rise and Fall Times Test -T_{R_CM}, T_{F_CM}
3.1.1.10: MHL Clock Duty Cycle Test – Normal Mode
3.1.1.11: MHL Clock Jitter Test – Normal Mode
3.1.1.12: MHL Data Eye Diagram Test – Normal Mode
3.1.1.14: MHL Clock Duty Cycle Test – Packed Pixel Mode
3.1.1.15: MHL Clock Jitter Test – Packed Pixel Mode

Table 3: Connection diagram 1 (Transmitter) (cont.)

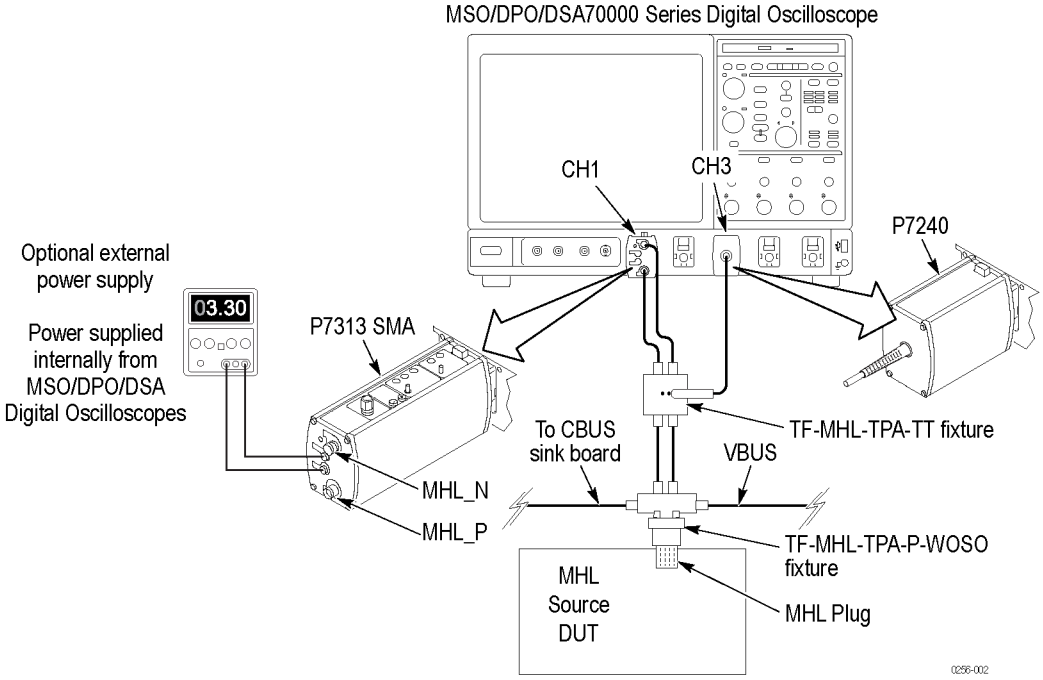
3.1.1.16: MHL Data Eye Diagram Test – PackedPixel Mode


Table 4: Connection diagram 2 (Transmitter)

3.1.1.2 Single-ended High Level Voltage — VSE_HIGH
3.1.1.3 Single-ended Low Level Voltages — VSE_LOW

Table 4: Connection diagram 2 (Transmitter) (cont.)

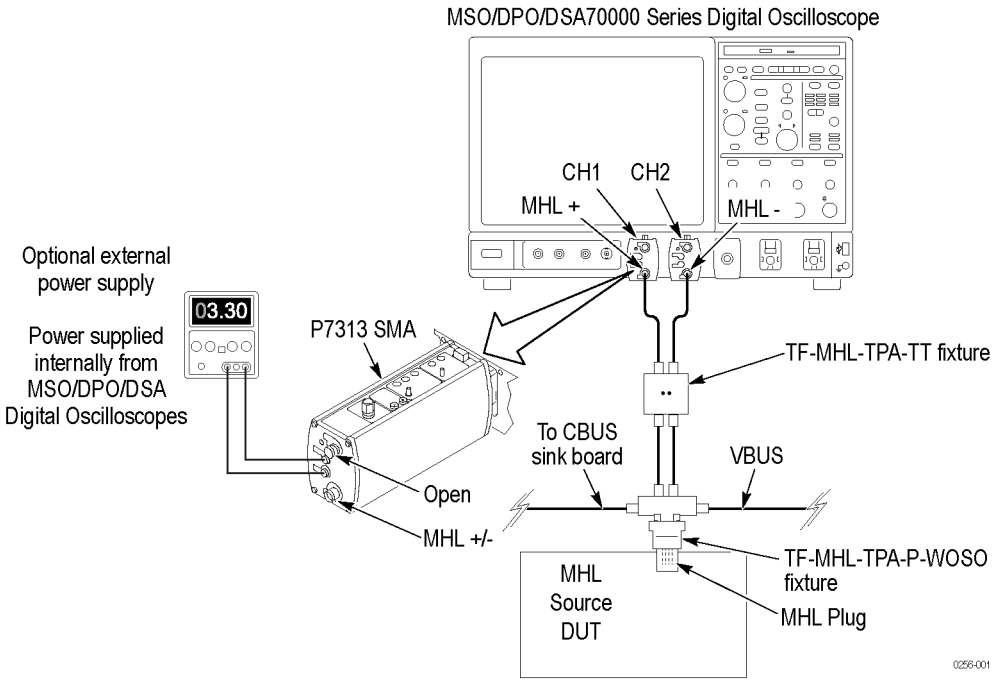
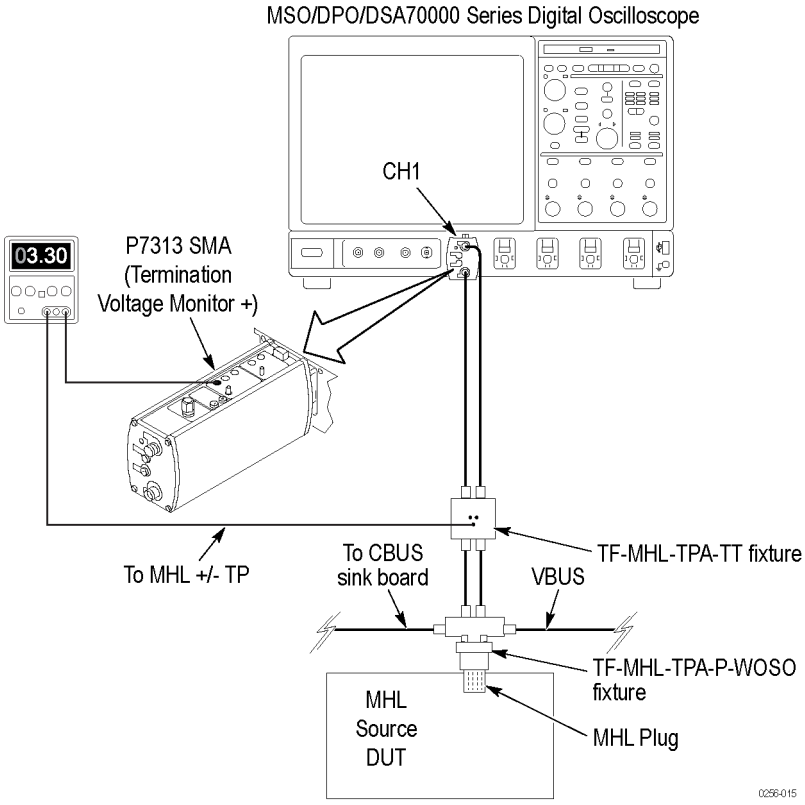
3.1.1.8 Differential Intra-Pair Skew — TSKEW_DF


Table 5: Connection diagram 3 (Transmitter)

3.1.1.1 Standby Output Voltage — V_{OFF}


See Also

- Equipment Connection Setup MHL Receiver (see page 18)
- About Algorithms (see page 153)
- View Connected Instruments (see page 37)

Equipment Connection Setup (MHL Receiver)

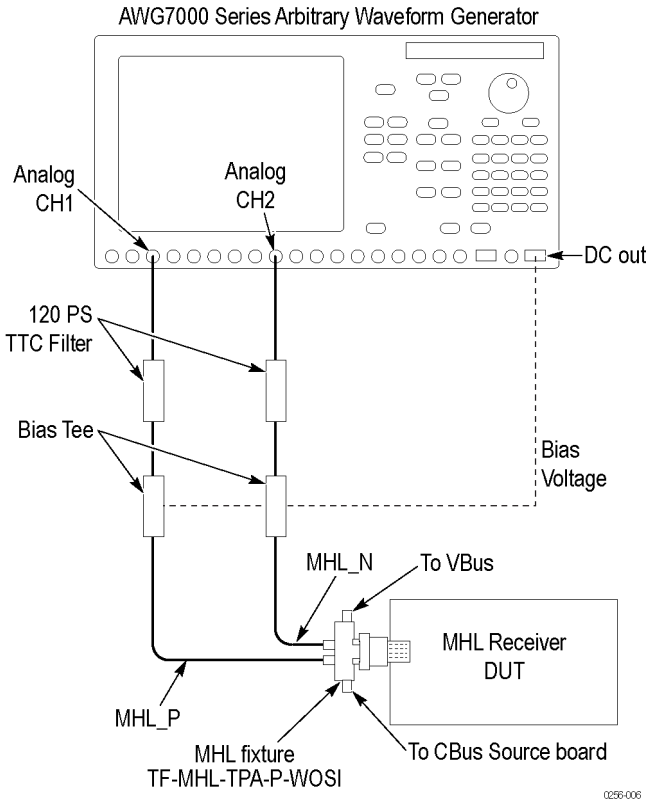
You need the following equipment (for details, see Minimum System Requirements (see page 31)):

- A supported Tektronix oscilloscope (see page 31)
- One Differential Probe – P7313SMA
- Device under test (DUT)
- Wilder Fixtures
- C-Bus Sink boards
- MHL Signal Generator: AWG 7122B/C with option 01 and 02 or 06 and 08

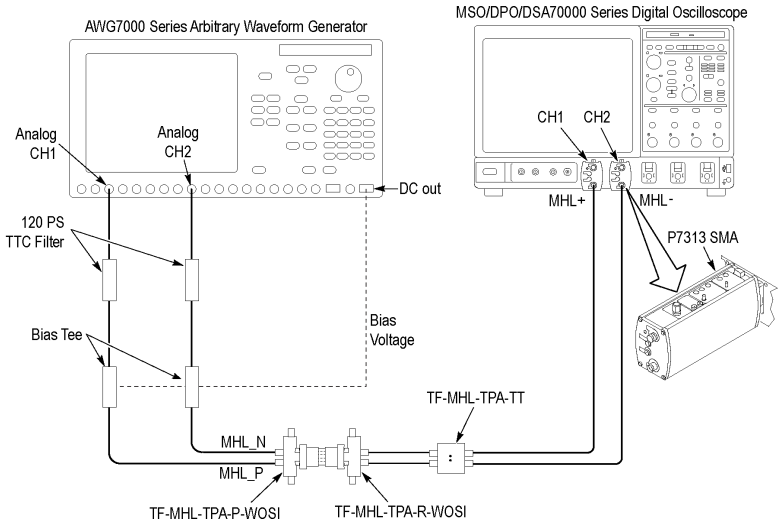
Connection diagrams and the tests they apply to are listed below.

Sink Tests

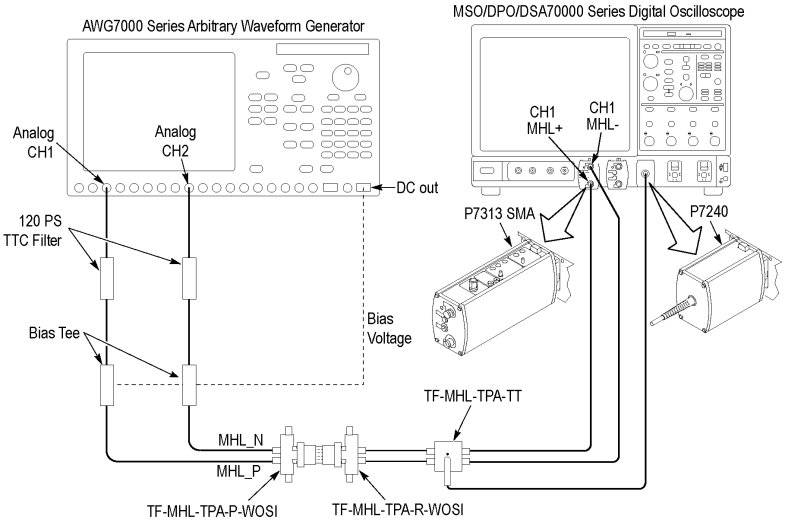
4.1.1.1: Input Signal DC Voltage Level Tolerance
4.1.1.3: Intra-Pair Skew Tolerance
CTS 1.X 4.1.1.4: Jitter Tolerance CTS 2.0 4.1.1.4: Jitter Tolerance in Normal Mode
CTS 2.0 4.1.1.8: Jitter Tolerance in Packed Pixel Mode



Sink-MHL Input Signal DC Voltage Level Test

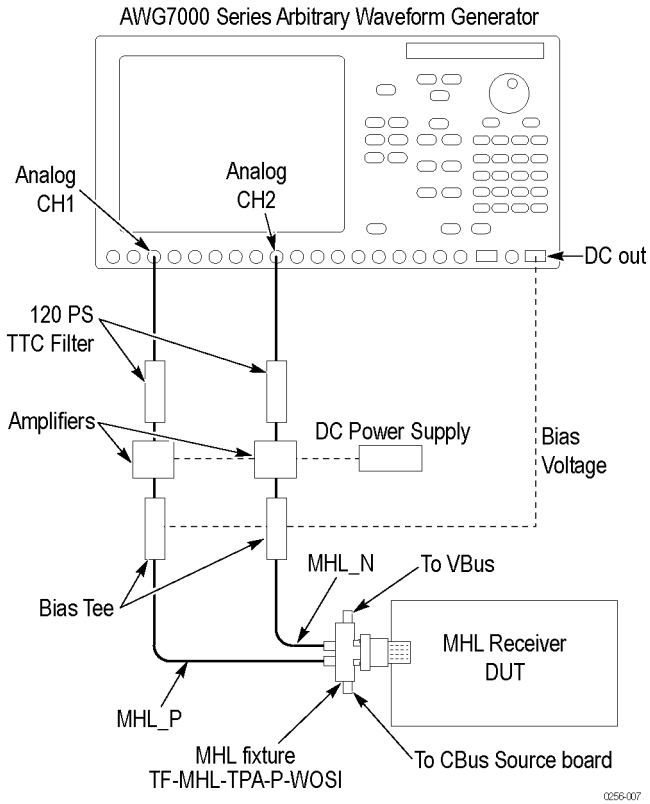


Single-Ended Signals Calibration Setup for Sink Test

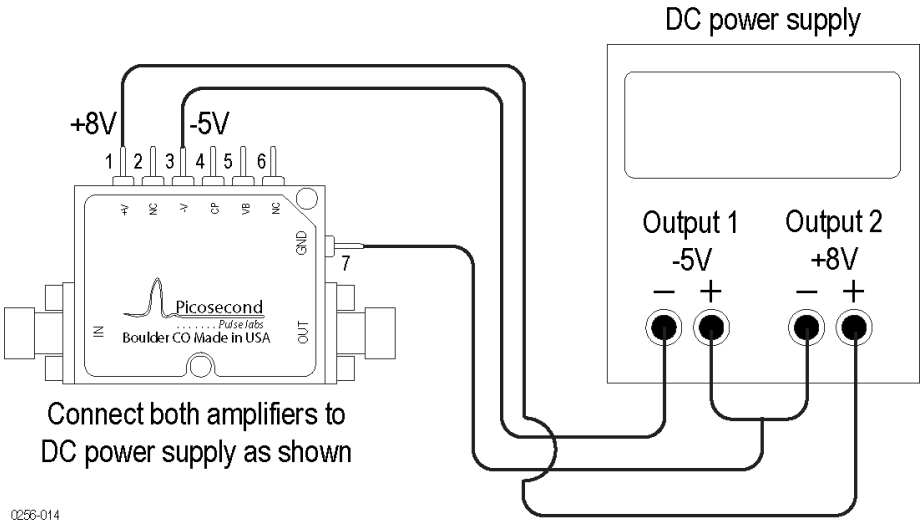


Differential and Common-Mode Signals Calibration Setup for Sink Test

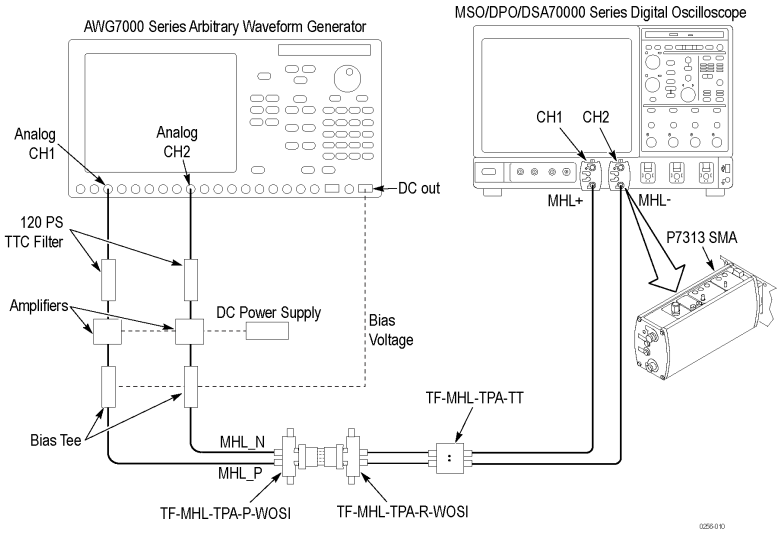
4.1.1.2: Input Signal Minimum and Maximum Swing Voltages Tolerance:



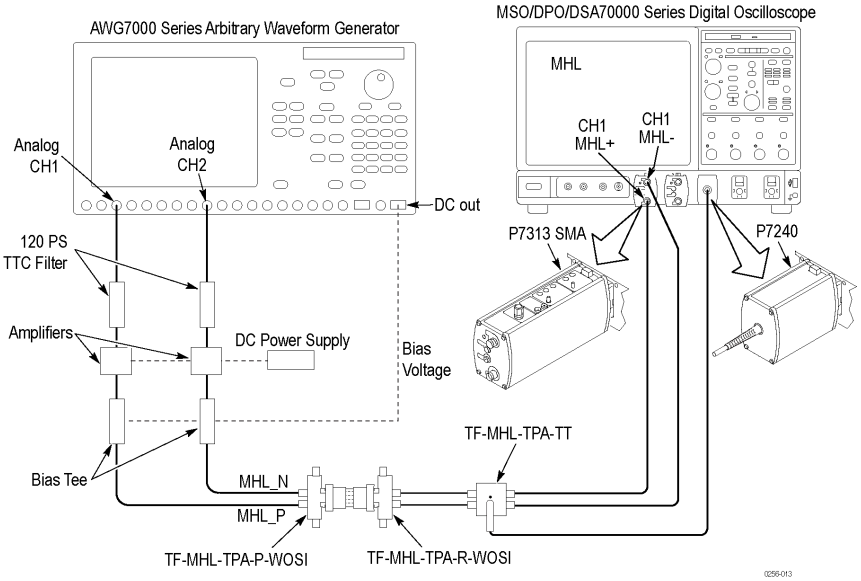
Sink — MHL Input Signal DC Voltage Level Min-Max Test



Amplifier Connections to Power Supply



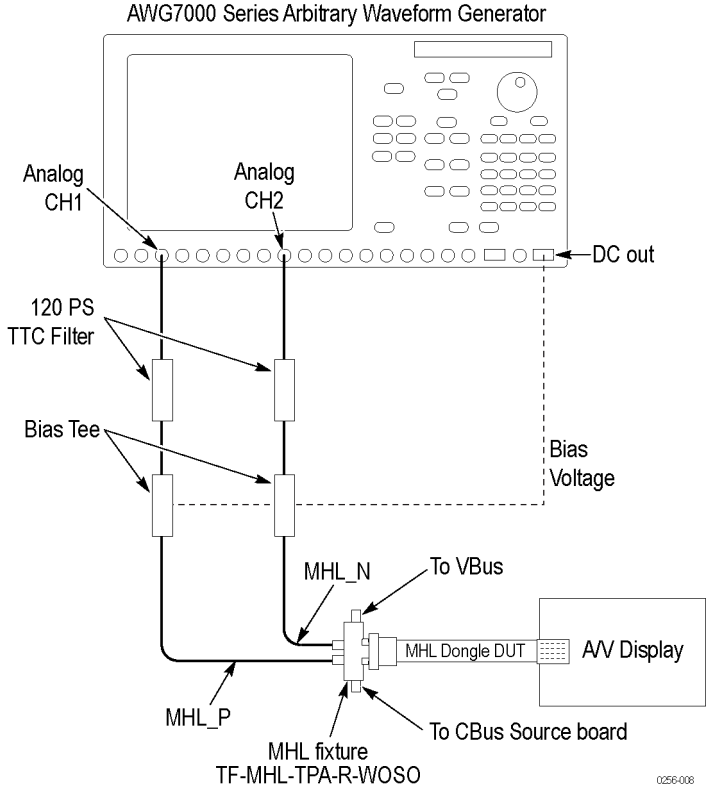
Single Ended Signals Calibration Setup for Sink Min-Max Test



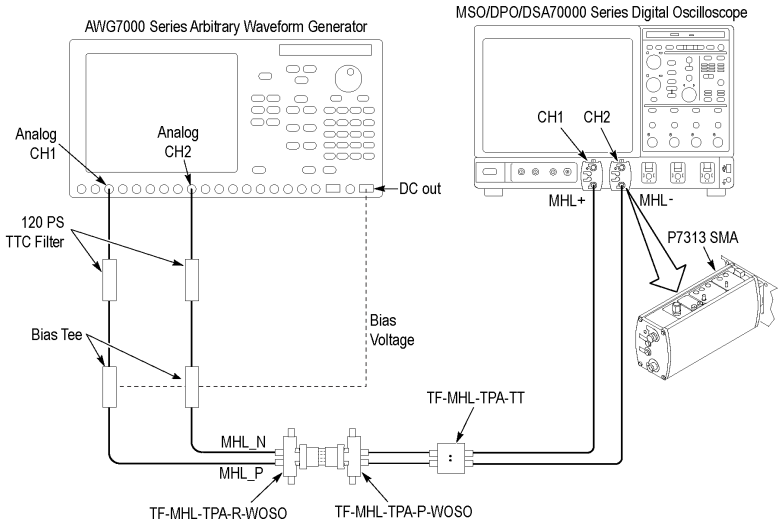
Differential and Common-Mode Signals Calibration Setup for Sink Min-Max Test

Dongle Tests

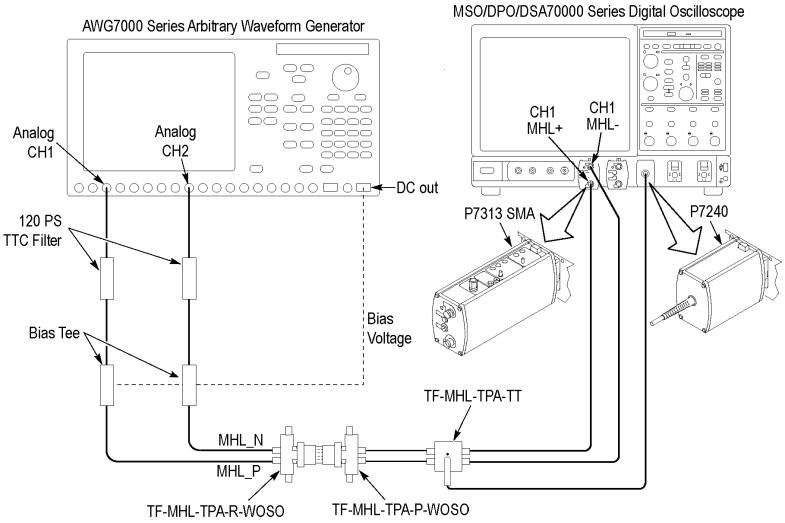
5.1.1.1: Input Signal Single-Ended Voltage Level Tolerance
5.1.1.3: Intra-Pair Skew Tolerance Test
CTS 1.X 5.1.1.4: Jitter Tolerance CTS 2.0 5.1.1.4: Jitter Tolerance in Normal Mode
CTS 2.0 5.1.1.9: Jitter Tolerance in Packed Pixel Mode



Dongle — Input Signal Single-Ended Voltage level Tolerance Test

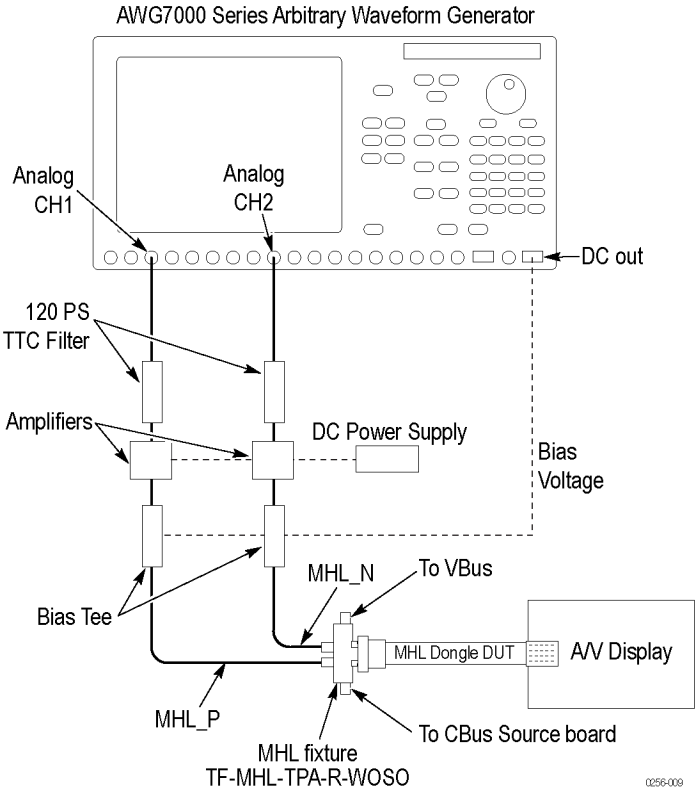


Single-Ended Signals Calibration Setup for Dongle Test

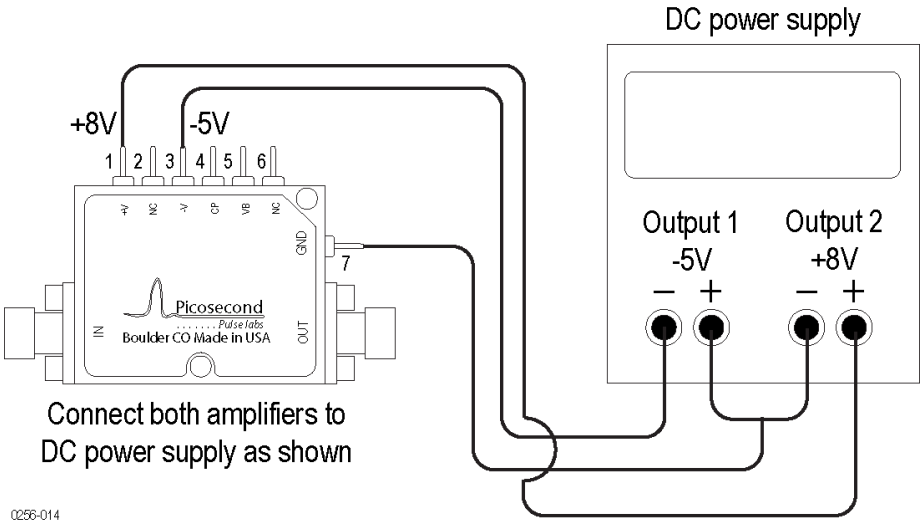


Differential and Common-Mode Signals Calibration Setup for Dongle Test

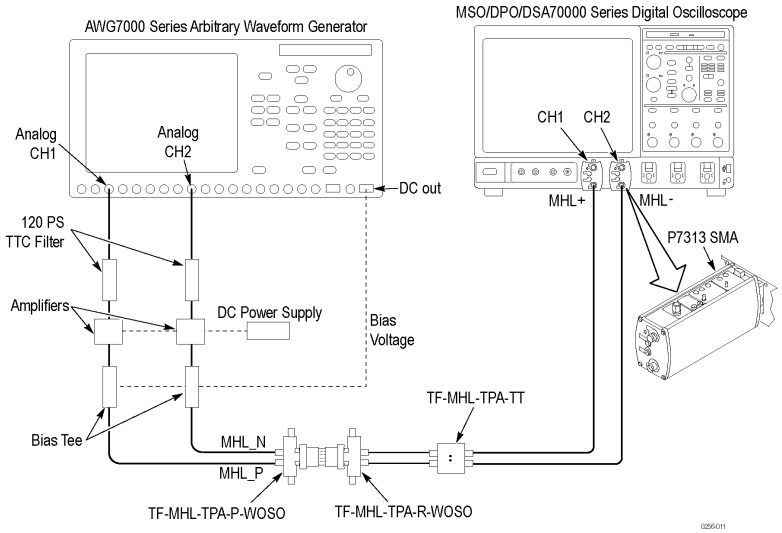
5.1.1.2: Input Signal Minimum and Maximum Swing Voltage Tolerance



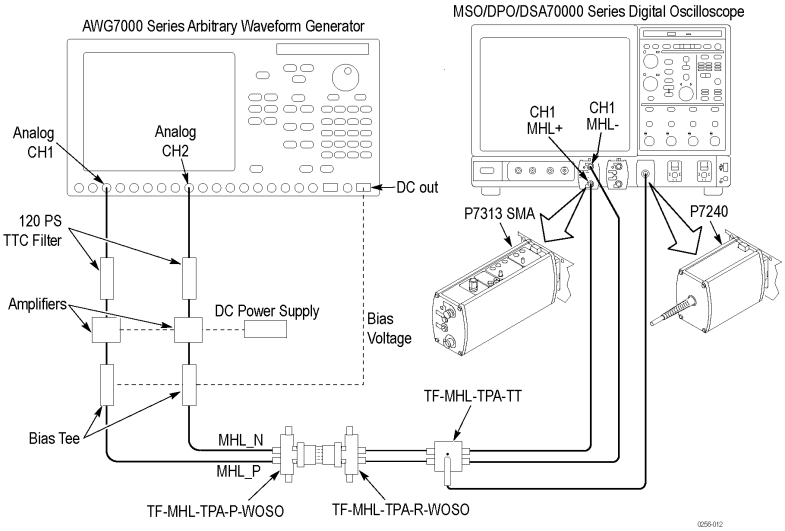
Dongle Min-Max Setup



Amplifier Connections to DC Power Supply



Single-Ended Signals Calibration Setup for Dongle Min Max Test



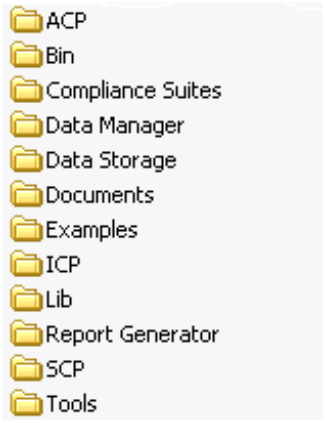
Differential and Common-Mode Signals Calibration Setup for Dongle Min Max Test

See Also

- Equipment Connection Setup MHL Transmitter (see page 14)
- About Algorithms (see page 153)
- View Connected Instruments (see page 37)

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

See Also

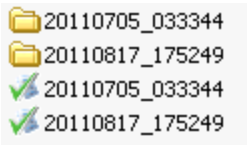
- View Test-Related Files (see page 28)

View Test-Related Files

Files related to MHL tests are stored in the `MHL` folder under the `My TekExpress` shared folder. In the `MHL` 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 MHL 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\MHL`. 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 change the report file location (see page 97) for a specific test.*

See Also

- File Name Extensions (see page 29)

File Name Extensions

The TekExpress MHL 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
.seq	The test sequence file
.xls	Microsoft Excel spreadsheet 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 format (see page 97).
.flt	Filter file used with transmitter tests

See Also

- Application Directories and Usage (see page 27)

Compatibility

The TekExpress MHL application runs on the following Tektronix oscilloscopes:

- DPO/DSA/MSO72004/B/C, DPO/DSA/MSO71604/B/C, DPO/DSA/MSO71254/B/C, and DPO/DSA/MSO70804/B/C Digital Oscilloscopes with Option DJA
- DPO/DSA73304D and DPO/DSA72504D Digital Oscilloscopes with Option DJA

See Also

- Minimum System Requirements (see page 31)

Minimum System Requirements

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

Table 7: System requirements

Oscilloscope	BW ≥ 8 GHz, 16M Record Length/Ch - Opt. 4M/2XL or more (for Eye Diagram and Jitter tests), and 20XL for MHL Protocol analysis software For a list of compatible oscilloscopes, see Compatibility (see page 31).
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 MHL Advanced Analysis and Compliance software ■ DPOJET, Jitter and Eye Diagram Analysis Tool, 3.5.1.4 or later ■ National Instruments TestStand engine 4.2.1 ■ Microsoft Excel 2002 or above ■ 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	■ Two Differential Probes – P7313SMA for Single-Ended and Differential tests ■ One Differential Probe – P7240 for Common-Mode tests

Table 7: System requirements (cont.)

MHL Sources	■ MHL Tx DUTs ■ One AWG7122 B/ C with Option 01, 02/06 and 08 which can be used as an MHL DUT ■ MHL Ref waveforms
TDR Tests	■ Oscilloscope – DSA8200 or equivalent with 80E03 and 80E04 modules and I-Connect software
MHL Generator for Sink and Dongle tests	■ AWG7122B/C with option 01 and 02 or 06 and 08
Test Fixtures	MHL fixture kits from Tektronix ■ TF-MHL-TPA-TEK (Complete MHL Fixture kit) ■ TF-MHL-TPA-TEK-SO (Source Fixture Only kit) ■ TF-MHL-TPA-TEK-SI (Sink Fixture kit) ■ TF-MHL-TPA-TEK-DG (Dongle Fixture kit) ■ TF-MHL-TPA-TEK-CB (Cable Fixture kit) ■ TF-MHL-TPA-TEK-RSEN (RSEN kit)
MHL Accessory Kit (to be used with AWG)	■ Two Bias Tees from Mini-Circuits – ZX85-12G-S+ ■ Two Amplifiers from Pico Second Pulse Labs – 5866 (needed only for Sink/Dongle Min/Max test) ■ Two 120PS TTC filters from Pico Second Pulse Labs – 5915-100-120PS
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] ■ 2 pair of SMA cables

1 If TekExpress is running on an instrument having a video resolution lower than 800 x 600 (for example, 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.

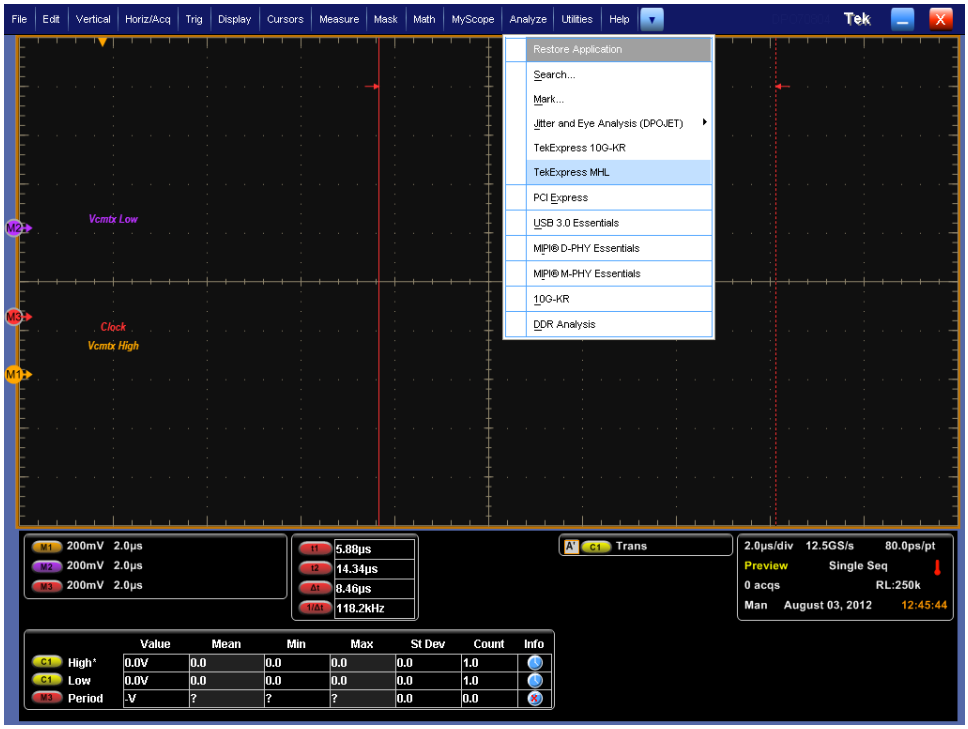
See Also

- Compatibility (see page 31)

Install the Software

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

1. Close all applications (including the TekScope application).
2. Go to the www.tek.com Web site and search for MHL to locate the installation file. Download the file `MHL WebInstaller.exe`.
3. Double-click the executable file to extract the installation files. After extraction, the installer launches and displays the InstallShield Wizard.
   - Check TekExpress application for MHL to install the TekExpress MHL application.
4. The software automatically installs in the following location:
   - Windows XP location: `C:\Program Files\Tektronix\TekExpress\TekExpress MHL`
   - Windows 7 location: `C:\Program Files (x86)\Tektronix\TekExpress\TekExpress MHL`
5. The installer updates the TekScope Analyze menu to include TekExpress MHL:



See Also

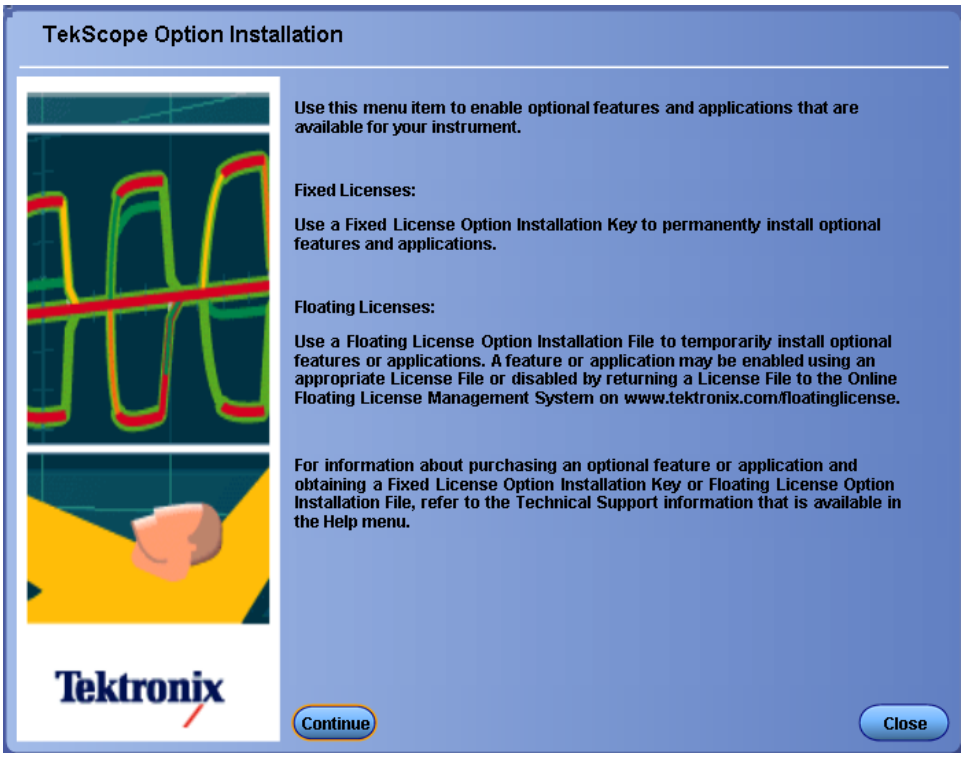
- Minimum System Requirements (see page 31)
- Compatibility (see page 31)

Activate the License

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

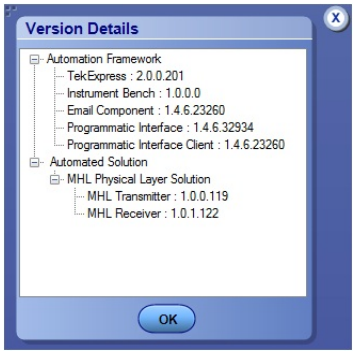
- View Version and License Information (see page 35)

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

See Also

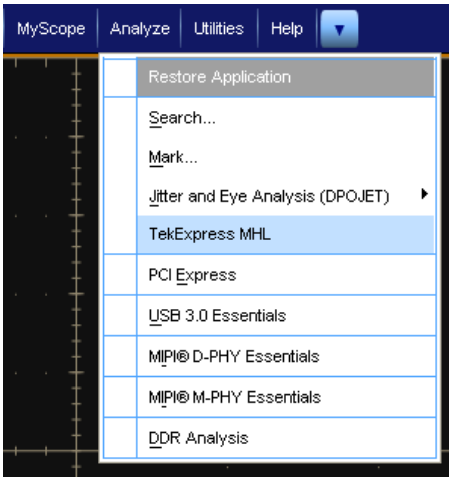
- Activate the License (see page 34)
- Options Menu (see page 36)

Start the Application

The first time you open the application after installation, you are required to enter a license key. The application also checks for a file called `Resources.xml` located in the `My TekExpress` folder. If this file is not found, instrument discovery is performed before launching TekExpress MHL. The Resources.xml file contains information regarding instruments available on your network.

NOTE. *When the application starts, it checks for the appropriate license key. If the valid license key is not present, the application switches to the Evaluation mode. If the application fails to detect the key, it continues to run in Evaluation mode. You are allowed 10 free trials in Evaluation mode. Each time you open the application without supplying a valid license key, one of the free trials is used.*

To run the MHL application, from the TekScope menu, select **Analyze** > **TekExpress MHL**.



While running the application, you can switch between the oscilloscope screen and MHL by clicking the desired window. To keep the application window on top, select **Keep On Top** from the **Options menu**.

NOTE. *If the application was not terminated properly during the last use, a dialog box asks to recall the previously unsaved session.*

Options Menu

Options menu

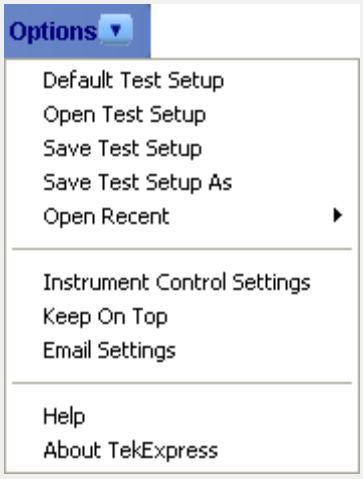
The Options menu (see page 37) is located in the upper right corner of the application.

It has the following selections:

Menu	Function
Default Test Setup	Opens an untitled test setup with defaults selected
Open Test Setup	Opens a saved test setup
Save Test Setup	Saves the current test setup selections
Save Test Setup As	Creates a new test setup based on an existing one
Open Recent	Displays a menu of recently opened test setups to select from

Menu	Function
Instrument Control Settings (see page 42)	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 MHL application on top of other open windows on the desktop
Email settings (see page 41)	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 license information (see page 35) for your MHL installation ■ Provides a link to the Tektronix Web site

Options Menu

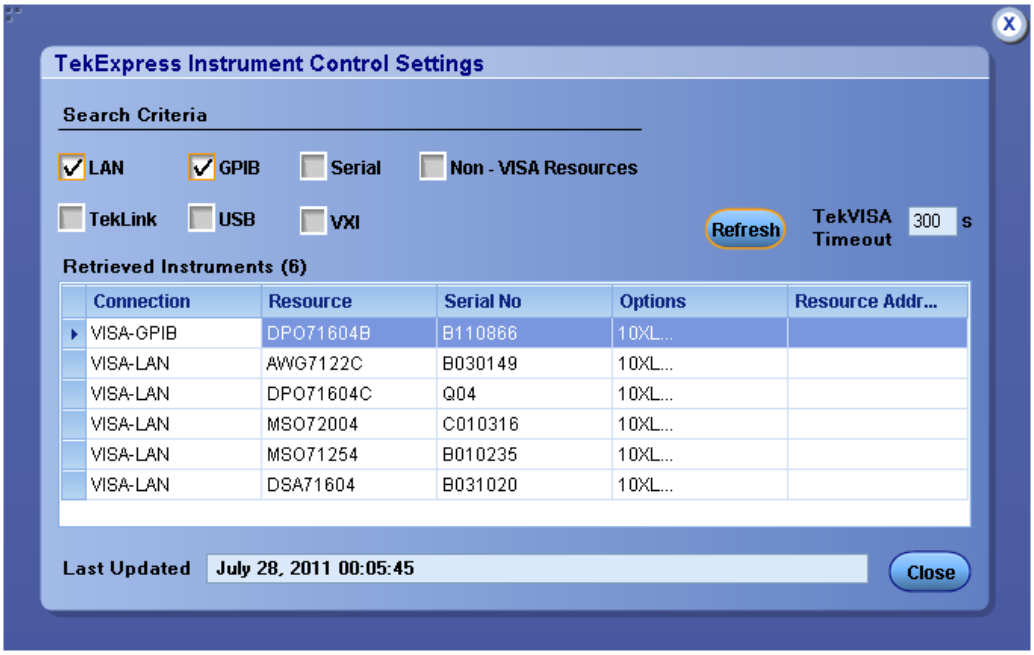


View Connected Instruments: the Instrument Control Settings

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

To refresh the list of connected instruments:

1. From the Options menu, select Instrument Control Settings.



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

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

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

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

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

See Also

- Equipment Connection Setup (MHL Transmitter) (see page 14)
- Equipment Connection Setup (MHL Receiver) (see page 18)

Application Controls

Table 8: Application controls descriptions

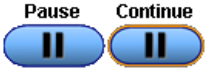
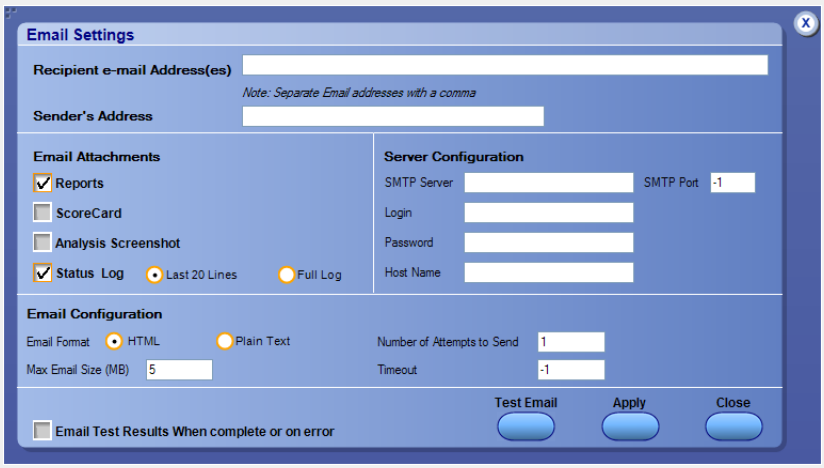
Item	Description
Options menu (see page 36)	Opens the Options menu for access to global controls
Panels	Visual frames with sets of related options. Some panels are further divided into tabs and other sections.
Start button	 Use the Start button to continuously acquire and accumulate measurements. If prior acquired measurements have not been cleared, the new measurements are added to the existing set.
Stop button	 Use the Stop button to abort a test in progress.
Pause \ Continue button	 Use the Pause button to interrupt the current acquisition. When a test is paused, the button name changes to Continue.

Table 8: Application controls descriptions (cont.)

Item	Description
Clear button	 Available only on the Results panel (see page 95). Use the Clear button to clear all existing measurement results. Note that adding or deleting a measurement, or changing a configuration parameter of an existing measurement also clears measurements. This is to prevent the accumulation of measurement statistics or sets of statistics that are not coherent.
Application window move icon	 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.

Email Settings

Use the Email Settings utility to configure email notifications (see page 87) if you want the application to notify you when a tests completes or produces an error. Select the type of test run information to be included in the notification, such as test reports and test logs, the email message format, and the email message size limit.

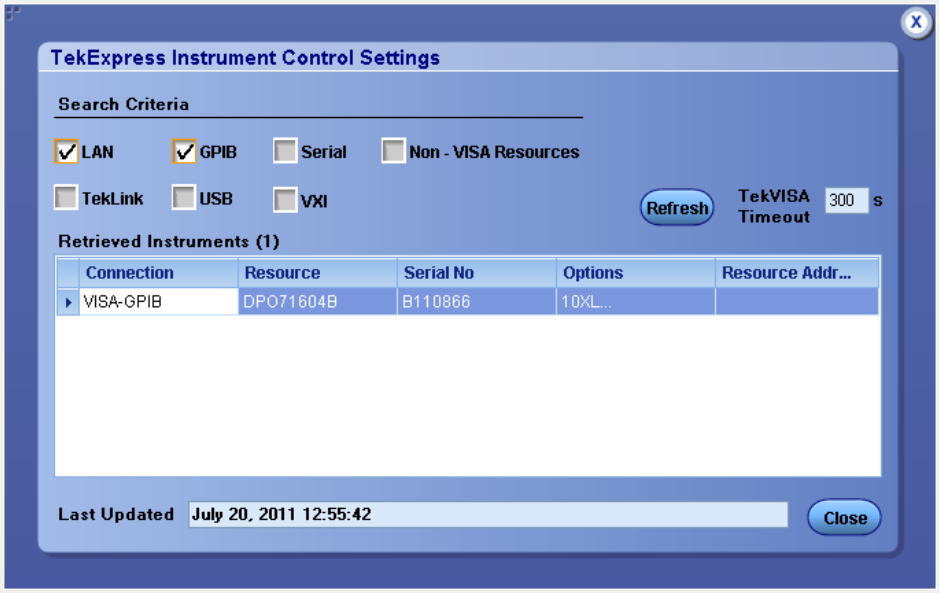


- Options Menu (see page 36)

Instrument Control Settings

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

- Discovers the connected instruments
- Confirms the instrument connection setup



Use the Instrument Control Settings feature to search for and view connected instruments (see page 37) and view instrument connection details. Connected instruments displayed here can be selected for use in the Global Settings section of the Setup panel's Configuration section.

See Also

- Options Menu (see page 36)

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:
   - A message box appears asking if you really want to exit TekExpress. To exit, click **Yes.** Otherwise, click **No.**
   - If you have an unsaved session or test setup open, you will be asked if you want to save it before exiting. To save it, click **Yes.** If you do not wish to save, click **No**. To remain in the session, click **Cancel.**

Application Basics

The TekExpress MHL software user interface is intuitive and easy to use. In addition to the UI, a programmatic interface (see page 105) is available.

The user interface has four main panels whose options differ depending upon whether the MHL Transmitter or MHL Receiver Suite has been selected.

- First, set up the tests. (see page 47)
- Then, prepare to run the tests (see page 85).
- Then, run the tests and view the progress of analysis (see page 89).
- Then, view the results of the tests (see page 95).
- You may configure and view reports (see page 97).
- You may easily save and recall test setups (see page 101).

See Also

- Overview and Key Specifications (see page 3)

Setting Up Tests: the Setup Panel

The Setup panel (see page 52) guides you through the MHL test setup process using tabs. The options selected in a tab affect the options available in the next tab down. The available options differ depending on whether you choose Version CTS 1.X or 2.0.

NOTE. *A check mark next to a tab title means you have made changes. However, if you make changes on a tab that significantly affect a prior tab, the check mark of the affected tab is replaced with its number, indicating that you should check the settings on that tab.*

- DUT tab (see page 52)

  Here, you can specify the DUT ID, device, applications suite (MHL Transmitter or Receiver), the CTS version, the view and device profile settings. These settings apply to all tests for the current session, and affect the test list in the **Test Selection** tab.

- Test Selection tab (see page 62)

  Here, select tests individually or by group, view a short description of a selected test, and view a schematic showing appropriate device connections.

- Acquisitions tab (see page 67) (MHL Transmitter only)

  View configurations for attached probes, view a list of signals and sources, and view the Acquisitions table.

- Configuration tab (see page 74) (Displays only when **View: Advanced** is selected in the DUT tab.)

  Here, select either **Compliance Mode** or **User Defined Mode**, view Global Settings, select measurement settings, and view or edit test parameter limits.

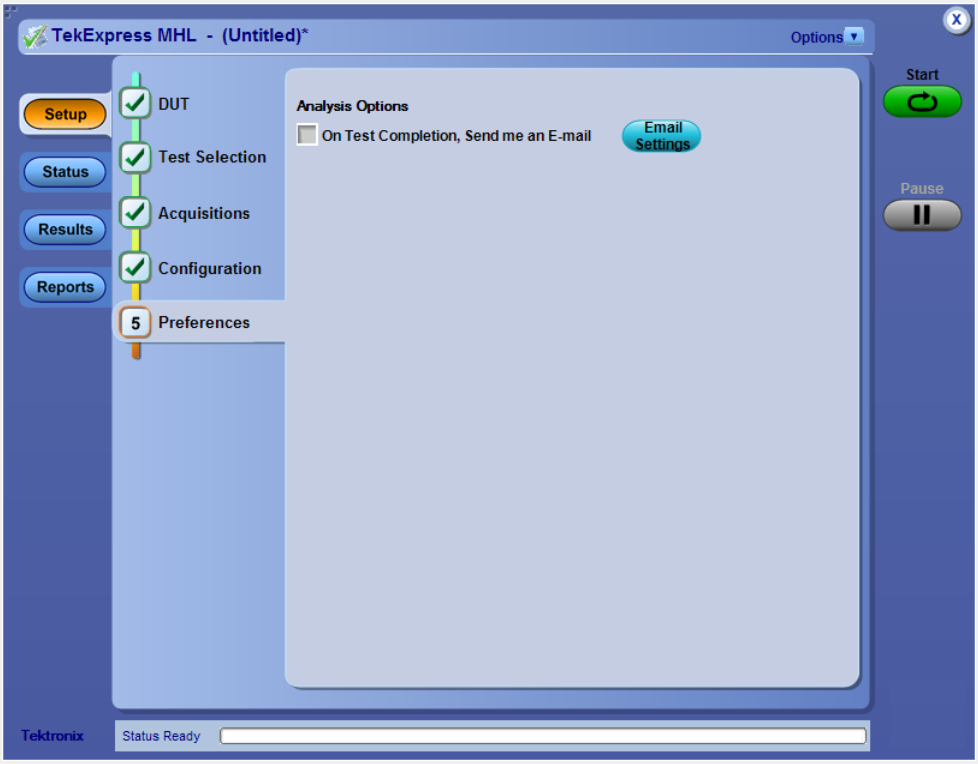
- Preferences tab (see page 48)

  Here, you may specify that an email be sent to you upon test completion.

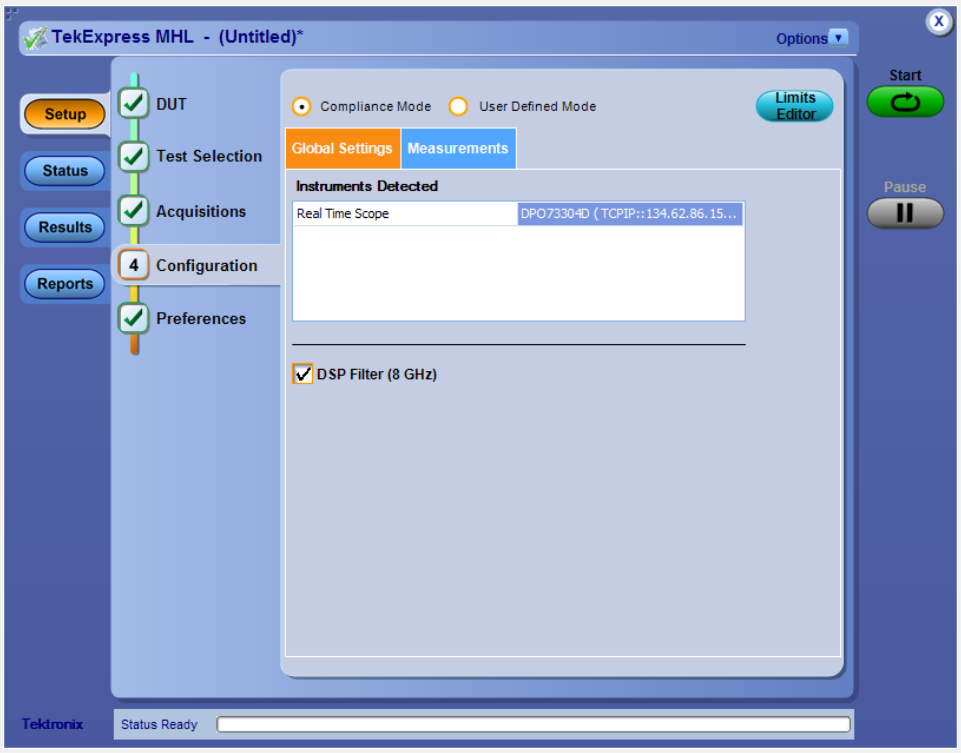
See also

- Saving a Test Setup (see page 101)
- Running the Tests and Viewing their Progress the Status Panel (see page 89)
- Viewing Test Results the Results Panel (see page 95)
- Configuring and Viewing Reports the Reports Panel (see page 97)

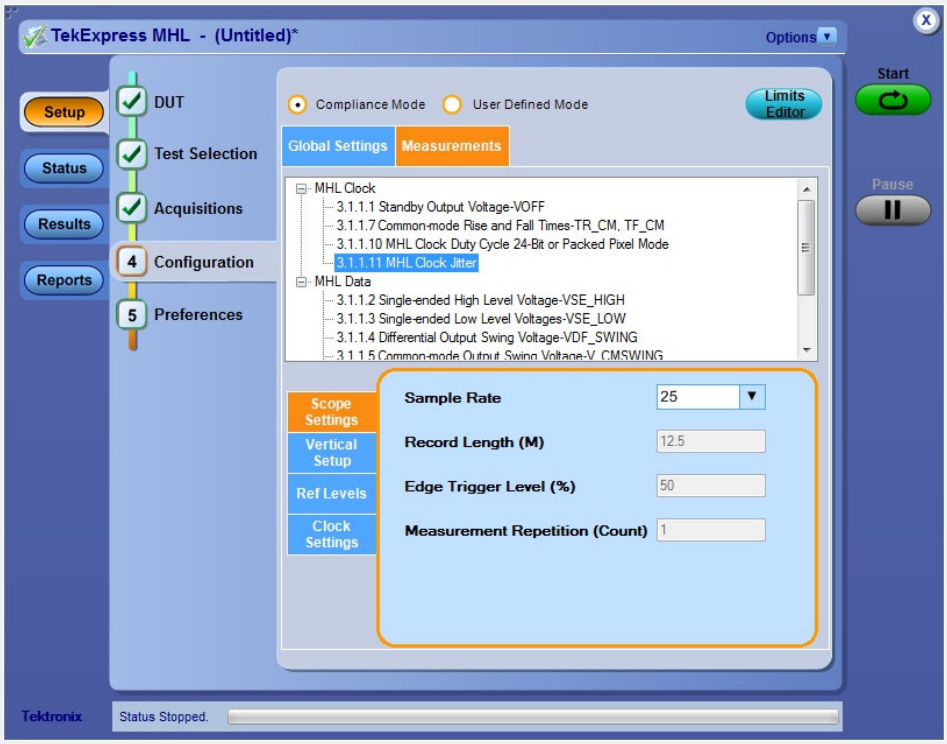
Preferences tab



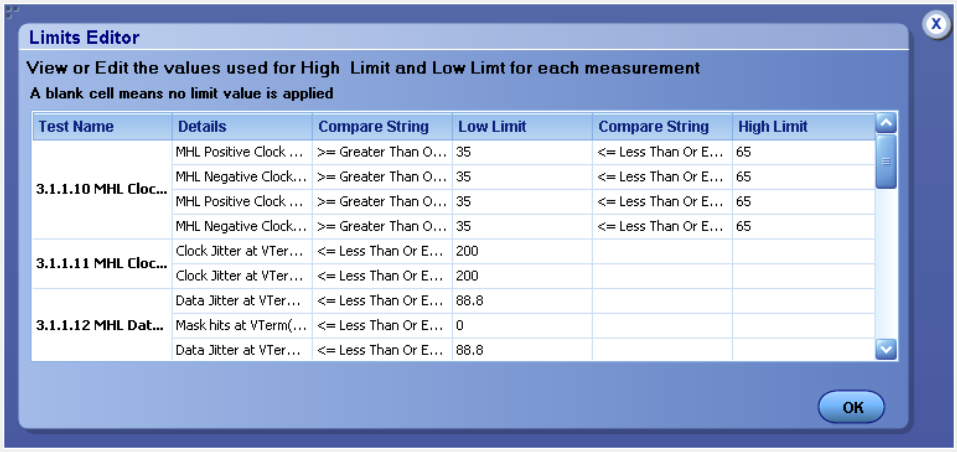
Configuration — Global Settings (Transmitter)



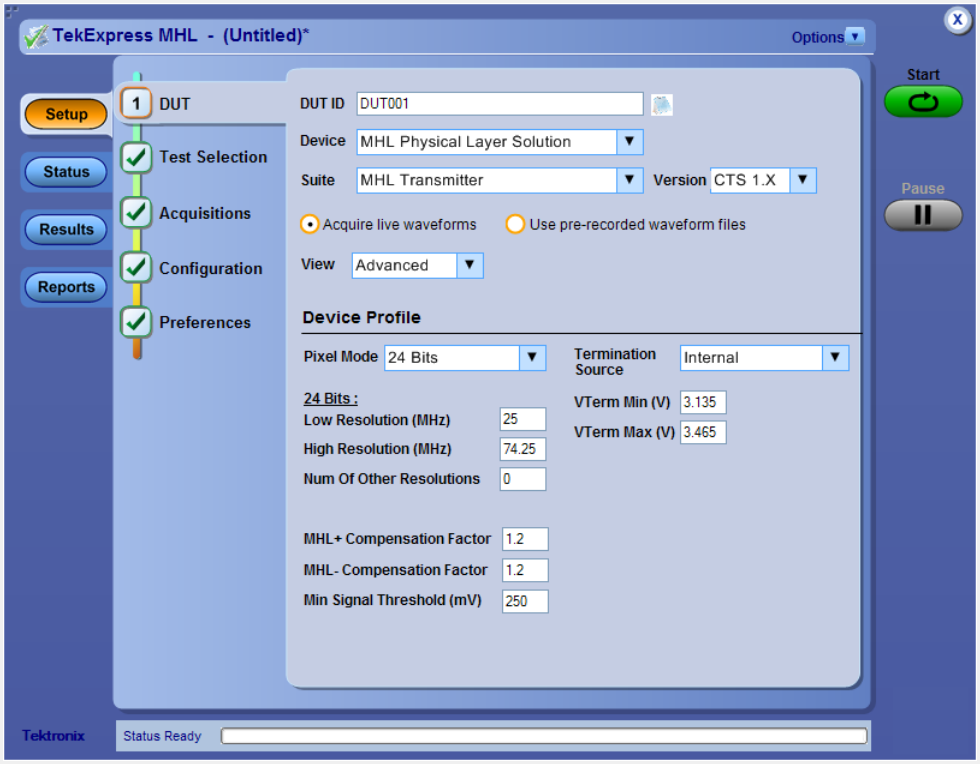
Configuration — Measurements (Transmitter)



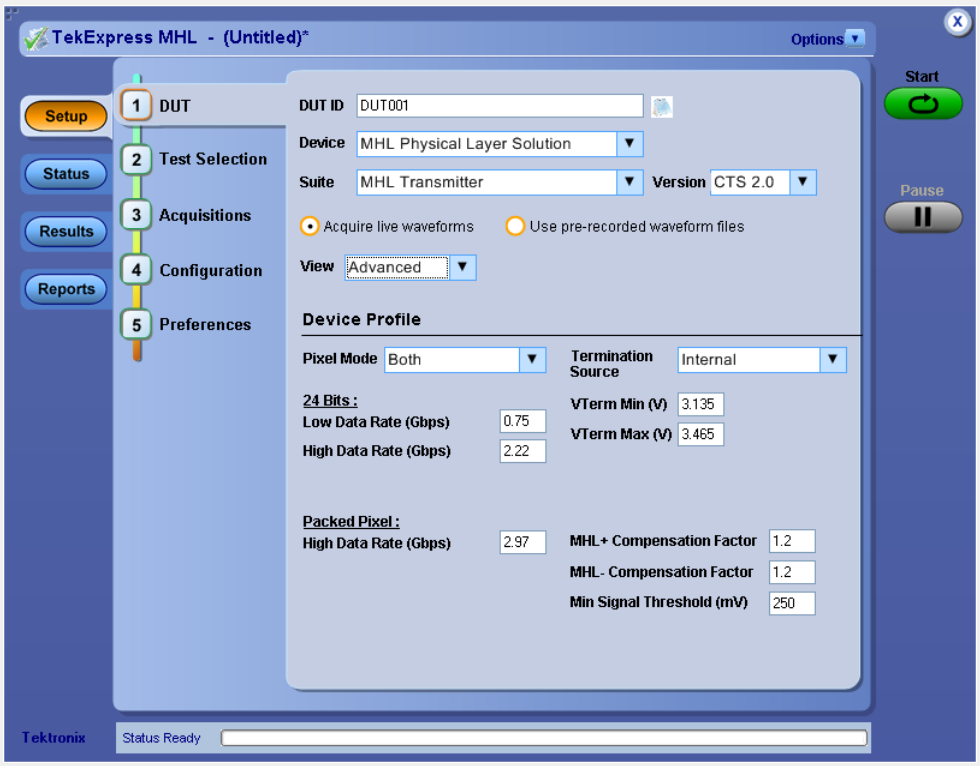
Limits Editor



Advanced View



Setup Panel



Selecting Device Parameters: the DUT tab

1. Click the DUT (see page 62) tab on the Setup panel. Here, specify the DUT ID, device, suite (MHL Transmitter or Receiver), CTS version, view, acquisition method and Device Profile settings.

   These settings apply to all current tests, and affect the test list in the **Test Selection** tab.

2. Enter the device **ID**. (The default value is DUT001.) This ID appears on test reports. To add comments to the top of a report, click and enter up to 256 characters.

3. For the Suite option, choose **MHL Transmitter** or **MHL Receiver.**

4. Select either **Version CTS 1.X** or **CTS 2.0**. This choice determines which Device Profile options appear on the screen, and which tests appear on the Test Selection tab.

5. *MHL Transmitter only:* Select either "Acquire Live Waveforms" or "Use pre-recorded waveform files" for testing.
6. *MHL Transmitter only:* For the View option, select either **Compliance** or **Advanced**.
   - If you choose the **Advanced** view, an additional **Configuration** tab appears.
   - To access configuration options for **Compliance** view, click the **Configure** button on the Test Selection tab.
7. Set the **Device Profile** options (again, these will differ depending upon the CTS version selected above).
   - *MHL Transmitter only:* If you select Version **CTS 1.X**, you will see this DUT screen. (see page 57) These Device Profile options will be available.

Table 9: Device Profile options for CTS 1.X (Transmitter)

Setting	Option	Description
Pixel Mode	24 Bits	This is the only option available.
Low Resolution (MHz)	25	Enter a value from 25 to 100.
High Resolution (MHz)	74.25	Low Resolution value should be less than or equal to High Resolution, and High Resolutions should be greater than or equal to Low resolution.
Num of Other Resolutions	0	If the DUT can support more than the three standard resolutions, enter the number here. Value can be from 0 to 10. Applicable tests will be run for the number of other resolutions specified here.
MHL+ Compensation Factor	1.2	Used for fixture compensation. Value should be between 1 and 2.
MHL– Compensation Factor	1.2	Used for fixture compensation. Value should be between 1 and 2.
Min Signal Threshold	250	Used for signal validation. Value should be between 50 and 600 mV.
Termination Source	Internal	Select this if the termination voltage control mode is INT (Internal).
	External Other	Select this if the termination voltage control mode is EXT (External) and you are using an external power supply other than Tek Power Supply to control probe termination.
	Tek Power Supply	Select this if the termination voltage control mode is EXT (External) and you are using a Tektronix Power Supply as the power source. ***NOTE.*** *Automated control of the Tek Power Supply is not available in this release.*

Setting	Option	Description
VTerm Min (V)	3.135	Value must be greater than or equal to (≥) 1V and less than or equal to (≤) VTerm Max.
VTerm Max (V)	3.465	Value must be greater than or equal to (≥) VTerm Min, and less than or equal to (≤) 5V.

- *MHL Transmitter only:* If you select Version **CTS 2.0**, the DUT screen and Device Profile options you will see will differ depending upon the Pixel Mode selected. This screen shows the DUT tab with Pixel Mode of Both (see page 59). These Device Profile options will be available.

Table 10: Device Profile options for CTS 2.0 (Transmitter)

Setting	Option	Description
Pixel Mode	24 Bits	Select this if the DUT uses 24 Bit mode.
	Packed Pixel	Select this if the DUT uses packed pixel mode.
	Both	Select this if the DUT uses both 24 Bit and packed pixel mode.
When Pixel Mode is 24 Bits:		
Low Data Rate (Gbps)	0.75	Enter a value from 0.75 – 2.22 Gbps.
High Data Rate (Gbps)	2.22	Low Data Rate value should be less than or equal to High Data Rate and High Data Rate values should be greater than or equal to Low Data Rate.
Num of Other Data Rates	0	If the DUT can support more than the three standard resolutions, enter the number here. Value can be from 0 to 10. Applicable tests will be run for the number of other resolutions specified here.
When Pixel Mode is Packed Pixel:		
Low Data Rate (Gbps)	2.25	Enter a value from 2.25 – 2.97 Gbps.
High Data Rate (Gbps)	2.97	Low Data Rate value should be less than or equal to High Data Rate and High Data Rate values should be greater than or equal to Low Data Rate.
Num of Other Data Rates	0	If the DUT can support more than the three standard resolutions; enter the number here. Value can be from 0 to 10. Applicable tests will be run for the number of other resolutions specified here.
When Pixel Mode is Both:		
24 Bits:		
Low Data Rate (Gbps)	0.75	Enter a value from 0.75 – 2.22 Gbps.
High Data Rate (Gbps)	2.22	Low Data Rate value should be less than or equal to High Data Rate and High Data Rate values should be greater than or equal to Low Data Rate.
Packed Pixel:		
High Data Rate (Gbps)	2.97	Low Data Rate value should be less than or equal to High Data Rate and High Data Rate values should be greater than or equal to Low Data Rate.

Setting	Option	Description
MHL+ Compensation Factor	1.2	Used for fixture compensation. Value should be between 1 and 2.
MHL– Compensation Factor	1.2	Used for fixture compensation. Value should be between 1 and 2.
Min Signal Threshold	250	Used for signal validation. Value should be between 50 and 600 mV.
Termination Source	Internal	Select this if the termination voltage control mode is INT (Internal).
	External Other	Select this if the termination voltage control mode is EXT (External) and you are using an external power supply other than Tek Power Supply to control probe termination.
	Tek Power Supply	Select this if the termination voltage control mode is EXT (External) and you are using a Tektronix Power Supply as the power source. ***NOTE.*** *Automated control of the Tek Power Supply is not available in this release.*
VTerm Min (V)	3.135	Value must be greater than or equal to (≥) 1V and less than or equal to (≤) VTerm Max.
VTerm Max (V)	3.465	Value must be greater than or equal to (≥) VTerm Min, and less than or equal to (≤) 5V.

- *MHL Receiver only:* If you select Version **CTS 1.X**, you will see this DUT tab screen. (see page 60) These Device Profile options will be available.

Table 11: Device Profile options for CTS 1.X (Receiver)

Setting	Option	Description
Pixel Mode	24 Bits	This is the only option available.
Refresh Rate	50 Hz	
	60 Hz	
	Both	
Low Resolution (MHz) when Refresh Rate is 50 Hz	27	Enter a value of 25 MHz, 27 MHz, or 74.25 MHz. Low Resolution value should be less than or equal to High Resolution, and High Resolutions should be greater than or equal to Low resolution.
Low Resolution (MHz) when Refresh Rate is 60 Hz	25	
Low Resolution (MHz) when Refresh Rate is Both	27	
High Resolution (MHz)	74.25	

- *MHL Receiver only:* If you select Version **CTS 2.0**, the DUT screen and Device Profile options you will see will differ depending upon the Pixel Mode selected. This screen shows the DUT tab with Pixel Mode of Both (see page 61). These Device Profile options will be available.

Table 12: Device Profile options for CTS 2.0 (Receiver)

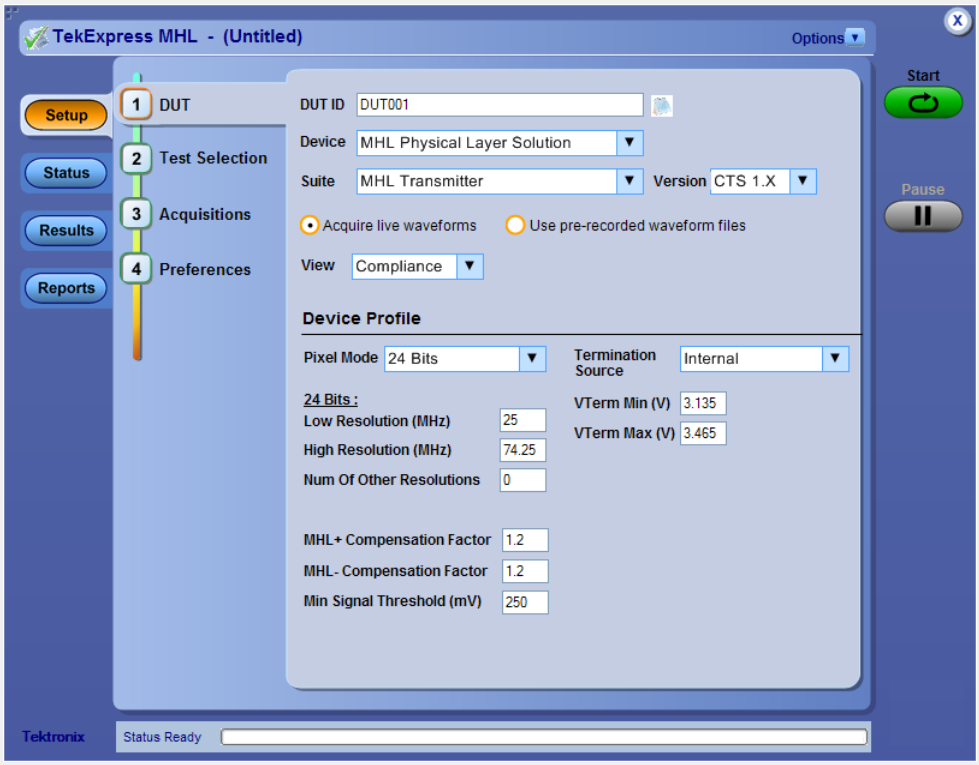
Setting	Option	Description
Pixel Mode	24 Bits	Select this if the DUT uses 24 Bit mode.
	Packed Pixel	Select this if the DUT uses packed pixel mode.
	Both	Select this if the DUT uses both 24 Bit and packed pixel mode.
Low Data Rate (Gbps) when Pixel Mode is 24 Bits, 50 Hz Refresh Rate:	0.81	Enter a value of 0.81 or 2.22.
Low Data Rate (Gbps) when Pixel Mode is 24 Bits, 60 Hz Refresh Rate:	0.75	Enter a value of 0.75, 0.81, or 2.22.
Low Data Rate (Gbps) when Pixel Mode is 24 Bits, Refresh Rate is Both:	0.81	Enter a value of 0.81 or 2.22.
Low Data Rate (Gbps) when Pixel Mode is Packed Pixel, 50 Hz Refresh Rate:	2.97	2.97
Low Data Rate (Gbps) when Pixel Mode is Packed Pixel, 60 Hz Refresh Rate:	2.97	2.97
Low Data Rate (Gbps) when Pixel Mode is Packed Pixel, Refresh Rate is Both:	2.97	2.97
Low Data Rate (Gbps) when Pixel Mode is Both, 50 Hz Refresh Rate:	0.81	Enter a value of 0.81 or 2.22.
Low Data Rate (Gbps) when Pixel Mode is Both, 60 Hz Refresh Rate:	0.75	Enter a value of 0.75, 0.81, or 2.22.
Low Data Rate (Gbps) when Pixel Mode is Both, Refresh Rate is Both:	0.81	Enter a value of 0.81 or 2.22.
High Data Rate (Gbps) when Pixel Mode is Both, Refresh Rate is Both:	2.22	Low Data Rate value should be less than or equal to High Data Rate and High Data Rate values should be greater than or equal to Low Data Rate.
High Data Rate (Gbps, Packed Pixel) when Pixel Mode is Both, Refresh Rate is Both:	2.97	2.97

See Also

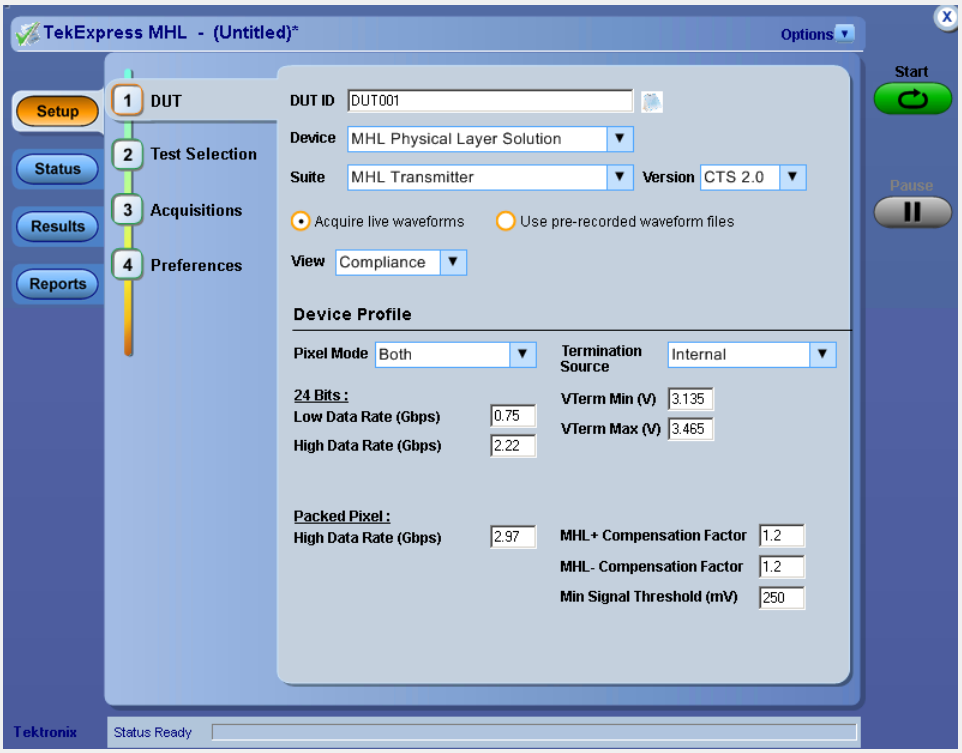
- Setting Up Tests the Setup Panel (see page 47)
- Choosing Tests the Test Selection tab (see page 62)
- Acquiring Waveforms the Acquisitions tab MHL Transmitter only (see page 67)

- Configuring Tests the Configure button or Configuration tab (see page 74)

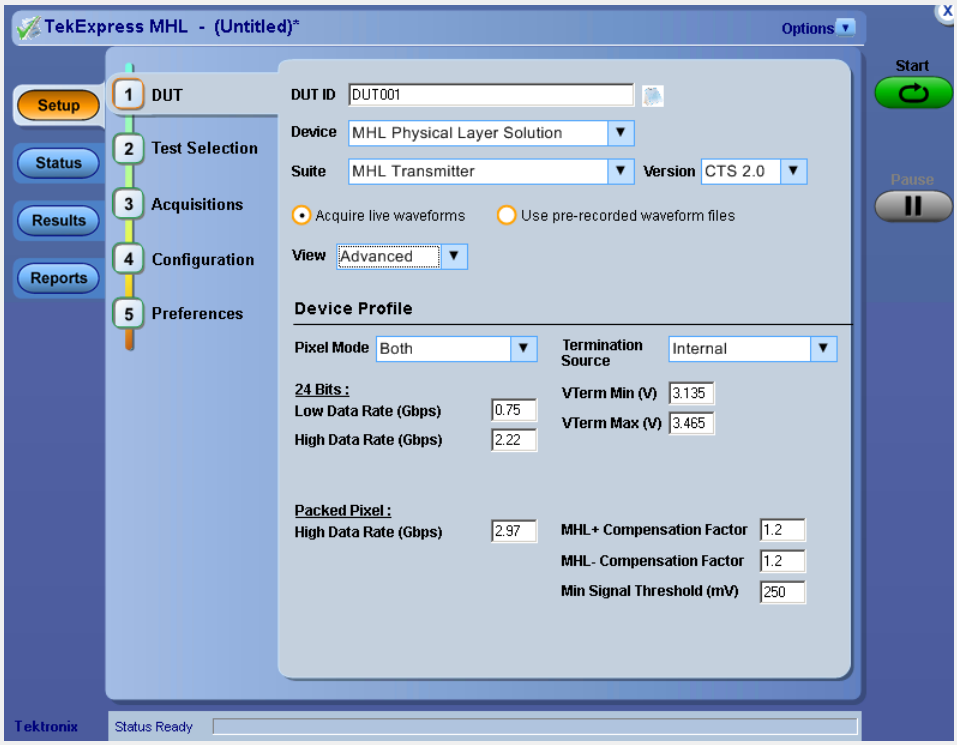
DUT tab – CTS 1.X (Transmitter)



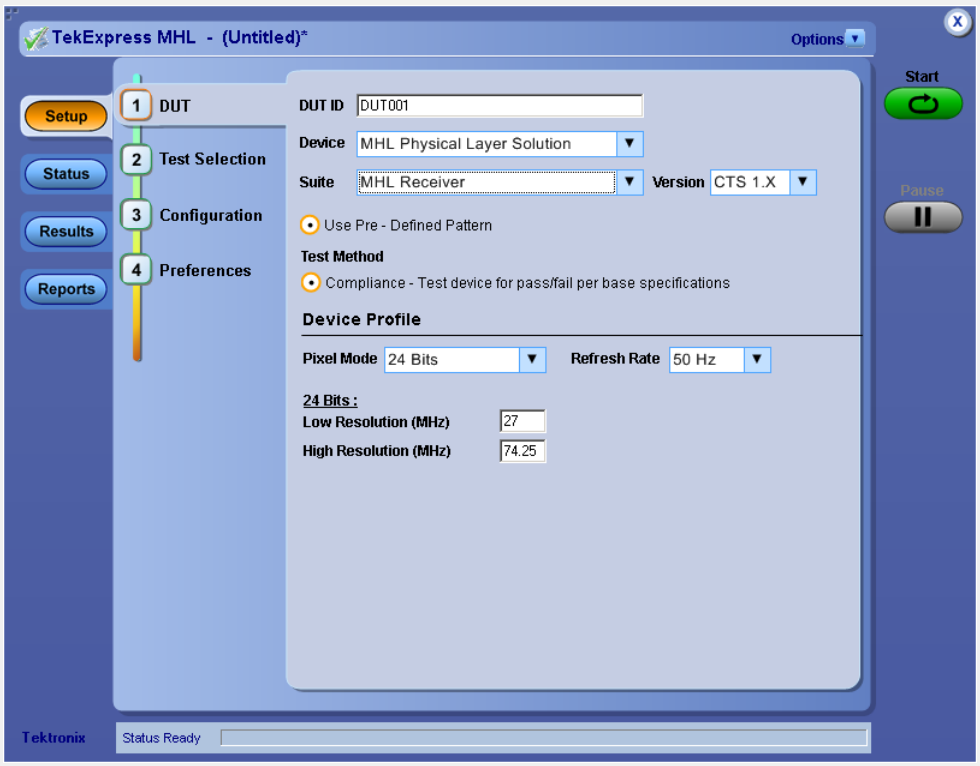
DUT tab – CTS 2.0 (Transmitter)



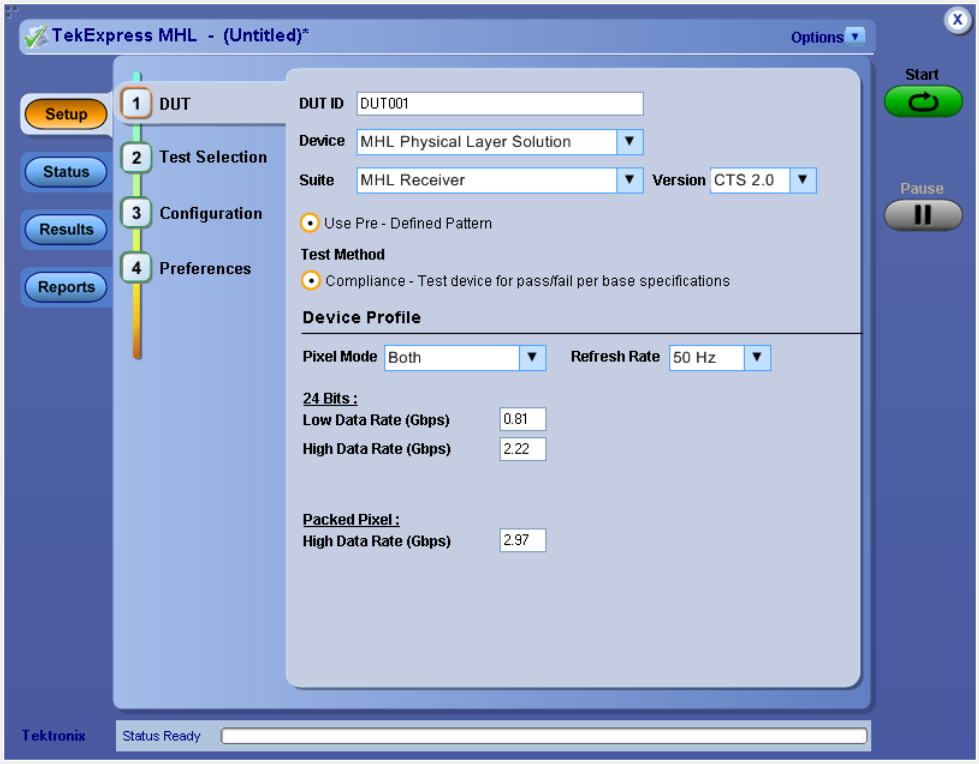
DUT tab with Pixel Mode of Both



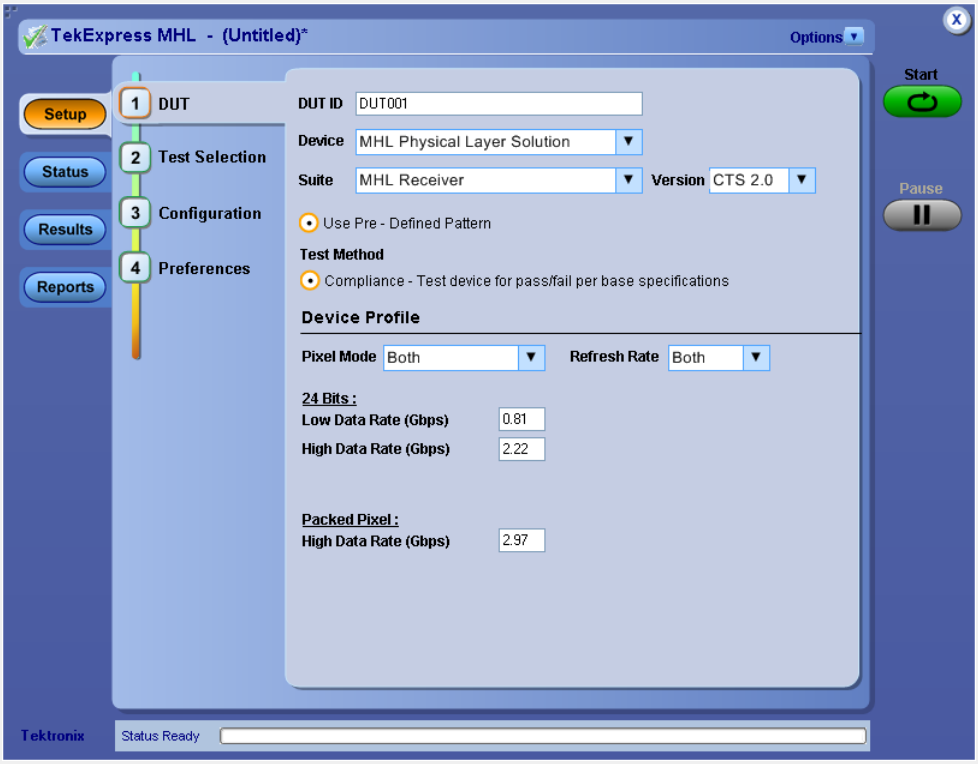
DUT tab – CTS 1.X (Receiver)



DUT tab – CTS 2.0 (Receiver)



DUT tab with Pixel Mode of Both



Choosing Tests: the Test Selection tab

Use these instructions to select the tests to run on the connected DUT. *MHL Transmitter only:* Tests that you select here impact the parameters available in the Acquisitions tab.

1. In the Setup panel, click the Test Selection tab.

   Different sets of tests are displayed depending upon which MHL solution and which CTS version you have selected in the DUT tab.

   If you selected MHL Transmitter and Version CTS 1.X, you will see this screen (see page 66). If you selected MHL Transmitter and Version CTX 2.0, you will see this screen (see page 64).

If you selected MHL Receiver and Version CTS 2.0, you will see this screen (see page 65). If you selected MHL Receiver and Version CTX 1.X, you will see this screen (see page 67).

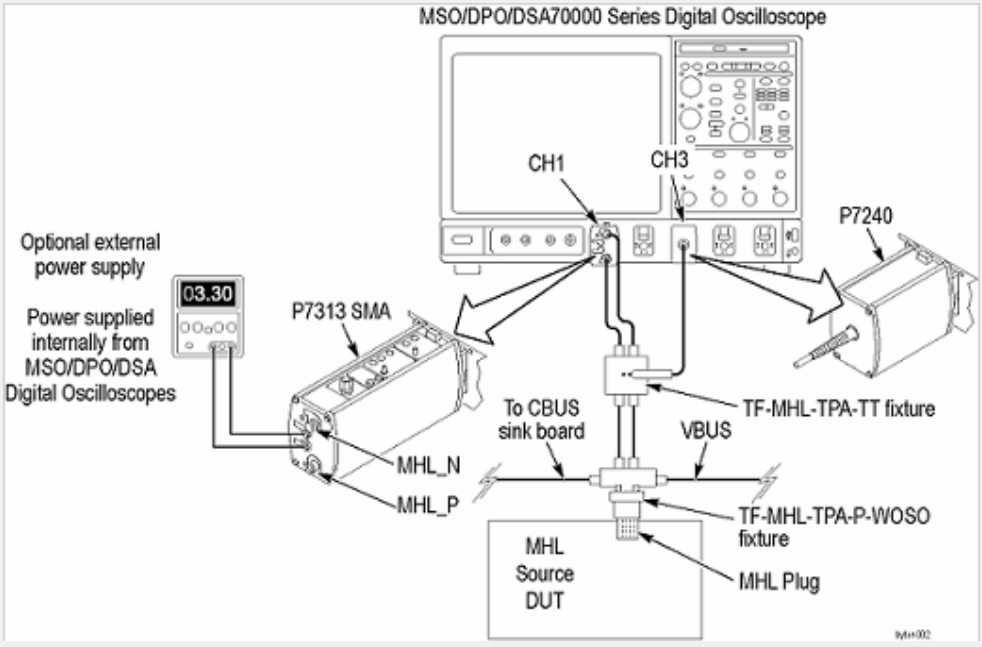
2. All tests are selected by default. Click **Deselect All** unless you want to run all tests.
3. Select the desired tests. For descriptions of the supported MHL Transmitter tests, click here. (see page 5) For descriptions of the supported MHL Receiver tests, click here. (see page 8)
   - Click the check boxes of individual tests or of entire groups of tests. To select all tests in the list, click the **Select All** button.
   - Click on the **Schematic** button to display the schematic document for the selected test (see page 63). Use to verify the test setup before running the test.

Once you have selected the tests, then you can configure the tests (see page 74).

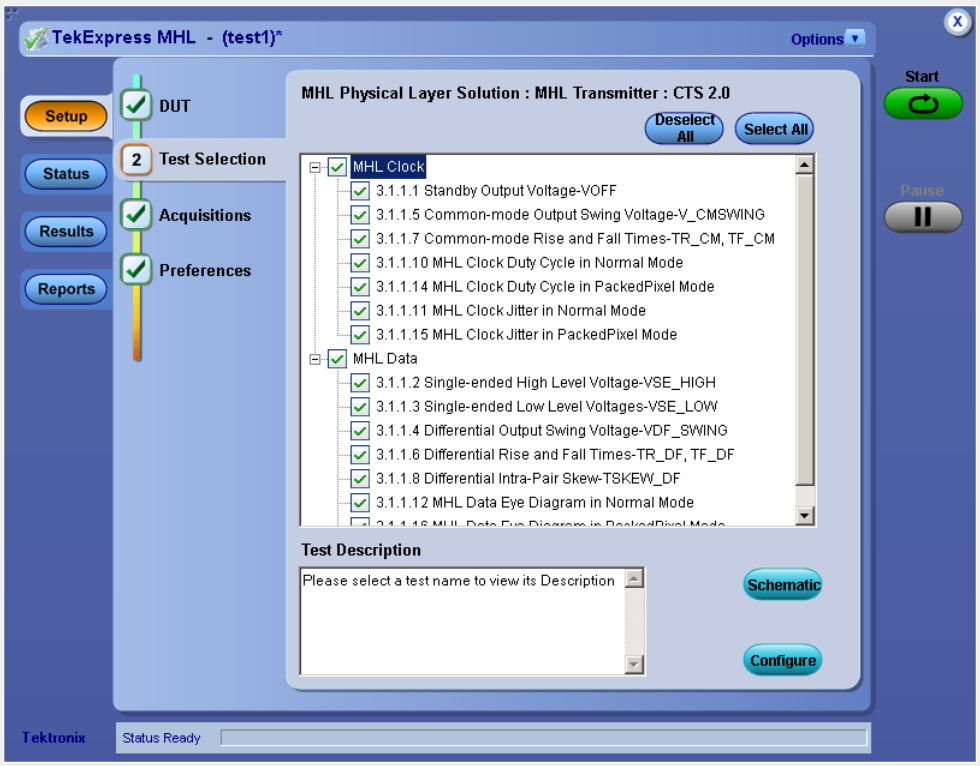
See Also

- Setting Up Tests the Setup Panel (see page 47)
- Acquiring Waveforms (see page 67)
- Configuring Tests the Configure button or Configuration tab (see page 74)
- Running the Tests and Viewing their Progress the Status Panel (see page 89)

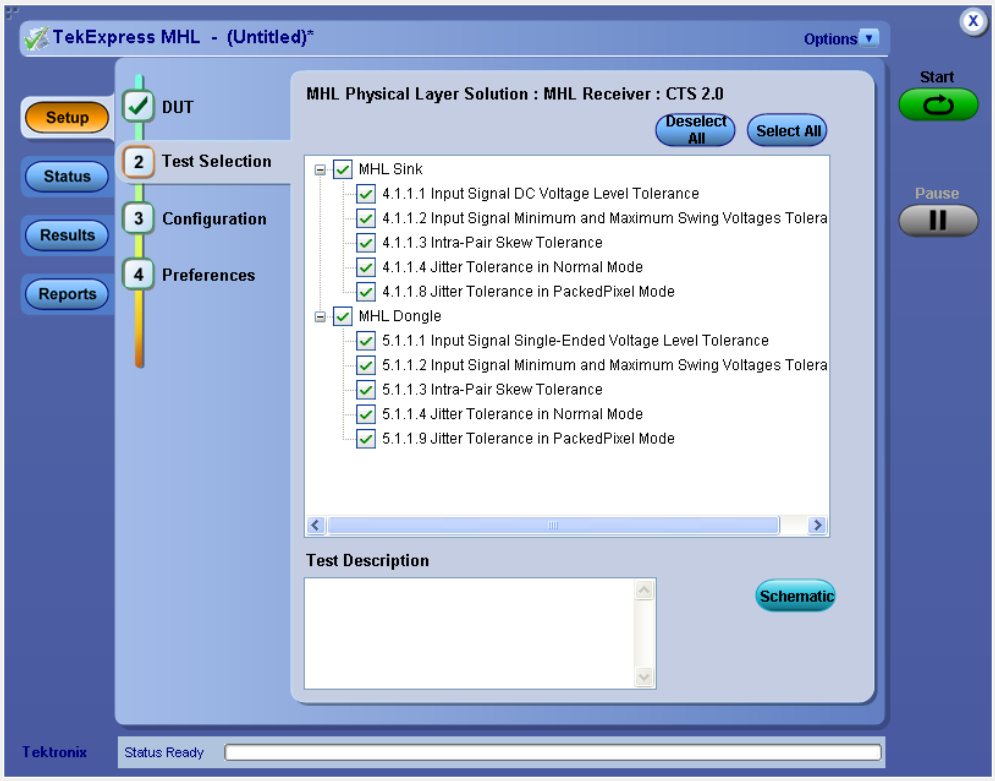
Example of Schematic Document



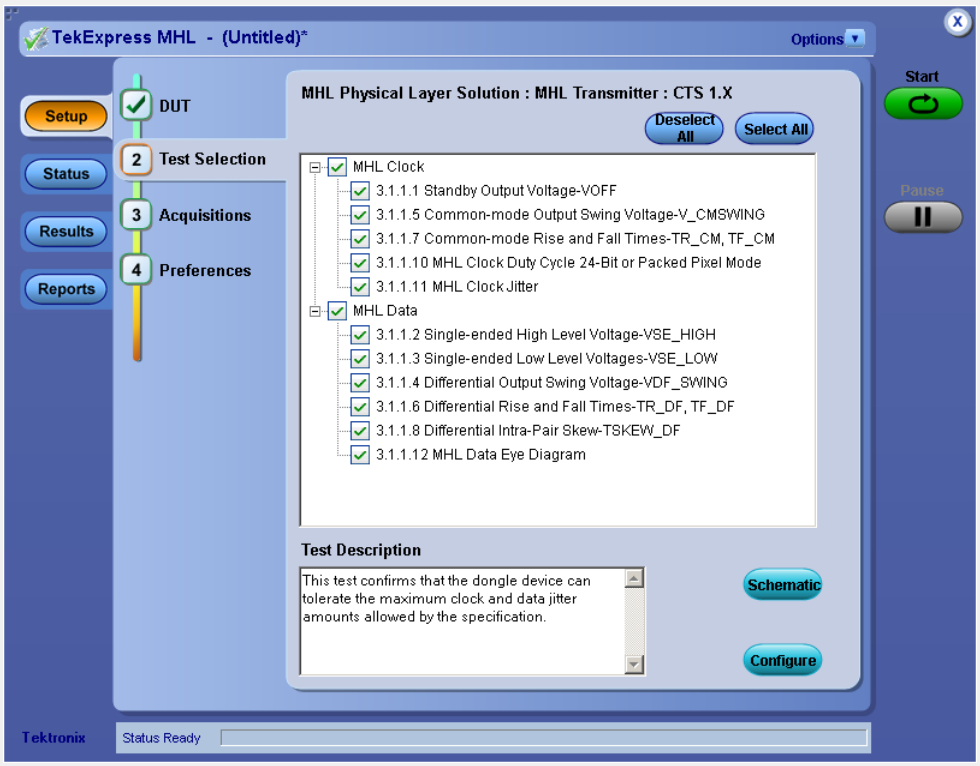
Test Selection tab – (MHL Transmitter with CTS 2.0 selected in DUT tab)



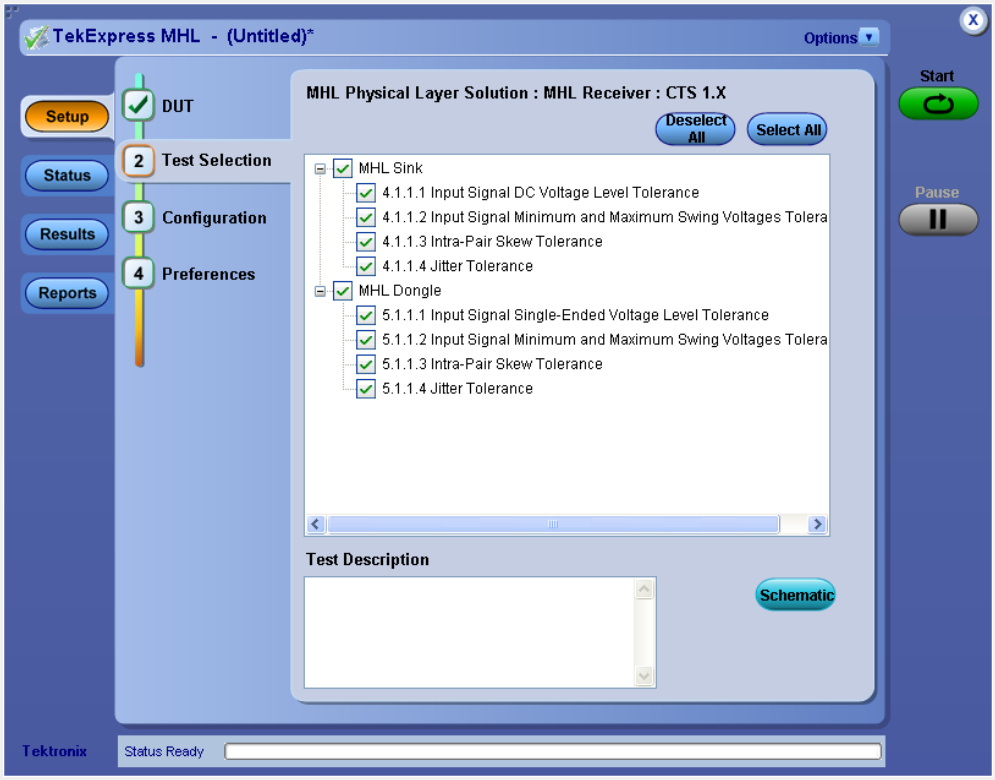
Test Selection tab – (MHL Receiver with CTS 2.0 selected in DUT tab)



Test Selection tab (MHL Transmitter with CTS 1.X selected in DUT tab)



Test Selection tab (MHL Receiver with CTS 1.X selected in DUT tab)



Acquiring Waveforms: the Acquisitions tab (MHL Transmitter only)

The Acquisitions (see page 70) in the Setup panel is used to view and set acquisitions parameters for the selected tests. Before you can do this, you must first:

- Choose either **Acquire Live Waveforms** or **Use Pre-recorded waveform files** in the Setup Panel on the **DUT** tab.
- The acquisition parameters displayed on the Acquisitions tab will differ depending on your choice on the DUT tab.

If you have chosen **Acquire Live Waveforms** on the DUT tab:

1. In the Setup panel, click the Acquisitions (see page 70) tab.
2. Select the probe source channel for each listed MHL signal in the Signal selection table.

MHL Signal Name	Source
MHL+ / Differential	CH1
MHL-	CH2
Common mode Clock	CH3

3. You may click the View Probes (see page 70) button to view probe configuration for each channel used.

4. You'll see the Acquisitions table. For information about the possible parameter columns displayed in the Acquisitions table, click here.

   The table below lists the possible parameter columns displayed in the Acquisitions table in the Acquisitions tab of the Setup panel. Columns displayed depend on the tests selected in the **Test Selection** tab, and whether or not the **Show Acquire Parameters** check box has been selected.

Column name	Function
Test Name	Displays the name of the selected test for performing acquisitions. One or more tests can perform the same acquisitions.
Acquisition	Updates the location of the named acquisition
Record Length	Displays the size of the record in M
Sample Rate (GS/s)	How often the oscilloscope takes a snapshot or sample of the signal
Diff Probe Vertical Scale (mV/Div)	Displays the vertical scale for the differential probe
Diff Probe Vertical Position (Div)	Displays the vertical position of the differential probe
CM Probe Vertical Scale (mV/Div)	Displays the vertical scale for the differential probe
CM Probe Vertical Position (Div)	Displays the vertical position of the differential probe
Probe attenuation (X)	Displays attenuation factor for the probe
Edge Trigger Level (%)	Displays the edge trigger level, in percentage
Measurement Repetition (count)	Displays the number of measurement repetitions
Population Count	Specifies a limit to the amount of waveform data that is analyzed
Vertical Scale	Displays the vertical scale for the non differential probe
Vertical Position	Displays the vertical position for the non differential probe
WaveForm FileName	Displays the name and location of the waveform file to be used for the measurement. Applies only to testing using prerecorded waveforms.

5. Underneath the table, you may check the Show Acquired Parameters check box. When selected, the acquisition parameters for each test display in additional columns in the Acquisitions table.

6. You may check the Acquire Step by Step check box. When selected, the software prompts you to continue after each phase of the test completes.

If you have chosen **Use Pre-recorded waveform files** on the DUT tab:

1. In the Setup panel, click the Acquisitions (see page 71) tab.

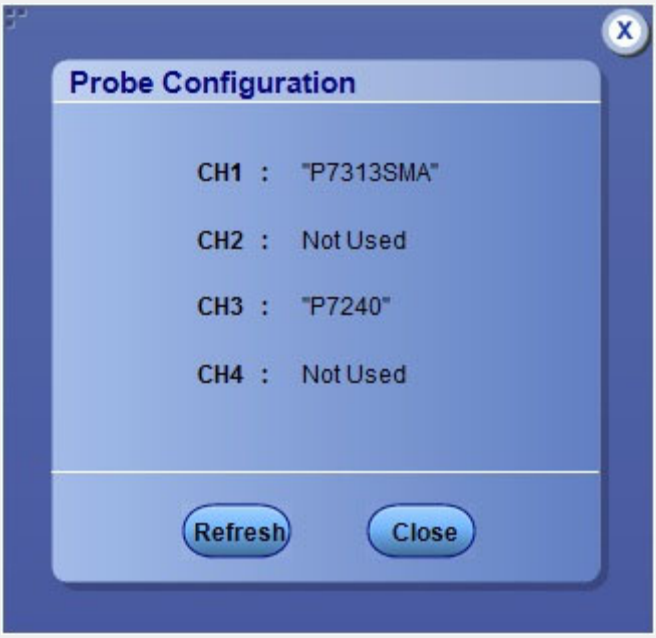
2. You'll see the Acquisitions table. Locate the row for the desired test and then click the ellipsis button () in the Waveform FileName column. Select a file. You can select more than one file for each test. For information about the possible parameter columns displayed in the Acquisitions table, click here.

The table below lists the possible parameter columns displayed in the Acquisitions table in the Acquisitions tab of the Setup panel. Columns displayed depend on the tests selected in the **Test Selection** tab, and whether or not the **Show Acquire Parameters** check box has been selected.

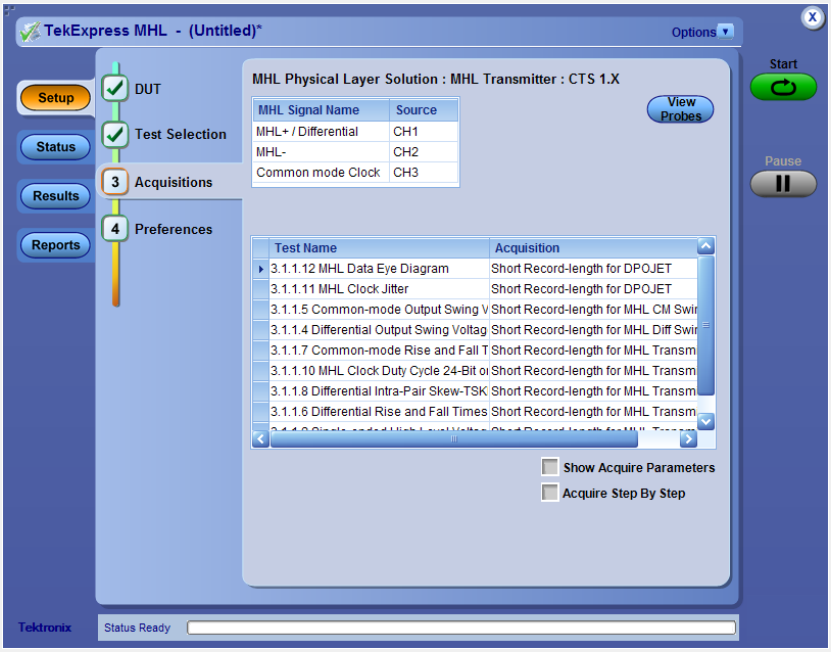
Column name	Function
Test Name	Displays the name of the selected test for performing acquisitions. One or more tests can perform the same acquisitions.
Acquisition	Updates the location of the named acquisition
Record Length	Displays the size of the record in M
Sample Rate (GS/s)	How often the oscilloscope takes a snapshot or sample of the signal
Diff Probe Vertical Scale (mV/Div)	Displays the vertical scale for the differential probe
Diff Probe Vertical Position (Div)	Displays the vertical position of the differential probe
CM Probe Vertical Scale (mV/Div)	Displays the vertical scale for the differential probe
CM Probe Vertical Position (Div)	Displays the vertical position of the differential probe
Probe attenuation (X)	Displays attenuation factor for the probe
Edge Trigger Level (%)	Displays the edge trigger level, in percentage
Measurement Repetition (count)	Displays the number of measurement repetitions
Population Count	Specifies a limit to the amount of waveform data that is analyzed
Vertical Scale	Displays the vertical scale for the non differential probe
Vertical Position	Displays the vertical position for the non differential probe
WaveForm FileName	Displays the name and location of the waveform file to be used for the measurement. Applies only to testing using prerecorded waveforms.

3. Underneath the Acquisitions table, you may check the Show Acquired Parameters check box. When selected, the acquisition parameters for each test display in additional columns in the Acquisitions table.

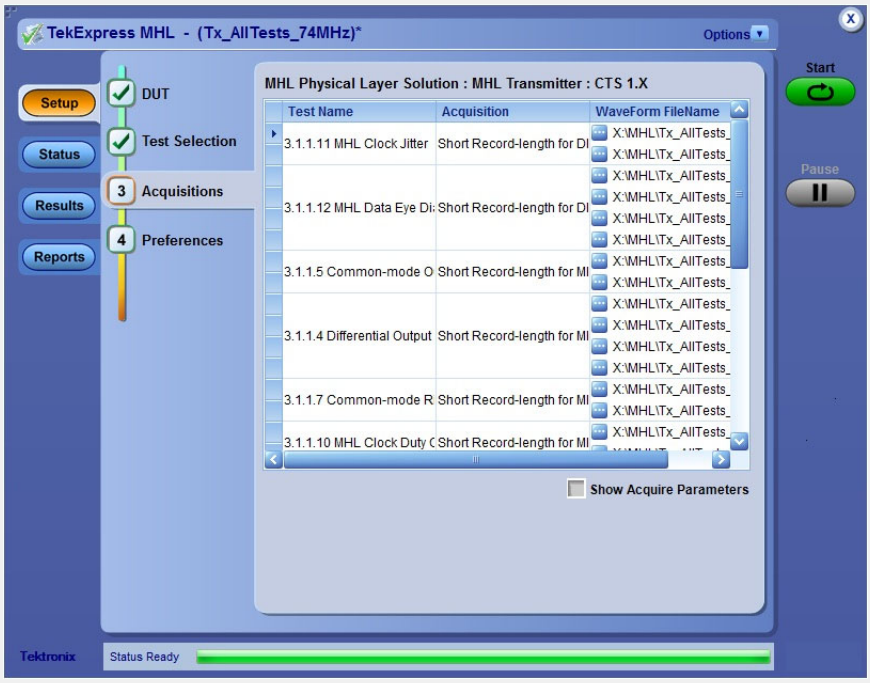
View Probes Button



Acquisitions Tab with Acquire Live Waveforms selected



Acquisitions Tab with Use Pre-recorded Waveform Files selected



Acquisitions Table

The table below lists the possible parameter columns displayed in the Acquisitions table in the Acquisitions tab of the Setup panel. Columns displayed depend on the tests selected in the **Test Selection** tab, and whether or not the **Show Acquire Parameters** check box has been selected.

Column name	Function
Test Name	Displays the name of the selected test for performing acquisitions. One or more tests can perform the same acquisitions.
Acquisition	Updates the location of the named acquisition
Record Length	Displays the size of the record in M
Sample Rate (GS/s)	How often the oscilloscope takes a snapshot or sample of the signal
Diff Probe Vertical Scale (mV/Div)	Displays the vertical scale for the differential probe
Diff Probe Vertical Position (Div)	Displays the vertical position of the differential probe
CM Probe Vertical Scale (mV/Div)	Displays the vertical scale for the differential probe
CM Probe Vertical Position (Div)	Displays the vertical position of the differential probe
Probe attenuation (X)	Displays attenuation factor for the probe
Edge Trigger Level (%)	Displays the edge trigger level, in percentage
Measurement Repetition (count)	Displays the number of measurement repetitions
Population Count	Specifies a limit to the amount of waveform data that is analyzed
Vertical Scale	Displays the vertical scale for the non differential probe
Vertical Position	Displays the vertical position for the non differential probe
WaveForm FileName	Displays the name and location of the waveform file to be used for the measurement. Applies only to testing using prerecorded waveforms.

See also

- Use Prerecorded Waveforms for Analysis MHL Transmitter only (see page 72)
- Acquire Live Waveforms for Analysis MHL Transmitter only (see page 73)
- Setting Up Tests the Setup Panel (see page 47)

Use Prerecorded Waveforms for Analysis (MHL Transmitter only)

NOTE. *If you are using the prerecorded waveform files option, it is recommended that you use a waveform file (.wfm) that was captured from a Tektronix oscilloscope. This eliminates the need to use an oscilloscope. You can manually select waveforms and perform the tests by clicking the* ***Start*** *button.*

1. Open a saved test setup (see page 102) or create a new one (see page 102).
2. In the Setup panel, select the **DUT** tab, select the desired DUT and then select **Use pre-recorded waveform files**.
3. In the **Test Selection** tab, select the desired tests (see page 62).
4. In the **Acquisitions** tab, locate the row for the desired test in the Acquisitions Table (see page 72). In the **Waveform FileName** column for that test, click the ellipses button () and select the desired waveform file.
5. Perform any additional desired test setup, such as test configuration (see page 74), and then click **Start** to run the test.

You can select a different waveform file for each test, deselect a test from the current run, or add a test to the current run. Once you click **Start**, data corresponding to that test run populates the other panels (such as the test status and test results).

See also

- Acquiring Waveforms the Acquisitions tab MHL Transmitter only (see page 67)
- Acquire Live Waveforms for Analysis MHL Transmitter only (see page 73)

Acquire Live Waveforms for Analysis (MHL Transmitter only)

Use these instructions to set up acquisition for a live waveform.

1. Open a saved test setup (see page 102) or create a new one (see page 102).
2. In the Setup panel, select the **DUT** tab, enter the desired DUT and then select **Acquire live waveforms**.
3. Select other DUT options (see page 52) as desired.
4. In the **Test Selection** tab, select the desired test(s) (see page 62).
5. In the **Acquisitions** tab, select or view the desired parameters in the Acquisitions Table (see page 72).
6. Configure the tests (see page 74) if you have not done so already. Then click **Start** to run the test.

See also

- Acquiring Waveforms the Acquisitions tab MHL Transmitter only (see page 67)
- Use Prerecorded Waveforms for Analysis MHL Transmitter only (see page 72)

Configuring Tests: the Configuration Screen

Once you've selected tests, you may use the Configuration screen (see page 75). There are two ways to reach the screen: the Configuration tab or the Configure button.

The MHL solution you are using determines how you may reach the Configuration screen.

*If you chose **MHL Transmitter** in the Setup panel DUT tab:*

- and you selected **Compliance** View in the Setup panel DUT tab, click on the **Test Selection** tab and select the desired test. Then click on the Configure (see page 77) button in the lower right corner of the Test Selection tab.
- and you selected **Advanced** View in the Setup panel DUT tab, the Configuration tab (see page 75) will appear on the Setup panel – click on it.

*If you chose **MHL Receiver** in the Setup panel DUT tab:* You may view the configuration settings by either clicking on the **Configuration** tab, or clicking on the **Configure** button on the Test Selection tab.

The **Configuration** screen shows **Global** parameters, which are common for all tests, and **Measurement** parameters, which are specific to selected tests, including acquisition, analysis, and limit parameters.

NOTE. *Test parameters that are grayed cannot be changed.*

Settings

- Compliance Mode or User Defined Mode (see page 78) *(MHL Transmitter only.)*
- Global Settings tab

  In the **Instruments Detected** section, click in the shaded area opposite **Real Time Scope** and select the desired instrument from the drop-down list. If you don't see the instrument, refresh the list by using the Instrument Control Settings (see page 37).
- DSP Filter (8 GHz) *(MHL Transmitter only.)*

  You may check this box if the DUT has a bandwidth greater than 8 GHz. This applies an 8 GHz low pass filter to all transmitter tests.
- Measurements tab
  - *If you are using MHL Transmitter:*

    You may change test parameters. Select the test whose parameters you'd like to change. The options below the test list change to reflect the selected test.

    To modify the parameters, select the desired tab and parameters. For parameter details, see Measurement Parameter Descriptions: MHL Transmitter (see page 79).
  - *If you are using MHL Receiver:*

You may not change test parameters, but you may view the Measurement Parameter Descriptions: MHL Receiver (see page 83).

- Limits Editor

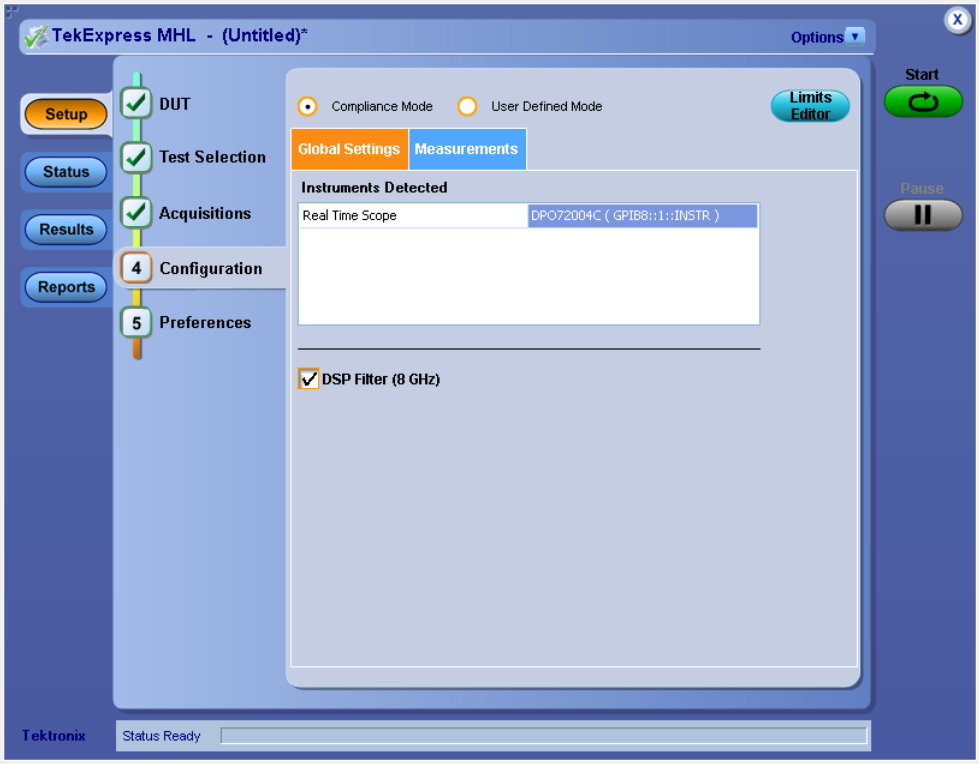
  Use the Limits Editor (see page 76) button in the upper right corner to view or change the High Limit and Low Limit values used for each measurement.

Next, prepare to run the tests. (see page 85)

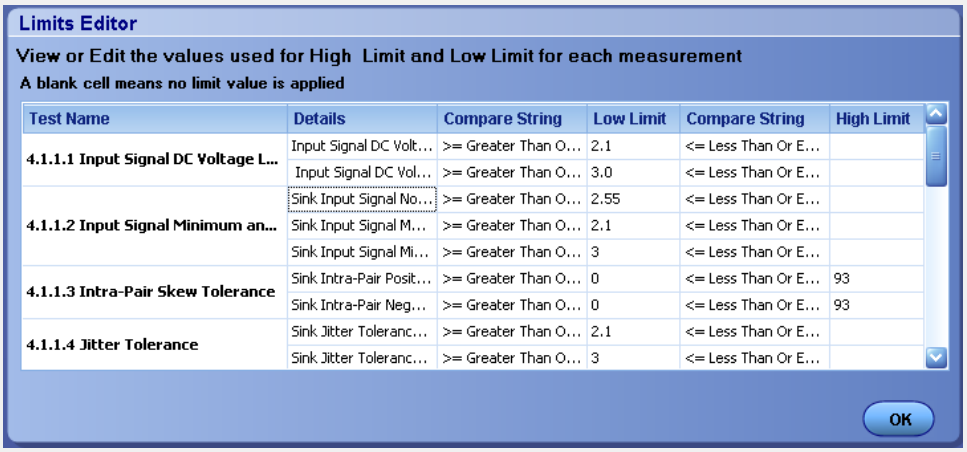
See Also

- Setting Up Tests: the Setup Panel (see page 47)
- Selecting Device Parameters: the DUT tab (see page 52)
- Choosing Tests: the Test Selection tab (see page 62)
- Acquiring Waveforms: the Acquisitions tab MHL Transmitter only (see page 67)
- About Saving and Recalling Test Setups (see page 101)

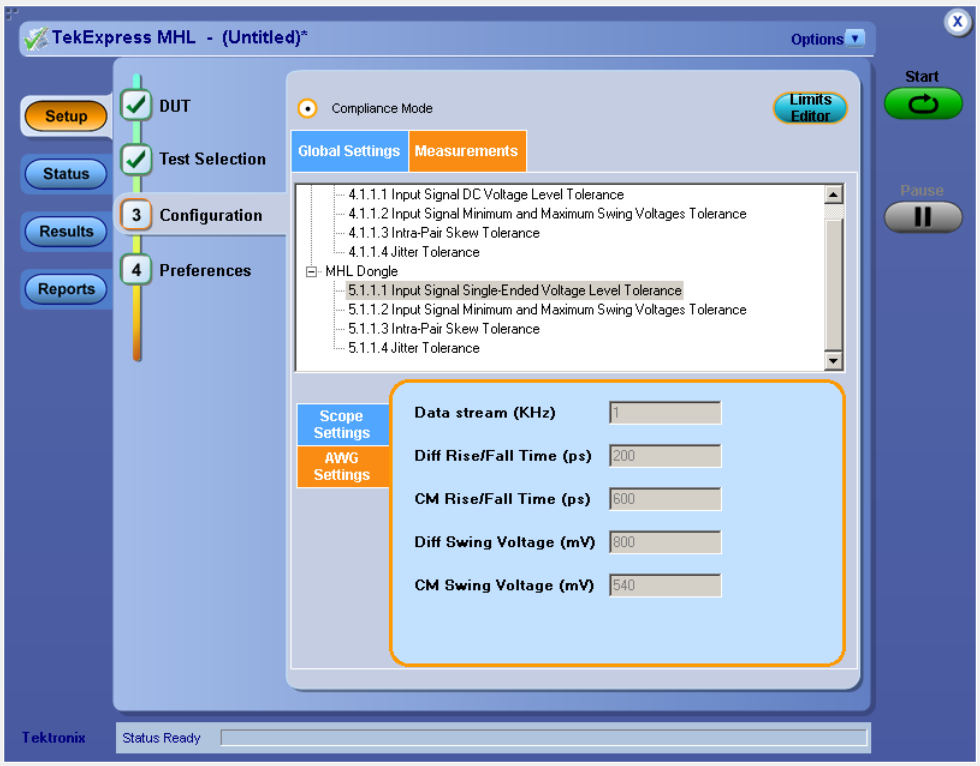
Configuration Screen



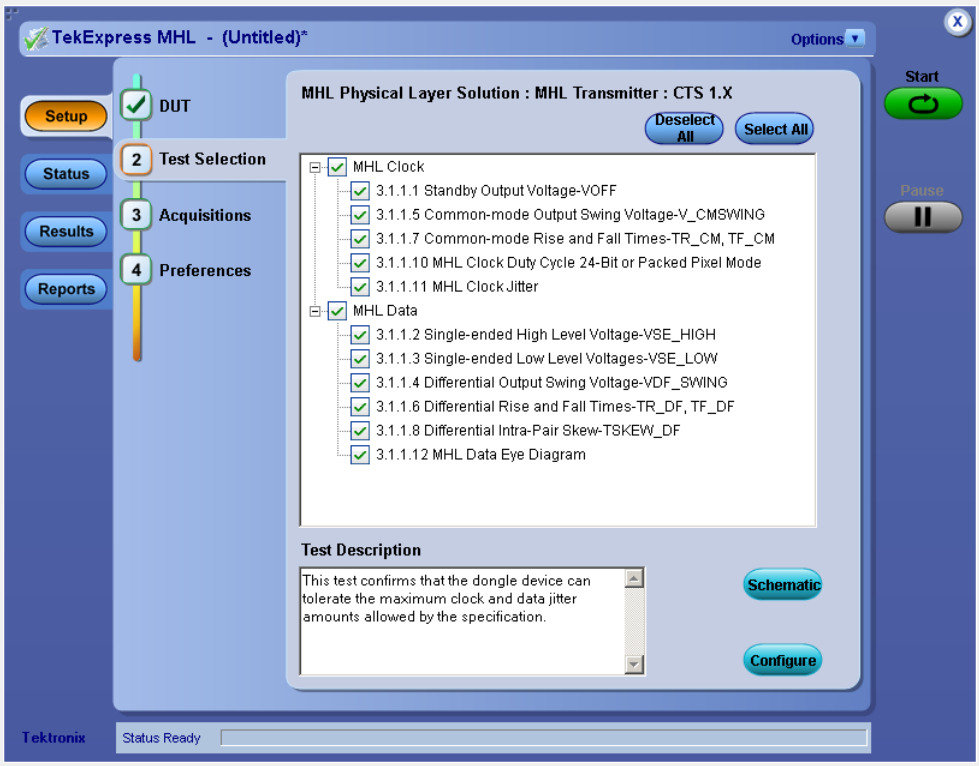
Limits Editor



Measurements Tab – MHL Receiver

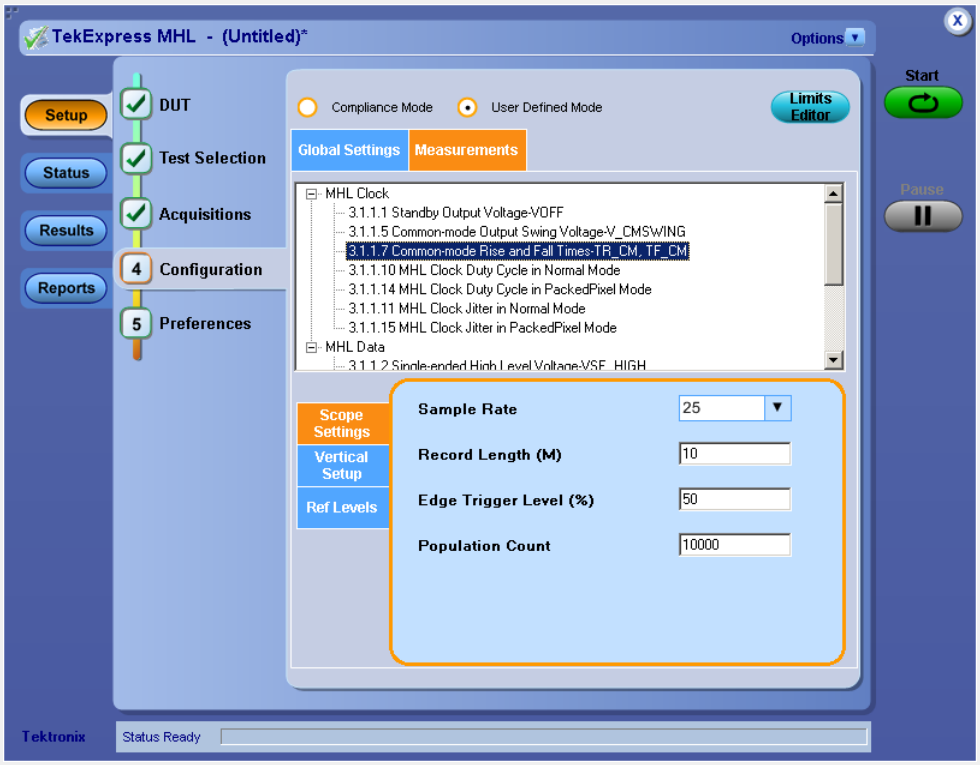


Configure button on Test Selection tab



Configuration tab

Measurements Tab – MHL Transmitter



Compliance Mode or User Defined Mode

From the Configuration screen, if you are using MHL Transmitter, you will have the option to select either Compliance Mode or User Defined Mode.

- **Compliance Mode**: Select to use Compliance Mode values. You cannot change most test parameters in Compliance Mode, but you can view the compliance parameters.
- **User Defined Mode**: Select to run tests using custom parameters. You may change parameters that are not grayed out.

Note that these modes are not to be confused with the two views available for MHL Transmitter: **Compliance View** and **Advanced View** (selected in the **DUT** tab of the Setup panel).

Pairing Modes and Views for Test Operation

Your selections of Modes and Views will change the way the tests operate. These pairings of views and modes are possible for MHL Transmitter.

- **Compliance View** selected with **Compliance Mode**:
  Tests will run automatically with little or no user intervention. You will not be able to change test parameters to anything that deviates from the compliance standards. To view configuration options, click on the Test Selection tab of the Setup panel and click the Configure button.
- **Compliance View** selected with **User Defined Mode**: Tests will run automatically but you will be able to change test parameters before starting the tests. To view configuration options, click on the Test Selection tab of the Setup panel and click the Configure button.
- **Advanced View** selected with **Compliance Mode**: Tests will run automatically with little or no user intervention. You will not be able to change test parameters to anything that deviates from the compliance standards. To view configuration options, click on the Configuration tab of the Setup panel.
- **Advanced View** selected with **User Defined Mode**: Tests will run automatically but you will be able to change test parameters before starting the tests. To view configuration options, click on the Configuration tab of the Setup panel.

Measurement Parameter Descriptions (MHL Transmitter)

View or change measurement parameters in the **Configuration** tab of the Setup panel. Measurement parameters are displayed for the test selected in the tree view section. Not all of the parameters listed apply to all tests, and some are only available when running tests in **User Defined** Mode. You cannot change most parameters if you selected **Compliance** Mode.

Table 13: MHL Transmitter Clock and MHL Data measurements

Parameter type	Parameter	Description	Applies to ...
Scope Settings	Sample Rate	Specified in GS/s. How often the digital oscilloscope takes a snapshot or sample of the signal.	All tests
	Record Length (M)	Set the record length to use.	All tests
	Edge Trigger Level (%)	Select the edge trigger level, in percentage.	All tests
	Population Count	Specifies a limit to the amount of waveform data that is analyzed.	3.1.1.2, 3.1.1.3, 3.1.1.4, 3.1.1.5, 3.1.1.6, 3.1.1.7, 3.1.1.8
	Measurement Repetition (Count)	Specifies the number of measurement repetitions used to measure the average value in the Clock Jitter test (3.1.1.11). If the Required Population count is not received from the acquired waveform, the application will take that count from this field.	3.1.1.11, 3.1.1.15

Table 13: MHL Transmitter Clock and MHL Data measurements (cont.)

Parameter type	Parameter	Description	Applies to ...
Vertical Setup	Probe Attenuation (X)	Select the appropriate attenuation factor for the probe.	All tests
	Diff Probe Vertical Position (Div)	Select the vertical position of the differential probe.	3.1.1.4, 3.1.1.5, 3.1.1.6, 3.1.1.7, 3.1.1.10, 3.1.1.11, 3.1.1.12, 3.1.1.14, 3.1.1.15, 3.1.1.16
	Diff Probe Vertical Scale (mV/Div)	Select the vertical scale for the differential probe.	3.1.1.4, 3.1.1.5, 3.1.1.6, 3.1.1.7, 3.1.1.10, 3.1.1.11, 3.1.1.12, 3.1.1.14, 3.1.1.15, 3.1.1.16
	CM Probe Vertical Position (Div)	Select the vertical position for the common mode probe.	3.1.1.4, 3.1.1.5, 3.1.1.6, 3.1.1.7, 3.1.1.10, 3.1.1.11, 3.1.1.12, 3.1.1.14, 3.1.1.15, 3.1.1.16
	CM Probe Vertical Scale (mV/Div)	Select the vertical scale for the common mode probe.	3.1.1.4, 3.1.1.5, 3.1.1.6, 3.1.1.7, 3.1.1.10, 3.1.1.11, 3.1.1.12, 3.1.1.14, 3.1.1.15, 3.1.1.16
	Vertical Position (Div)	Used in vertical setup to position the waveform. Sets the number of divisions of the signal. Adjusts the display of the graticule relative to the vertical acquisition window.	3.1.1.2, 3.1.1.3, 3.1.1.8
	Vertical Scale (mV/Div)	Used in vertical setup to position the waveform. Specifies the waveform vertical scale.	3.1.1.2, 3.1.1.3, 3.1.1.8

Table 13: MHL Transmitter Clock and MHL Data measurements (cont.)

Parameter type	Parameter	Description	Applies to ...
Ref Levels	Absolute	Use to manually set the reference levels.	All Tests
	Percentage	Use to set the reference levels as a percentage.	All tests
	Mid Level	A reference voltage level that defines when the waveform state transition occurs at a given threshold.	All tests
	Hysteresis	Used to prevent small amounts of noise in a waveform from producing multiple threshold crossings. Use when the rising and falling thresholds for a given reference voltage level are set to the same value.	All tests
Clock Settings	PLL Model	Selects between Type 1 and Type 2 phase-locked loop.	3.1.1.11, 3.1.1.12, 3.1.1.15, 3.1.1.16
	Clock Recovery Method	■ PLL-Custom BW ■ Constant Clock-Mean ■ Explicit Clock: (Applies to test 3.1.1.12 only) This method derives the reference clock from a channel other than the one upon which the measurement is defined.	3.1.1.11 and 3.1.1.12, 3.1.1.15, 3.1.1.16
	Clock Multiplier	Specifies the number of edges to be used in the measurement.	3.1.1.12, 3.1.1.16
	Loop Band Width (MHz)	Sets the bandwidth of the clock recovery PLL.	3.1.1.12, 3.1.1.16
	Loop Band Width Low Res. (MHz)	Sets the bandwidth of the clock recovery PLL.	3.1.1.11, 3.1.1.15
	Loop Band Width High Res. (MHz)	Sets the bandwidth of the clock recovery PLL.	3.1.1.11, 3.1.1.15
	Clock Edge	Specifies whether the rising, falling, or both edges of the selected source should be considered in the measurement.	3.1.1.11, 3.1.1.12, 3.1.1.15, 3.1.1.16

See Also

- Measurement Parameter Descriptions: MHL Receiver (see page 83)
- Configuring Tests: the Configure button or Configuration tab

Measurement Parameter Descriptions (MHL Receiver)

View or change measurement parameters in the **Configuration** tab of the Setup panel. Measurement parameters are displayed for the test selected in the tree view section. Not all of the parameters listed apply to all tests.

Table 14: MHL Receiver Sink and MHL Dongle measurements

Parameter type	Parameter	Description	Applies to ...
Scope Setting	VTerm (V)	VTerm voltage of the fixture	All Tests
AWG Setting	Data Streams (KHz)	Data Stream of AWG Pattern	All Tests
	Diff Rise/Fall Time (ps)	Differential Rise/Fall Time of AWG Pattern	All Tests
	CM Rise/Fall Time (ps)	Common Mode Rise/Fall Time of AWG Pattern	All Tests
	Diff Swing Voltage (mV)	Differential Swing voltage of AWG Pattern	4.1.1.1, 4.1.1.3, 4.1.1.4 (CTS 1.X only), 5.1.1.1, 5.1.1.3, 5.1.1.4 (CTS 1.X only)
	CM Swing Voltage (mV)	Common Mode Swing voltage of AWG Pattern	4.1.1.1, 4.1.1.3, 4.1.1.4 (CTS 1.X only), 5.1.1.1, 5.1.1.3, 5.1.1.4 (CTS 1.X only)

See Also

- Measurement Parameter Descriptions: MHL Transmitter (see page 79)
- Configuring Tests: the Configure button or Configuration tab (see page 74)

Before You Click Start

Before Running a Test for the First time

Before you run a test for the first time, review these steps:

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

   After you install and launch TekExpress MHL, it creates the following folders on the oscilloscope:

   - `\Program Files\Tektronix\TekExpress\TekExpress MHL` (for WINXP scopes)
   - `\Program Files (x86)\Tektronix\TekExpress\TekExpress MHL` (for WIN7 scopes)
   - `\My Documents\My TekExpress\MHL`
   - `\My Documents\My TekExpress\MHL\Untitled Session`

     Every time you launch `TekExpress MHL.exe`, an `Untitled Session` folder is created in the `MHL` folder. The `Untitled Session` folder is automatically deleted when you exit the `MHL` application.

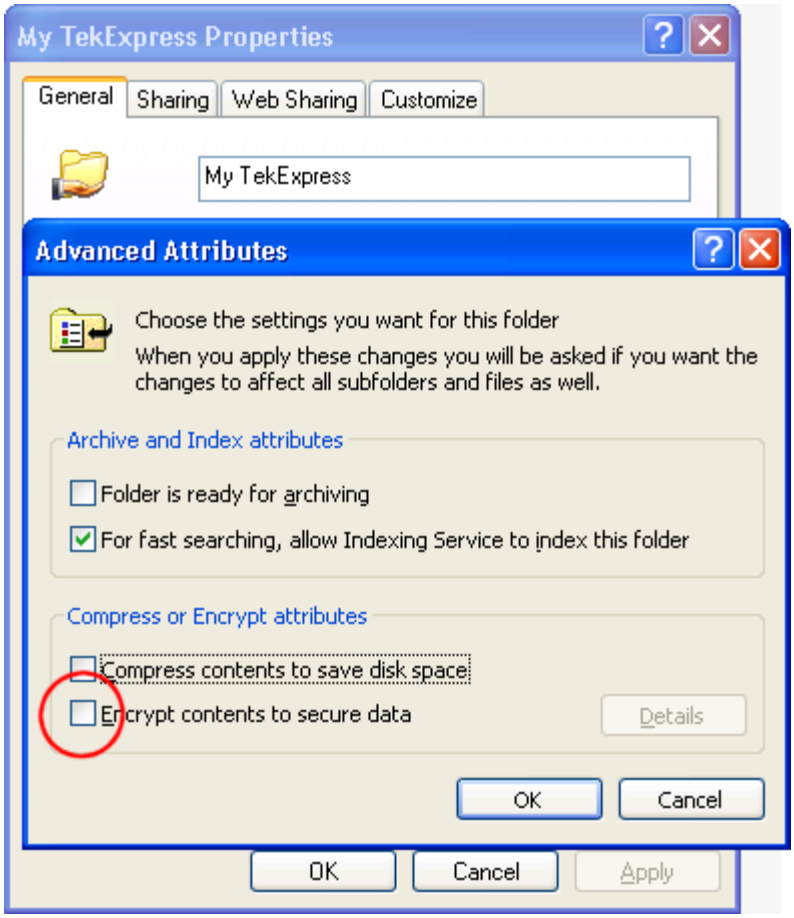
CAUTION. *Do not modify any of the session files or folders because this may result in loss of data or corrupted session files. Each session has multiple files associated with it. When you save a session, a .TekX file, and a folder named for the session that contains the associated files, is created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

2. Map the shared `My TekExpress` folder as X: (see page 191)

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

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

      Example:



Before running any test

1. Review the Pre Run Check List (see page 87).
2. Configure the Email notification options if you want the application to notify you by email when a test completes or produces an error. Access the email options either from the Options menu in the upper right corner, or from the **Preferences tab** on the Setup panel.
3. Select the Report Options (see page 97).

See Also

- PreRun Check List (see page 87)
- Configure Email Notification (see page 87)

- Running the Tests and Viewing their Progress (see page 89)

Pre-Run Check List

Do the following before you click Start to run a test. If this is the first time you are running a test on a setup, then refer also to the guidelines above.

1. Ensure that all the required instruments are properly warmed up (about 20 minutes).
2. Perform the Signal Path Compensation (SPC).
   a. On the oscilloscope main menu, select the **Utilities** menu.
   b. Select **Instrument Calibration**.
3. Deskew any cables.
4. Ensure that the application is able to find the DUT. If it cannot, perform a search for connected instruments (see page 37).

   To find the DUT:
   a. Launch the TekExpress MHL application.
   b. Select the **Setup** panel and then click the **DUT** tab. For **Suite**, select MHL Transmitter.
   c. Click the **Test Selection** tab. Select any test and then click **Configure**.
   d. In the Configuration section, click **Global Settings**.
   e. In the Instruments Detected section, click in the shaded area opposite Real Time Scope and make sure that the oscilloscope with the (GPIB8::1::INSTR) designation is in the drop-down list.

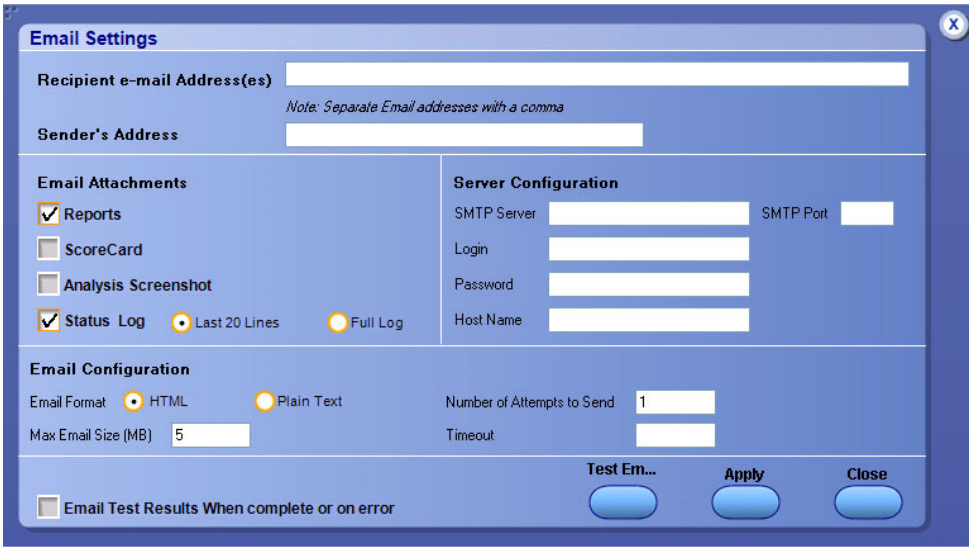
See also

- Before You Click Start (see page 85)
- Configure Email Notification (see page 87)
- Running the Tests and Viewing their Progress (see page 89)

Configure Email Notification

Set up these email settings if you want the application to notify you by email when a test completes or produces an error. Configure email from the Options menu.

1. From the **Options** menu in the upper right corner, select **Email Settings** to open the Email Settings dialog box, or click the **Preferences** tab on the Setup panel.



2. (Required) For **Recipient email Address(es)**, enter your email address. You can include multiple addresses as long as you 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 the email server used. For example: DPO72004C_B130099@yourcompany.com.
4. (Required) In the **Server Configuration** section, type the SMTP Server address of the Mail server configured at the client location, and the SMTP Port number, in the corresponding fields.

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

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

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

NOTE. *The ScoreCard and Analysys Screenshot options are not available in MHL.*

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

See also

- Before You Click Start (see page 85)
- PreRun Check List (see page 87)
- Running the Tests and Viewing their Progress (see page 89)

Running the Tests and Viewing their Progress: the Status Panel

Once you've configured the tests and gone through the Pre-run Check List (see page 87), from any screen, click the green **Start** button.

The application acquires and analyzes the data, then displays a report when the tests are complete. While the tests are running, other applications may display windows in the background. The TekScope application takes precedence over other applications, but you can switch to other applications using the Alt+Tab key combination. To keep the TekExpress MHL application on top, select Keep On Top from the Options menu in the upper right corner.

Viewing the Progress of Analysis

The Status panel (see page 91) displays a record of the test as it is executed. By default, the application switches to this panel after you click the Start button to run a test. You can choose from the following two views by selecting the named tab, even while a test is in progress:

Test Status view

The Test Status tab (see page 92) presents a collapsible table with information about each test as it is running. To collapse and expand the table rows, click the expand ()/() collapse button.

Table 15: Status tab table

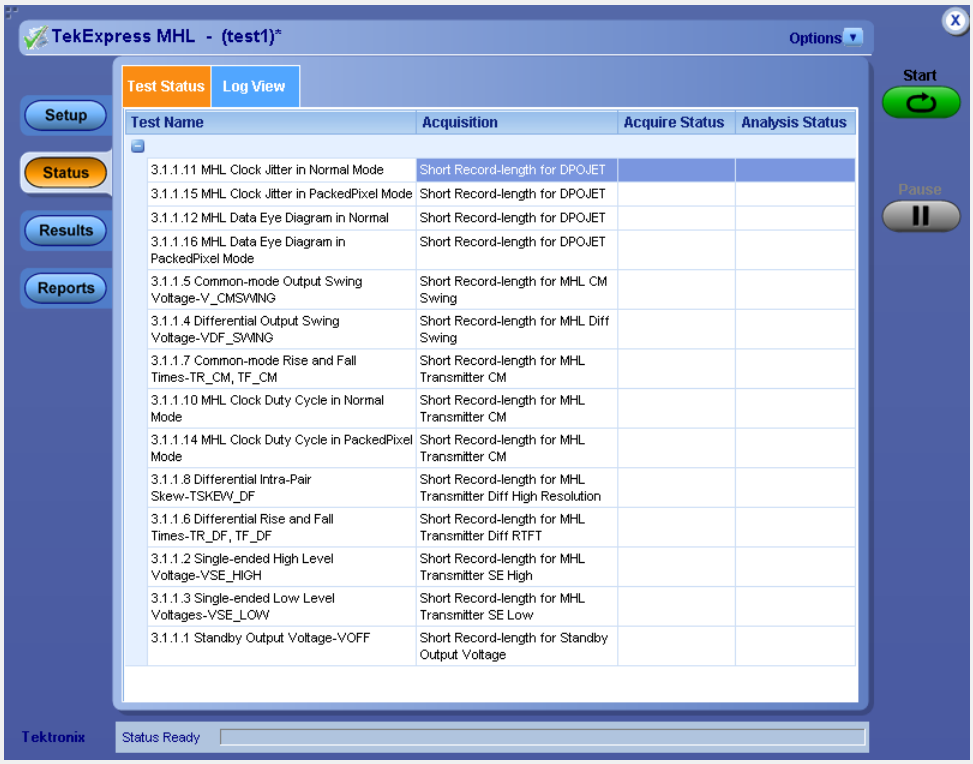
Column	Description
Test Name	Name of the test
Acquisition	Describes the type of data being acquired
Acquire Status	Progress of the acquisition: ■ To be started ■ Started Acquisition ■ Completed Acquisition ■ Pre-Recorded Mode
Analysis Status	Progress state of the analysis: ■ To be started ■ In Progress ■ Completed ■ Aborted

Log View

The Log View tab (see page 93) provides a list of actions that happen as the test executes. You can use this tab to review or troubleshoot tests.

Table 16: Log View options

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

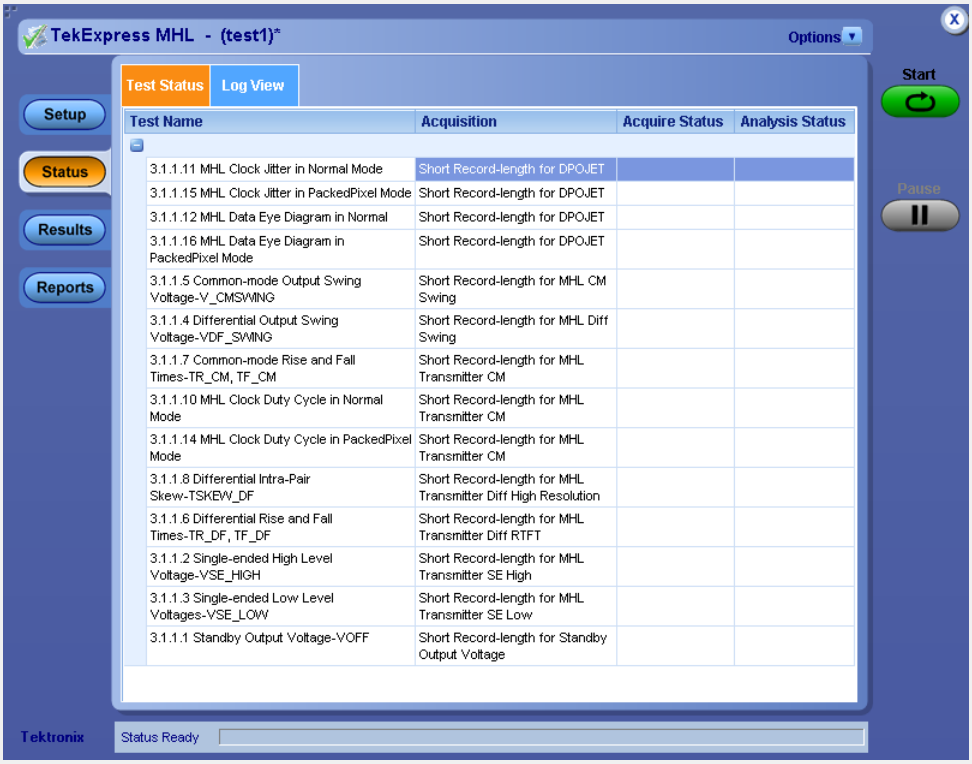


Status Panel

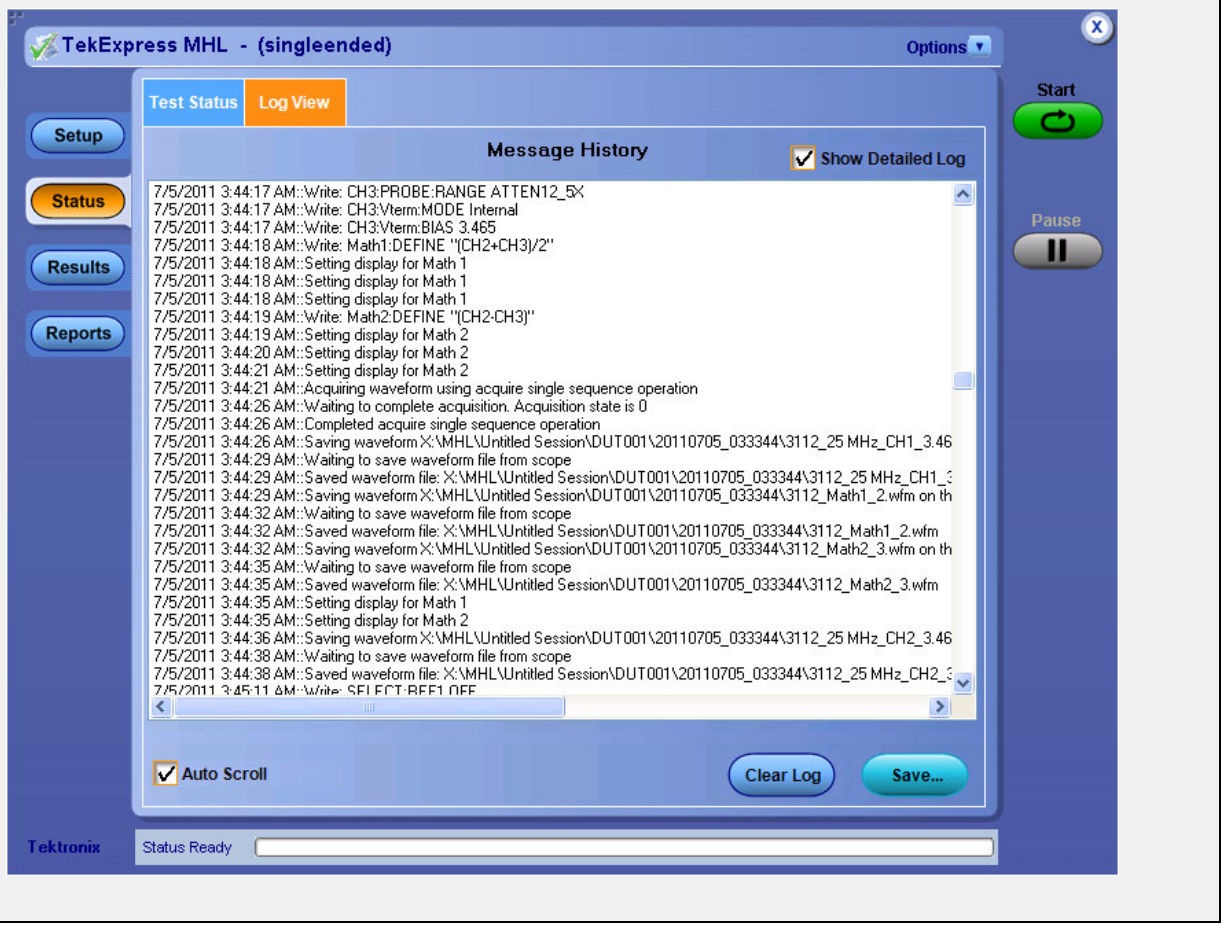
See also

- Before You Click Start (see page 85)
- PreRun Check List (see page 87)
- Configure Email Notification (see page 87)

Test Status Tab



Log View on Status Panel



Viewing Test Results: the Results Panel

When a test completes running, the application switches to the Results Panel (see page 96).

Set result viewing preferences from the Preferences (see page 96) menu in the upper right corner.

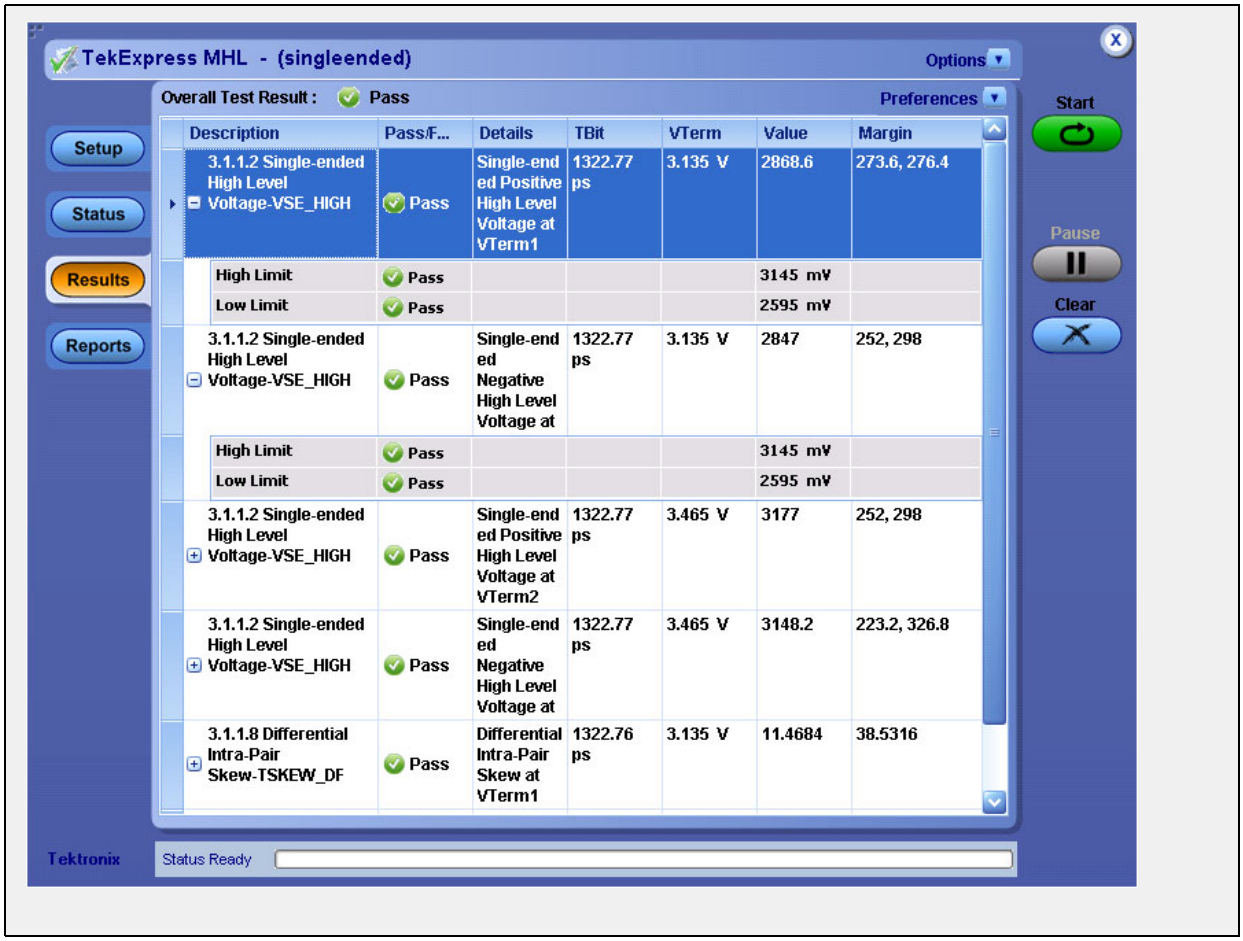
Each test result occupies a row in the Results table. By default, results are displayed in summary format, with the measurement details collapsed, and with the Pass/Fail column visible. You can change the display view.

- To expand all test rows listed, from the **Preferences** menu in the upper right corner, select **View Results Details**.
- To expand a collapsed test row, click the plus button (⊞) to the left of the test row.
- To collapse all expanded test rows, select **Preferences > View Results Summary**.
- To collapse a single expanded test row, click the minus button (⊟) to the left of the test row.
- 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 one 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** 

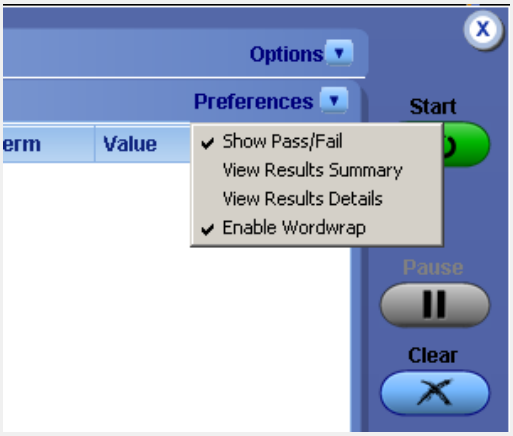
.

See Also

- Setting Up Tests the Setup Panel (see page 47)
- Configuring and Viewing Reports: the Reports Panel (see page 97)
- Running the Tests and Viewing their Progress: the Status Panel (see page 89)



Results Preferences



Configuring and Viewing Reports: the Reports Panel

Use the Reports panel (see page 99) to browse for, name, and save reports, select report viewing options, and to view reports.

Naming a Report

Use the Reports Panel to select the naming convention to use for the report. By default, the test report file is located in the session folder to which it belongs, and gets overwritten each time you run the test under the same device name if you don't change the test report name before running the test.

If you do not want your test results to be overwritten each time you run any test, always give the report a unique name, or select to have the 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.

Selecting Report Options

- Report Name

  Displays the default name and location where the report will be saved when generated.

  To change the report name or location, type over the current folder path and name and then save the test setup. 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\MHL\DUT001_group1.mht.

  To open an existing report, click **Browse**, locate the report file and then click **View**.

  NOTE. *You cannot change the file location using the Browse button.*

- Save As Type

  Saves a report in a file type different from the default. 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

  If the application finds a report with the same name as the one being generated, the application automatically increments the name of the report. For example: DUT001, DUT002, DUT003

- 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.

- Append Reports

  This option adds new report data to the end of an existing report of the same name. This option is deselected by default.

- View Report After Generating

  Automatically opens the report in your Web browser when the test completes. This option is selected by default.

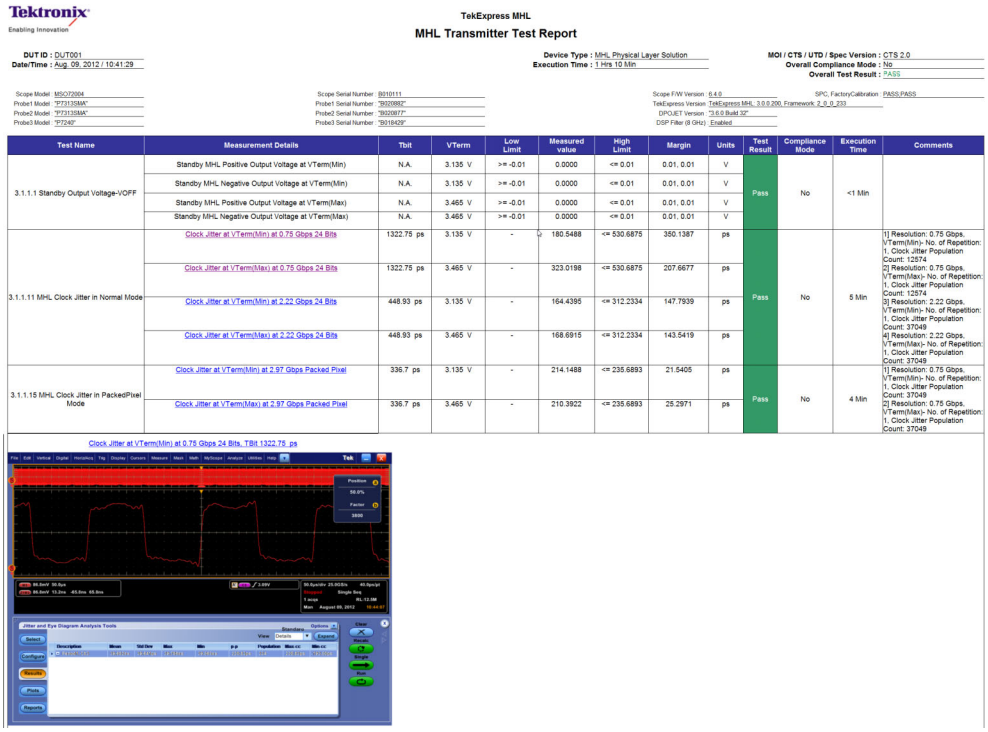
NOTE. *If you unchecked this option before running a test but now would like to view the report, then when analysis is complete, click the* ***Browse*** *button at the top of the Reports panel and navigate to the report file.*

Viewing a Report

You can view any report by locating and opening the report file, which ends in .mht unless you changed the file type before running the report.

The top of the report displays information about the instruments and probes used, the duration of the test, software versions, and some summary test information. Below that is a table that shows the test name, measurement details, various measurements, test result (pass/fail), Compliance Mode status (Yes/No), and analysis time. Additional test parameter information that does not fall into the other columns is put in the Comments column.

Report example



Tektronix
Enabling Innovation

TekExpress MHL

MHL Transmitter Test Report

DUT ID : DUT001
Date/Time : Aug. 09, 2012 / 10:41:29

Device Type : MHL Physical Layer Solution
Execution Time : 1 Hrs 10 Min

MOI / CTS / UTD / Spec Version : CTS 2.0
Overall Compliance Mode : No
Overall Test Result : PASS

Scope Model: MSO72004
Probe1 Model: "P73135MA"
Probe2 Model: "P73135MA"
Probe3 Model: "P7240"

Scope Serial Number: B010111
Probe1 Serial Number: "B020992"
Probe2 Serial Number: "B020977"
Probe3 Serial Number: "B018429"

Scope F/W Version : 6.4.0
TekExpress Version : TekExpress MHL 3.0.0.200, Framework 2_0_0_233
DPOJET Version : "3.6.0 Build 10"
DSP Filter (8 GHz) : Enabled

SPC, FactoryCalibration : PASS,PASS

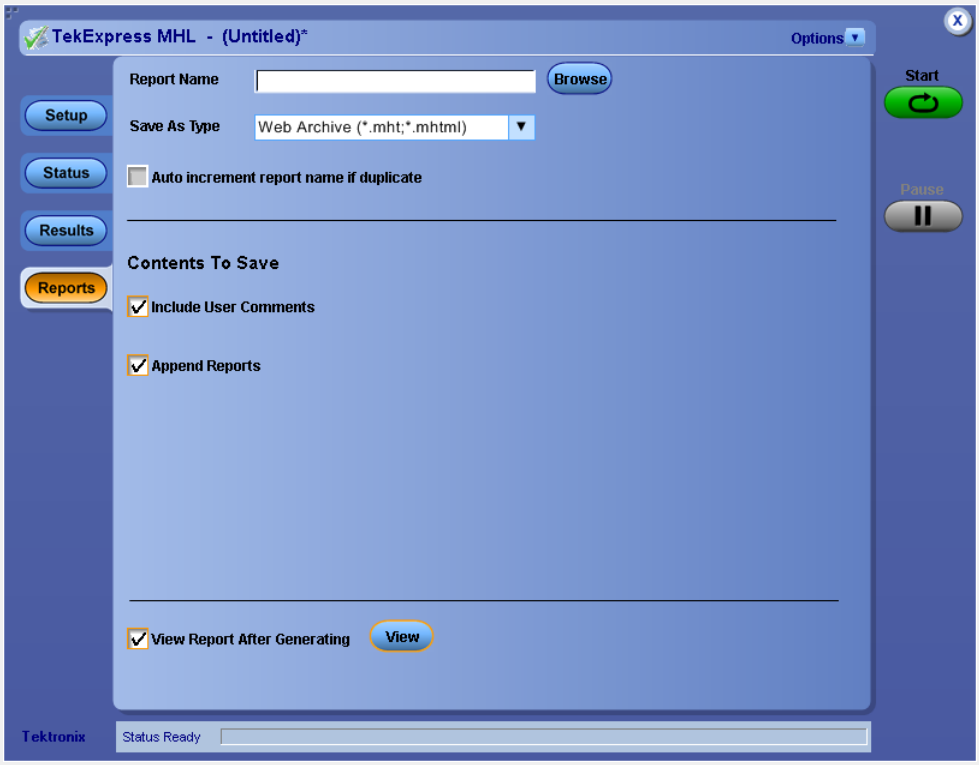
Test Name	Measurement Details	Tbit	VTerm	Low Limit	Measured value	High Limit	Margin	Units	Test Result	Compliance Mode	Execution Time	Comments
3.1.1.1 Standby Output Voltage-VOFF	Standby MHL Positive Output Voltage at VTerm(Min)	N.A.	3.135 V	>= -0.01	0.0000	<= 0.01	0.01, 0.01	V	Pass	No	<1 Min	
	Standby MHL Negative Output Voltage at VTerm(Min)	N.A.	3.135 V	>= -0.01	0.0000	<= 0.01	0.01, 0.01	V				
	Standby MHL Positive Output Voltage at VTerm(Max)	N.A.	3.465 V	>= -0.01	0.0000	<= 0.01	0.01, 0.01	V				
	Standby MHL Negative Output Voltage at VTerm(Max)	N.A.	3.465 V	>= -0.01	0.0000	<= 0.01	0.01, 0.01	V				
3.1.1.11 MHL Clock Jitter in Normal Mode	Clock Jitter at VTerm(Min) at 0.75 Gbps 24 Bits	1322.75 ps	3.135 V	-	180.5488	<= 530.6875	350.1387	ps	Pass	No	5 Min	1] Resolution: 0.75 Gbps, VTerm(Min)- No. of Repetition: 1, Clock Jitter Population Count: 12574 2] Resolution: 0.75 Gbps, VTerm(Max)- No. of Repetition: 1, Clock Jitter Population Count: 12574 3] Resolution: 2.22 Gbps, VTerm(Min)- No. of Repetition: 1, Clock Jitter Population Count: 37049 4] Resolution: 2.22 Gbps, VTerm(Max)- No. of Repetition: 1, Clock Jitter Population Count: 37049
	Clock Jitter at VTerm(Max) at 0.75 Gbps 24 Bits	1322.75 ps	3.465 V	-	323.0198	<= 530.6875	207.6677	ps				
	Clock Jitter at VTerm(Min) at 2.22 Gbps 24 Bits	448.93 ps	3.135 V	-	164.4395	<= 312.2334	147.7939	ps				
	Clock Jitter at VTerm(Max) at 2.22 Gbps 24 Bits	448.93 ps	3.465 V	-	168.6915	<= 312.2334	143.5419	ps				
3.1.1.15 MHL Clock Jitter in PackedPixel Mode	Clock Jitter at VTerm(Min) at 2.97 Gbps Packed Pixel	336.7 ps	3.135 V	-	214.1488	<= 235.6893	21.5405	ps	Pass	No	4 Min	1] Resolution: 0.75 Gbps, VTerm(Min)- No. of Repetition: 1, Clock Jitter Population Count: 37049 2] Resolution: 0.75 Gbps, VTerm(Max)- No. of Repetition: 1, Clock Jitter Population Count: 37049
	Clock Jitter at VTerm(Max) at 2.97 Gbps Packed Pixel	336.7 ps	3.465 V	-	210.3922	<= 235.6893	25.2971	ps				

Clock Jitter at VTerm(Min) at 0.75 Gbps 24 Bits, TBit 1322.75 ps

See Also

- Setting Up Tests: the Setup Panel (see page 47)
- Running the Tests and Viewing their Progress: the Status Panel (see page 89)
- Viewing Test Results: the Results Panel (see page 95)
- About Saving and Recalling Test Setups (see page 101)

Reports Panel



About Saving and Recalling Test Setups

TekExpress MHL opens with the default test setup selected. Any time you want to create a new test setup, you can select the default test setup to clear the previous test setup selections and take the settings back to their defaults.

You can run a test before or after saving a setup. When you save a setup, the selected oscilloscope, general parameters, acquisition parameters, measurement limits, prerecorded waveform files (if applicable), test, and other configuration settings are all saved under the setup name. You can open a setup and click **Start** without having to do any other setting up except ensuring that the oscilloscope is detected and ready. For details, see Before You Click Start (see page 85).

See Also

- Save a Test Setup (see page 101)
- Recall a Saved Test Setup (see page 102)
- Create a New Test Setup Based on an Existing One (see page 102)
- Delete a Test Setup (see page 103)
- Setting Up Tests: the Setup Panel (see page 47)

Saving a Test Setup

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

1. From the **Options** menu, select **Default Test Setup**.
2. Select the desired options in the Setup panel. (see page 47)
3. Select the desired report options (see page 97).
4. If desired, run the test to ensure that it captures the information you want. If it does not, edit the parameters.
5. From the **Options** menu, select **Save Test Setup**.
6. Name the test and then click **Save**.

See Also

- About Test Setups (see page 101)
- Recall a Saved Test Setup (see page 102)

- Create a New Test Setup Based on an Existing One (see page 102)
- Delete a Test Setup (see page 103)

Recalling a Saved Test Setup

These instructions are for recalling saved test setups.

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

- About Test Setups (see page 101)
- Save a Test Setup (see page 101)
- Create a New Test Setup Based on an Existing One (see page 102)
- Delete a Test Setup (see page 103)

Creating a New Test Setup Based on an Existing One

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

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

- About Test Setups (see page 101)
- Save a Test Setup (see page 101)
- Recall a Saved Test Setup (see page 102)
- Delete a Test Setup (see page 103)

Deleting a Test Setup

If you no longer need a test setup, delete it from the test setup list in the Options menu using these instructions.

Each saved test setup consists of two main parts, the test setup file and the test setup folder, both named for the test session.

1. Make sure the setup you want to delete is not currently selected in TekExpress MHL.
2. Navigate to the MHL folder where test setup files are stored. For example, `X:\MHL\(test setup name)`.
3. Locate the test setup file and then delete it. This removes the setup from the list in the **Options** menu.
4. Locate the test setup folder. If you want to keep any of the session files, move them out of the test setup folder and then delete the test setup folder.

- About Test Setups (see page 101)
- Save a Test Setup (see page 101)
- Recall a Saved Test Setup (see page 102)
- Create a New Test Setup Based on an Existing One (see page 102)

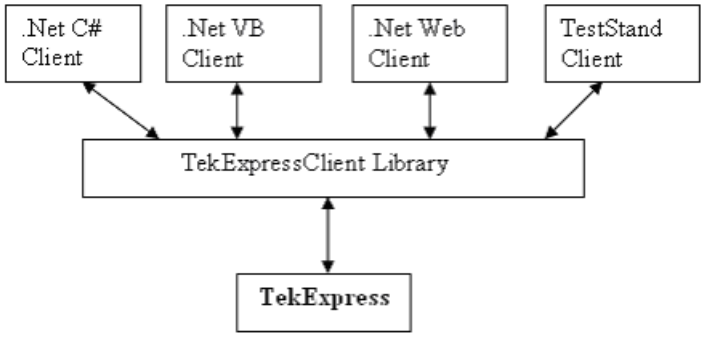
About the Programmatic Interface

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

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

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

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



See Also

- Requirements for Developing TekExpress Client (see page 105)
- Remote Proxy Object (see page 107)
- Client Proxy Object (see page 107)

Requirements for Developing TekExpress Client

While developing 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 are the steps that a client needs to follow to use `TekExpressClient.dll` to programmatically control the server:

A client UI must be developed to access the interfaces exposed through the server. This client needs to load `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 must do the following to run a remote operation:

1. The client needs to provide the IP address of the PC at which the server is running in order to connect to the server.
2. The client needs to lock 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. Note that this does not lock the UI.

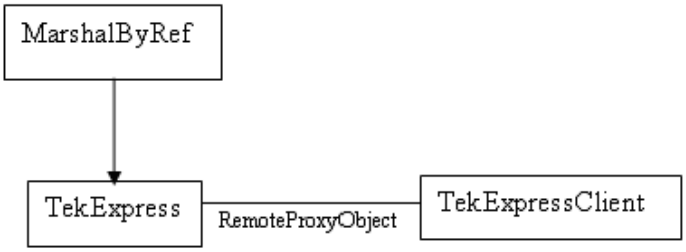
   If any other client tries to access a server that is locked, it will get a notification that the server is locked by another client.
3. When the client has connected to and locked the server, the client can access any of the programmatic controls to run the remote automations.
4. After the client operations are completed, the server needs to be unlocked by the client.

See Also

- About MHL Application Commands (see page 116)

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.

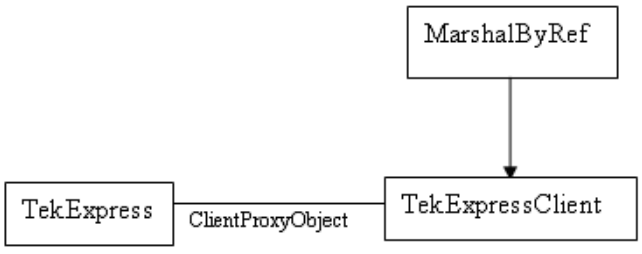
For example,

//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.

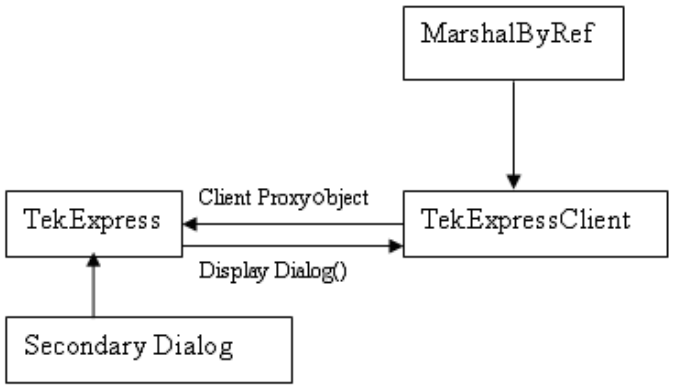
Examples

```
clientObject.clientIntf.DisplayDialog(caption, msg,iconType, btnType);

clientObject.clientIntf.TransferBytes(buffer, read, fileLength);
```

For more information, click the topic links listed below.

Secondary Dialog Message Handling



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

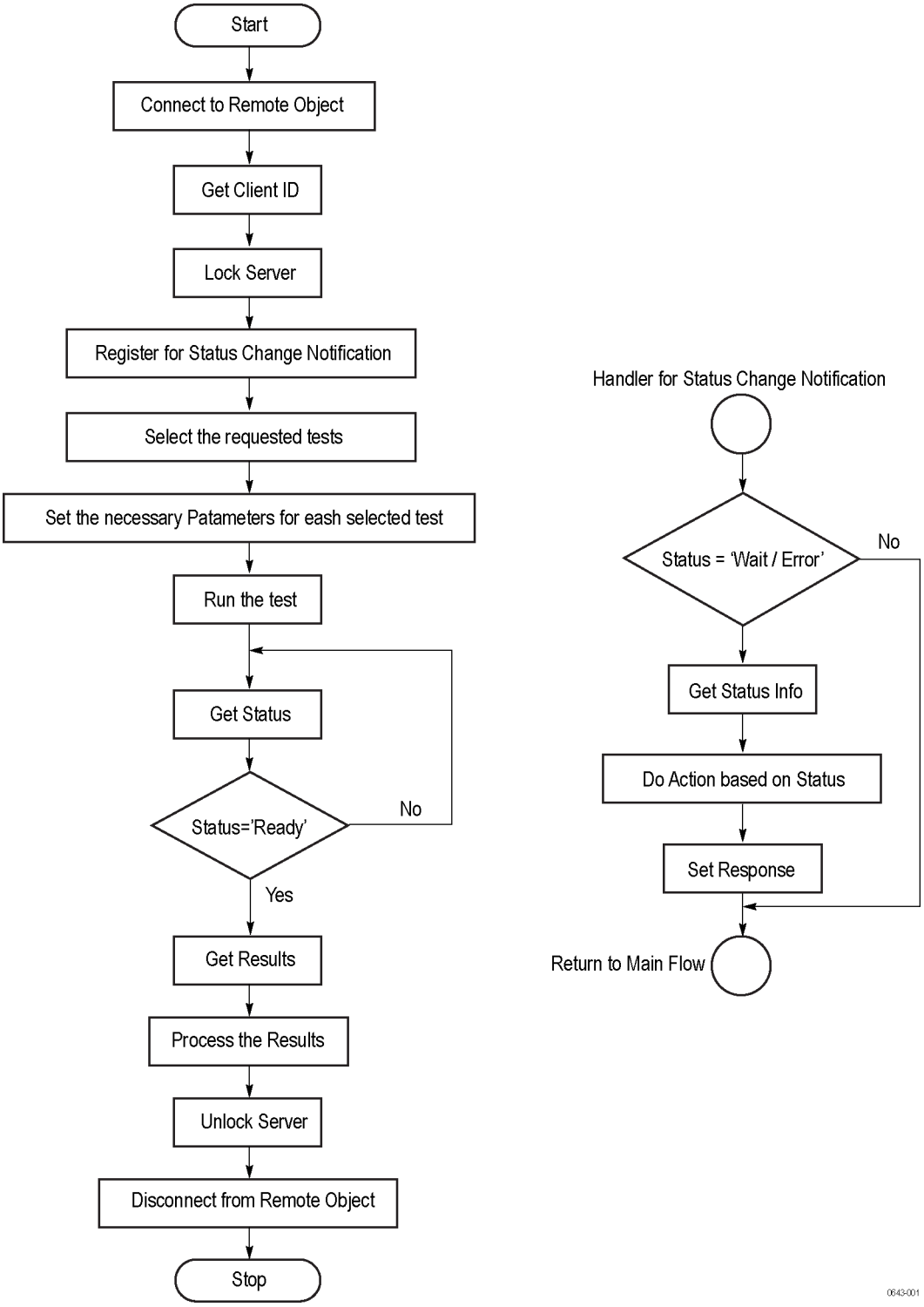
In the secondary dialog library, the assembly that is calling for the dialog box to be displayed is checked and if a remote connection is detected, the messages are directed to the remote end.

File Transfer Events

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

Client Programmatic Interface Example

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


Start
Connect to Remote Object
Get Client ID
Lock Server
Register for Status Change Notification
Select the requested tests
Set the necessary Patameters for eash selected test
Run the test
Get Status
Status='Ready'
No
Yes
Get Results
Process the Results
Unlock Server
Disconnect from Remote Object
Stop
Handler for Status Change Notification
Status = 'Wait / Error'
No
Get Status Info
Do Action based on Status
Set Response
Return to Main Flow
0643-001

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

NOTE. *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.

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 that you want 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 when 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 you complete all the tasks.
12. Disconnect from the remote object.

Handler of Status Change Notification

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

See Also

- About MHL Application Commands (see page 116)
- Program Example (see page 112)

Program Example

This program example shows how to communicate between a PC and TekExpress MHL remotely, using typical steps.

For detailed information about each command, see MHL Application Commands (see page 116) section.

1. **Start the application.**
2. **Connect through an IP address.**

```
{'Set String Details
string devicename = ”MHL Physical Layer Solution”
string suitename = ”MHL Transmitter”
m_Client.Connect("localhost")'True or False
clientID = m_Client.getClientID }
```

3. **Lock the server.**

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

4. **Disable the Popups.**

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

5. **Set the Dut ID.**

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

6. **Set Pixel Mode.**

```
m_Client.SetGeneralParameter(clientID, devicename, suitename,
string.Empty,"Pixel Mode$24 Bits");
```

7. **Set Resolutions.**

```
m_Client.SetGeneralParameter(clientID, devicename, suitename,
string.Empty,"24Bits Lowest Resolution (MHz)$25");
m_Client.SetGeneralParameter(clientID, devicename, suitename,
string.Empty,"24Bits Highest Resolution (MHz)$74.25");
m_Client.SetGeneralParameter(clientID, devicename, suitename,
string.Empty,"24Bits Number of Other Resolution (Count)$0");
```

8. **Select Voltage Termination Method.**

```
m_Client.SetGeneralParameter(clientID, devicename, suitename,
string.Empty,"Voltage termination$Internal");
```

9. **Set Voltage.**

```
m_Client.SetGeneralParameter(clientID, devicename, suitename,
string.Empty,"VTerm1 (V)$3.1");

m_Client.SetGeneralParameter(clientID, devicename, suitename,
string.Empty,"VTerm2 (V)$3.5");
```

10. **Select a measurement.**

```
m_Client.SelectTest(clientID, devicename, suitename, "3.1.1.12 MHL Data Eye
Diagram", True)
```

11. **Select channels.**

```
m_Client.SetGeneralParameter(clientID, devicename, suitename, String.Empty,
"P7313SMA Diff mode / MHL Positive Connected to$CH1");

m_Client.SetGeneralParameter(clientID, devicename, suitename, String.Empty,
"P7313SMA MHL Negative Connected to$CH2");

m_Client.SetGeneralParameter(clientID, devicename, suitename,
String.Empty,"P7240 Common Mode Connected to$CH3");
```

12. **Configure the selected measurement(Acquire Parameters).**

```
m_client.SetAcquireParameter(clientID, devicename, suitename,"3.1.1.12 MHL
Data Eye Diagram","Short Record-length for DPOJET$Sample Rate (GS/s)$25")

m_client.SetAcquireParameter(clientID, devicename, suitename,"3.1.1.12 MHL
Data Eye Diagram","Short Record-length for DPOJET$Record Length (M)$12.5")

m_client.SetAcquireParameter(clientID, devicename, suitename,"3.1.1.12 MHL
Data Eye Diagram","Short Record-length for DPOJET$Edge trigger level
(%)$50")

m_client.SetAcquireParameter(clientID, devicename, suitename,"3.1.1.12 MHL
Data Eye Diagram","Short Record-length for DPOJET$Probe attenuation
(X)$12.5")

m_client.SetAcquireParameter(clientID, devicename, suitename,"3.1.1.12 MHL
Data Eye Diagram","Short Record-length for DPOJET$Diff Probe Vertical
Position (Div)$0")

m_client.SetAcquireParameter(clientID, devicename, suitename,"3.1.1.12 MHL
Data Eye Diagram","Short Record-length for DPOJET$Diff Probe Vertical Scale
(mV/Div)$125")

m_client.SetAcquireParameter(clientID, devicename, suitename,"3.1.1.12 MHL
Data Eye Diagram","Short Record-length for DPOJET$CM Probe Vertical
Position (Div)$0")
```

```
m_client.SetAcquireParameter(clientID, devicename, suitename,"3.1.1.12 MHL
Data Eye Diagram","Short Record-length for DPOJET$CM Probe Vertical Scale
(mV/Div)$80")
```

13. **Configure the selected measurement(Analyze Parameters).**

```
m_client.SetAnalyzeParameter(clientID, devicename, suitename,"3.1.1.12 MHL
Data Eye Diagram","Reference levels$Absolute")

m_client.SetAnalyzeParameter(clientID, devicename, suitename,"3.1.1.12 MHL
Data Eye Diagram","Reference Level - Mid Level (%)$2")

m_client.SetAnalyzeParameter(clientID, devicename, suitename,"3.1.1.12 MHL
Data Eye Diagram","Reference Level - Hysteresis (%)$1")

m_client.SetAnalyzeParameter(clientID, devicename, suitename,"3.1.1.12 MHL
Data Eye Diagram","PLL model type$2")

m_client.SetAnalyzeParameter(clientID, devicename, suitename,"3.1.1.12 MHL
Data Eye Diagram","Clock recovery method$PLL-Custom BW")

m_client.SetAnalyzeParameter(clientID, devicename, suitename,"3.1.1.12 MHL
Data Eye Diagram","Clock Multiplier$15")

m_client.SetAnalyzeParameter(clientID, devicename, suitename,"3.1.1.12 MHL
Data Eye Diagram","Loop bandwidth (MHz)$4")

m_client.SetAnalyzeParameter(clientID, devicename, suitename,"3.1.1.12 MHL
Data Eye Diagram","Clock Edge$RISE")
```

14. **Run with set configurations.**

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

15. **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, WaitingMsbBxMes-
sage, WaitingMsbBxButtontexts)

'Send the Response

mClient.SendResponse(clientID, WaitingMsbBxCaption, WaitingMsbBxMessage,
WaitingMsbBxResponse)

End Select

Loop Until status = "Ready"
```

16. **After the test is complete:**

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

17. **Unlock the server.**

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

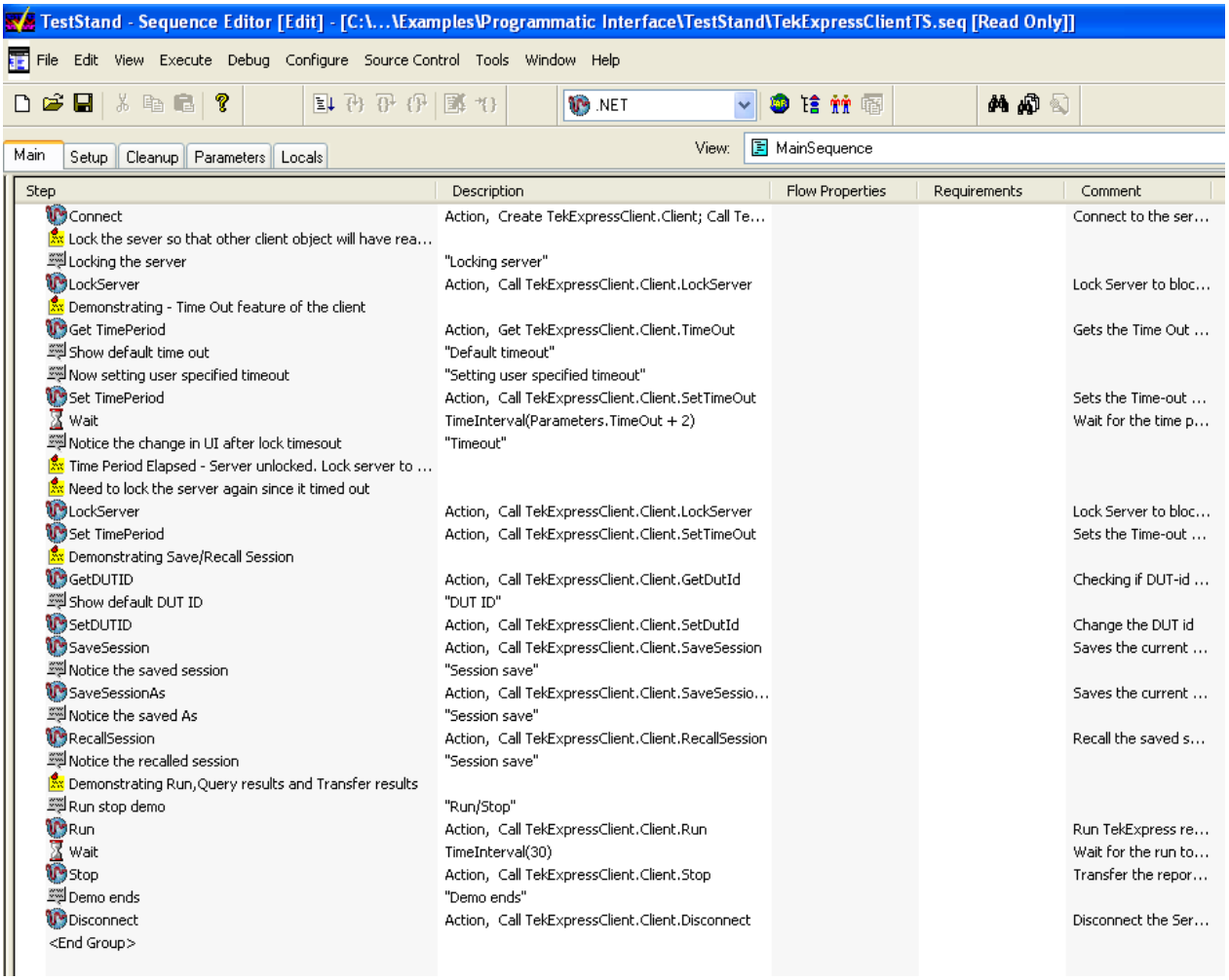
18. **Disconnect from the server.**

```
m_Client.Disconnect()
```

19. **Exit the application.**

NI TestStand Client Example

The following is an example for NI TestStand Client available in the path: `C:\Program Files\Tektronix\TekExpress\TekExpress MHL\Examples\Programmatic Interface\TestStand`



About TekExpress MHL Application Commands

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

Connect through an IP address (see page 122)

Lock the server (see page 123)

Disable the popups (see page 124)

Set or get the DUT ID (see page 125)

Set the configuration parameters for a suite or measurement (see page 126)

Query the configuration parameters for a suite or measurement (see page 129)

Select a measurement (see page 131)

Select a suite (see page 138)

Set Pixel Mode (see page 138)

Set Resolution (see page 139)

Set Termination Source Method (see page 141)

Set Termination Voltage (see page 142)

Run with set configurations or stop the run operation (see page 143)

Handle Error Codes (see page 144)

Get or set the timeout value (see page 145)

Wait for the test to complete (see page 145)

After the test is complete (see page 148)

Save, recall, or check if a session is saved (see page 151)

Unlock the server (see page 152)

Disconnect from server (see page 152)

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 has 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 to. This is required to establish the connection between the server and the client.

out string clientID

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

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

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

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

string dutId

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

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

string device

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

string suite

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

string test

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

string parameterString

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

int rowNr

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

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

out string[] status

Name	Type	Direction	Description
status	string array	OUT	The list of status messages generated during 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. Else, 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 automatically unlocked.*

out string caption

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

out string message

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

out string[] buttonTexts

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

string response

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

Connect Through an IP Address

Command name	Parameters	Description	Return value	Example
Connect()	string ipAddress (see page 118) out string clientID (see page 118)	This method connects the client to the server. Note (see page 118) The client provides the IP address to connect to the server. The server provides a unique client identification number when connected to it.	Return value is either True or False.	m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as boolean returnval = m_Client.Connect(ipaddress,m_clientID)

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

NOTE. *This method does not lock the UI, but you need this method to set the value that gives the status of the operation after it has been performed.*

Command name	Parameters	Description	Return value	Example
LockSession()	out string clientID (see page 118)	This method locks the server. Note (see page 119) The client must call this method before running any of the remote automations. The server can be locked by only one client.	String value that gives the status of the operation after it has been performed. The return value is "Session Locked..." on success.	m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string returnval = m_Client.LockServer(clientID)

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

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

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

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

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

Disable the Popups

Command name	Parameters	Description	Return value	Example
SetVerboseMode()	string clientID (see page 118) bool _verbose (see page 121)	This method sets the verbose mode to either True or False. When the value is set to True, any of the message boxes appearing during the application run will be routed to the client machine that is controlling TekExpress. When the value is set to False, then all the message boxes are shown on the server machine.	String that displays 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.SetVerbose-Mode(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 118) string dutName (see page 117)	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 118)
GetDutId()	string clientID (see page 118) string dutId (see page 118)	This method gets the DUT ID of the current set up.	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 118) string device (see page 119) string suite (see page 119) string test (see page 119) string parameterString (see page 119)	This method sets the general parameters.	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 returnval = m_Client.SetGeneralParameter(clientID, devicename, suitename, string.Empty, parameterstring); SetGeneralParameters Examples (see page 128)
SetAnalyzeParameter()	string clientID (see page 118) string device (see page 119) string suite (see page 119) string test (see page 119) string parameterString (see page 119)	This method sets the parameter values in the Ref Levels and Clock Settings tabs in the test configuration section.	String that displays the status of the operation after it has been performed. The return value is "" (an empty String) on success.	mClient = new Client() //m_Client is a reference to the Client class in the Client DLL returnval as string Select Analyze parameter: returnval = mClient.SetAnalyzeParameter(clientID, devicename, suitename, test, parameterstring) SetAnalyzeParameter Examples (see page 127)
SetAcquireParameter()	string clientID (see page 118) string device (see page 119) string suite (see page 119) string test (see page 119) string parameterString (see page 119)	This method sets the parameter values in the Vertical Setup and the Scope Settings tabs in the test configuration section.	String that displays the status of the operation after it has been performed. The return value is " (an empty string) on success.	mClient = new Client() //m_Client is a reference to the Client class in the Client DLL returnval as string Set Acquire Parameter: returnval = mClient.SetAcquireParameter(clientID, devicename, suitename, test, parameterstring) SetAcquireParameter Example (see page 128)

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...".

SetAnalyzeParameter Examples

This example uses MHL test 3.1.1.12 MHL Data Eye Diagram.

Parameter	Example
Reference Level	returnval =mClient.SetAnalyzeParameter(clientID, devicename, suitename,"3.1.1.12 MHL Data Eye Diagram","Reference levels$Absolute")
Mid Level	returnval =mClient.SetAnalyzeParameter(clientID, devicename, suitename,"3.1.1.12 MHL Data Eye Diagram","Reference Level - Mid Level (%)$2")
Hysteresis	returnval =mClient.SetAnalyzeParameter(clientID, devicename, suitename,"3.1.1.12 MHL Data Eye Diagram","Reference Level - Hysteresis (%)$1")
PLL Model Type	returnval =mClient.SetAnalyzeParameter(clientID, devicename, suitename,"3.1.1.12 MHL Data Eye Diagram","PLL model type$2")
Clock Recovery Method	returnval =mClient.SetAnalyzeParameter(clientID, devicename, suitename,"3.1.1.12 MHL Data Eye Diagram","Clock recovery method$PLL-Custom BW")
Clock Multiplier	returnval =mClient.SetAnalyzeParameter(clientID, devicename, suitename,"3.1.1.12 MHL Data Eye Diagram","Clock Multiplier$15")
Loop bandwidth	returnval =mClient.SetAnalyzeParameter(clientID, devicename, suitename,"3.1.1.12 MHL Data Eye Diagram","Loop bandwidth (MHz)$4")
Clock Edge	returnval =mClient.SetAnalyzeParameter(clientID, devicename, suitename,"3.1.1.12 MHL Data Eye Diagram","Clock Edge$RISE")

SetAcquireParameter Examples

This configuration example uses MHL test 3.1.1.12 MHL Data Eye Diagram.

Parameter	Example
Sample Rate (GS/s)	returnval = mClient.SetAcquireParameter(clientID, devicename, suitename,"3.1.1.12 MHL Data Eye Diagram","Short Record-length for DPOJET$Sample Rate (GS/s)$25")
Record Length	returnval = mClient.SetAcquireParameter(clientID, devicename, suitename,"3.1.1.12 MHL Data Eye Diagram","Short Record-length for DPOJET$Record Length (M)$12.5")
Edge Trigger	returnval = mClient.SetAcquireParameter(clientID, devicename, suitename,"3.1.1.12 MHL Data Eye Diagram","Short Record-length for DPOJET$Edge trigger level (%)$50")
Probe Attenuation	returnval = mClient.SetAcquireParameter(clientID, devicename, suitename,"3.1.1.12 MHL Data Eye Diagram","Short Record-length for DPOJET$Probe attenuation (X)$12.5")
Diff Vertical Position	returnval = mClient.SetAcquireParameter(clientID, devicename, suitename,"3.1.1.12 MHL Data Eye Diagram","Short Record-length for DPOJET$Diff Probe Vertical Position (Div)$0")
Diff Vertical Scale	returnval = mClient.SetAcquireParameter(clientID, devicename, suitename,"3.1.1.12 MHL Data Eye Diagram","Short Record-length for DPOJET$Diff Probe Vertical Scale (mV/Div)$125")
CM Vertical Position	returnval = mClient.SetAcquireParameter(clientID, devicename, suitename,"3.1.1.12 MHL Data Eye Diagram","Short Record-length for DPOJET$CM Probe Vertical Position (Div)$0")
CM Vertical Scale	returnval = mClient.SetAcquireParameter(clientID, devicename, suitename,"3.1.1.12 MHL Data Eye Diagram","Short Record-length for DPOJET$CM Probe Vertical Scale (mV/Div)$80")

SetGeneralParameter Examples

```
returnval = mClient.SetGeneralParameter(clientID,devicename, suitename,
String.Empty, "P7313SMA Diff mode / MHL Positive Connected to$CH1");

returnval = mClient.SetGeneralParameter(clientID, devicename, suitename,
String.Empty, "P7313SMA MHL Negative Connected to$CH2");

returnval = mClient.SetGeneralParameter(clientID, devicename, suitename,
String.Empty,"P7240 Common Mode Connected to$CH3");
```

Query the Configuration Parameters for a Suite or Measurement

Command name	Parameters	Description	Return value	Example
GetGeneralParameter()	string clientID (see page 118) string device (see page 119) string suite (see page 119) string test (see page 119) string parameterString (see page 119)	This method gets the general configuration parameters for a given suite or measurement.	The return value is the general configuration parameter for a given suite or measurement that is set.	m_Client = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string
GetAnalyzeParameter()	string clientID (see page 118) string device (see page 119) string suite (see page 119) string test (see page 119) string parameterString (see page 119)	This method queries the parameter values in the Ref Level and Clock Settings tabs in the test configuration section.	The return value is the configuration parameter for a given suite or measurement.	mClient = new Client() //m_Client is a reference to the Client class in the Client DLL returnval as string Get Analyze parameter: returnval = mClient.GetAnalyzeParameter(clientID, devicename, suitename, test, parameterstring) GetAnalyzeParameter Examples (see page 131)
GetAcquireParameter()	string clientID (see page 118) string device (see page 119) string suite (see page 119) string test (see page 119) string parameterString (see page 119)	This method queries the parameter values in the Vertical Setup and Scope Settings tabs in the test configuration section.	The return value is the configuration parameter for a given suite or measurement.	mClient = new Client() //m_Client is a reference to the Client class in the Client DLL returnval as string Get Acquire Parameter: returnval = mClient.GetAcquireParameter(clientID, devicename, suitename, test, parameterstring) GetAcquireParameter Examples (see page 130)

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...".

GetAcquireParameter Examples

This example uses MHL test 3.1.1.12 MHL Data Eye Diagram.

Parameter	Example
Sample Rate (GS/s)	returnval = mClient.GetAcquireParameter(clientID, devicename, suitename,"3.1.1.12 MHL Data Eye Diagram","Short Record-length for DPOJET$Sample Rate (GS/s)")
Record Length	returnval = mClient.GetAcquireParameter(clientID, devicename, suitename,"3.1.1.12 MHL Data Eye Diagram","Short Record-length for DPOJET$Record Length (M)")
Edge Trigger	returnval = mClient.GetAcquireParameter(clientID, devicename, suitename,"3.1.1.12 MHL Data Eye Diagram","Short Record-length for DPOJET$Edge trigger level (%)")
Probe Attenuation	returnval = mClient.GetAcquireParameter(clientID, devicename, suitename,"3.1.1.12 MHL Data Eye Diagram","Short Record-length for DPOJET$Probe attenuation (X)")
Diff Prove Vertical Position	returnval = mClient.GetAcquireParameter(clientID, devicename, suitename,"3.1.1.12 MHL Data Eye Diagram","Short Record-length for DPOJET$Diff Probe Vertical Position (Div)")
Diff Probe Vertical Scale	returnval = mClient.GetAcquireParameter(clientID, devicename, suitename,"3.1.1.12 MHL Data Eye Diagram","Short Record-length for DPOJET$Diff Probe Vertical Scale (mV/Div)")
CM Probe Vertical Position	returnval = mClient.GetAcquireParameter(clientID, devicename, suitename,"3.1.1.12 MHL Data Eye Diagram","Short Record-length for DPOJET$CM Probe Vertical Position (Div)")
CM Probe Vertical Scale	returnval = mClient.GetAquireParameter(clientID, devicename, suitename,"3.1.1.12 MHL Data Eye Diagram","Short Record-length for DPOJET$CM Probe Vertical Scale (mV/Div)")

GetAnalyzeParameter Examples

This example uses MHL test 3.1.1.12 MHL Data Eye Diagram.

Parameter	Example
Reference Level	returnval =mClient.GetAnalyzeParameter(clientID, devicename, suitename,"3.1.1.12 MHL Data Eye Diagram","Reference levels")
Mid Level	returnval =mClient.GetAnalyzeParameter(clientID, devicename, suitename,"3.1.1.12 MHL Data Eye Diagram","Reference Level - Mid Level (%)")
Hysteresis	returnval =mClient.GetAnalyzeParameter(clientID, devicename, suitename,"3.1.1.12 MHL Data Eye Diagram","Reference Level - Hysteresis (%)")
PLL Model Type	returnval =mClient.GetAnalyzeParameter(clientID, devicename, suitename,"3.1.1.12 MHL Data Eye Diagram","PLL model type")
Clock Recovery Method	returnval =mClient.GetAnalyzeParameter(clientID, devicename, suitename,"3.1.1.12 MHL Data Eye Diagram","Clock recovery method")
Clock Multiplier	returnval =mClient.GetAnalyzeParameter(clientID, devicename, suitename,"3.1.1.12 MHL Data Eye Diagram","Clock Multiplier")
Loop Bandwidth	returnval =mClient.GetAnalyzeParameter(clientID, devicename, suitename,"3.1.1.12 MHL Data Eye Diagram","Loop bandwidth (MHz)")
Clock Edge	returnval =mClient.GetAnalyzeParameter(clientID, devicename, suitename,"3.1.1.12 MHL Data Eye Diagram","Clock Edge")

Select a Measurement

Command name	Parameters	Description	Return value	Example
SelectTest()	string clientID (see page 118) string device (see page 119) string suite (see page 119) string test (see page 119) bool isSelected (see page 120)	This method selects or deselects a given test.	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 **Select test: (test name)** returnval=mClient.SelectTest(clientID, devicename, suitename, testname, isSelected);) Select a specific test.

Select a Specific Test

Test	Command
3.1.1.1 Standby Output Voltage — VOFF	For selecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "3.1.1.1 Standby Output Voltage-VOFF", True);` For deselecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "3.1.1.1 Standby Output Voltage-VOFF", False)`
3.1.1.2 Single-ended High Level Voltage — VSE_HIGH	For selecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "3.1.1.2 Single-ended High Level Voltage-VSE_HIGH", True);` For deselecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "3.1.1.2 Single-ended High Level Voltage-VSE_HIGH", False);`
3.1.1.3 Single-ended Low Level Voltages — VSE_LOW	For selecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "3.1.1.3 Single-ended Low Level Voltages-VSE_LOW", True);` For deselecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "3.1.1.3 Single-ended Low Level Voltages-VSE_LOW", False);`
3.1.1.4 Differential Output Swing Voltage — VDF_SWING	For selecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "3.1.1.4 Differential Output Swing Voltage-VDF_SWING", True);` For deselecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "3.1.1.4 Differential Output Swing Voltage-VDF_SWING", False);`
3.1.1.5 Common-mode Output Swing Voltage — V_CMSWING	For selecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "3.1.1.5 Common-mode Output Swing Voltage-V_CMSWING", True);` For deselecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "3.1.1.5 Common-mode Output Swing Voltage-V_CMSWING", False);`

Test	Command
3.1.1.6 Differential Rise and Fall Times-TR_DF, TF_DF	For selecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "3.1.1.6 Differential Rise and Fall Times-TR_DF, TF_DF", True);` For deselecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "3.1.1.6 Differential Rise and Fall Times-TR_DF, TF_DF", False);`
3.1.1.7 Common-mode Rise and Fall Times — TR_CM, TF_CM	For selecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "3.1.1.7 Common-mode Rise and Fall Times-TR_CM, TF_CM", True);` For deselecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "3.1.1.7 Common-mode Rise and Fall Times-TR_CM, TF_CM", False);`
3.1.1.8 Differential Intra-Pair Skew —TSKEW_DF	For selecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "3.1.1.8 Differential Intra-Pair Skew-TSKEW_DF", True);` For deselecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "3.1.1.8 Differential Intra-Pair Skew-TSKEW_DF", False);`
3.1.1.10 MHL Clock Duty Cycle – 24 Bit or Packed Pixel (CTS 1.X)	For selecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "3.1.1.10 MHL Clock Duty Cycle 24-Bit or Packed Pixel Mode", True);` For deselecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "3.1.1.10 MHL Clock Duty Cycle 24-Bit or Packed Pixel Mode", False);`
3.1.1.10 MHL Clock Duty Cycle – Normal Mode (CTS 2.0)	For selecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "3.1.1.10 MHL Clock Duty Cycle Normal Mode", True);` For deselecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "3.1.1.10 MHL Clock Duty Cycle Normal Mode", False);`

Test	Command
3.1.1.11 MHL Clock Jitter (CTS 1.X)	For selecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "3.1.1.11 MHL Clock Jitter", True);` For deselecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "3.1.1.11 MHL Clock Jitter", False);`
3.1.1.11 MHL Clock Jitter – Normal Mode (CTS 2.0)	For selecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "3.1.1.11 MHL Clock Jitter Normal Mode", True);` For deselecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "3.1.1.11 MHL Clock Jitter Normal Mode", False);`
3.1.1.12 MHL Data Eye Diagram (CTS 1.X)	For selecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "3.1.1.12 MHL Data Eye Diagram", True);` For deselecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "3.1.1.12 MHL Data Eye Diagram", False);`
3.1.1.12 MHL Data Eye Diagram Normal Mode (CTS 2.0)	For selecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "3.1.1.12 MHL Data Eye Diagram Normal Mode", True);` For deselecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "3.1.1.12 MHL Data Eye Diagram Normal Mode", False);`
3.1.1.14 MHL Clock Duty Cycle – Packed Pixel Mode (CTS 2.0)	For selecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "3.1.1.14 MHL Clock Duty Cycle Packed Pixel Mode", True);` For deselecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "3.1.1.14 MHL Clock Duty Cycle Packed Pixel Mode", False);`
3.1.1.15 MHL Clock Jitter – Packed Pixel Mode (CTS 2.0)	For selecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "3.1.1.15 MHL Clock Jitter Packed Pixel Mode", True);` For deselecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "3.1.1.15 MHL Clock Jitter Packed Pixel Mode", False);`

Test	Command
3.1.1.16 MHL Data Eye Diagram – Packed Pixel Mode (CTS 2.0)	For selecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "3.1.1.16 MHL Data Eye Diagram Packed Pixel Mode", True);` For deselecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "3.1.1.16 MHL Data Eye Diagram Packed Pixel Mode", False);`
4.1.1.1. Input Signal DC Voltage Level Tolerance Test	For selecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "4.1.1.1 Input Signal DC Voltage Level Tolerance", True);` For deselecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "4.1.1.1 Input Signal DC Voltage Level Tolerance", False);`
4.1.1.2 Input Signal Minimum and Maximum Swing Voltages Tolerance	For selecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "4.1.1.2 Input Signal Minimum and Maximum Swing Voltages Tolerance", True);` For deselecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "4.1.1.2 Input Signal Minimum and Maximum Swing Voltages Tolerance", False);`
4.1.1.3 Intra-Pair Skew Tolerance Test	For selecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "4.1.1.3 Intra-Pair Skew Tolerance", True);` For deselecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "4.1.1.3 Intra-Pair Skew Tolerance", False);`
4.1.1.4 Jitter Tolerance Test (CTS 1.X)	For selecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "4.1.1.4 Jitter Tolerance", True);` For deselecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "4.1.1.4 Jitter Tolerance”, False);`
4.1.1.4 Jitter Tolerance in Normal Mode Test (CTS 2.0)	For selecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "4.1.1.4 Jitter Tolerance Normal Mode", True);` For deselecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "4.1.1.4 Jitter Tolerance Normal Mode”, False);`

Test	Command
4.1.1.8 Jitter Tolerance in Packed Pixel Mode Test (CTS 2.0)	For selecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "4.1.1.8 Jitter Tolerance Packed Pixel Mode", True);` For deselecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "4.1.1.8 Jitter Tolerance Packed Pixel Mode", False);`
5.1.1.1 Input Signal Single-Ended Voltage Level Tolerance Test	For selecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "5.1.1.1 Input Signal Single-Ended Voltage Level Tolerance", True);` For deselecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "5.1.1.1 Input Signal Single-Ended Voltage Level Tolerance", False);`
5.1.1.2 Input Signal Minimum and Maximum Swing Voltage Tolerance Test	For selecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "5.1.1.2 Input Signal Minimum and Maximum Swing Voltages Tolerance", True);` For deselecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "5.1.1.2 Input Signal Minimum and Maximum Swing Voltages Tolerance", False);`
5.1.1.3 Intra-Pair Skew Tolerance Test	For selecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "5.1.1.3 Intra-Pair Skew Tolerance", True);` For deselecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "5.1.1.3 Intra-Pair Skew Tolerance", False);`
5.1.1.4 Jitter Tolerance Test (CTS 1.X)	For selecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "5.1.1.4 Jitter Tolerance", True);` For deselecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "5.1.1.4 Jitter Tolerance", False);`

Test	Command
5.1.1.4 Jitter Tolerance in Normal Mode Test (CTS 2.0)	For selecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "5.1.1.4 Jitter Tolerance Normal Mode", True);` For deselecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "5.1.1.4 Jitter Tolerance Normal Mode", False);`
5.1.1.9 Jitter Tolerance in Packed Pixel Mode Test (CTS 2.0)	For selecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "5.1.1.9 Jitter Tolerance Packed Pixel Mode", True);` For deselecting the test: `returnval=m_Client.SelectTest(clientID, devicename, suitename, "5.1.1.9 Jitter Tolerance Packed Pixel Mode", 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...".

Select a Suite

Command name	Parameters	Description	Return value	Example
SelectSuite()	string clientID (see page 118) string device (see page 119) string suite (see page 119) bool isSelected (see page 120)	This method selects or deselects a given suite. Setting parameter is selected to True, you can select a suite. Setting parameter is selected 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 = new Client() //m_Client is a reference to the Client class in the Client DLL. returnval as string **Select Suite (Default):** returnval=m_Client.SelectTest(clientID, "MHL Physical Layer Solution", "MHL Transmitter", 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...".

Set Pixel Mode

Command name	Parameters	Description	Return value	Example
SetGeneralParameter()	string clientID (see page 118) string device (see page 119) string suite (see page 119) string test (see page 119) string parameterString (see page 119)	This method sets the Pixel Mode for the given suite or 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 Select Pixel Mode: returnval = m_Client.SetGeneralParameter(clientId, devicename, suitename, string.Empty, parameterstring); Set Pixel Mode to 24 Bits Example (see page 139) Set Pixel Mode to Packed Pixel Example (see page 139)

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 Pixel Mode to 24 Bits Example

```
returnval = m_Client.SetGeneralParameter(clientID, devicename, suitename,
string.Empty,"Pixel Mode$24 Bits");
```

Set Pixel Mode to Packed Pixel Example

```
returnval = m_Client.SetGeneralParameter(clientID, devicename, suitename,
string.Empty,"Pixel Mode$Packed Pixel");
```

Set Resolution

Command name	Parameters	Description	Return value	Example
SetGeneralParameter()	string clientID (see page 118) string device (see page 119) string suite (see page 119) string test (see page 119) string parameterString (see page 119)	This method can be used to set the Low Resolution, High Resolution, or Number of Other Resolutions parameters for a given suite or 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 Set Resolution: returnval = m_Client.SetGeneralParameter(clientId, devicename, suitename, string.Empty, parameterstring); Set Resolution to 24 Bits Pixel Mode Examples (see page 140) Set Resolution for Packed Pixel Mode Examples (see page 140)

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 Resolution to 24 Bits Pixel Mode Examples

```
returnval = m_Client.SetGeneralParameter(clientID, devicename, suitename,
string.Empty,"24Bits Lowest Resolution (MHz)$25");

returnval = m_Client.SetGeneralParameter(clientID, devicename, suitename,
string.Empty,"24Bits Highest Resolution (MHz)$74.25");

returnval = m_Client.SetGeneralParameter(clientID, devicename, suitename,
string.Empty,"24Bits Number of Other Resolution (Count)$0");
```

Set Resolution for Packed Pixel Mode Examples

```
returnval = m_Client.SetGeneralParameter(clientID, devicename, suitename,
string.Empty,"PackedPixel Lowest Resolution (MHz)$25");

returnval = m_Client.SetGeneralParameter(clientID, devicename, suitename,
string.Empty,"PackedPixel Highest Resolution (MHz)$74.25");

returnval = m_Client.SetGeneralParameter(clientID, devicename, suitename,
string.Empty,"PackedPixel Number of Other Resolution (Count)$0");
```

Set Termination Source Method

Command name	Parameters	Description	Return value	Example
SetGeneralParameter()	string clientID (see page 118) string device (see page 119) string suite (see page 119) string test (see page 119) string parameterString (see page 119)	This method sets the Type of Termination source to be used for a given suite or 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 Set Termination Source Method:returnval = m_Client.SetGeneralParameter(clientId, devicename, suitename, string.Empty, parameterstring); Set Termination Source Method Examples (see page 141)

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 Termination Source Method Examples

```
returnval = m_Client.SetGeneralParameter(clientId, devicename, suitename,
string.Empty,"Voltage termination$Internal");

returnval = m_Client.SetGeneralParameter(clientId, devicename, suitename,
string.Empty,"Voltage termination$External Other");

returnval = m_Client.SetGeneralParameter(clientId, devicename, suitename,
string.Empty,"Voltage termination$Tek Power Supply");
```

Set Termination Voltage

Command name	Parameters	Description	Return value	Example
SetGeneralParameter()	string clientID (see page 118) string device (see page 119) string suite (see page 119) string test (see page 119) string parameterString (see page 119)	This method sets the VTerm (Min) and VTerm (Max) voltages for a given suite or measurement. The Limit is between 1V and 5V.	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 Set Termination Voltage: returnval = m_Client.SetGeneralParameter(clientId, devicename, suitename, string.Empty, parameterstring); Set Termination Voltage Examples (see page 142)

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 Termination Voltage Examples

```
returnval = m_Client.SetGeneralParameter(clientId, devicename, suitename,
string.Empty,"VTerm1 (V)$3.1");

returnval = m_Client.SetGeneralParameter(clientId, devicename, suitename,
string.Empty,"VTerm2 (V)$3.5");
```

Run with Set Configurations or Stop the Run Operation

Command name	Parameters	Description	Return value	Example
Run()	string clientID (see page 118)	Runs the selected tests. Note (see page 120) Once the server is set up and is configured, it can be run 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_Clientis a reference to the Client class in the Client DLL. returnval as string returnval=m_Client.Run(clientID)
Stop()	string clientID (see page 118)	Stops the currently running tests. Note (see page 121)	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 is changed to a string value depending on its code and returned to the client. The values of OP_STATUS are as follows:

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

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

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

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

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

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

Get or Set the Timeout Value

Command name	Parameters	Description	Return value	Example
GetTimeOut()	string clientID (see page 118)	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 118) string time (see page 121)	Sets a timeout period specified by client. After expiry of this timeout period, the server is automatically unlocked.	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 are executed while tests are running. The GetCurrentStateInfo() and SendResponse() commands are executed when application is running and in wait state.

Command name	Parameters	Description	Return value	Example
ApplicationStatus()	string clientID (see page 118)	This method gets the status of the server application. The states at a given time are Ready, Running, Paused, Wait, or Error. (see page 117)	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 118) out string[] status (see page 119)	This is 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 returnVal=m_Client.QueryS-tatus(clientID, out statusMes-sages) if ((OP_STATUS)returnVal == OP_STATUS.SUCCESS) return “Status updated...” else return CommandFailed(re-turnVal)

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 118) out string caption (see page 121) out string message (see page 122) out string[] buttonTexts (see page 122)	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 fills up 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. m_Client.GetCurrentState-Info(clientID, caption,message, buttonTexts)
SendResponse() ***NOTE.*** *This command is used when the application is running and is in the wait or error state.*	string clientID (see page 118) out string caption (see page 121) out string message (see page 122) string response (see page 122)	After receiving the additional information using the method GetCurrentState-Info(), the client can decide on the response to send and send the response to the application using this function. The response should be one of the strings that was earlier received as a string array in the GetCurrentState-Info 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. m_Client.SendResponse(clientID, caption,message, response)

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

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

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

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

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

After the Test is Complete

Command name	Parameters	Description	Return value	Example
GetPassFailStatus()	string clientID (see page 118) string device (see page 119) string suite (see page 119) string test (see page 119)	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, "3.1.1.3 Single-ended Low Level Voltages-VSE_LOW") //Pass or Fail
GetResultsValue()	string clientID (see page 118) string device (see page 119) string suite (see page 119) string test (see page 119) string parameterString (see page 119)	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, "MHL Physical Layer Solution", "MHL Transmitter", "3.1.1.3 Single-ended Low Level Voltages-VSE_LOW", "Measured Value")
GetResultsValueForSubMeasurements()	string clientID (see page 118) string device (see page 119) string suite (see page 119) string test (see page 119) string parameterString (see page 119) int rowNr (see page 119)	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 **Single-ended Low Level Voltages DP (nS)** returnval=m_Client.GetResultsValueForSubMeasurements(clientID, "MHL Physical Layer Solution", "MHL Transmitter", "3.1.1.3 Single-ended Low Level Voltages-VSE_LOW", "3.1.1.3 Single-ended Low Level Voltages-VSE_LOW ", "Measured Value",0) //For DP wfm returnval=m_Client.GetResultsValueForSubMeasurements(clientID, "MHL Physical Layer Solution", "MHL Transmitter", "3.1.1.3 Single-ended Low Level Voltages-VSE_LOW", "Measured Value",1) //For DN wfm

Command name	Parameters	Description	Return value	Example
GetReportParameter()	string clientID (see page 118) string device (see page 119) string suite (see page 119) string test (see page 119) string parameterString (see page 150)	This method gets the general report details such as oscilloscope model, TekExpress version, and MHL version.	The return value is the oscilloscope model, TekExpress version, and MHL 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") **MHL Version** returnval=m_Client.GetReportParameter(clientID,"Application Version")
TransferReport()	string clientID (see page 118) string filePath (see page 121)	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")

Command name	Parameters	Description	Return value	Example
TransferWave-forms()	string clientID (see page 118) string filePath (see page 121)	This method transfers all the waveforms from the folder for the current run. ***NOTE.*** *For each click of Run button, a folder is created in the X: drive. Transfer the waveforms before clicking the Run button.*	String that gives the status of the operation after it has been performed. Transfers all the waveforms in the form of a string. 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 returnval=m_Client.Transfer-Waveforms(clientID,"C:\Wave-forms")
TransferImages()	string clientID (see page 118) string filePath (see page 121)	This method transfers all the images (screenshots) from the folder for the current run (for a given suite or measurement). ***NOTE.*** *For each click of Run button, a folder is created in the X: drive. Transfer the waveforms before clicking the Run button.*	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.TransferIm-ages(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 MHL version.

Save, Recall, or Check if a Session is Saved

Command name	Parameters	Description	Return value	Example
CheckSession-Saved()	string clientID (see page 118) out bool saved (see page 118)	This method is called when a check is to be made to know if 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 118) string name (see page 120)	Recalls a saved session. The name of the session is provided by the client.	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 118) string name (see page 120)	Saves the current session. The name of the session is provided by the client.	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 118) string name (see page 120)	Saves the current session in a different name every time this method is called. The name of the session is provided by the client.	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 118)	This method unlocks the server from the client. The ID of the client to be unlocked must be provided. Note (see page 121)	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 118)	This method disconnects the client from the server it is connected to. Note (see page 118)	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.

Measurement Procedure

Signal validation is an automatic, default feature in MHL.

Before each measurement, the signal is validated.

3.1.1.1: Standby Output Voltage Test - V_{OFF}

This MHL Transmitter test measures that the MHL source output voltage is within the specified level limits when the source device is in Standby State or power off mode as specified in the CDF.

Required Test Equipment

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

- Digital Multimeter: DMM4020
- Digital Storage Oscilloscope: DPO/DSA/MSO 70000/B/C/D Series Real time Oscilloscope with BW greater than or equal to 8GHz with Opt DJA, Opt 2XL and Opt. MHD (MHL Advanced Analysis and Compliance Software)
- One Probe: P7313SMA
- Test Fixture: TF-MHL-TPA-P-WOSO*TF-MHL-TPA-TT or TF-MHL-TPA-P-TTSO
- Power supply: PWS4205 or PWS2185 (if using external Power Supply option)
- CBus Sink Board: Simplay Labs MHL Rx/Tx Analyzer SL-860 (Source DUT Port) or SOSI Tektronix board

Connect the equipment as shown in Source — VOFF Setup for Clock Test (see page 155).

Connect the CBus Sink board to the CBus and VBus of the test fixture to enable device discovery and CBus information exchange.

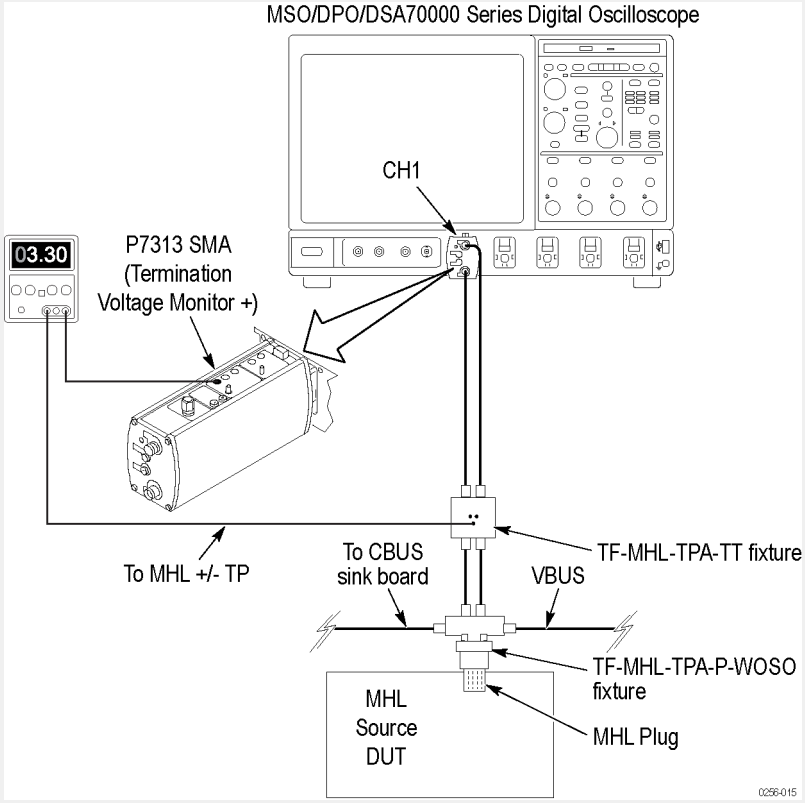
Set the DUT to operate in normal mode.

Measurement Algorithm

Do the following test for all stand-by and power off modes specified in CDF of the DUT.

1. Connect the equipment as shown in Source — VOFF Setup for Clock Test (see page 155) and unplug the DUT from the Test Fixture. Connect the CBus and VBus from the Test Fixture to the Source DUT port of the SL-860 (Source DUT Port) or SOSI Tektronix board.
2. Turn on the DC Power Supply and provide power to VTERM_IN if external Power Supply is used for VTERM, else the Tektronix Oscilloscope provides the VTERM supply internally through the probes.
3. Set VTERM = VTERM{min} 3.135V using Oscilloscope probe control settings if internal power supply is used, else use the external power supply.
4. Measure and record the voltages (VBASELINE) between MHL+ and VTERM and between MHL- and VTERM.
5. Set VTERM = VTERM{min} 3.465V using Oscilloscope probe control settings if internal power supply is used, else use the external power supply.
6. Measure and record the voltages (VBASELINE) between MHL+ and VTERM and between MHL- and VTERM.
7. Plug the DUT to the Test Fixture as shown in Source — VOFF Setup for Clock Test (see page 155). Connect the CBus and VBus from the Test Fixture to the Source DUT port of the SL-860 (Source DUT Port) or SOSI Tektronix board.
8. Put the DUT in the power mode following the instruction in CDF. VBus provision should follow the instruction in CDF.
9. Set VTERM = VTERM{min} 3.135V.
10. Measure the voltage (VOFF_BEFORE_COMPENSATION) between MHL+ and VTERM.
11. Measure the voltage (VOFF_BEFORE_COMPENSATION) between MHL- and VTERM.
12. Record VOFF values for MHL+ and MHL-; VOFF = VOFF_BEFORE_COMPENSATION - VBASELINE.
13. Set VTERM = VTERM{max} 3.465V and repeat steps 10 – 12.
14. If all recorded VOFF values are within +/- 10mV (inclusive), PASS. Otherwise FAIL.

Source — V_{OFF} Setup for Clock Test



3.1.1.5: Common-Mode Output Swing Voltage Test - $V_{CMSWING}$

This MHL Transmitter test confirms that common-mode output voltage swing amplitude is within the specified limits when the DUT operates in normal mode.

Required Test Equipment

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

- Two probes: one P313SMA and one P7240
- Test Fixture: TPA-P Source with a micro USB, pluggable to be used for Source testing
- Power supply: PWS4205 or PWS2185 (to be used if an external power supply-based test setup is used)
- CBUS: CBUS Sink Adapter board

Connect the equipment as shown in Connection Diagram 1 (see page 14).

Measurement Algorithm

TekExpress MHL automatically executes the calculations described below.

1. Acquire the waveform.
2. Consider the data point in the region of 3 unit intervals from the Rise transition for V_{High} and Fall transition for V_{Low}.
3. Average the respective V_{High} and V_{Low} for 10,000 regions and get V_{High} and V_{Low}. measurements.
4. Calculate $V_{CMSWING}$ for the given V_{TERM},

   Where:

   - V_{TERM} = 3.135 V (lowest)
   - V_{TERM} = 3.465 V (highest)

If 360 mV ≤ $V_{CMSWING}$ ≤ Minimum (720 mV, 0.85 $V_{DFSWING}$) for all recorded $V_{CMSWING}$ values, then PASS. Otherwise, FAIL.

3.1.1.7: Common-mode Rise and Fall Times Test - T_{R_CM}, T_{F_CM}

This MHL Transmitter test confirms that the rise time and fall time of the common-mode output signal are within the specified limits.

Rise time and Fall time are measured on the Transition of the common-mode signal by measuring the time difference between the 20% and 80% level during the Transition. The Reference Level is calculated based on the Common-mode Swing measurement. This measurement is performed by averaging the measured RT and FT for 10,000 or more transitions.

Required Test Equipment

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

- 2 Probes: one P7313SMA and one P7240
- Test Fixture TPA-P Source with a micro USB pluggable for source testing
- DC Power Supply: PWS4205 or PWS2815 (to be used if an external power supply-based test setup is used)
- CBUS Sink Adapter board

Connect the equipment as shown in Connection Diagram 1 (see page 14).

Measurement Algorithm

TekExpress MHL automatically executes the calculations described below.

The Rise Time measurement is the time difference between when the VRefHi reference level is crossed and the VRefLo reference level is crossed on the rising edge of the waveform. The Rise Time algorithm uses the VRef values as the reference voltage level. Each edge is defined by the slope, voltage reference level (threshold), and hysteresis. The application calculates this measurement using the following equations:

Where:

- T^{Rise} is the Rise Time
- T^{Hi+} is the VRefHi crossing on the rising edge
- T^{Lo+} is the VRefLo crossing on the rising edge

V_{RefHi} is 80% of the Ref level and V_{RefLo} is 20% of the Ref level.

$T_{nFall} = T_{nHi} - T_{nLo}$

$$T_{Fall} = \frac{\sum_{1}^{n} T_{Fall}(n)}{n}$$

$T_{nRIse} = T_{nHi} - T_{nLo}$

$$T_{Rise} = \frac{\sum_{1}^{n} T_{Rise}(n)}{n}$$

3.1.1.10: MHL Clock Duty Cycle Test – 24 Bit or Packed Pixel Mode (CTS 1.X); Normal Mode (CTS 2.0)

This test confirms that the MHL clock duty cycle in 24-bit or packed pixel mode does not exceed the limits allowed by the specification. (CTS 1.X)

This test confirms that the MHL clock duty cycle does not exceed the limits allowed by the specification in Normal Mode. (CTS 2.0)

Required Test Equipment

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

- Two probes: one P7313SMA and one P7240
- Test Fixture: TPA-P Source with a micro USB pluggable to be used for source testing
- Power supply: PWS4205 or PWS2185 (to be used if an external power supply-based test setup is used)
- CBUS: CBUS Sink Adapter board

Connect the equipment as shown in Connection Diagram 1 (see page 14).

Measurement Algorithm

TekExpress MHL automatically executes the calculations described below.

The +Duty Cycle and –Duty Cycle measurements calculate the ratio of the positive (or negative) portion of the cycle relative to the period. The application calculates these measurements as follows:

Where:

- D+ is the positive duty cycle
- D– is the negative duty cycle
- W+ is the positive pulse width
- W– is the negative pulse width
- P_{Clock} is the period

3.1.1.11: MHL Clock Jitter Test (CTS 1.X) – in Normal Mode (CTS 2.0)

This MHL Transmitter test confirms that the MHL Clock output does not contain excessive jitter greater than the limit allowed by the specification.

Required Test Equipment

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

- Two probes: one P313SMA and one P7240
- Test Fixture: TPA-P Source with a micro USB pluggable to be used for source testing
- Power supply: PWS4205 or PWS2185 (to be used if an external power supply-based test setup is used)
- CBUS: CBUS Sink Adapter board

Connect the equipment as shown in Connection Diagram 1 (see page 14).

Measurement Algorithm

TekExpress MHL automatically executes the calculations described below.

TIE (Time Interval Error) is the difference in time between an edge in the source waveform and the corresponding edge in a reference clock. The reference clock is usually determined by a clock recovery process performed on the source waveform. The source waveform is filtered using a 2-GHz low-pass filter. For Explicit-Clock clock recovery, the process is performed on an explicitly identified source.

If the signal is Clock Type,

The application calculates the Clock TIE measurement using the following equation:

$$TIE_n^{Clock} = T_n^{Clock} - T_n{}'^{Clock}$$

Where:

- TIE^{Clock} is the VRefMid crossing time for the specified clock edge.
- T'^{Clock} is the corresponding edge time for the specified reference clock after recovering it using PLL with a bandwidth of 4 MHz.

3.1.1.14: MHL Clock Duty Cycle Test - Packed Pixel Mode

This test confirms that the MHL clock duty cycle in packed pixel mode does not exceed the limits allowed by the specification.

NOTE. *This test is only available in CTS 2.0.*

For this test, use the same Required Test Equipment and Measurement Algorithm as

3 1 1 10MHL Clock Duty Cycle 24 Bit or Packed Pixel Mode CTS 1 X Normal Mode CTS 2 0 (see page 158)

3.1.1.15: MHL Clock Jitter Test in Packed Pixel Mode

NOTE. *This test is only available in CTS 2.0.*

This test confirms that the MHL Clock output does not contain excessive jitter larger than the limit allowed by the specification in Packed Pixel Mode.

For this test, use the same Required Test Equipment and Measurement Algorithm as

3. 1 .1 .11: MHL Clock Jitter Test in Normal Mode (see page 159).

3.1.1.2: Single-Ended High Level Voltage Test - V_{SE_HIGH}

This MHL Transmitter test confirms that the single-ended high output voltage level is within the specified limits when the DUT is in normal mode.

Required Test Equipment

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

- Two Differential probes: P7313SMA
- Test Fixture: TPA-P source with a micro USB, pluggable to be used for source testing
- Power supply: PWS4205 or PWS2185 (to be used if an external power supply-based test setup is used)
- CBUS: CBUS Sink Adapter board

Connect the equipment as shown in Connection Diagram 2 (see page 14).

Measurement Algorithm

TekExpress MHL automatically executes the calculations described below.

The single-ended High and Low are measured using a customized algorithm to find regions that were identified based on the Bit stream.

1. Acquire the waveform by capturing at least 1000 regions using a real-time oscilloscope.
2. Find the Region of LHHHH transition on the region of single-ended measurement.
3. Measure and record the Histogram mode value at the middle HH pattern of the MHL-P signal when the common-mode voltage is high and it is MHLP-SEHIGH.
4. Measure and record the Histogram mode value at the middle HLLLL of the MHL-N signal when the common-mode signal is high and it is MHLN-SEHIGH.
5. Compare the measured values with the limit values:

   If $V_{TERM} - 540\ mV \leq V_{SE\_HIGH} \leq V_{TERM} + 10\ mV$ for all recorded V_{SE_HIGH} values, then PASS. Otherwise, FAIL.

   Where:

   - $V_{TERM} = 3.135$ V (lowest)
   - $V_{TERM} = 3.465$ V (highest)
   - Limit Values are ≥ 2595 mV and ≤ 3145 mV

3.1.1.3: Single-Ended Low Level Voltages Test - V_{SE_LOW}

This MHL Transmitter test confirms that the single-ended low output voltage level is within the specified limits when the source device is in normal mode.

Required Test Equipment

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

- Two Differential Probes: P7313SMA
- Test Fixture: TPA-P Source with a micro USB, pluggable to be used for source testing
- Power supply: PWS4205 or PWS2185 (to be used if an external power supply-based test setup is used)
- CBUS: CBUS Sink Adapter board

Connect the equipment as shown in Connection Diagram 2 (see page 14).

Measurement Algorithm

TekExpress MHL automatically executes the calculations described below.

The single-ended High and Low are measured using a customized algorithm to find regions of LHHHH transition on single-ended measurement regions.

1. Acquire the waveform by capturing at least 1000 regions using a real-time oscilloscope.
2. Find the region of LHHHH transition on the region of single-ended measurement
3. Measure and record the Histogram mode value at the middle HH on the MHL-N signal when the common-mode signal is low, and it is MHLN-SE-LOW.
4. Measure and record the Histogram mode value at the middle HLLLL pattern on the MHL-P signal using the Histogram technique when the common-mode voltage is low, and it is MHLP-SELOW.
5. Compare the measured values with limit values:

   If $V_{TERM} - 1760\ mV \leq V_{SE\_LOW} \leq V_{TERM} - 700\ mV$ for all recorded V_{SE_Low} values, then PASS. Otherwise, FAIL

   Where:

   - V_{TERM} = 3.135 V (lowest)
   - V_{TERM} = 3.465 V (highest)
   - Limit values are ≥ 1375 and ≤ 2435

3.1.1.4: Differential Output Swing Voltage Test - V_{DF_SWING}

This MHL Transmitter test confirms that the differential output voltage swing amplitude is within the specified limits when the source device is in normal mode.

Required Test Equipment

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

- Two probes: one P313SMA and one P7240
- Test Fixture: TPA-P Source with a micro USB, pluggable to be used for source testing
- Power supply: PWS4205 or PWS2185 (to be used if an external power supply-based test setup is used)
- CBUS: CBUS Sink Adapter board

Connect the equipment as shown in Connection Diagram 1 (see page 14).

Measurement Algorithm

TekExpress MHL automatically executes the calculations described below.

1. Acquire the waveform. Slice and overlap data points for approximately 5 UI and then measure the minimum and maximum using a common-mode clock waveform.
2. Consider the histogram vertical window for the data point of value > minimum + 0.75* (maximum – minimum) and measure the mode as V_{High}.
3. Consider the histogram vertical window for the data point of value > maximum – 0.75* (maximum – minimum) and measure the mode as V_{Low}.
4. Calculate the average ($V_{CMSWING}$) of the measured common-mode High and Low values.
5. Record the histogram mode value for the common-mode Low.
6. Calculate $V_{CMSWING}$ for the given V_{TERM}.

   Where:

   - V_{TERM} = 3.135 V (lowest)
   - V_{TERM} = 3.465 V (highest)

If 600 mV ≤ $V_{DFSWING}$ ≤ 1000 mV for all recorded $V_{DFSWING}$ values, then Pass. Otherwise, Fail

The measurement takes into consideration all the data points on the differential signal for 10,000 clock periods.

3.1.1.6: Differential Rise and Fall Times Test - T_{R_DF}, T_{F_DF}

This MHL Transmitter test confirms that the rise and fall times of the differential output signal are equal to or larger than the minimum limit.

Required Test Equipment

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

- Two probes: one P313SMA and one P7240
- Test Fixture: TPA-P Source with a micro USB, pluggable to be used for Source testing
- Power supply: PWS4205 or PWS2185 (to be used if an external power supply-based test setup is used)
- CBUS: CBUS Sink Adapter board

Connect the equipment as shown in Connection Diagram 1 (see page 14).

Measurement Algorithm

TekExpress MHL automatically executes the calculations described below.

1. Perform the Differential Swing measurement.
2. Arrive at 20% and 80% levels of the signal.
3. Perform the Rise time and Fall time calculations similar to the Common-mode Rise time and Fall time on the data using the Reference level arrived at in step 2.

3.1.1.8: Differential Intra-Pair Skew Test - T_{SKEW_DF}

This MHL Transmitter test confirms that the timing skew in the differential signal pair is below the specified limits.

Required Test Equipment

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

- Two Differential probes: P7313SMA
- Test Fixture: TPA-P Source with a micro USB, pluggable to be used for Source testing
- Power supply: PWS4205 or PWS2185 (to be used if an external power supply-based test setup is used)
- CBUS: CBUS Sink Adapter board

Connect the equipment as shown in Connection Diagram 2 (see page 14).

Measurement Algorithm

This measurement is performed using a custom algorithm that evaluates the qualified region for the measurement. The skew is the average of 1000 qualified regions.

3.1.1.12: MHL Data Eye Diagram Test (CTS 1.X) – in Normal Mode (CTS 2.0)

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

NOTE. *If the MHL Data Eye diagram test fails due to mask hits, TekExpress MHL Solution provides the option to the user to move the mask horizontally to achieve zero or minimal hits. Refer to the* *Mask Movement (see page 189)* *procedure.*

Required Test Equipment

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

- Two probes: one P313SMA and one P7240
- Test Fixture: TPA-P Source with micro USB, pluggable to be used for Source testing
- Power supply: PWS4205 or PWS2185 (to be used if an external power supply-based test setup is used)
- CBUS: CBUS Sink Adapter board

Connect the equipment as shown in Connection Diagram 1 (see page 14).

Measurement Algorithm

TekExpress MHL automatically executes the algorithm described below.

1. Acquire the Waveform Differential Data and Common-mode clock waveforms.
2. Filter the clock signal using the 2-GHz low-pass filter.
3. Recover the clock using 4 MHz PLL
4. Plot the Eye diagram using the Differential data signal with the recovered clock from the common-mode signal.
5. The Jitter is measured as:

   Data period – Eye opening (EYE width)

3.1.1.16: MHL Data Eye Diagram Test in Packed Pixel Mode

NOTE. *This test is only available in CTS 2.0.*

This test confirms that the MHL Data output has signal quality that meets the eye opening required by the specification in Packed Pixel Mode.

For this test, use the same Required Test Equipment and Measurement Algorithm as

3. 1 .1 .12: MHL Data Eye Diagram Test in Normal Mode (see page 166).

4.1.1.1: Input Signal DC Voltage Level Tolerance Test

This MHL Receiver test confirms that the sink device supports input signal DC voltage level allowed by the specification.

Make the connection after signal calibration is done. Set the DUT to operate in normal mode.

Required Test Equipment

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

- MHL Signal Generator: 7122C with option 01, 02 or 06 and 08, and Option SDX with Direct Synthesis capability to simulate Cable emulator effect and rise time filter effect.
- Test Fixture: 1 set TF-MHL-TPA-P-WOSI
- Rise time Filter: 2 Picosecond Pulse labs 120PS
- Bias Tees: 2 Mini Circuits Bias Tees ZX85–12G-S+
- CBUS Source Board: Simplay Labs MHL Rx/Tx Analyzer SL-860 (Sink DUT Port) or SOSI Tektronix board

Connect the equipment as shown in MHL Input Signal DC Voltage Level Test (see page 169).

Measurement Algorithm

1. Connect the equipment as shown for Single-Ended Signals Calibration Setup for Sink Tests (see page 169) and Differential & Common-Mode Signals Calibration Setup for Sink Tests (see page 170).

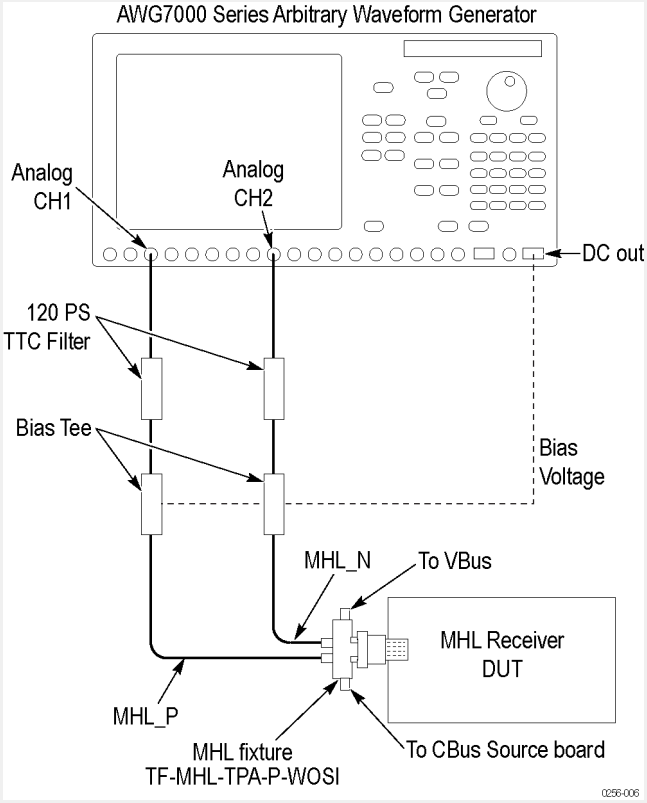
Set VTERM = 3.3V, AWG Ch1 transmitting MHL+ and AWG Ch2 transmitting MHL- signals. Ensure the Bias Tees are used at the output of the AWG channels along with 120ps filters.

2. Calibrate the Signal Generator AWG7122C output for the following settings.
   a. Frequency: Highest frequency supported by the Sink DUT
   b. Pattern: MHL Gray Ramp data streams
   c. Rise/fall times:
      - Less than or equal to 2.25Gbps

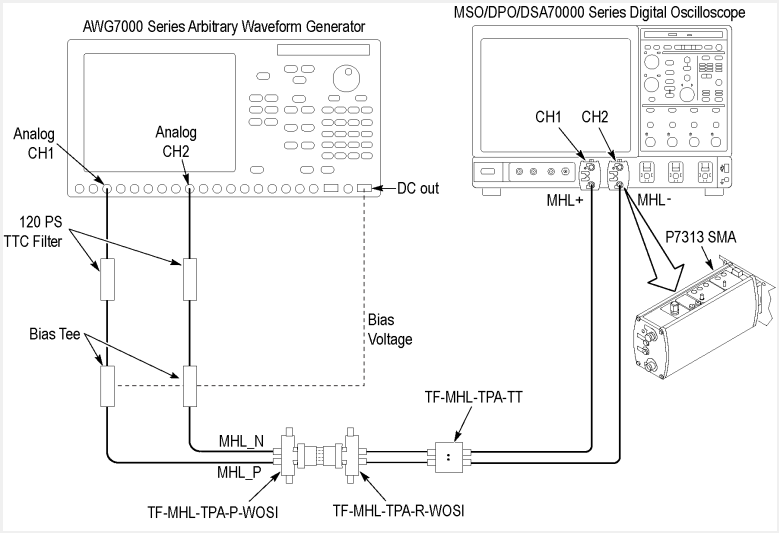
        Differential signal 200ps (20% - 80%), Common-mode signal 600ps (20% - 80%)
      - Less than or equal to 2.25Gbps

        Differential signal 200ps (20% - 80%), Common-mode signal 600ps (20% - 80%)
   d. Swing voltages: Differential signal 800mV, Common-mode signal 540mV
3. DC level of the output single-ended signal: V_{TERM} –300mV
4. Connect the equipment MHL Input Signal DC Voltage Level Test (see page 169) with the CBus Source board connected to the Tektronix fixture CBus and VBus ports.
5. Turn on the DUT and enable MHL mode through Device Discovery and CBus Information Exchange with the CBus Source board.
6. Examine the output of the Sink through the display image.
7. Record errors on the display if any for the given V_{TERM} frequency and DC voltage level.
8. Set DC level of the output single-ended signal: V_{TERM} –1200mV
9. Turn on the DUT and enable MHL mode through Device Discovery and CBus Information Exchange with the CBus Source board.
10. Examine the output of the Sink through the display image.
11. Record errors on the display if any for the given V_{TERM} , frequency and DC voltage level.
12. If there is no error on the display in all recorded results, then PASS. Otherwise FAIL.

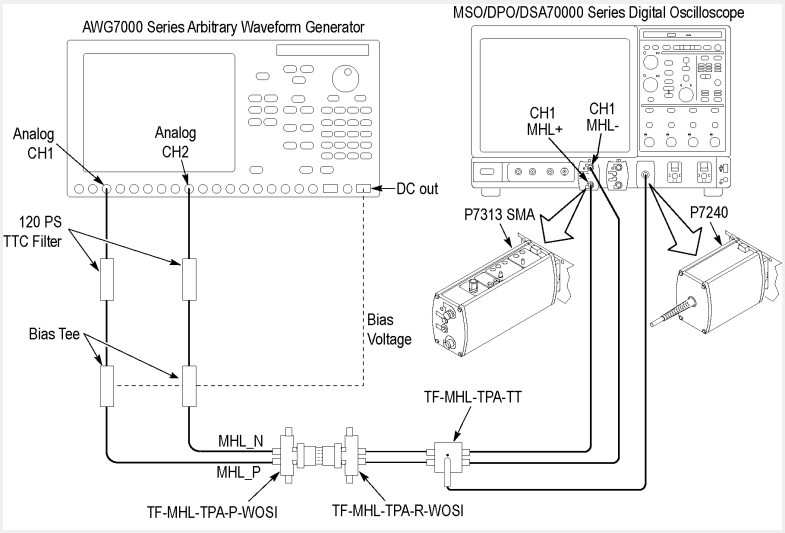
MHL Input Signal DC Voltage Level Test



Single-ended Signals Calibration Setup for Sink Test



Differential & Common-Mode Signals Calibration Setup for Sink Test



4.1.1.2: Input Signal Minimum and Maximum Swing Voltages Tolerance Test

This MHL Receiver test confirms that the sink device supports input signal DC voltage level and swing voltage allowed by the specification.

Required Test Equipment

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

- MHL Signal Generator: AWG7122C with option 01, 02 or 06 and 08 and Option SDX with Direct Synthesis capability, to simulate Cable emulator effect and Rise time filter effect.
- Test Fixture: 1 set TF-MHL-TPA-P-WOSI
- Rise time Filter: 2 Picosecond Pulse labs 120PS
- Bias Tees: 2 Mini Circuits Bias Tees ZX85–12G-S+
- CBUS Source Board: Simplay Labs MHL Rx/Tx Analyzer SL-860 (Sink DUT Port) or SOSI Tektronix board

Make the connection as shown in Sink MHL Input Signal DC Voltage Level Min-Max Test (see page 172) and Amplifier Connections to DC Power Supply (see page 173) after signal calibration is done. Set the DUT to operate in normal mode.

Measurement Algorithm

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

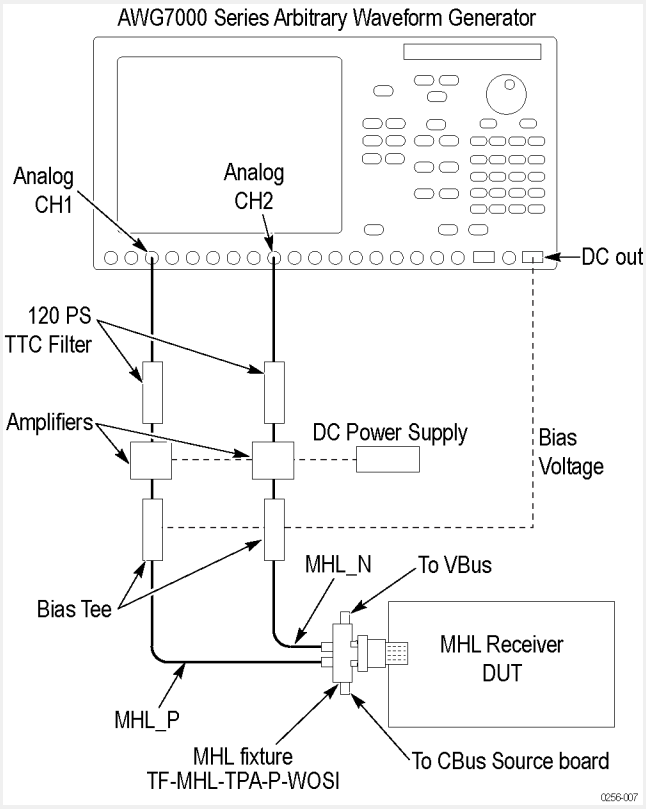
1. Connect the equipment as shown for Single Ended Signals Calibration Setup for Sink Min Max Test (see page 173), and Differential and Common Mode Signals Calibration Setup for Min Max Test (see page 174). Set V_{TERM} = 3.3V, AWG Ch1 transmitting MHL+ and AWG Ch2 transmitting MHL- signals. Ensure the Bias Tees are used at the output of the AWG channels along with 120ps filters.
2. Calibrate the Signal Generator AWG7122C output for the following settings.
   a. Frequency: Highest frequency supported by the Sink DUT
   b. Pattern: MHL Gray Ramp data streams
   c. Rise/fall times:
      - Less than or equal to 2.25Gbps

        Differential signal 200ps (20% - 80%), Common-mode signal 600ps (20% - 80%
      - Above 2.25Gbps

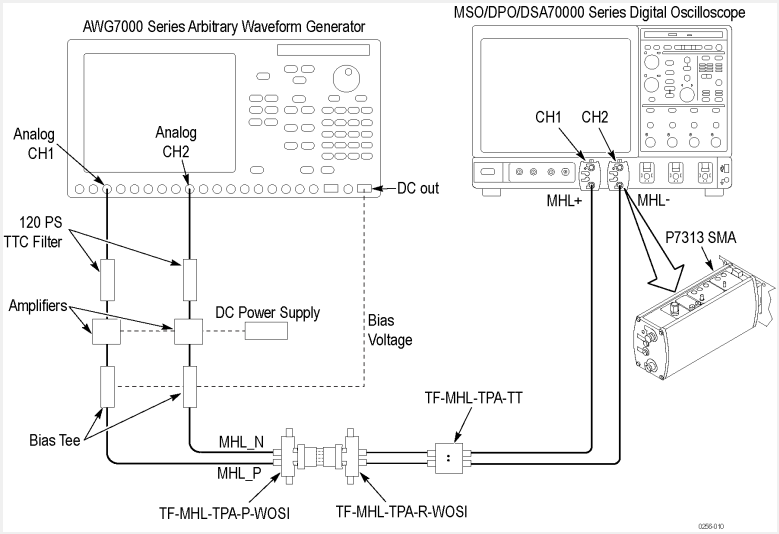
        Differential signal 100ps (20% - 80%), Common-mode signal 600ps (20% - 80%
   d. Swing voltages: Differential signal 800mV, Common-mode signal 540mV
   e. DC level of the output single-ended signal: V_{TERM} – 750mV
3. Connect the equipment as shown in Sink MHL Input Signal DC Voltage Level Min-Max Test (see page 172) with CBus Source board connected to Tektronix fixture CBus and VBus ports.
4. Turn on the DUT and enable MHL mode through Device Discovery and CBus Information Exchange with the CBus Source board.
5. Examine the output of the Sink through the display image.
6. Record errors on the display if any for the given V_{TERM}, frequency and differential and common-mode swing voltages.
7. Adjust the differential swing voltage to 1000mV, the common-mode swing voltage to 720mV and the DC level of the output single-ended signal to V_{TERM} – 1200mV.
8. Connect the equipment as shown in as shown in Sink MHL Input Signal DC Voltage Level Min-Max Tests (see page 172).
9. Turn on the DUT and enable MHL mode through Device Discovery and CBus Information Exchange with the CBus Source board.
10. Examine the output of the Sink through the display image.
11. Record errors on the display if any for the given V_{TERM}, frequency and differential and common-mode swing voltages.
12. Adjust the differential swing voltage to 200mV, the common-mode swing voltage to 170mV and the DC level of the output single-ended signal to V_{TERM} – 300mV.
13. Connect the equipment as shown in as shown in Sink MHL Input Signal DC Voltage Level Min-Max Tests (see page 172).

14. Turn on the DUT and enable MHL mode through Device Discovery and CBus Information Exchange with the CBus Source board.
15. Examine the output of the Sink through the display image.
16. Record errors on the display if any for the given V_{TERM}, frequency and differential and common-mode swing voltages.
17. If there is no error on the display in all recorded results, then PASS. Otherwise FAIL.

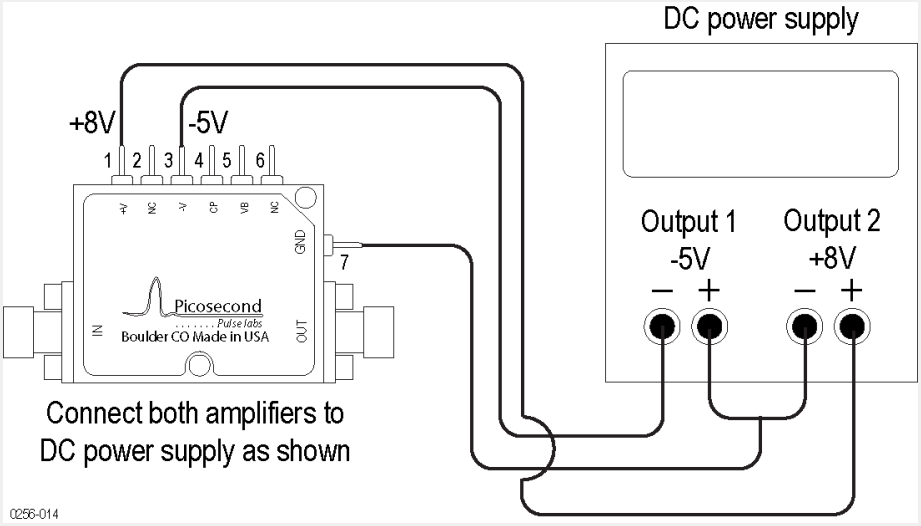
Sink MHL Input Signal DC Voltage Level Min Max Test



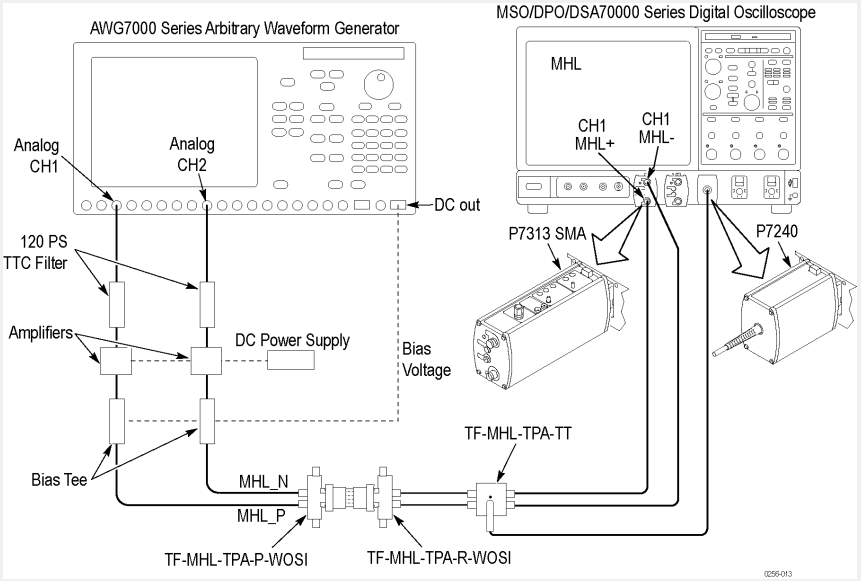
Single Ended Signals Calibration Setup for Sink Min Max Test



Amplifier Connections to DC Power Supply



Differential and Common Mode Signals Calibration Setup for Sink Min Max Test



4.1.1.3: Intra-Pair Skew Tolerance Test

This MHL Receiver test confirms that the sink device supports input signal DC voltage level and swing voltage allowed by the specification.

Required Test Equipment

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

- MHL Signal Generator: AWG7122C with option 01, 02 or 06 and 08 and Option SDX with Direct Synthesis capability, to simulate Cable emulator effect and Rise time filter effect.
- Test Fixture: 1 set TF-MHL-TPA-P-WOSI
- Rise time Filter: 2 Picosecond Pulse labs 120PS
- Bias Tees: 2 Mini Circuits Bias Tees ZX85–12G-S+
- CBUS Source Board: Simplay Labs MHL Rx/Tx Analyzer SL-860 (Sink DUT Port) or SOSI Tektronix board

Connect the equipment as shown in MHL Input Signal DC Voltage Level Test (see page 169).

Measurement Algorithm

1. Connect the equipment as shown for Single-Ended Signals Calibration Setup for Sink Test (see page 169) and Differential & Common-Mode Signals Calibration Setup for Sink Test (see page 170).

Set V_{TERM} = 3.3V, AWG Ch1 transmitting MHL+ and AWG Ch2 transmitting MHL- signals. Ensure the Bias Tees are used at the output of the AWG channels along with 120ps filters.

2. Calibrate the Signal Generator AWG7122C output for the following settings.
   a. Frequency: Highest frequency supported by the Sink DUT
   b. Pattern: MHL Gray Ramp data streams
   c. Rise/fall times:
   - Less than or equal to 2.25Gbps

     Differential signal 200ps (20% - 80%), Common-mode signal 600ps (20% - 80%)
   - Less than or equal to 2.25Gbps

     Differential signal 200ps (20% - 80%), Common-mode signal 600ps (20% - 80%)

   a. Swing voltages: Differential signal 800mV, Common-mode signal 540mV
   b. DC level of the output single-ended signal: V_{TERM} – 750mV
   c. Intra-pair skew: Differential skew 0ps, Common-mode skew 0ps
3. Connect the equipment as shown in MHL Input Signal DC Voltage Level Test (see page 169) with the CBus Source board connected to the Tektronix fixture CBus port.
4. Turn on the DUT and enable MHL mode through Device Discovery and CBus Information Exchange with the CBus Source board.
5. Examine the output of the Sink through the display image.
6. Confirm that there is no error on the display. If there is any error, STOP. The test Failed.
7. Increase the intra-pair skew of differential signal and common-mode signal by the same amount to the maximum where error appears on the display. Make the positive channel leads the negative channel in time, which is called positive skew direction.
8. Record the maximum intra-pair skew for the given V_{TERM}, frequency and skew direction.
9. Set the differential and common-mode intra-pair skews to 0ps.
10. Increase the intra-pair skew of differential signal and common-mode signal by the same amount to the maximum where error appears on the display. Make the positive channel lags the negative channel in time, which is called negative skew direction.
11. Record the maximum intra-pair skew for the given V_{TERM}, frequency and skew direction.

4.1.1.4: Jitter Tolerance Test (CTS 1.X) – in Normal Mode (CTS 2.0)

This MHL Receiver test confirms that the sink device can tolerate the maximum clock and data jitter amounts allowed by the specification.

Required Test Equipment

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

- MHL Signal Generator: AWG7122C with option 01, 02 or 06 and 08 and Option SDX with Direct Synthesis capability, to simulate Cable emulator effect and Rise time filter effect.
- Test Fixture: 1 set TF-MHL-TPA-P-WOSI
- Rise time Filter: 2 Picosecond Pulse labs 120PS
- Bias Tees: 2 Mini Circuits Bias Tees ZX85–12G-S+
- CBUS Source Board: Simplay Labs MHL Rx/Tx Analyzer SL-860 (Sink DUT Port) or SOSI Tektronix board

Connect the equipment as shown in MHL Input Signal DC Voltage Level Test (see page 169).

Measurement Algorithm

1. Connect the equipment as shown for Single-Ended Signals Calibration Setup for Sink Tests (see page 169) and Differential & Common-Mode Signals Calibration Setup for Sink Tests (see page 170). Set V_{TERM} = 3.3V, AWG Ch1 transmitting MHL+ and AWG Ch2 transmitting MHL- signals. Ensure the Bias Tees are used at the output of the AWG channels along with 120ps filters.
2. Calibrate the Signal Generator AWG7122C output for the following settings.
   a. Frequency: Lowest frequency supported by the Sink DUT
   b. Pattern: MHL Gray Ramp data streams
   c. Rise/fall times:
      - Less than or equal to 2.25Gbps

        Differential signal 200ps (20% - 80%), Common-mode signal 600ps (20% - 80%)
      - Less than or equal to 2.25Gbps

        Differential signal 200ps (20% - 80%), Common-mode signal 600ps (20% - 80%)
   d. Swing voltages: Differential signal 800mV, Common-mode signal 360mV
   e. DC level of the output single-ended signal: V_{TERM} – 540mV
3. Set the clock jitter frequency to 10MHz and the data jitter frequency to 500KHz. Inject clock jitter and data jitter to the TP1 maximum amount specified in MHL Spec for the given clock frequency.
4. Adjust the differential signal swing to get the marginal eye diagram specified in MHL Spec.
5. Generate MHL signals with Cable emulator effects using the innovative Direct Synthesis capability of AWG7122C.
6. Connect the equipment as shown in MHL Input Signal DC Voltage Level Test (see page 169) with the CBus Source board connected to the Tektronix fixture CBus and VBus ports.
7. Turn on the DUT and enable MHL mode through Device Discovery and CBus Information Exchange with the CBus Source board.

8. Examine the output of the Sink through the display image.
9. Record error on the display if any for the given V_{TERM}, clock frequency, jitter frequency and clock edge location.
10. Change the common-mode clock edge location by $0.1T_{bit}$ step with respect the differential data edge for 10 times.
11. Examine the display at each step and record error on the display if any for the given V_{TERM}, clock frequency, jitter frequency and clock edge location.
12. Repeat the above steps 4-11 for clock jitter frequency 7MHz and data jitter frequency 1MHz.
13. Repeat the above steps 2-12 for the minimum resolution supported by the DUT.
14. If there is no error on the display in all recorded results, then PASS. Otherwise FAIL.

4.1.1.8: Jitter Tolerance Test in Packed Pixel Mode

NOTE. *This test is only available in CTS 2.0.*

This test confirms that the sink device can tolerate the maximum clock and data jitter amounts allowed by the specification in Packed Pixel Mode.

For this test, use the same Required Test Equipment and Measurement Algorithm as

4. 1.1.4 Jitter Tolerance Test in Normal Mode (see page 175)

5.1.1.1: Input Signal Single-Ended Voltage Level Tolerance

This MHL Receiver test confirms that the dongle device supports input signal single-ended voltage level allowed by the specification.

Required Test Equipment

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

- MHL Signal Generator: AWG7122C with option 01, 02 or 06 and 08 and Option SDX with Direct Synthesis capability, to simulate Cable emulator effect and Rise time filter effect.
- Rise time Filter: 2 Picosecond Pulse labs 120PS
- Test Fixture: 1 set TF-MHL-TPA-R-WOSO
- DC Power Supply: PWS4205 or PWS2185
- Bias Tees: 2 Mini Circuits Bias Tees ZX85–12G-S+

- CBUS Source Board: Simplay Labs MHL Rx/Tx Analyzer SL-860 (Sink DUT Port) or SOSI Tektronix board
- A/V Display

Make the connection as shown in Dongle – Input Signal Single-Ended Voltage Level Tolerance Test (see page 179) after calibration is done. Set the DUT to operate in normal mode.

Measurement Algorithm

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

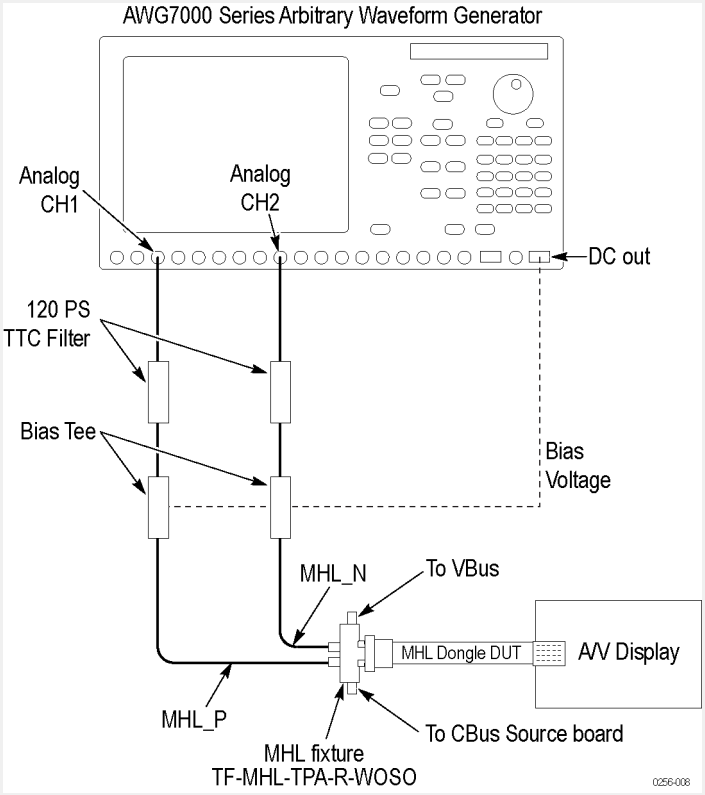
1. Connect the equipment as shown in Single-Ended Signals Calibration Setup for Dongle Tests (see page 180) and as shown in Differential & Common-Mode Signals Calibration Setup for Dongle Tests (see page 180). Set V_{TERM} = 3.3V, AWG Ch1 transmitting MHL+ and AWG Ch2 transmitting MHL- signals. Ensure the Bias Tees are used at the output of the AWG channels along with 120ps filters.
2. Calibrate the MHL Signal Generator AWG7122C output for the following settings. (Calibration of differential and common-mode signals).
   a. Frequency: Highest frequency supported by the Dongle DUT
   b. Pattern: MHL Gray Ramp data streams
   c. Rise/fall times:
      - Less than or equal to 2.25Gbps

        Differential signal 200ps (20% - 80%), Common-mode signal 600ps (20% - 80%)
      - Less than or equal to 2.25Gbps

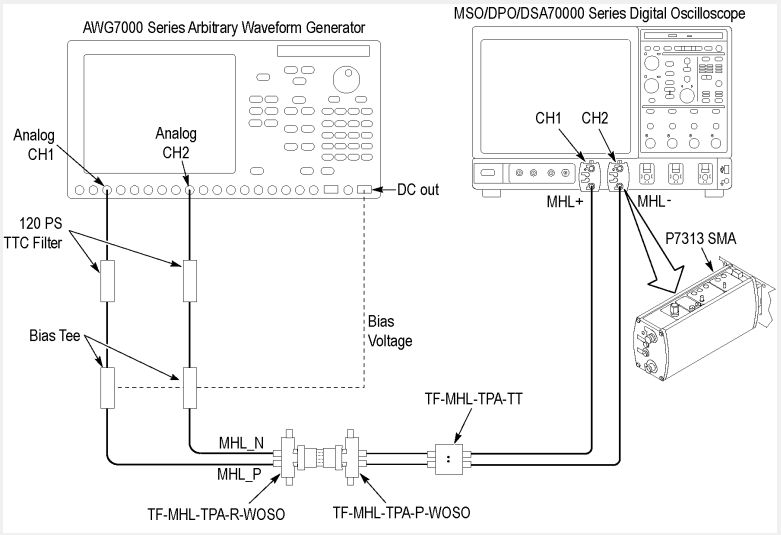
        Differential signal 200ps (20% - 80%), Common-mode signal 600ps (20% - 80%)
   d. Swing voltages: Differential signal 800mV, Common-mode signal 540mV e.
3. Adjust the DC level of the MHL Signal Generator AWG7122C so that $V_{SE\_HIGH} = V_{TERM} + 10$ mV, $V_{SE\_HIGH} = V_{TERM} – 540$ mV, $V_{SE\_LOW} = V_{TERM} – 700$ mV and $V_{SE\_LOW} = V_{TERM} – 1760$ mV. Record the DC level setting for each single-ended value setting. (Calibration of single-ended signals)
4. Connect the equipment as shown in Dongle – Input Signal Single-Ended Voltage Level Tolerance Test (see page 179) with CBus Source board connected to Tektronix fixture CBus port. Provide VBus if DUT is unpowered dongle. Do not provide VBus if DUT is powered dongle.
5. Turn on the DUT and enable the MHL mode through Device Discovery and CBUS Information Exchange with the CBUS Source board.

6. Adjust the DC level of the MHL Signal Generator AWG7122C output so that $V_{SE\_HIGH} = V_{TERM} + 10$ mV.
7. Examine the output of the Dongle DUT on the Display.
   - Record errors on the display if any for the given V_{TERM}, frequency and single-ended voltage.
8. Adjust the DC level of the output so that $V_{SE\_HIGH} = V_{TERM} - 540$ mV.
9. Examine the output of the Dongle DUT on the Display.
   - Record errors on the display if any for the given V_{TERM}, frequency and single-ended voltage.
10. Adjust the DC level of the output so that $V_{SE\_LOW} = V_{TERM} - 700$ mV.
11. Examine the output of the Dongle DUT on the Display.
    - Record errors on the display if any for the given V_{TERM}, frequency and single-ended voltage.
12. Adjust the DC level of the output so that $V_{SE\_LOW} = V_{TERM} - 1760$ mV.
13. Examine the output of the Dongle DUT on the Display.
    - Record errors on the display if any for the given V_{TERM}, frequency and single-ended voltage.
14. If there is no error on the display in all recorded results, then PASS. Otherwise FAIL.

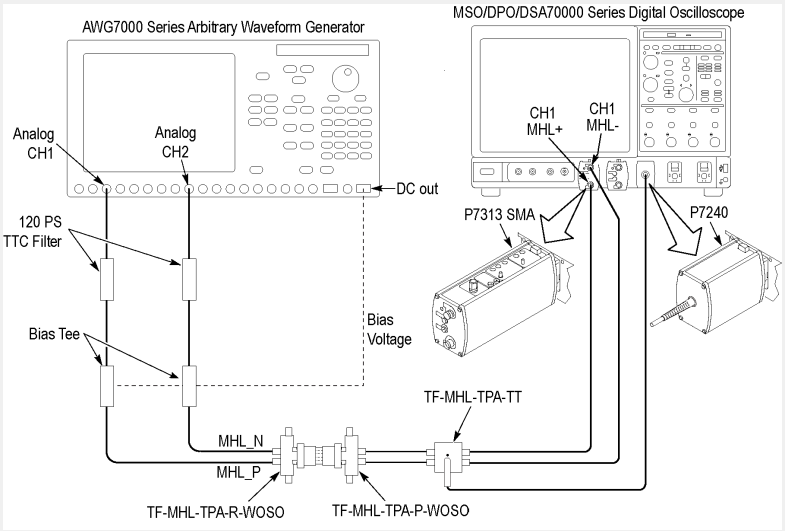
Dongle — Input Signal Single-ended Voltage Level Tolerance Test



Single-Ended Signals Calibration Setup for Dongle Tests



Differential & Common-Mode Signals Calibration Setup for Dongle Test



5.1.1.2: Input Signal Minimum and Maximum Swing Voltage Tolerance Test

This MHL Receiver test confirms that the dongle device supports input signal minimum and maximum swing voltages allowed by the specification.

Required Test Equipment

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

- MHL Signal Generator: AWG7122C with option 01, 02 or 06 and 08 and Option SDX with Direct Synthesis capability, to simulate Cable emulator effect and Rise time filter effect.
- Rise time Filter: 2 Picosecond Pulse labs 120PS
- Test Fixture: 1 set TF-MHL-TPA-R-WOSO
- DC Power Supply: PWS4205 or PWS2185
- Bias Tees: 2 Mini Circuits Bias Tees ZX85–12G-S+
- CBUS Source Board: Simplay Labs MHL Rx/Tx Analyzer SL-860 (Sink DUT Port) or SOSI Tektronix board
- A/V Display

Make the connection as shown in Dongle Min-Max Setup (see page 183) and Amplifier Connections to DC Power Supply (see page 184) after calibration is done. Set the DUT to operate in normal mode.

Measurement Algorithm

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

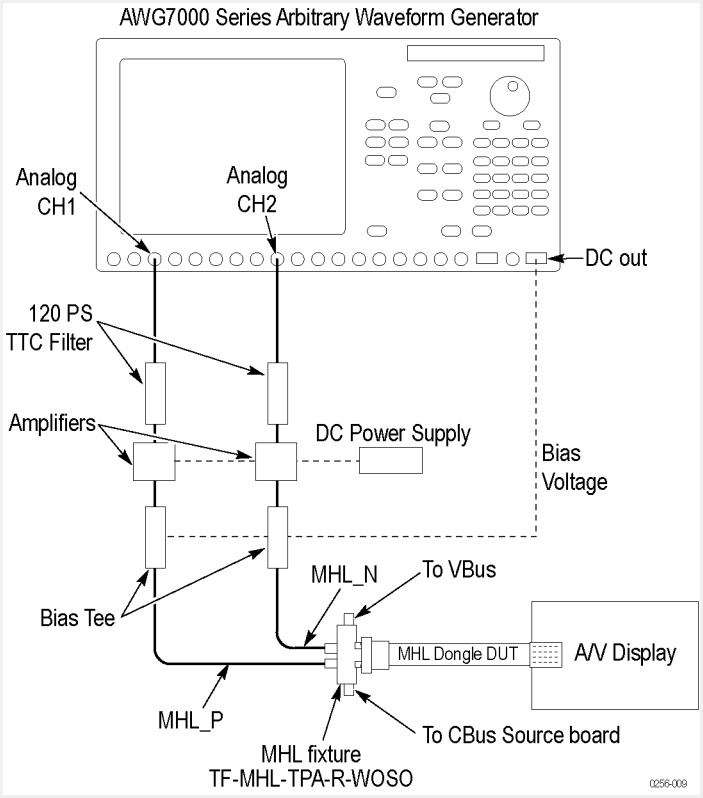
1. Connect the equipment as shown in Single Ended Signals Calibration Setup For Dongle Min Max Test (see page 183) and as shown in Differential and Common-Mode Signals Calibration Setup for Dongle Min Max Test (see page 184). Set V_{TERM} = 3.3V, AWG Ch1 transmitting MHL+ and AWG Ch2 transmitting MHL- signals. Ensure the Bias Tees are used at the output of the AWG channels along with 120ps filters.
2. Calibrate the MHL Signal Generator AWG7122C output for the following settings.
   a. Frequency: Highest frequency supported by the Dongle DUT
   b. Pattern: MHL Gray Ramp data streams
   c. Rise/fall times:
      - Less than or equal to 2.25Gbps

        Differential signal 200ps (20% - 80%), Common-mode signal 600ps (20% - 80%)
      - Less than or equal to 2.25Gbps

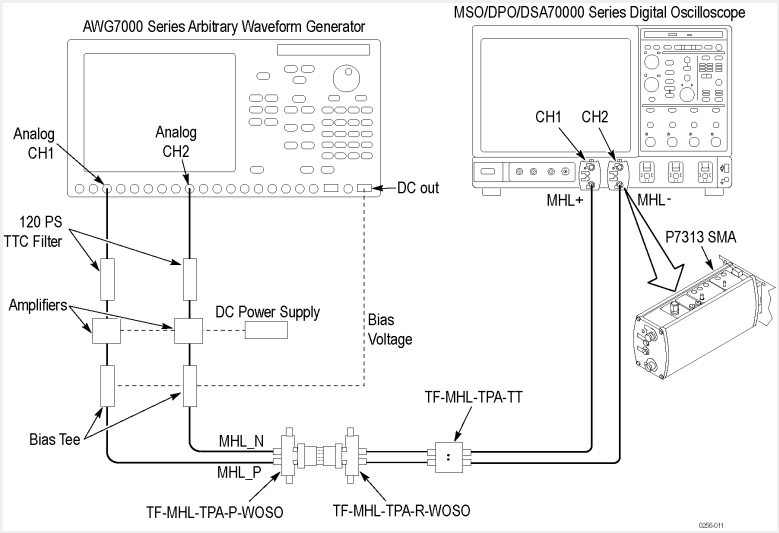
        Differential signal 200ps (20% - 80%), Common-mode signal 600ps (20% - 80%)

3. Adjust the DC level and swing amplitudes of the MHL Signal Generator AWG7122C so that $V_{SE\_HIGH} = V_{TERM} + 10$ mV, $V_{SFSWING} = 600$ mV and $V_{CMSWING} = 360$ mV. Record the DC level and swing amplitude settings.
4. Adjust the DC level and swing amplitudes of the MHL Signal Generator AWG7122C so that $V_{SE\_LOW} = V_{TERM} - 1760$ mV, $V_{DFSWING} = 1000$ mV and $V_{CMSWING} = 720$ mV. Record the DC level and swing amplitude settings.
5. Connect the equipment as shown in Single-Ended Signals Calibration Setup for Dongle Tests (see page 180) with the CBus Source board connected to the Tektronix fixture CBus port. Provide VBus if DUT is unpowered dongle. Do not provide VBus if DUT is powered dongle.
6. Turn on the DUT and enable the MHL mode through Device Discovery and CBUS Information Exchange with the CBUS Source board.
7. Adjust the DC level of the MHL Signal Generator so that $V_{SE\_HIGH} = V_{TERM} + 10$ mV, $V_{DFSWING} = 600$ mV and $V_{CMSWING} = 360$ mV.
8. Examine the output of the Dongle DUT on the Display.
   - Record errors on the display, if any, for the given V_{TERM}, frequency and single-ended voltage.
9. Adjust the DC level and swing amplitudes of the MHL Signal Generator so that $V_{SE\_LOW} = V_{TERM} - 1760$ mV, $V_{DFSWING} = 1000$ mV and $V_{CMSWING} = 720$ mV.
10. Examine the output of the Dongle DUT on the Display.
    - Record errors on the display if any for the given V_{TERM}, frequency and single-ended voltage.
11. If there is no error on the display in all recorded results, then PASS. Otherwise FAIL.

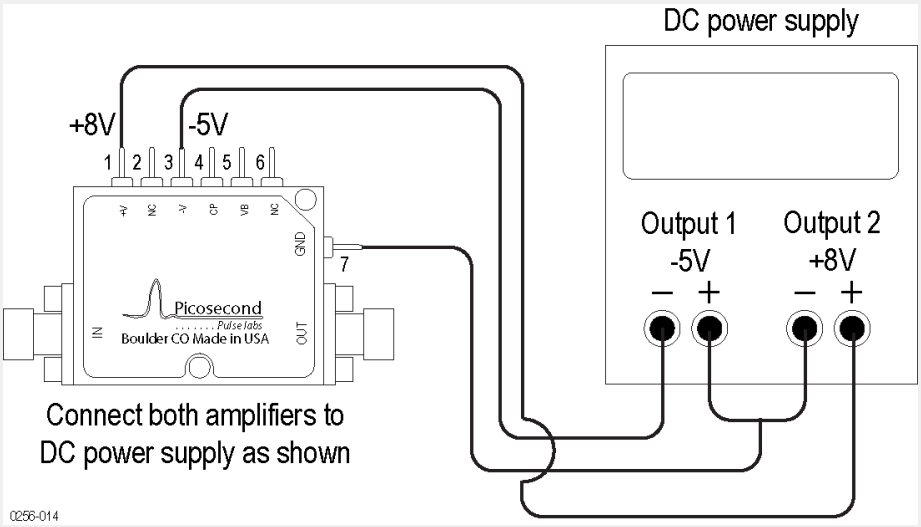
Dongle Min-Max Setup



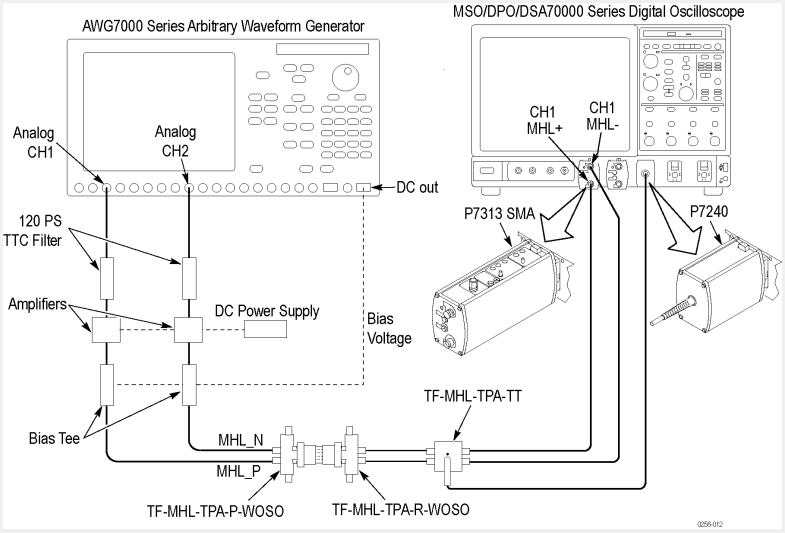
Single Ended Signals Calibration Setup For Dongle Min Max Test



Amplifier Connections to Power Supply



Differential and Common-Mode Signals Calibration Setup for Dongle Min Max Test



5.1.1.3: Intra-Pair Skew Tolerance Test

This MHL Receiver test confirms that the dongle device can tolerate the maximum intra-pair skew allowed by the specification.

Required Test Equipment

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

- MHL Signal Generator: AWG7122C with option 01, 02 or 06 and 08 and Option SDX with Direct Synthesis capability, to simulate Cable emulator effect and Rise time filter effect.
- Rise time Filter: 2 Picosecond Pulse labs 120PS
- Test Fixture: 1 set TF-MHL-TPA-R-WOSO
- DC Power Supply: PWS4205 or PWS2185
- Bias Tees: 2 Mini Circuits Bias Tees ZX85–12G-S+
- CBUS Source Board: Simplay Labs MHL Rx/Tx Analyzer SL-860 (Sink DUT Port) or SOSI Tektronix board
- A/V Display

Make the connection as shown in Dongle – Input Signal Single-Ended Voltage Level Tolerance Test (see page 179) after calibration is done.

Measurement Algorithm

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

1. Connect the equipment as shown in Single-Ended Signals Calibration Setup for Dongle Tests (see page 180) and as shown in Differential & Common-Mode Signals Calibration Setup for Dongle Tests (see page 180). Set V_{TERM} = 3.3V, AWG Ch1 transmitting MHL+ and AWG Ch2 transmitting MHL- signals. Ensure the Bias Tees are used at the output of the AWG channels along with 120ps filters.
2. Calibrate the MHL Signal Generator AWG7122C output for the following settings.
   a. Frequency: Highest frequency supported by the Dongle DUT
   b. Pattern: MHL Gray Ramp data streams
   c. Rise/fall times:
      - Less than or equal to 2.25Gbps

        Differential signal 200ps (20% - 80%), Common-mode signal 600ps (20% - 80%)
      - Less than or equal to 2.25Gbps

        Differential signal 200ps (20% - 80%), Common-mode signal 600ps (20% - 80%)
   d. Swing voltages: Differential signal 800mV, Common-mode signal 540mV
   e. DC level of the output single-ended signal: V_{TERM} – 750mV
   f. Intra-pair skew: Differential skew 0ps, Common-mode skew 0ps

3. Connect the equipment as shown in Dongle – Input Signal Single-Ended Voltage Level Tolerance Test (see page 179) with the CBus Source board connected to the Tektronix fixture CBus port. Provide VBus if DUT is unpowered dongle. Do not provide VBus if DUT is powered dongle.
4. Turn on the DUT and enable the MHL mode through Device Discovery and CBUS Information Exchange with the CBUS Source board.
5. Examine the output of the Dongle DUT on the Display.
6. Confirm that there is no error on the display. If there is any error on the display, STOP. The test FAILed.
7. Increase the intra-pair skew of differential signal and common-mode signal by the same amount to the maximum where error appears on the display. Make the positive channel leads the negative channel in time, which is called positive skew direction.
8. Record the maximum intra-pair skew for the given V_{TERM}, frequency and skew direction.
9. Set the differential and common-mode intra-pair skews to 0ps.
10. Increase the intra-pair skew of differential signal and common-mode signal by the same amount to the maximum where error appears on the display. Make the positive channel lags the negative channel in time, which is called negative skew direction.
11. Record the maximum intra-pair skew for the given V_{TERM}, frequency and skew direction.
12. If the all recorded maximum intra-pair skew value is larger than the spec limit, then PASS. Otherwise FAIL

5.1.1.4: Jitter Tolerance Test (CTS 1.X) – in Normal Mode (CTS 2.0)

This MHL Receiver test confirms that the dongle device can tolerate the maximum clock and data jitter amounts allowed by the specification.

Required Test Equipment

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

- MHL Signal Generator: AWG7122C with option 01, 02 or 06 and 08 and Option SDX with Direct Synthesis capability, to simulate Cable emulator effect and Rise time filter effect.
- Rise time Filter: 2 Picosecond Pulse labs 120PS
- Test Fixture: 1 set TF-MHL-TPA-R-WOSO
- DC Power Supply: PWS4205 or PWS2185
- Bias Tees: 2 Mini Circuits Bias Tees ZX85–12G-S+
- CBUS Source Board: Simplay Labs MHL Rx/Tx Analyzer SL-860 (Sink DUT Port) or SOSI Tektronix board
- A/V Display

Make the connection as shown in Dongle – Input Signal Single-Ended Voltage Level Tolerance Test (see page 179) after calibration is done.

Measurement Algorithm

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

1. Connect the equipment as shown in Single-Ended Signals Calibration Setup for Dongle Tests (see page 180) and as shown in Differential & Common-Mode Signals Calibration Setup for Dongle Tests (see page 180). Set V_{TERM} = 3.3V, AWG Ch1 transmitting MHL+ and AWG Ch2 transmitting MHL- signals. Ensure the Bias Tees are used at the output of the AWG channels along with 120ps filters.
2. Calibrate the MHL Signal Generator AWG7122C output for the following settings.
   a. Frequency: Highest frequency supported by the Dongle DUT
   b. Pattern: MHL Gray Ramp data streams
   c. Rise/fall times:
      - Less than or equal to 2.25Gbps

        Differential signal 200ps (20% - 80%), Common-mode signal 600ps (20% - 80%)
      - Less than or equal to 2.25Gbps

        Differential signal 200ps (20% - 80%), Common-mode signal 600ps (20% - 80%)
   d. Swing voltages: Differential signal 800mV, Common-mode signal 360mV
   e. DC level of the output single-ended signal: V_{TERM} – 540mV
3. Set the clock jitter frequency to 10MHz and the data jitter frequency to 500KHz. Inject clock jitter and data jitter with TP1 maximum amount specified in MHL Spec for the given frequency. 4. Adjust the differential signal swing to the marginal TP1 eye opening specified in MHL Spec for the given frequency.
4. Adjust the differential signal swing to the marginal TP1 eye opening specified in MHL Spec for the given frequency.
5. Connect the equipment as shown in Dongle – Input Signal Single-Ended Voltage Level Tolerance Test (see page 179) with the CBus Source board connected to the Tektronix fixture CBus port. Provide VBus if DUT is unpowered dongle. Do not provide VBus if DUT is powered dongle.
6. Turn on the DUT and enable the MHL mode through Device Discovery and CBUS Information Exchange with the CBUS Source board.
7. Examine the output of the Dongle DUT on the Display.
8. Record error on the Display if any for the given V_{TERM}, clock frequency, jitter frequency and clock edge location.
9. Change the common-mode clock edge location by 0.1Tbit step with respect the differential data edge for 10 times and examine the Display at each step.

10. Record error on the Display if any for the given V_{TERM}, clock frequency, jitter frequency and clock edge location.
11. Repeat the above test for clock jitter frequency 7MHz and data jitter frequency 1MHz.
12. Repeat the above test for the minimum resolution supported by the DUT.
13. If there is no error on the display in all recorded results, then PASS. Otherwise FAIL.

5.1.1.9: Jitter Tolerance Test in Packed Pixel Mode

NOTE. *This test is only available in CTS 2.0.*

This test confirms that the dongle device can tolerate the maximum clock and data jitter amounts allowed by the specification in Packed Pixel Mode.

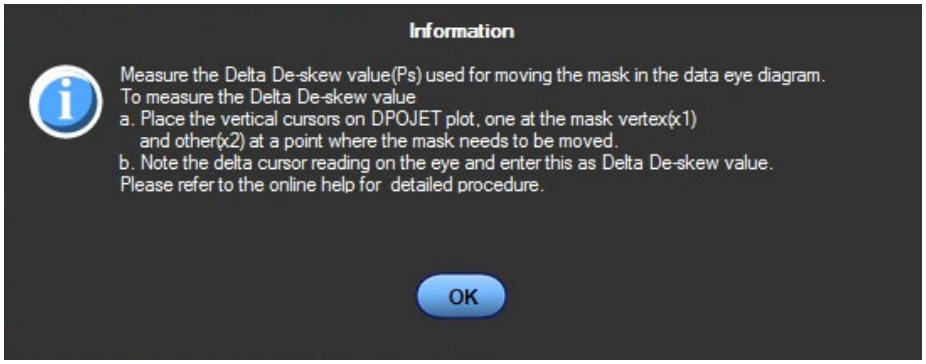
For this test, use the same Required Test Equipment and Measurement Algorithm as

5. 1. 1. 4 Jitter Tolerance Test in Normal Mode (see page 186)

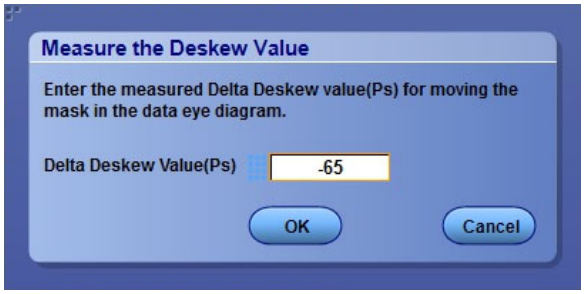
MHL Data Eye Diagram Test - Mask Movement

When performing the MHL Data Eye Diagram test (Transmitter), 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, and 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 tests 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 pop-up dialog box displays. Once you determine how the amount of time (in pico seconds) to shift the mask horizontally, enter the value here. To determine the amount of deskew time, use the procedure Moving the Mask (see page 190).

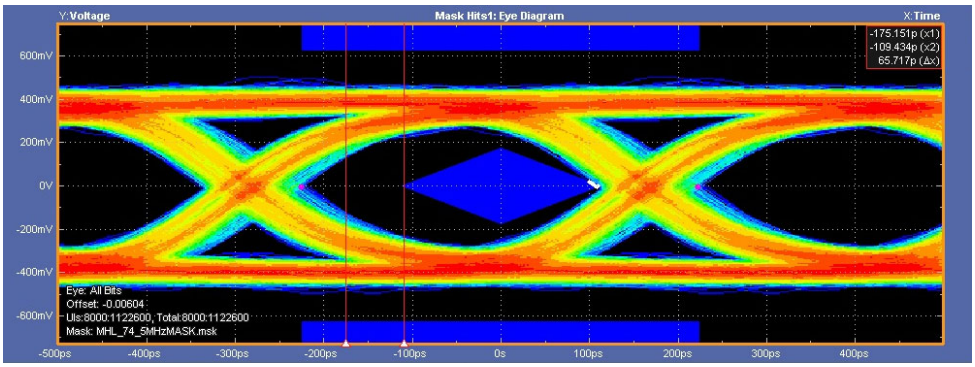


Moving the Mask

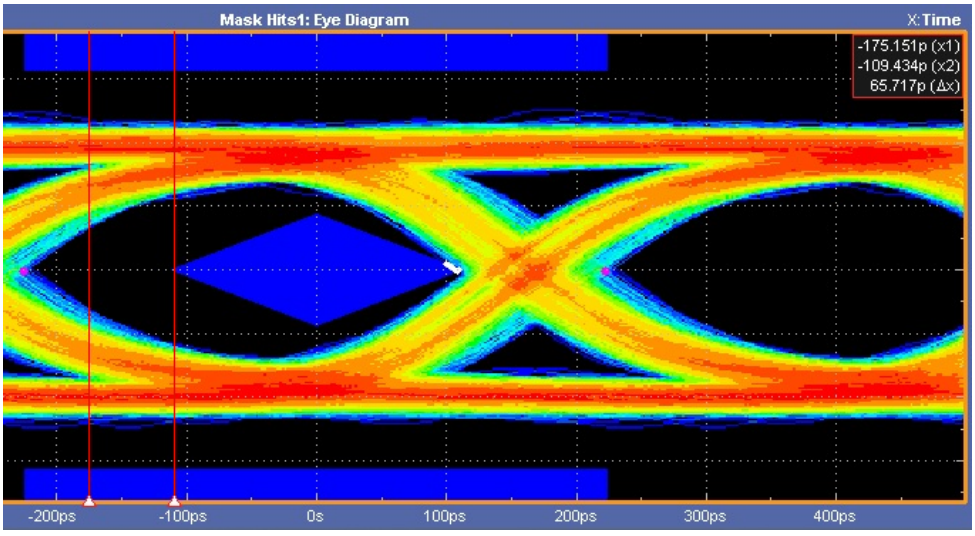
Before you enter a value in the Delta Deskew Value pop-up dialog box, you first need to measure the amount of timing shift required to move the mask horizontally.

1. Observe the eye diagram plot and locate the mask hits.

   In the example, you can see the mask hits. You can move the mask towards the right or left horizontally by entering the positive or negative de-skew values appropriately.

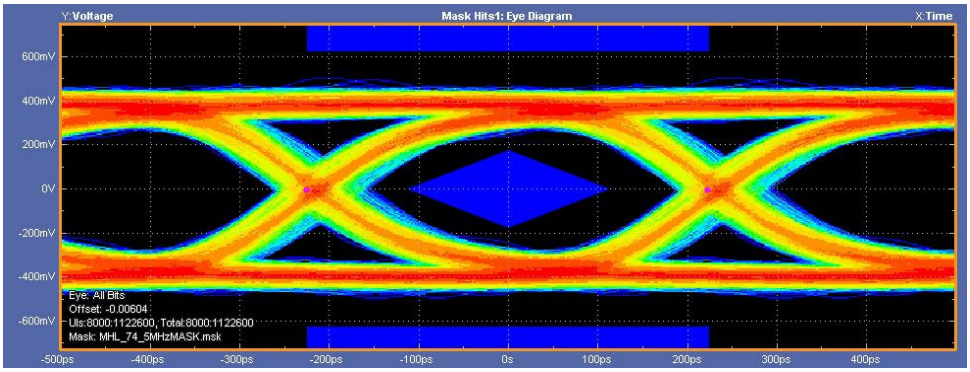


2. Using the delta cursors, measure the amount of time you want to shift the mask. In the example, we've measured that we want to shift the mask -65 ps.



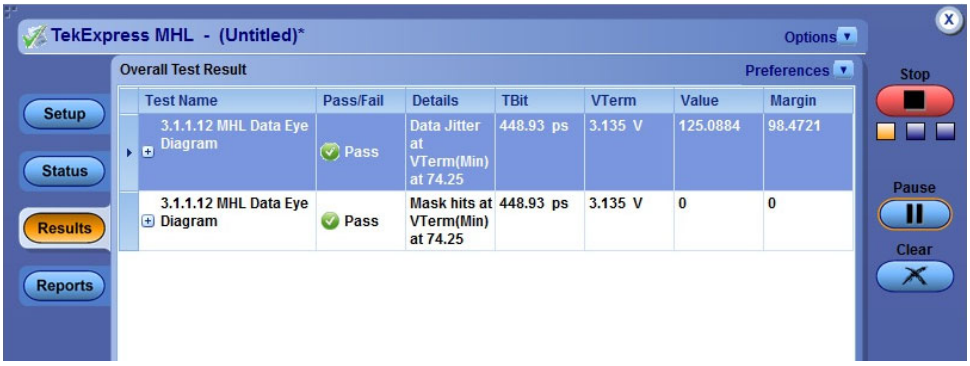
3. Enter the Delta Deskew Value in the pop-up dialog box and click OK. This causes the MHL TekExpress solution to perform the test again.

   Observe that the Eye Diagram is rendered again after shifting the mask. You can see that there are no mask hits.



NOTE. *If the test fails again due to mask hits, you will have two more attempts (three total) to adjust the position of the mask.*

4. Viewing the Results screen shows that the MHL Data Eye Diagram test now passes.



See also

- 3. 1. 1. 12: MHL Data Eye Diagram Test (see page 166)

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

Symbols and Numbers

5.1.1.9, 188

A

Acquire Status, 90
Acquisition source
  selecting live waveforms as, 73
  selecting prerecorded waveforms as, 72
Acquisitions, 67
Activating the license, 34
Advanced View, 52
Algorithms, 153
  Common-mode Output Swing Voltage Test, 155
  Common-mode Rise and Fall Times, 156
  Differential Intra-Pair Skew-TSKEW_DF, 165
  Differential Output Swing Voltage-VDF_SWING, 163
  Differential Rise and Fall Times-TR_DF, TF_DF, 164
  MHL Clock Duty Cycle Test 24-Bit or Packed Pixel Mode, 158
  MHL Clock Jitter Test, 159
  MHL Data Eye Diagram Parameters, 166
  Single-Ended High Level Voltage, 167
  Single-ended Low Level Voltages-VSE_LOW, 161
  Standby Output Voltage Test, 153
Analysis Status, 90
Appending reports, 97
Application controls, 39
Application window, 39
ApplicationStatus(), 146
Auto Scroll, 90

C

CheckSessionSaved(), 151
Clear Log, 90
Clearing test results displayed, 95
Client Proxy Object, 107
Clock Settings, 79
Commands
  ApplicationStatus(), 145
  CheckSessionSaved(), 151
  Connect(), 122
  Disconnect(), 152
  GetAcquireParameter(), 129
  GetAnalyzeParameter(), 129
  GetCurrentStateInfo(), 145
  GetDUTId(), 125
  GetGeneralParameter(), 129
  GetPassFailStatus(), 148
  GetReportParameter(), 148
  GetResultsValue(), 148
  GetResultsValueForSubMeasurements(), 148
  GetTimeOut(), 145
  LockSession(), 123
  Pixel Mode, 138
  QueryStatus(), 145
  RecallSession(), 151
  Run(), 143
  SaveSession(), 151
  SaveSessionAs(), 151
  selecting a channel, 112
  SelectSuite(), 138
  SelectTest(), 131
  SendResponse(), 145
  SetAcquireParameter(), 126
  SetAnalyzeParameter(), 126
  SetDUTId(), 125
  SetGeneralParameter(), 126
  SetTimeOut(), 145
  setting Resolution, 139
  setting Termination Source method, 141
  setting Termination Voltage, 142
  SetVerboseMode(), 124
  Stop(), 143
  TransferImages(), 148
  TransferReport(), 148
  TransferWaveforms(), 148
  UnlockSession(), 152
Commands List, 116
Comments, 52
Common-mode Output Swing Voltage Test algorithm, 155
Common-mode Rise and Fall Times algorithm, 156
Compliance Mode, 74
Compliance View, 52
Configuring tests, 74
Connect(), 122
Connected instruments
  selecting, 74
  viewing, 37
Connection requirements, 18

D

Default directory, 27
Default test setup, 101
Deskew value, 189
Device options, 52
Device Profile, 52
Differential Intra-Pair Skew-TSKEW_DF algorithm, 165
Differential Output Swing Voltage-VDF_SWINGalgorithm, 163
Differential Rise and Fall Times-TR_DF, TF_DF algorithms, 164
Directories, 27
Disconnect(), 152

DSP Filter
8 GHz, 74
DUT settings, 47

E

Edge Trigger Level, 67
Email Notifications, 87
Email Settings, 37
Equipment setup, 14
Exiting the application, 43
Eye diagram test, 189
Eye diagram to mask alignment, 189

F

File name extensions, 29
Fixtures, 31
Flowchart for Client Programmatic Interface, 109

G

GetAcquireParameter(), 129
GetAnalyzeParameter(), 129
GetCurrentStateInfo(), 147
GetDutId(), 125
GetGeneralParameter(), 129
GetPassFailStatus(), 148
GetReportParameter(), 149
GetResultsValue(), 148
GetResultsValueForSubMeasurements(), 148
GetTimeOut(), 145

I

Installing the software, 32
Instrument Control Settings, 37
Instrument Control Settings dialog box, 37
Instruments
searching for, 37
selecting, 74
viewing connected, 37
Interface, 105
Interface error codes, 144

K

Keep On Top, 35

L

License, 35
License activation, 34
Limits Editor, 74
Live waveforms, 73
LockSession(), 123
Log View, 89

M

Mapping My TekExpress folder, 191
Mask alignment, 189
Mask movement, 189
Measurement parameters, 83
Menus, 39
Options, 39
Preferences, 95
Message History, 90
MHL Application Commands, 116
MHL Clock Duty Cycle Test 24-Bit or Packed Pixel Mode algorithm, 158
MHL Clock Jitter Test algorithm, 159
MHL Data Eye Diagram algorithms, 166
MHL data eye diagram to mask alignment, 189
Mode
selecting, 74
My TekExpress folder, 85
mapping, 191

O

Options Menu, 39
Oscilloscope setup, 153
Oscilloscopes supported, 31
Overview, 3

P

Panels
Setup, 47
Parameters, 83
Pixel Mode, 52
Pixel Mode test, 158
Population Count, 67
Pre-run check list, 85
Preferences Menu, 95
Prerecorded waveform files, 72
Probe parameters, 67
Probe setup illustration, 18
Probes, 31
Program example, 112
Programmatic Interface, 105

Q

QueryStatus(), 146

R

RecallSession(), 151
Record Length, 79
Ref Levels, 79
Remote Proxy Object, 107
Report Options, 97
Report sections, 97
Reports
appending, 97
naming, 97
viewing, 97
Resolution controls, 52
Results Panel
using, 95
Run(), 143

S

Sample Rate, 79
SaveSession(), 151
SaveSessionAs(), 151
Saving Log files, 90
Saving reports, 97
Scope settings, 79
SelectSuite(), 138
SelectTest(), 131
SendResponse(), 147
Server, 105
Session files, 28
Session folders, 28
SetDutId(), 125
SetGeneralParameter(), 126

SetTimeOut(), 145
Setup Panel, 47
SetVerboseMode(), 124
Show Detailed Log, 90
Signal Path Compensation (SPC), 87
Single-Ended High Level Voltage algorithm, 180
Single-Ended Low Level Voltages-VSE_LOWalgorithm, 161
Specifications, 3
Standby Output Voltage Test algorithm, 153
Status Panel
using, 89
Stop(), 143
System requirements, 31

T

Technical support, 1
TekExpress Client, 105
TekExpress Client requirements, 105
TekExpress Server, 105
Termination source, 52
Test groups, 62
Test limits
changing, 74
Test Reports
selecting options, 97
viewing, 97
Test results
viewing, 95
Test Results Summary, 95
Test setup files, 28
Test setups, 101
creating based on existing, 102
deleting, 103
recalling, 102
saving, 101
Test Status, 89
Tests
configuring, 74
selecting, 62
TestStand Client example, 116
TransferImages(), 150
TransferReport(), 149
TransferWaveforms(), 150

U

UnlockSession(), 152
Untitled Session folder, 85
User Comments
including in test reports, 97
viewing in reports, 97
User Defined Mode, 78

V

Version, 35
Vertical Scale, 67
Vertical Setup, 79
View option, 52

W

Waveforms
acquiring live, 73
using prerecorded, 72