

**TekExpress USB
Automated Solutions
Online Help**



**TekExpress USB
Automated Solutions
Online Help**

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

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

TEKTRONIX and TEK are registered trademarks of Tektronix, Inc.

TekExpress is a registered trademark of Tektronix, Inc.

TekExpress USB Automated Solutions Online help, 076-0196-01.

Contacting Tektronix

Tektronix, Inc.
14200 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

General Safety Summary	iii
Introduction	
Using Online Help	1
Related Documentation	1
Conventions	2
Technical Support	3
Getting Started	
What is new in this release	5
Accessories	5
Minimum System Requirements	6
Application Directories and Usage	6
File Name Extensions	8
How To Activate the License	8
Before You Click Run	10
Operating Basics	
TekExpress Application Overview	13
Starting the Application	14
Resizing the Application Window	14
Exiting the Application	15
Global Controls	15
Menus	
File Menu	16
View Menu	17
Tools Menu	17
Help Menu	19
How To	
Deskew Real Time Oscilloscopes	21
Select the Test(s)	22
Configure and Run the Test(s)	24
View and Select Connected Instruments	28
Use the Prerecorded Waveform for Analysis	30
View the Progress of Analysis	32
View the Report	34
View Test Related Files	35
Use Filters	36

Using the Nonstandard Filter	36
Using the Nonstandard Waveform Mask	37

Application Examples

Testing a Device Transmitter Using DPOJET Application

Set Up the Equipment	39
Testing Deterministic Jitter on a Device Front Panel	40

TekExpress Programmatic Interface

About the Programmatic Interface	45
Server and Client Proxy Objects	47
Remote Proxy Object	47
Client Proxy Object	48
Programmatic Interface APIs	
Connect()	49
Disconnect()	51
LockSession()	52
UnlockSession()	53
SetTimeOut()	54
GetTimeOut()	55
GetDutId()	56
ChangeDutId()	57
SaveSession()	58
SaveSessionAs()	59
RecallSession()	60
Run()	61
Stop()	62
QueryStatus()	63
TransferReport()	64
CheckSessionSaved()	65
ApplicationStatus()	66
ErrorCodes	67
Example	67

Troubleshooting

Instrument Connectivity	69
TestStand Run time Engine Installation	69

Reference

Shortcut Keys	71
---------------------	----

Index

General Safety Summary

Review the following safety precautions to avoid injury and prevent damage to this product or any products connected to it.

To avoid potential hazards, use this product only as specified.

Only qualified personnel should perform service procedures.

While using this product, you may need to access other parts of a larger system. Read the safety sections of the other component manuals for warnings and cautions related to operating the system.

To Avoid Fire or Personal Injury

Connect and disconnect properly. Connect the probe output to the measurement instrument before connecting the probe to the circuit under test. Connect the probe reference lead to the circuit under test before connecting the probe input. Disconnect the probe input and the probe reference lead from the circuit under test before disconnecting the probe from the measurement instrument.

Observe all terminal ratings. To avoid fire or shock hazard, observe all ratings and markings on the product. Consult the product manual for further ratings information before making connections to the product.

Do not operate without covers. Do not operate this product with covers or panels removed.

Do not operate with suspected failures. If you suspect that there is damage to this product, have it inspected by qualified service personnel.

Avoid exposed circuitry. Do not touch exposed connections and components when power is present.

Terms in This Manual

These terms may appear in this manual:



WARNING. *Warning statements identify conditions or practices that could result in injury or loss of life.*



CAUTION. *Caution statements identify conditions or practices that could result in damage to this product or other property.*

Using Online Help

Select Help from the menu to open the help file. You can also find an electronic copy of the help file in the Documents directory on the 063-4068-XX DVD.

Tables of Contents (TOC) tab — Organizes the Help into book-like sections. Select a book icon to open a section; select any of the topics listed under the book.

Index tab — Enables you to scroll a list of alphabetical keywords. Select the topic of interest to bring up the appropriate help page.

Search tab — Allows a text-based search.

Follow these steps:

1. Type the word or phrase you want to find in the search box. If the word or phrase is not found, try the Index tab.
2. Choose a topic in the lower box, and then select the Display button.

General Help Functions

- Select the Print button from the Help topics menu bar to print a topic.
- To return to the previous window, select the Back button.
- Use hyperlinks to jump from one topic to another.
- If the back button is grayed out or a jump is not available, choose the Help Topics button to return to the originating help folder.

Related Documentation

In addition to TekExpress USB Online Help, the following documentation is included with the software:




- *DPOJET SuperSpeed (USB 3.0) Setup Library Methods of Implementation (MOI) for Verification, Debug and Characterization*, Tektronix part number 077-0266-xx.
- TekExpress USB Online Help (PDF version), Tektronix part number 077-0350-xx.

Conventions

The online help uses the following conventions:

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

Table 1: Icon descriptions

Icon	Meaning
	This icon identifies important information.
	This icon identifies conditions or practices that could result in loss of data.
	This icon identifies additional information that will help you use the application more efficiently.

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.

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 application setup files, log.xml and status messages text file.
- If possible, save the waveform on which you are performing the measurement as a .wfm file.

Forward the information to technical support using one of these methods:

- E-mail – techsupport@tektronix.com
- FAX – (503) 627-5695

What is new in this release

This version of TekExpress USB provides the following feature enhancements:

- The USB transmitter solution support test automation in the TekExpress framework providing customers with a complete solution for USB 3.0 verification, characterization, debug, and compliance.

Accessories

About the Test Fixture

For Host Testing. TF-USB3-A-P (for best signal quality) or for more mechanical flexibility use TF-USB-B-R (with included 13 cm USB 3.0 Cable - Part number 174-5772-00). For precision De-Embed of TF-USB3-A plug fixture, order TF-USB3-AB-KIT (includes Cal Kit).

For Device Testing. TF-USB3-A-R (includes short USB 3.0 Cable).

Supported Probes

The following probes support TekExpress USB application:

- P7313 SMA differential probe
- P7500 Tri-Mode probe

Minimum System Requirements

The minimum system requirements for a PC to run the application are as follows:

Table 2: System requirements

Processor	Pentium 4/M or equivalent processor.
Operating System	Windows XP Service Pack 2.
Memory	512 MB of memory.
Hard Disk	Approximately 2 GB of available hard-disk space for the recommended installation, which includes full TekExpress installation and distributed components.
Drive	DVD drive
Display	Super VGA resolution or higher video adapter (800x600 minimum video resolution for small fonts or 1024x768 minimum video resolution for large fonts). The application is best viewed at 96 dpi display settings ¹ .
Software	<ul style="list-style-type: none"> ■ TekExpress Framework (v1.3.4 or later) installed. ■ DPOJET Jitter and Eye Analysis Tool (v2.5.0.7 or later) with Advanced Jitter and Eye analysis (DJA option) installed. ■ SDLA software for Channel De-Embed for custom filter development (optional). ■ Microsoft Internet Explorer 6.0 SP1 or later. ■ Adobe Reader 6.0 or equivalent software for viewing portable document format (PDF) files.
Other Devices	<ul style="list-style-type: none"> ■ Microsoft compatible mouse or compatible pointing device. ■ Four USB ports (2 USB ports minimum). ■ PCI-GPIB or equivalent interface for instrument connectivity².

¹ If TekExpress is running on an instrument that has a video resolution lower than 800x600 (for example, sampling oscilloscope), it is recommended to connect a secondary monitor and this has to be enabled before launching the application.

² If TekExpress is installed on a Tektronix oscilloscope, the virtual GPIB port cannot be used by TekExpress for communicating with oscilloscope applications. If external devices like USB-GPIB or equivalent are used for instrument connectivity, ensure that the Talker Listener utility is enabled in the DPO/DSA oscilloscope's GPIB menu.

Application Directories and Usage

Related Topics

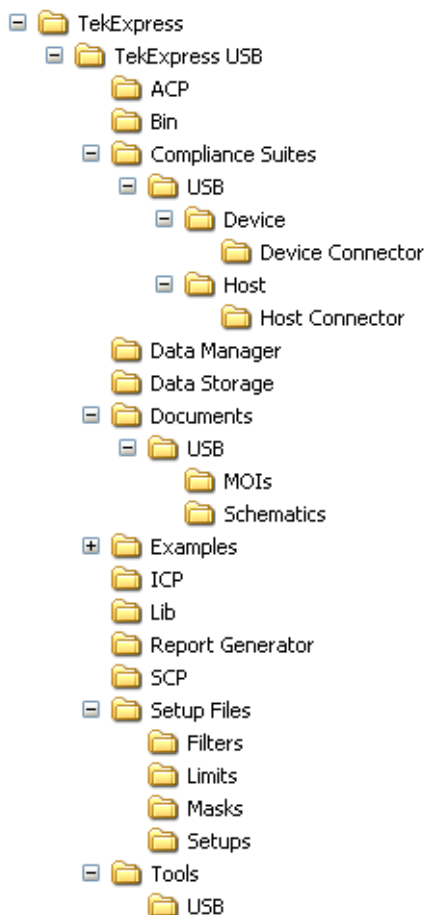
[File Name Extensions](#)

[How To Activate the License](#)

[View Test Related Files](#)

The complete path of the directory structure is C:\Program Files\Tektronix\TekExpress\Tek-Express USB.

The application directory and associated files are organized as follows:



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

Table 3: Default directory names and their usage

Directory names	Usage
InstallDir\TekExpress	Contains the TekExpress application and associated files.
\TekExpress\TekExpress USB	Contains files specific to TekExpress USB.
\TekExpress USB\Compliance Suites	Contains compliance specific sequence files. The folders under this directory represent the devices to be tested.
\TekExpress USB\Compliance Suites\USB	Includes the Device and Host Transmitter folders.
\TekExpress USB\Compliance Suites\USB\Device	Includes the "Device Connector" folder.
\TekExpress USB\Compliance Suites\USB\Device\Device Connector	Contains application specific files for Device.
\TekExpress USB\Compliance Suites\USB\Host	Includes the "Host Connector" folder.

Table 3: Default directory names and their usage (cont.)

Directory names	Usage
\TekExpress USB\Compliance Suites\USB\Host\Host Connector	Contains application specific files for Host.
\TekExpress USB\ACP	Includes instrument and application specific interface libraries of TekExpress.
\TekExpress USB\SCP	
\TekExpress USB\ICP	
\TekExpress USB\Data Manager	Includes the result management specific libraries of TekExpress are present in these folders.
\TekExpress USB\Data Storage	
\TekExpress USB\Report Generator	
\TekExpress USB\Documents	Includes the Method of Implementation documents and technical documentation for the application.
\TekExpress USB\Bin	Includes the miscellaneous libraries of TekExpress.
\TekExpress USB\Lib	
\TekExpress USB\Tools	

File Name Extensions

Related Topics

[Application Directories and Usage](#)

[How To Activate the License](#)

The software uses the following file name extensions:

File name extension	Description
.TekX	The session file will be saved in this format.
.seq	The test sequence file.
.xml	The encrypted XML file that contains the test specific configuration information. The log file extension is also xml.
.mht	This file details the test report.
.PDF	The PDF file that details the method of implementation for the test.
.msk	The mask file.
.flt	The filter file.
.html	The html file.

How To Activate the License

Related Topics

[Application Directories and Usage](#)

[File Name Extensions](#)

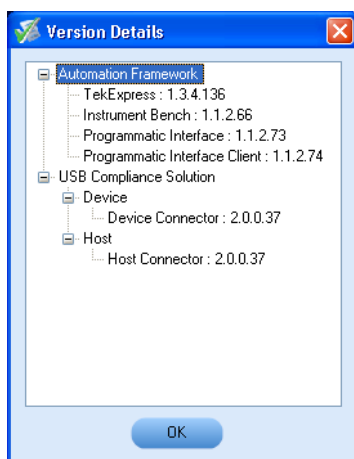
Follow the steps below to activate the license:

NOTE. *Ensure that your TekExpress USB dongle is installed on your host system.*

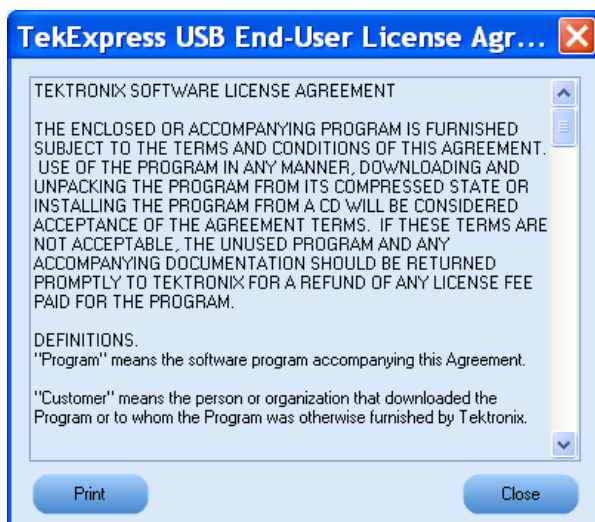
1. Click **Help** > **About** to view the license information.




2. Click the **View Version Details** link to check the version numbers of the installed test suites.

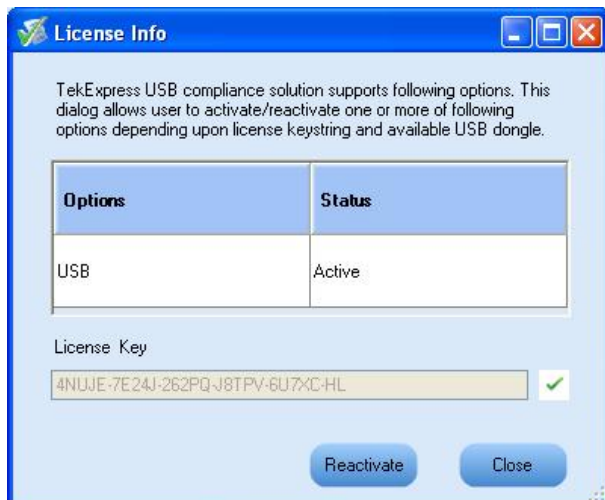


3. Click the **View End-User License Agreement** link to open the following Tektronix Software License Agreement window. Click **Print** to print the License Agreement.



4. Click **License Info** to view the available software options. This window shows the license key and the various options with their status (active or inactive) with the current license key.

- If you are activating the license for the first time, the license key field will be empty. To activate the license, connect the USB dongle to your computer, enter the license key provided in the license key certificate, and click **Activate**. If the activation is successful, a  sign is displayed next to the license key field.



- If you are reactivating the license, click **Reactivate**, enter the new license key and click **Activate**.

Before You Click Run

After you first launch TekExpress, it creates the following folders on your computer:

- \My Documents\My **TekExpress**

NOTE. Ensure that the “My TekExpress” folder has read and write access.

NOTE. If a user with a different Windows login ID launches TekExpress, a new My TekExpress folder is created under that user’s My Documents folder

- \My Documents\My TekExpress\USB.
- \My Documents\My TekExpress\USB\Untitled Session. Every time the USB.exe is launched an Untitled Session folder is created under USB folder. The Untitled Session folder is deleted when you exit TekExpress.



CAUTION. Each session has multiple files associated with it. Do not modify any of the session files and/or folders as this may result in loss of data or corrupted session files.

- The My TekExpress folder is created as a shared folder with share name as <domain><user ID> My TekExpress (or if the PC is not connected to domain then share name is <Computer name><user ID> My TekExpress).
- The above shared folder is mapped as X: (X drive) on to the PC where TekExpress is running.

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

Do the following before you click Run:

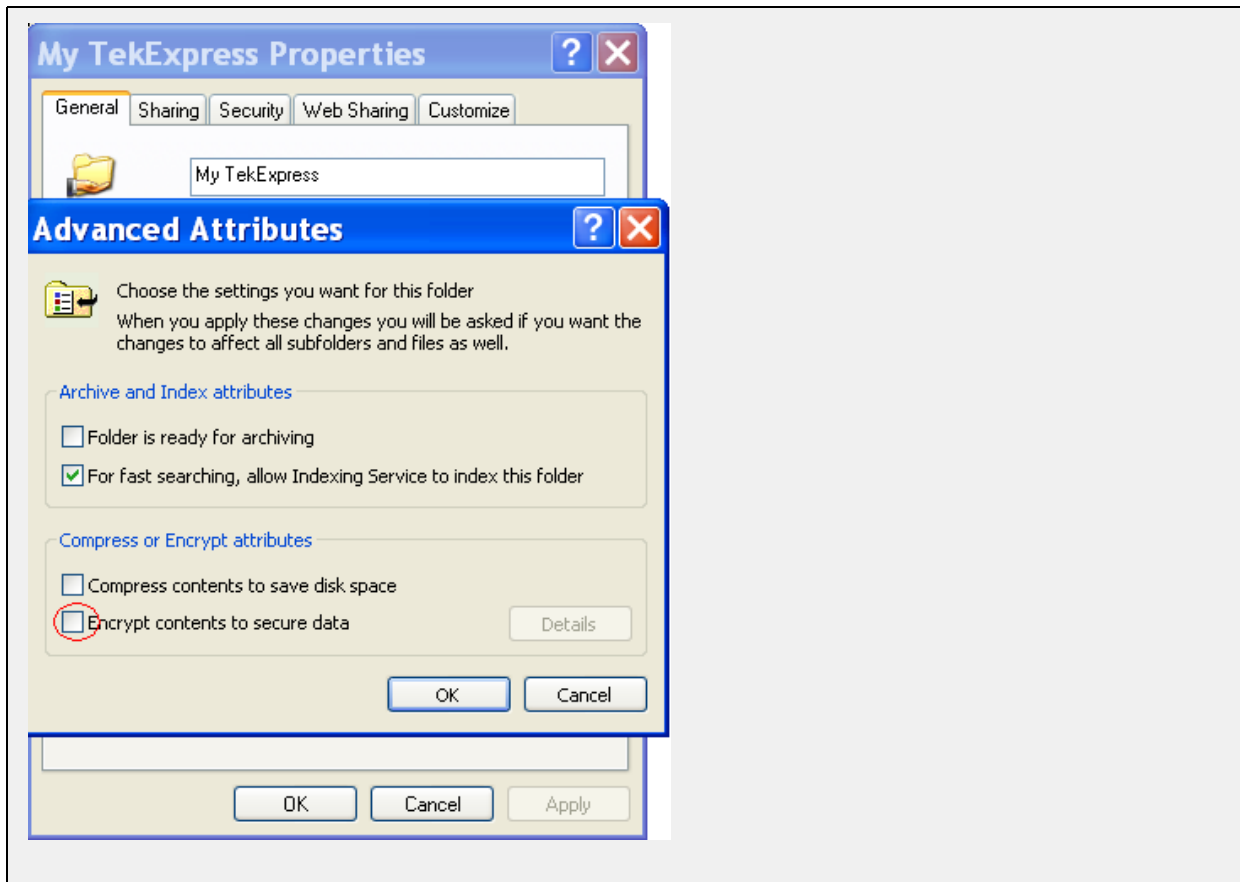
NOTE. *Ensure that the network connectivity is enabled on the PC running the TekExpress.*

1. [Map](#) the shared My TekExpress folder as X: (X drive) on all the instruments used in test setup running Microsoft Windows Operating System. This share folder is used to save the waveform files or any other file transfer operations.
2. Right click on the My TekExpress folder and open the **Properties** dialog box. Select **General** tab and then **Advanced**. In the **Advanced Attributes** window, ensure that the option **Encrypt contents to secure data** is NOT selected. Click [here](#) to view the picture.
3. Ensure that all the required instruments are properly warmed up, and that [Signal Path Compensation \(SPC\)](#) is performed.

Mapping My TekExpress folder

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

1. Open Windows Explorer.
2. From the Windows Explorer menu, select **Tools > Map Network drive**.
3. 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).
4. In the Folder field, enter remote My TekExpress folder path (for example, \\192.158.97.65\My TekExpress)
5. You can determine the IP address of the PC where “My TekExpress” folder exists by doing the following:
 - Select **Start > Run** menu on the PC where My TekExpress folder exists.
 - Enter cmd and click **Enter**.
 - At command prompt, type ipconfig.



You can find SPC by following the steps below:

1. On the oscilloscope main menu, click **Utilities** menu.
2. Click **Instrument Calibration** option.

TekExpress Application Overview

TekExpress is the Tektronix Compliance Test Automation Framework, developed to support current and future test automation needs of customers. Developed using National Instruments' TestStand, TekExpress leverages on the capabilities of Microsoft .NET framework. It is a highly modular architecture that enables deploying automated test solutions for various serial standards in a relatively short time.

The TekExpress USB application (Opt. USB-TX) is the automated version of USB3.0 measurements from DPOJET Timing and Analysis tool. With TekExpress USB, Tektronix provides Fully-Automated Tx solution for verification, characterization, and debug.

Key Features

The following are the key features of TekExpress USB application:


- Comprehensive test coverage
- Precise debugging and troubleshooting
- Accurate and reliable results

Starting the Application

Related Topics

- [Resizing the Application Window](#)
- [Exiting the Application](#)

To start the application, you can do one of the following:

- Click **Start > Programs > Tektronix > TekExpress > TekExpress USB**. Other applications follow similar pattern.
- Double click the icon  on the desktop.
- If you have previously saved a session, you can double-click the session file stored under My TekExpress\USB.

When the application is launched it displays the splash screen providing launch information. The application also checks for the presence and validity of the USB dongle.







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

Resizing the Application Window

Related Topics

- [Starting the Application](#)
- [Exiting the Application](#)

- To minimize the application, click  on the application title bar. To restore the application to its previous size, select  in the Windows task bar.
- To maximize the application, click . To restore it to previous size, click  on the application title bar.


Exiting the Application

Related Topics

[Starting the Application](#)

[Resizing the Application Window](#)

To exit the application, do one of the following:

- Click **File > Exit**.
- Click  on the application title bar.

Global Controls

The menus and controls that appear outside the individual tabs are called “Global Controls”. These are used to specify the devices to be tested.

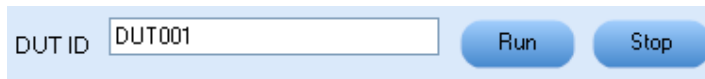



Table 4: Controls and Functions

Control name	Function
DUT ID	The device ID is specified at the global level and the information is stored in the default location for all data files. This field cannot be empty and does not allow these special characters (.,,.,.,.,.,\,/:?*<> *). The maximum length of characters allowed is 32.
	You will be able to run, pause, resume and stop the tests.

File Menu

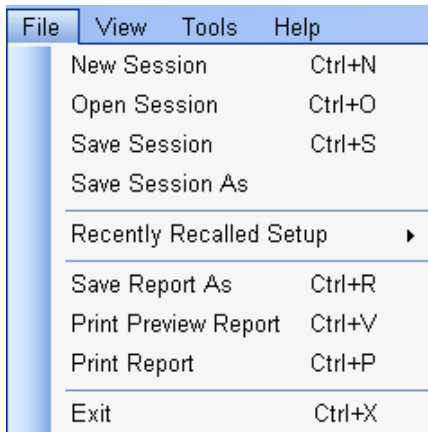
Related Topics

[View Menu](#)

[Tools Menu](#)

[Help Menu](#)

Click **File** on the application menu bar.



The File menu has the following selections:

Menu	Function
New Session	Starts a default session of TekExpress.
Open Session	Opens a saved session.
Save Session	Saves the session.
Save Session As	Saves a session in a different name.
Recently Recalled Setup	Lists all the recent and previously recalled setup files.
Save Report As	Saves the report in user specified location.
Print Preview Report	Previews the report before printing.
Print Report	Opens the Windows "Print" dialog box.
Exit	Closes the application.

View Menu

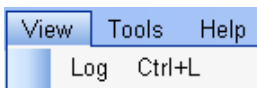
Related Topics

[File Menu](#)

[Tools Menu](#)

[Help Menu](#)

Click **View** on the application menu bar.



The View menu has the following selections:

Menu	Function
Log	Opens the log (log.xml) file in the default viewer.

Tools Menu

Related Topics

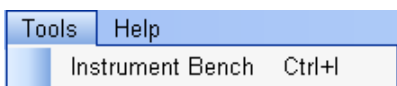
[File Menu](#)

[View Menu](#)

[Help Menu](#)

[Deskewing Real Time Oscilloscopes](#)

Click **Tools** on the application menu bar.



The Tools menu has the below selection:

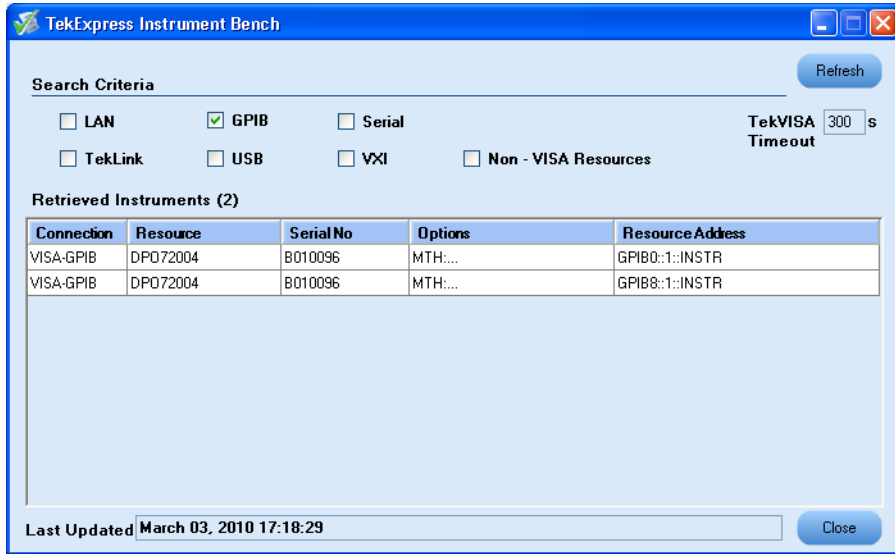
Menu	Function
Instrument Bench	Opens a dialog box showing the list of instruments attached to the test setup.

Tools > Instrument Bench

The Instrument Bench window shows the list of VISA and Non-VISA resources found on different interfaces/connections. It serves two purposes at the launch of TekExpress:

- Discovers the connected instruments.
- Confirms the instrument connection setup.

When you click **Tools > Instrument Bench**, the following dialog box is displayed:



- **Search Criteria:** The various connections on which you can search. **Non-VISA Resources** are the instruments that cannot be searched using TekVISA.
- **Retrieved Instruments:** Displays the count and details of instruments that were discovered.
- **Last Updated:** Displays the time when the last time search was performed.
- **TekVISA Refresh Timeout (Seconds):** This time out specifies the maximum time that TekExpress can wait for TekVISA update.

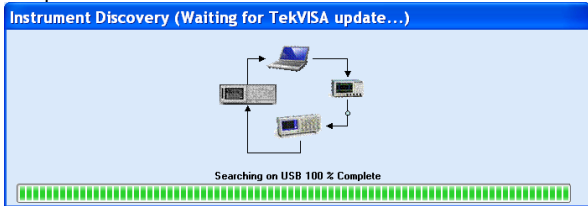
NOTE. TekExpress uses TekVISA for instrument search. Ensure that TekVISA is running on your system before you refresh the instrument bench window.

Table 5: Retrieved Resources properties in the Instrument Bench window

Title	Description
Connection	Shows the type of connection with the instrument.
Resource	Shows the name of the resource.
Serial Number	Shows the serial number of the resource.
Options	Shows the options available on the instrument. ¹
Resource Address	Shows IP Address/Port number of the resource.

¹ The options column displays the options that fit in the field. To view complete options on the instrument, move the mouse cursor over the option.

Table 6: Button controls on Instrument Bench dialog box

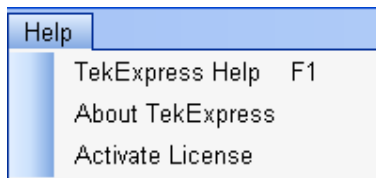
Button	Function
Refresh	The application searches on the selected connection for resources. While searching resources it shows the Instrument Bench discovery window. The Instrument Discovery window shows the connection currently being scanned and the percentage of task completed.
	
Close	Closes the dialog box.

Help Menu

Related Topics

- [File Menu](#)
- [View Menu](#)
- [Tools Menu](#)

Click **Help** on the application menu bar.



The Help menu has the following selections:

Selection	Function
TekExpress Help	Displays TekExpress Help (F1).
About TekExpress	Displays application details such as software name, version number and copyright.
Activate License	Displays available software options and also about license activation.

Deskew Real Time Oscilloscopes

The following procedure is used for deskewing direct input SMA channels on a real time oscilloscope.

NOTE. *If you are using DPOJET, it has an automatic deskew option under **Analyze > Jitter And Eye Analysis > Deskew**. Refer to your DPOJET online help on how to deskew the channels.*

1. Run Signal Path Compensation (SPC) on the oscilloscope.
2. Connect a SMA Power Splitter (preferred) or SMA 50 ohm coaxial “T” connector to the Fast Edge output of the oscilloscope.
3. Connect SMA cables from each of the two channels to be deskewed to the power splitter (or SMA coaxial “T” connector). It is best to use matched cables when making high speed serial measurements. **It is important to use the same cables that will be used for subsequent measurements.**
4. Select **Default Setup**, and then select **Autoset** on the oscilloscope front panel.
5. Set the oscilloscope for 70% to 90% full screen amplitude on both channels. Center both traces so that they overlap.
6. Make sure that volts/div, position, and offset are identical for the two channels being deskewed.
7. Set the time/div to approximately 100 ps/div or less, with sample rate at 1 ps/pt. These settings are not critical, but should be close.
8. Set the horizontal acquisition mode to average, which provides a more stable display.
9. Select **Deskew** from the **Vertical** menu.
10. Verify that the reference channel (typically CH1 or CH2) is set to 0 ps deskew.
11. In the deskew control window, select the channel to deskew (typically CH3 or CH4). Adjust the deskew to overlay the rising edge as best as possible.

NOTE. *Typical values are in the 10's of ps or less with cables connected directly from Fast Edge to SMA inputs. If you are using a switch box (for example, Keithley), deskew the complete path from where the test fixture connects, through the switch, and into the oscilloscope. Deskew values in these cases may be as much as 30 ps or more.*

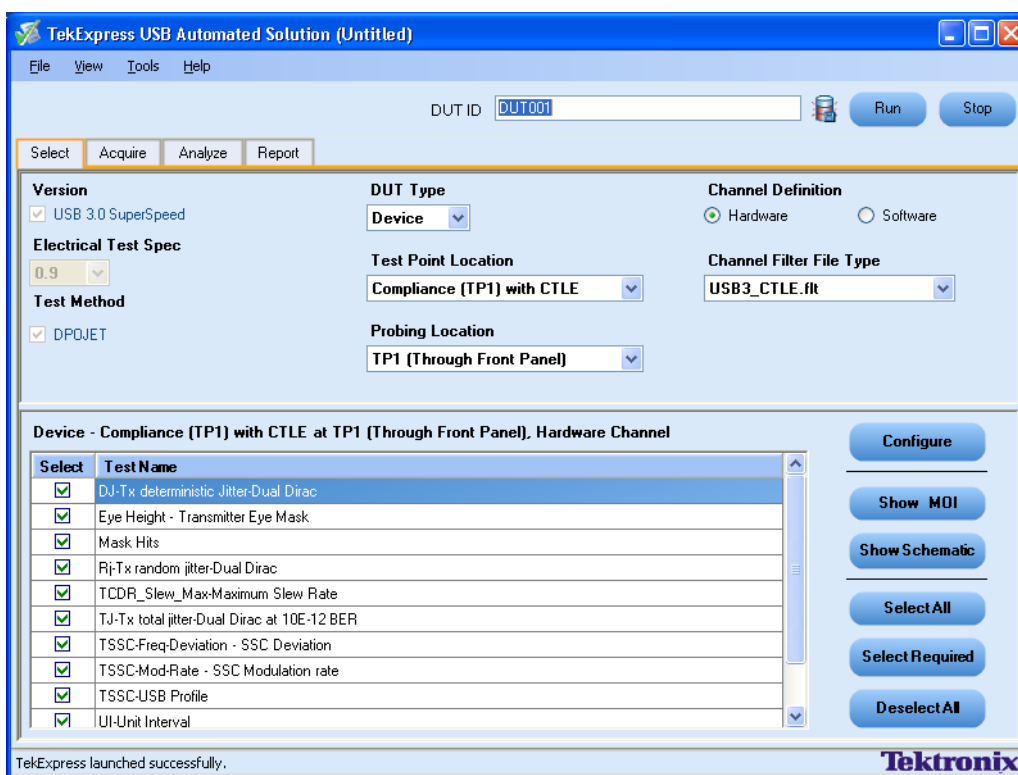
There are sometimes significant differences in the skew between two TCA-SMA adapters. If you find that a system requires a very large correction, it might be better to find a pair of TCA-SMA adapters that match each other better.

Select the Test(s)

Related Topics

- [View and Select Connected Instruments](#)
- [Configure and Run the Tests](#)
- [View the Progress of Analysis](#)
- [View the Report](#)
- [View Test Related Files](#)

The application tests USB transmitter devices for compliance. The Select panel allows you to select tests to configure and run.



This panel provides the following functions:

Version

USB 3.0 SuperSpeed is the default version selected.

Electrical Test Spec

0.9 is the default value for electrical test specification.

Test Method

By default, DPOJET is selected for running the selected measurements.

Select DUT type

Select the Device or Host as the device to be tested.

Test Point Selection

You can select the appropriate test point location. For example, Compliance[TP1] with CTLE, Connector and so on.

Probing Location

You can select the appropriate probing location from where the DUT is tested. In the example screen shown above, the device DUT is being tested with a test fixture at the Device Connector (Near End connection) and the Channel Configuration being emulated is the Host's Back Panel or Front Panel.

Channel Definition

Select Hardware or Software option.

Channel Filter File Type

Select the filter file from the drop-down menu. Use can also choose your own filter file when Custom Filter File is selected.

The table has the following two columns:

- **Select:** You can include or exclude any test for analysis.
- **Test Name:** Displays the name of the test.

NOTE. *If any of the check boxes in the Select column is grayed, you cannot make any changes. It implies that the test is mandatory.*

Once you select a row, the following options are available:

Table 7: Button controls on the Select panel








Button	Description
	Opens the configuration panel for the selected test.
	Opens the PDF of method of implementation (MOI) for the selected test.

Table 7: Button controls on the Select panel (cont.)

Button	Description
	Opens the schematic for the selected test. This is useful if you want to verify the test setup before running the test.
	Selects all tests in the table.
	Selects all the required tests in the grid and deselects all informative tests in the table.
	Deselects all tests in the table.
	Indicates system on low disk space.

Configure and Run the Test(s)

Related Topics

[Select the Tests](#)

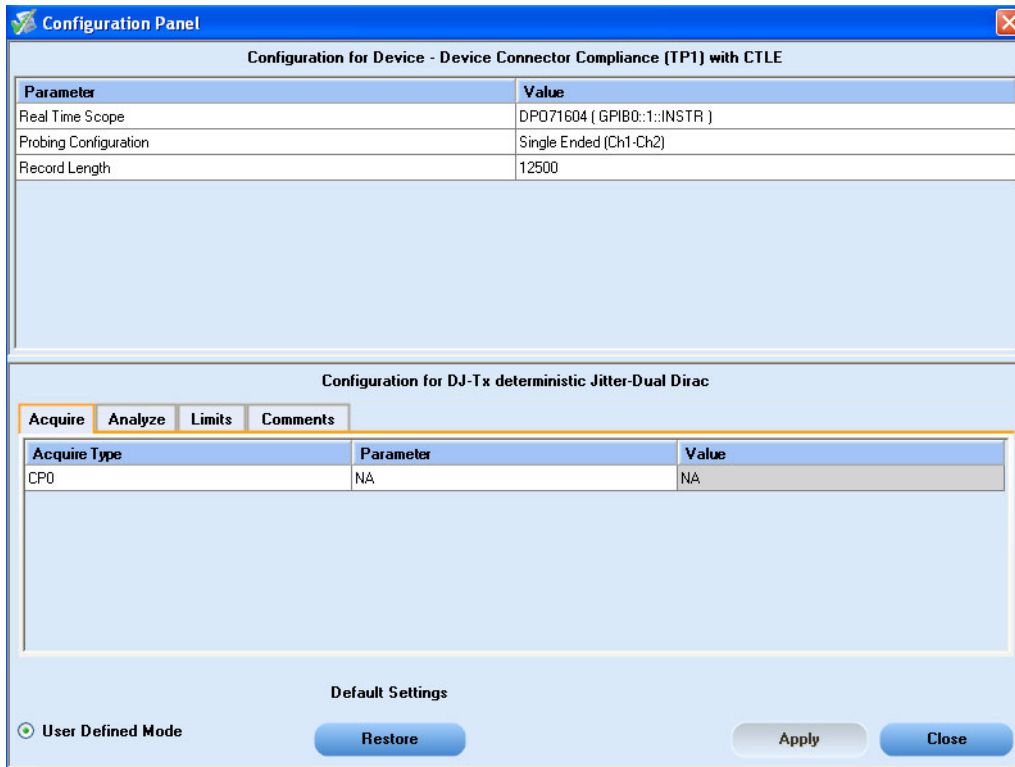
[View and Select Connected Instruments](#)

[View the Progress of Analysis](#)

[View the Report](#)

[View Test Related Files](#)

The configuration panel is used to create, view, and edit the parameters associated with the acquisition and the analysis of the selected test. In the Select panel, click **Configure**.



In the Configure panel, you have the following options:

- To change the parameters associated with the configuration of acquisition.
- To change the parameters associated with analysis configuration.

NOTE. You can select different filter files under Filter Selection option for different test suites.

The upper part of the Configure panel has general parameters that are common for all the tests under the selected test suite that are editable.

- Select the appropriate instrument.
- Select the [Probing Configuration](#).

The lower part of the Configure panel has test specific parameters.

NOTE. If any of the test parameters are grayed, it means that these parameters cannot be modified in the compliance mode. When you switch to the user defined mode, these parameters are editable.

Table 8: Test parameters

Parameters to configure

Description

Acquire

Shows the various parameters related to acquisition of a selected test. These parameters can vary from one test to another.

Acquire Type	Parameter	Value
CP0	NA	NA

Analyze

Shows the various parameters related to analysis of a selected test. The Analyze parameters shows the default analysis parameters for the selected test.

Refer to:

[Using NonStandard Waveform Mask](#)

Parameter	Value
MaskFile path	USB_3_0_Fix_Normative.msk

Limits

Applies to a specific test. It shows the measurement limits using different types of comparisons.

Details	Value1	Compare String	Value2
DJ	86e-12	Less Than Or Equal To (<=)	0

Comments

You can specify a comment up to 256 characters long for the selected test.

Acquire	Analyze	Limits	Comments

Compliance settings

Restore

Restores the default settings.

Apply

Accepts all changes that you made.

Close

Dismisses the dialog box and does not apply changes.

Table 8: Test parameters (cont.)

Click **Run** in the Select panel to run the selected tests.

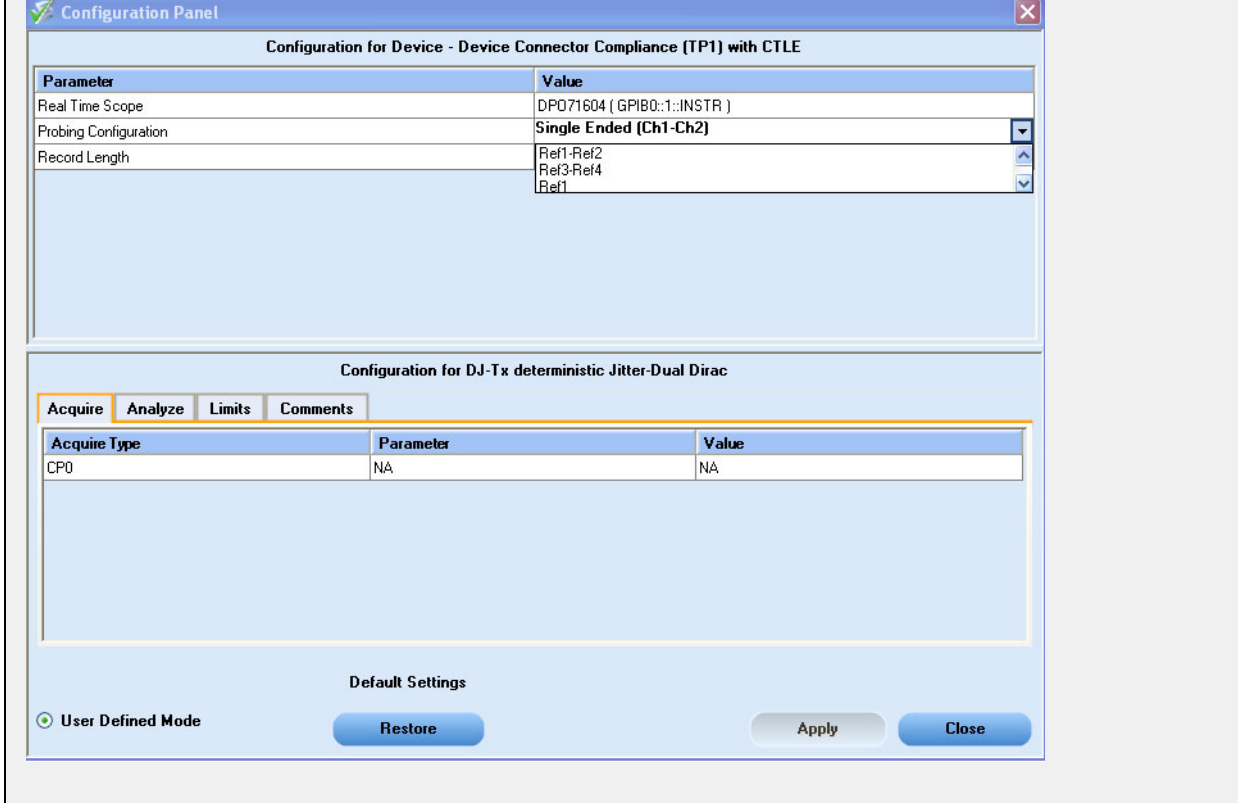
Refer to the following table for different test limit comparisons:

Table 9: Different test limit comparisons

Comparison string	Description
EQ(==)	Equal to
NE(!=)	Not equal to
GT(>)	Greater than
LT(<)	Less than
GE(>=)	Greater than or Equal to
LE(<=)	Less than or Equal to
GTLT(><)	Greater than and Less than
GELT(>=<)	Greater than or equal to and Less than
GTLE(><=)	Greater than and Less or equal to
LTGT(<>)	Less than and Greater than
LEGE(<= >=)	Less than or equal to and Greater than or equal to
LEGT(<= >)	Less than or equal to and Greater than
LTGE(< >=)	Less than and Greater than or equal to
GELE(>= =<)	Greater than or equal to and Less than or equal to

Using Reference Waveforms

If you want to use an existing reference waveform from your storage location for analysis and run, click Probing Configuration field to select the different waveforms for example, ref1, or ref1-ref2 and so on from the drop-down list.



View and Select Connected Instruments

Related Topics

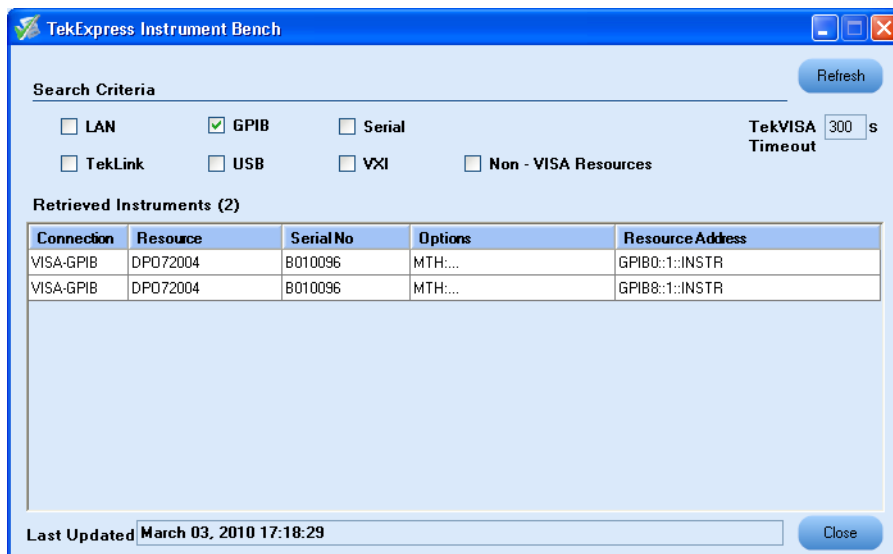
- [Select the Tests](#)
- [Configure and Run the Tests](#)
- [View Test Related Files](#)
- [View the Progress of Analysis](#)
- [View the Report](#)

Viewing Connected Instruments

The **Tools > Instrument Bench** menu item is used to discover connected instruments required for the tests. The application uses TekVISA to discover the connected instruments. Once the search is done, the Instrument Bench dialog box resumes operation and lists the instrument-related details based on the selected search criteria.

NOTE. When the TekVISA Instrument Manager checks for connected instruments, the Instrument Bench dialog box does not respond.

For example, if you select LAN and GPIB as the search criteria in the Instrument Bench dialog box and click Refresh, the TekVISA Instrument Manager checks for the instruments available over LAN and GPIB and the details of the instrument are displayed in the **Retrieved Instruments** table.

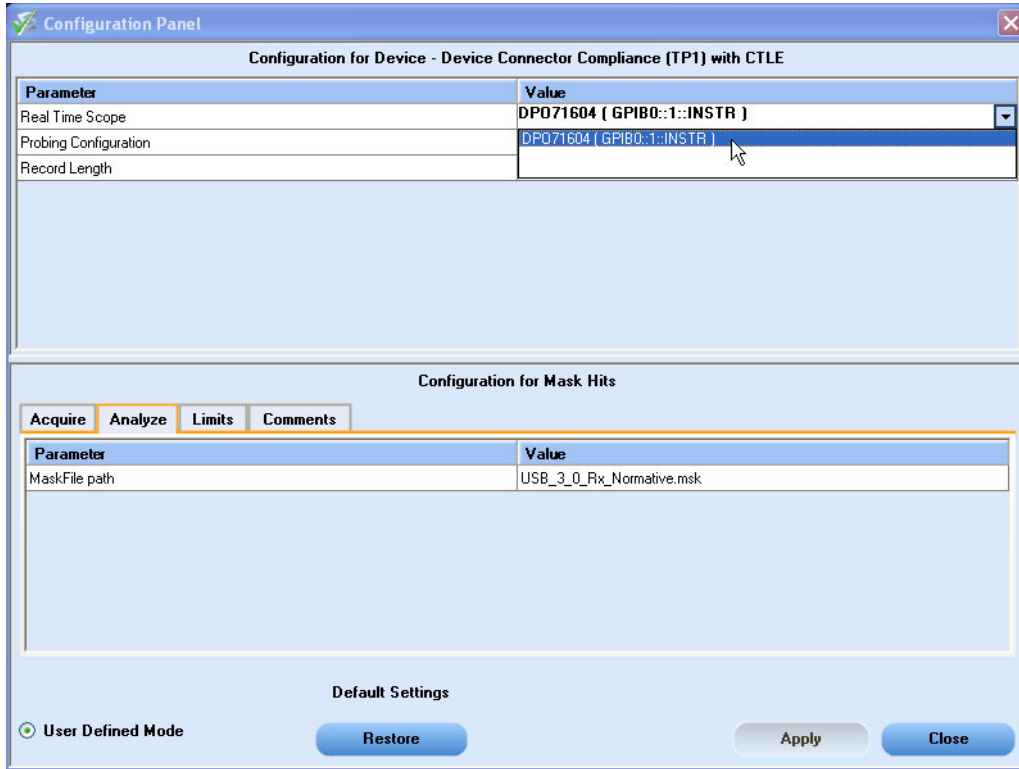


You can provide the time in the **TekVISA Timeout (Seconds)** field, within which if the TekVISA Instrument Manager does not find the instruments, the TekExpress application resumes operation.

If you choose Non-VISA resources, all the instruments supported by TekExpress but not communicating over the VISA layer can be searched.

Selecting Connected Instruments

You can view the instruments connected in the Configuration panel. The upper part of the panel displays the general parameters for the tests under the selected test suite.

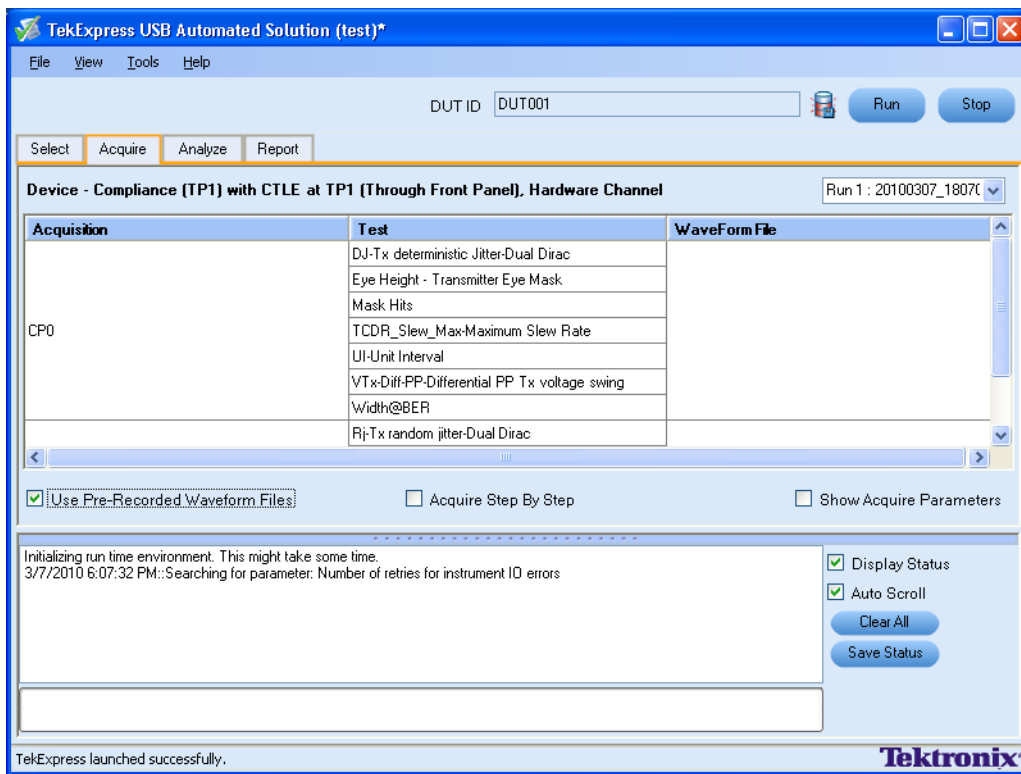


NOTE. The list of instruments displayed is specific to the selected test suite. It does not show all the connected instruments.

Use the Prerecorded Waveform for Analysis

From the Acquire tab, you can see unique acquisitions, acquisition parameters, acquisition status, and prerecorded waveform files of the selected test suite. The Acquire panel is specific to a suite and gets updated every time the selected test suite is changed. This panel shows the acquisition details for the tests in the currently selected suite. The tests with the common acquisition parameters are grouped together and shown as a single acquisition.

Column name	Function
Status	Test acquisition status of the running test passed at intervals.
Test	Name of the tests performing acquisitions. One or more test can perform the same acquisitions.
Acquisition	Acquisition name
Waveform File(s)	Prerecorded waveform files of unique acquisitions. You can select waveform files by selecting browse on individual cells. This allows you to select any waveform file using the standard file open window.



The following Acquire source options are available:

- **Use Pre-Recorded Waveform files:** Enabling or disabling the option shows or hides the waveform file column in the acquisition table. When you save a session and then select this option, the DUT ID text box changes to a drop-down list, in which you can select the DUT ID up to the point where the session was saved. A drop-down box appears above the Waveform file column, showing the run details, including the date and time of each run. If you select a run, corresponding data for that run will be populated (such as the selected test, the test configuration settings, and the test summary status).
- **Acquire Step by Step:** Selecting this option prompts you at the end of each acquisition before proceeding to the next one.
- **Show Acquire Parameters:** On enabling this option, the acquisition parameters for each unique acquisition are displayed in the acquisition table.

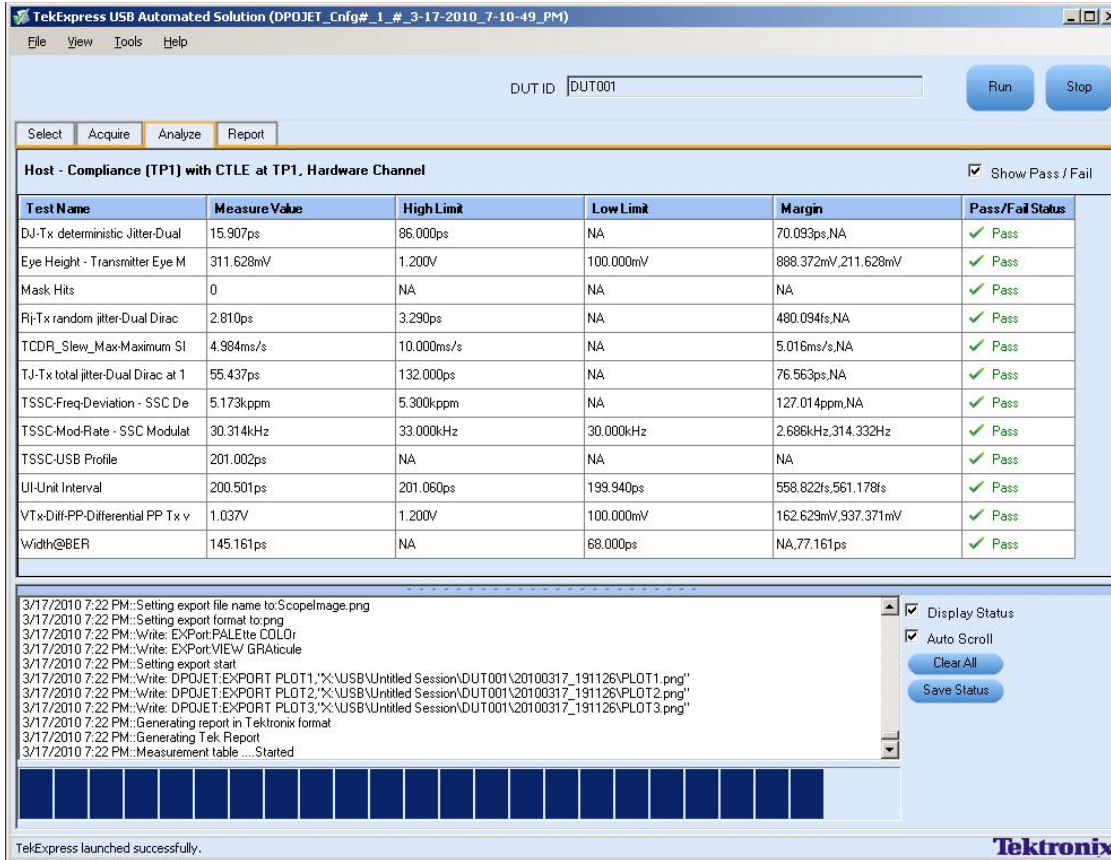
When you select “Use Pre-Recorded Waveform Files”, the first column shows the waveform type and the second column shows the tests that use that waveform type for analysis.

View the Progress of Analysis

Related Topics

- [Select the Tests](#)
- [View and Select Connected Instruments](#)
- [Configure and Run the Tests](#)
- [View the Report](#)
- [View Test Related Files](#)

You can view the progress of the analysis in the Analyze panel. As the analysis of each test is complete, the result value is updated.



Analysis Table

The table contains the following:

- The test name.
- The status of the tests that are being run.
- The measured, limit and margin values of the tests.

The tests that are not yet started are shown with a “To be Started” status. A summarized status of the currently running test is shown on the Status Messages panel.

The **Status Messages** window timestamps all runtime messages and displays them. You can do the following:

- **Display Status:** Enable/Disable status messages.
- **Auto Scroll:** Scrolls status messages automatically.
- **Clear All:** Clear all status messages in Status Window.
- **Save Status:** Save all status messages in text file. Displays a standard save file window and saves the status messages in the user specified file.

NOTE. *The Status Messages window is dockable and can be resized.*

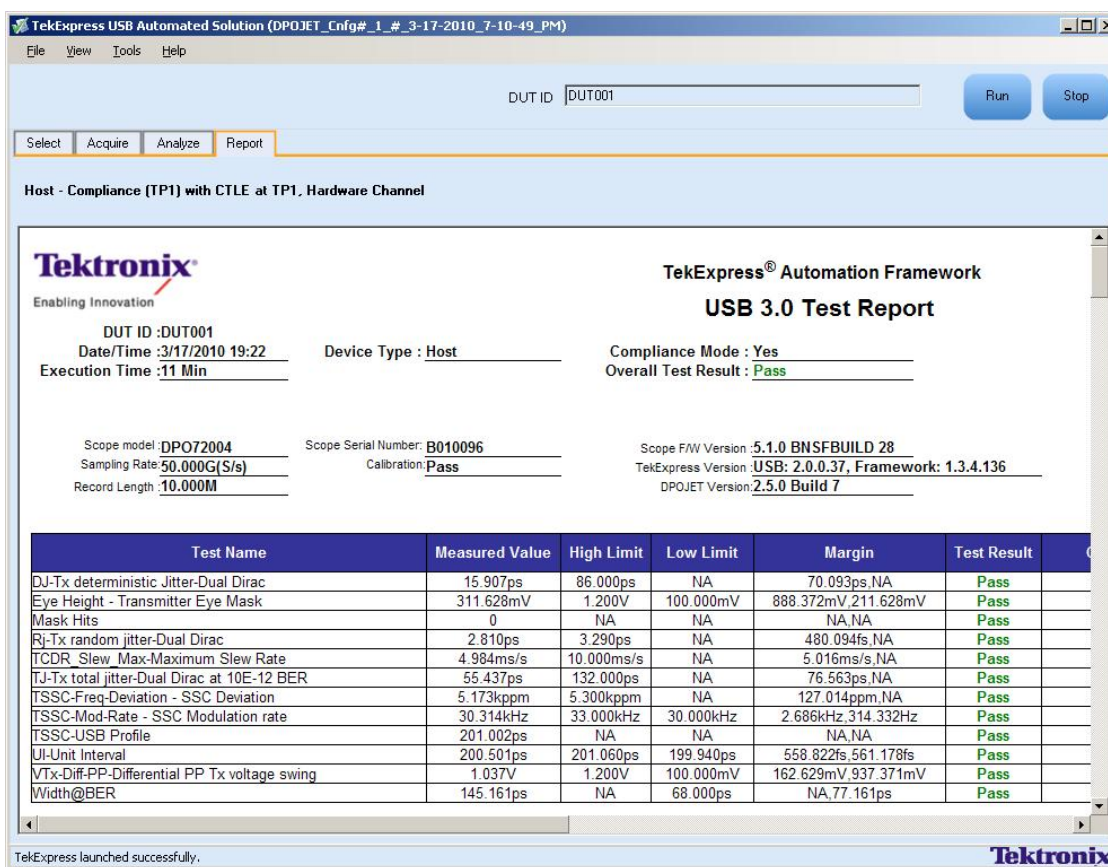
View the Report

Related Topics

- [Select the Tests](#)
- [View and Select Connected Instruments](#)
- [Configure and Run the Tests](#)
- [View the Progress of Analysis](#)
- [View Test Related Files](#)

After the analysis, a report is automatically generated and displayed in the report panel. The report shows the results of the tests, including device information and pass/fail status of each test. The following screen is an example report of a test run using only DPOJET software.

The Report View Area contains an mht report. You can select any area of the report and copy it to the clipboard to make it available to other application.



View Test Related Files

Related Topics

[Application Directories and Usage](#)

All the test related files for currently selected tests are always saved under My Documents\My TekExpress\USB\Untitled Session.

When you save a session, it is saved with the session name under the path My Documents\My TekExpress\USB\SessionName for future references.

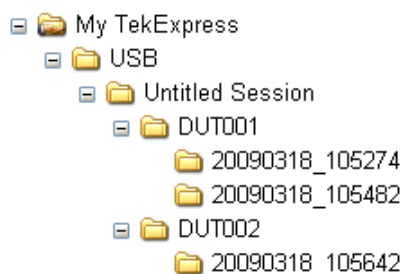
The session that is currently running will be stored in the same path as “Untitled Session” until you save it.



WARNING. Do not save a session named “Untitled Session” or “Backup” because there are application-specific files and are deleted when you exit the application.

A session folder can contain results for more than one DUT, and a DUT folder can contain more than one run data folder marked by date-time stamp as folder name.

Here is an example image of data storage:

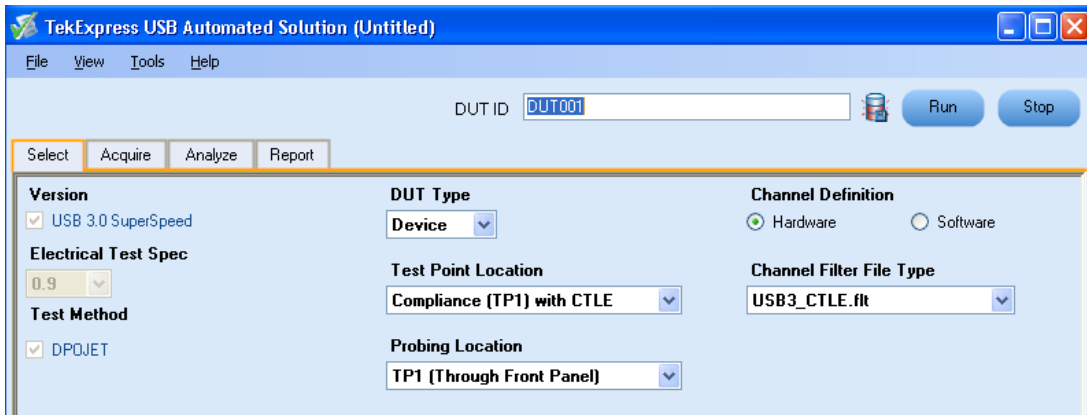


Use Filters

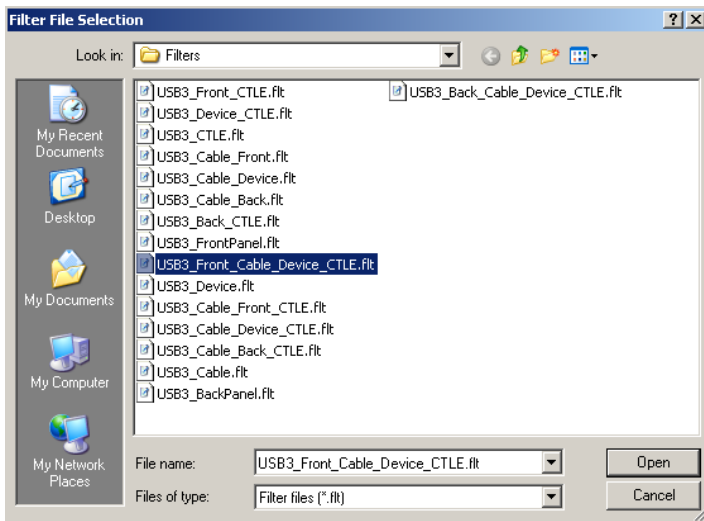
Using the Nonstandard Filter

The following procedure explains how to select and enter any filter file in the configuration panel.

1. Click the **Channel Filter File Type** option in the Select panel and select **Custom Filter File** option from the drop-down menu.



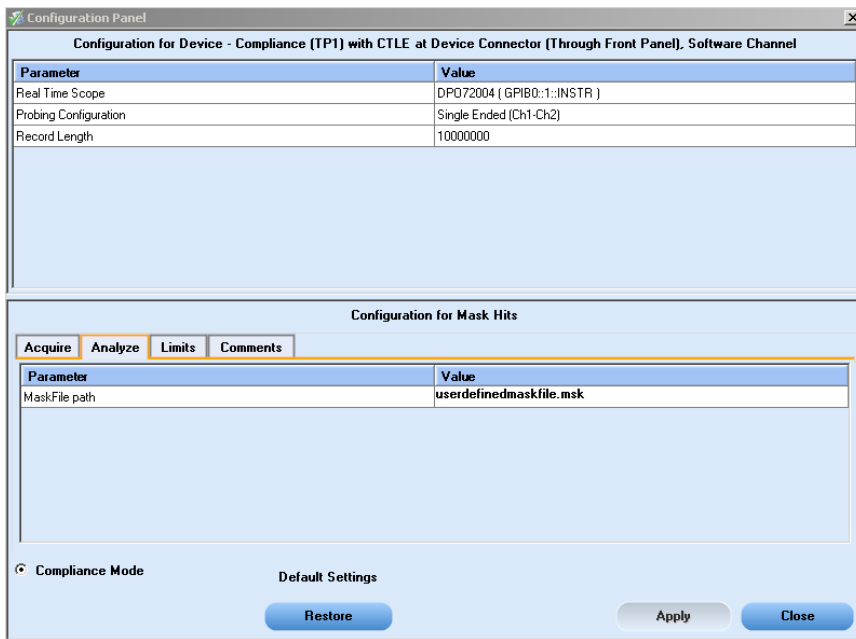
2. Click **Browse** to view the location of filter files from where you can choose from. Select the file and click **Open**.



Using the Nonstandard Waveform Mask

If you want to use a nonstandard mask file, follow this procedure:

1. Enter the limits file name in the Analyze tab of the Configuration panel as shown. Click **Apply** and **Close**.



2. Browse to the folder, C:\Program Files\Tektronix\TekExpress\TekExpress USB\Setupfiles\Masks and copy userdefinedmaskfile.msk to this folder. The user defined mask file is ready to use.

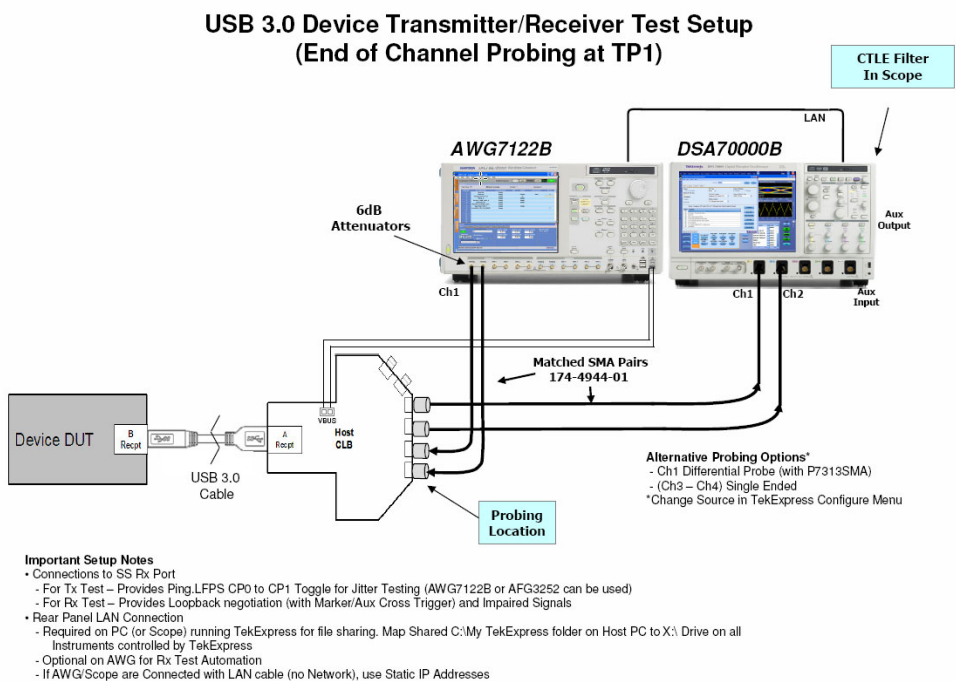
Set Up the Equipment

You need the following equipment:

Table 10: Required equipment

Resource	Model supported
Real-time oscilloscope	Tektronix DPO/DSA71254/B and above. Also DPO/DSA70804/B (8 GHz) model is suitable for Normative measurements.
Probes	Two TCS-SMA or one P7300SMA/P7500 differential probe.
Test Fixture	TF-USB3-A-R (includes short USB 3.0 cable)
AWG (optional)	AWG7122B with options 1,6,8 (v3.0.136.60.2 or later).

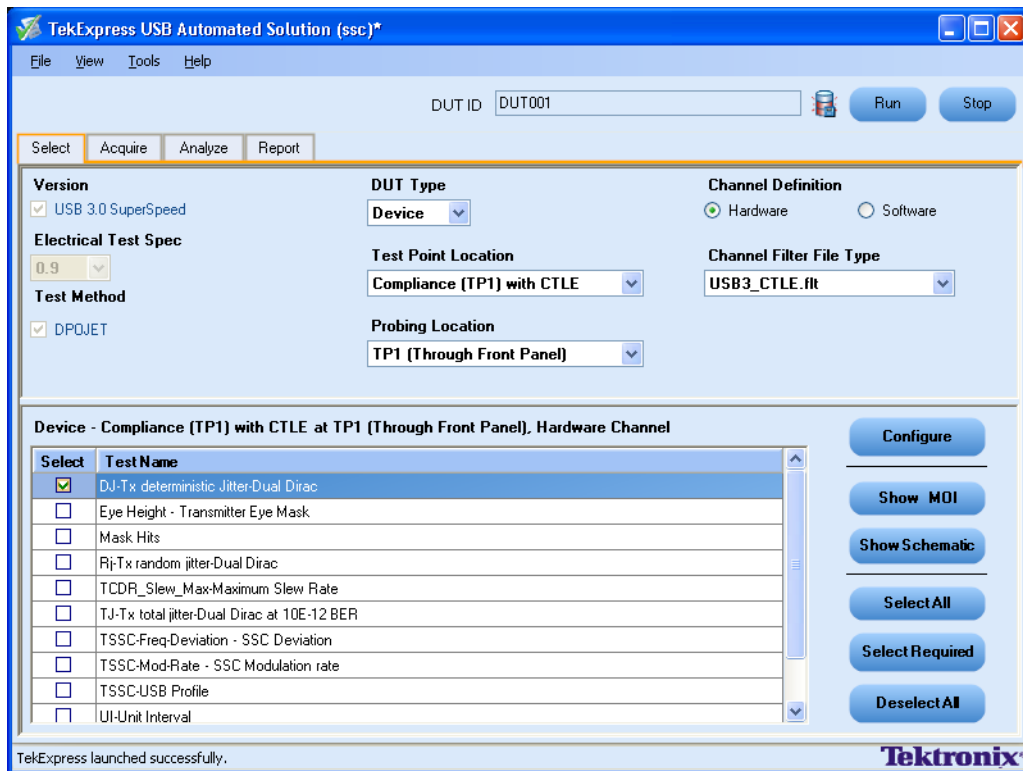
Connect the equipment as shown in the following diagram:



Testing Deterministic Jitter on a Device Front Panel

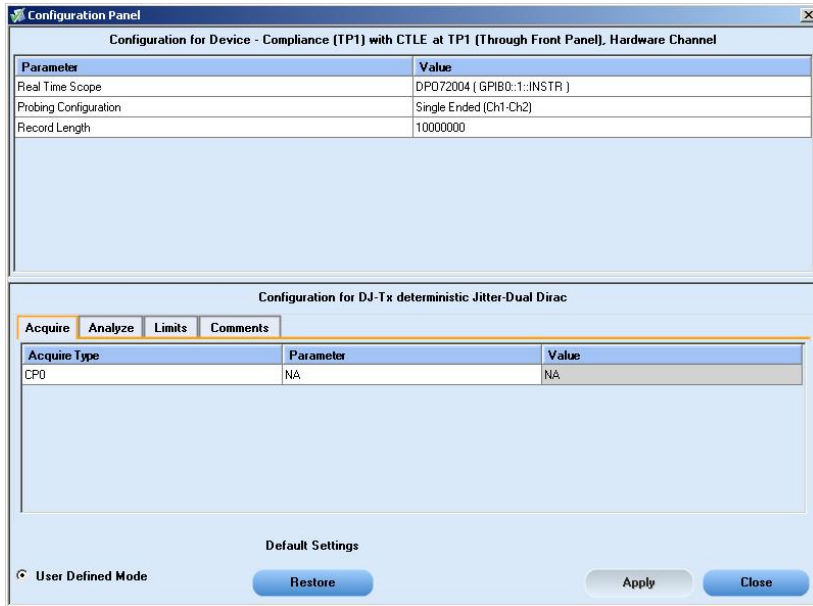
The following procedure discusses how to use TekExpress to test the Device back panel.

1. Select **Device** as the DUT type.
2. Select **TP1 (Through Front Panel)** as the probing location.
3. Select **Compliance[TP1] with CTLE** as the test point.
4. Select **DJ-Tx deterministic Jitter-Dual Dirac** as the test to be run.

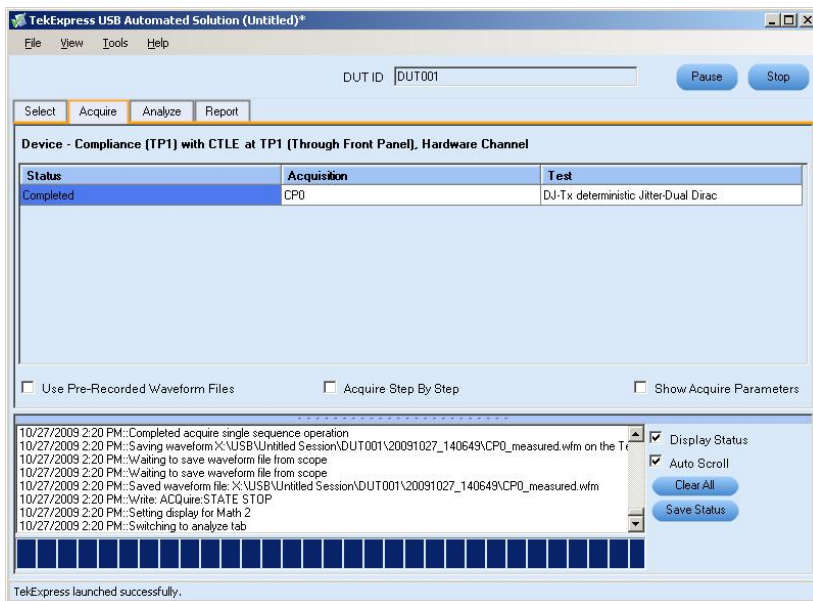


5. Click **Configure** to configure the test parameters.
6. You can configure any test to help you analyze measurement results. To refer to the *DPOJET SuperSpeed (USB 3.0) Setup Library Methods of Implementation (MOI)* document for information on how to configure the Deterministic Jitter test, click **Show MOI** in the Select panel.
7. Once you change the parameters, click **Apply** to apply the new settings for the selected test. If you want to restore the default settings, click **Restore**.

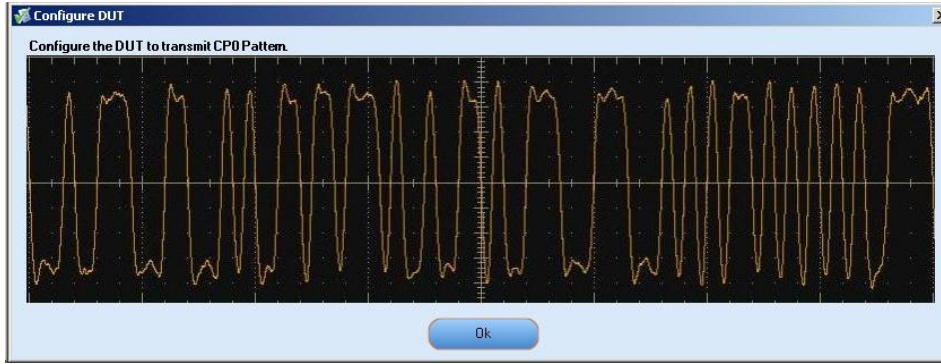
Click **Close** to close the dialog box.



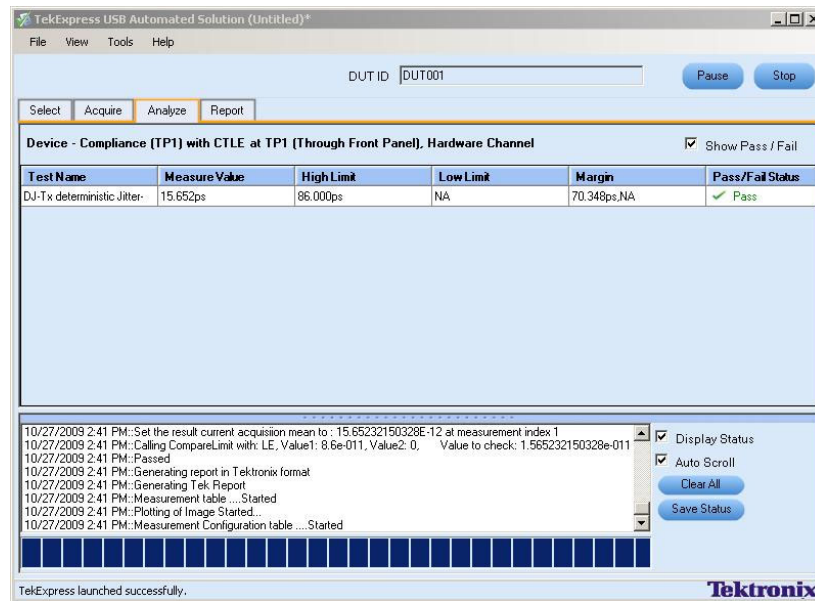
- Click **Run** in the Select panel to run the selected test.
- The Acquire panel shows the status of the waveform acquisition. To know more about the Acquire panel, click [here](#).



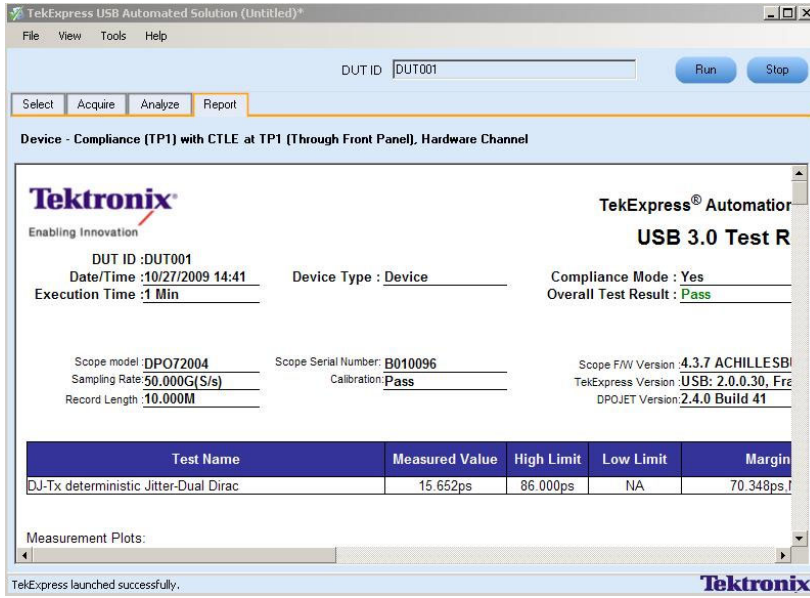
- Click **Ok** in the following dialog box to continue.



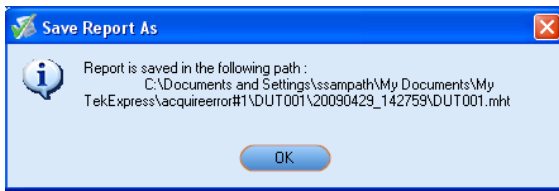
11. The Analyze tab displays the Measured, High, and Low limit values.



12. After the tests complete, a report is generated and displayed in the Report panel.



You can save the report using the **File > Save Report As** option. The following dialog box is displayed.



About the Programmatic Interface

Related Topics

[Server and Client Proxy Objects](#)

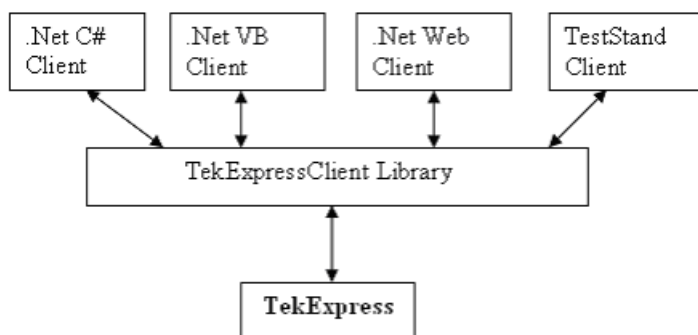
The Programmatic interface allows you to seamlessly integrate the TekExpress test automation application with the high-level automation layer. This also allows you to control the state of TekExpress application running on a local or a remote PC. The following operations can be performed using the programmatic interface exposed by TekExpress:

- Query DUT ID
- Set DUT ID
- SaveSession
- RecallSession
- Run the TekExpress execution
- Stop the TekExpress execution
- Query execution status
- Transfer result files
- Check the application status

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.



Click the following links to get details on them:

What does one need to have to develop TekExpress Client ?

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

References required

TekExpressClient.dll has 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 above mentioned DLL files. It is essential to have these DLLs (IRemoteInterface.dll, IIdlglib.dll and TekDotNetLib.dll) in same folder location as that of TekExpressClient.dll.*

What steps does a client need to follow ?

The following are the steps that a client needs to follow to use the TekExpressClient.dll to programmatically control the server:

A client UI has to be developed to access the interfaces exposed through the server. This client needs to load `TekExpressClient.dll` to access the interfaces. Once the TekExpressClient.dll is loaded, the client UI can call the specific functions to run the operations requested by the client. Once the client is up and running, it has to do the following to run a remote operation:

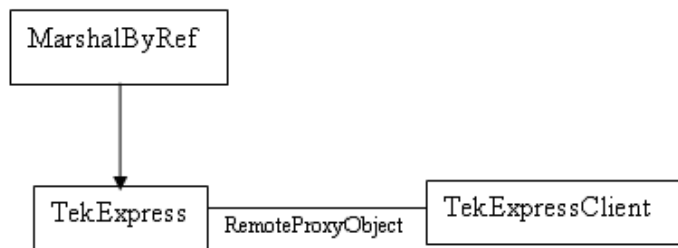
1. The client needs to provide the IP address of the PC at which the server is running in order to connect to the server.
2. The client needs to lock the server application to avoid conflict with any other Client that may try to control the server simultaneously. “Lock” would also disable all user controls on server so that server state cannot be changed by manual operation.
3. If any other client tries to access a server which is locked, it will get a notification that the server is locked by another client.
4. When the client has connected to and locked the server, the client can access any of the programmatic controls to run the remote automations.

5. Once the client operations are completed, the server needs to be “unlocked” by the client.

Server and Client Proxy Objects

Remote Proxy Object

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



The following is an example:

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

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

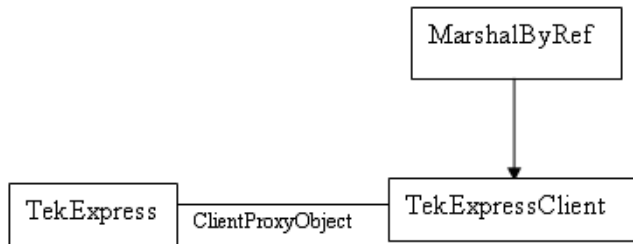
For example,

```
//Get a reference to the remote object
```

```
remoteObject = (IRemoteInterface)Activator.GetObject(typeof(IRemoteInterface), URL.ToString());
```

Client Proxy Object

Client exposes a proxy object to receive certain information.



For example,

```
//Register the client proxy object
```

```
wellKnownServiceTypeEntry[] e = RemotingConfiguration.GetRegisteredWellKnownServiceTypes();
```

```
clientInterface = new ClientInterface();
```

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

```
//Expose the client proxy object through marshalling
```

```
RemotingServices.Marshal(clientInterface, "Remote Client Interface");
```

The client proxy object is used for the following:

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

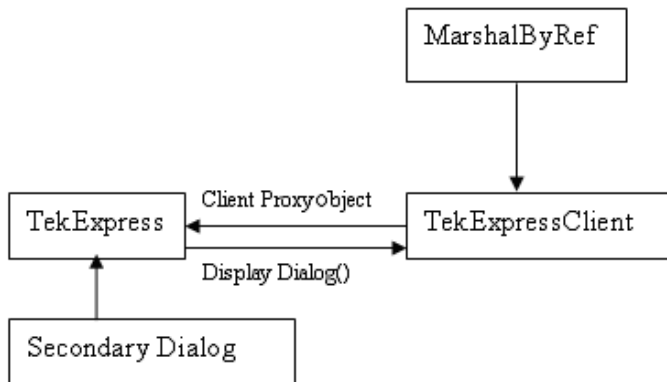
Click here to see examples.

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

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


To know more on the topics below, click the links.

Secondary Dialog Message Handling



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

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

File Transfer Events

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

Connect()

Connect(*string* ipAddress, *out string* clientID)

This method connects the client to the server. The client provides the ip address of the server to connect to the server. The server provides a unique clientID when the client is connected to it.

Parameters

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.
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

Return Value

Value that suggests the status if the connection was established or an error occurred. The return value is an object that can be a boolean value, returning true or a string, returning the error message.

Example

```
try {  
    IPAddress[] hostIPAddr = Dns.GetHostAddresses(Dns.GetHostName());  
    // Connect to the remoter Server  
    remoteObject.Connect(hostIPAddr, clientInterface, out clientID);  
    return true;  
}  
catch (Exception error)  
{  
    return error;  
}
```

Comments

The server has to 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. Each client will get a unique id.

Disconnect()

Disconnect(*string* id)

This method disconnects the client from the server it is connected to.

Parameters

Name	Type	Direction	Description
id	String	IN	Identifier of the client that is performing the remote function.

Return Value

Integer value that suggests the status of the operation after it has been performed.

1 – Success

-1 – Failure

Example

```
try
{
    string returnVal = UnlockServer (clientId);
    remoteObject.Disconnect (clientId);
    return 1;
}
```

Comments

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

LockSession()

LockSession(*string* id)

This method locks the server. The client has to call this method before running any of the remote automations. The server can be locked by only one client.

Parameters

Name	Type	Direction	Description
id	String	IN	Identifier of the client that is performing the remote function.

Return Value

String value that suggests the status of the operation after it has been performed.

Example

```
if (!locked)
    return "Session has already been locked!";
returnVal = remoteObject.LockSession(clientId);
if ((OP_STATUS)returnVal == OP_STATUS.SUCCESS)
{
    locked = true;
    return "Session Locked...";
}
```

Comments

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 has to 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.

UnlockSession()

UnlockSession(*string* id)

This method unlocks the server from the client. The client id of the client to be unlocked has to be provided.

Parameters

Name	Type	Direction	Description
id	string	IN	Identifier of the client that is performing the remote function.

Return Value

String that suggests the status of the operation after it has been performed.

Example

```
returnVal = remoteObject.UnlockSession(clientId);  
if ((OP_STATUS)returnVal == OP_STATUS.SUCCESS)  
{  
    locked = false;  
    return "Session Un-Locked...";  
}
```

Comments

When the client is disconnected, it is automatically unlocked.

SetTimeout()

SetTimeout (*string* id, *string* time)

Parameters

Name	Type	Direction	Description
id	string	IN	Identifier of the client that is performing the remote function.
time	string	IN	The time in seconds which refers to the timeout period.

Return Value

String that suggests the status of the operation after it has been performed.

Example

```
if (ClientId == ClientID)
{
    if (locked == true)
    {
        tempTime = Int32.Parse(time);
    }
    if (tempTime < 0)
        return "Enter a valid Timeout Period";
    else
    {
        timeout = tempTime;
        return "Timeout Period Changed";
    }
}
```

Comments

The time parameter gives the timeout period, that 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.

GetTimeOut()

GetTimeOut (*string* id)

Parameters

Name	Type	Direction	Description
id	string	IN	Identifier of the client that is performing the remote function.

Return Value

String that suggests the status of the operation after it has been performed.

Example

```
if (ClientId == ClientID)
{
    ResetTimer();
    return timeOut.ToString();
}
```

Comments

The timeout period is a positive integer.

GetDutId()

GetDutId(*string* id, *string* dutId)

This method gives the DUT id of the current set-up.

Parameters

Name	Type	Direction	Description
id	string	IN	Identifier of the client that is performing the remote function.
dutId	string	OUT	The DUT id of the set-up.

Return Value

String that gives the timeout period (in seconds) of the client.

Example

```
returnVal = remoteObject.GetDutId(clientId, out id);  
if ((OP_STATUS)returnVal == OP_STATUS.SUCCESS)  
{  
    return id;  
}  
else  
    return CommandFailed(returnVal);
```

Comments

The dutId is an OUT parameter whose value is set after the server processes the request.

ChangeDutId()

ChangeDutId(*string* id, *string* dutName)

This method changes the DUT id of the set-up. The client has to provide a valid DUT id.

Parameters

Name	Type	Direction	Description
id	string	IN	Identifier of the client that is performing the remote function.
dutName	string	IN	The new DUT id of the set-up.

Return Value

String that suggests the status of the operation after it has been performed.

Example

```
If (dut Id.Length <=0 && locked == true)
    return "Enter a valid DUT-ID";
returnVal = remoteObject.ChangeDutId(clientId, dutId);
if ((OP_STATUS)returnVal == OP_STATUS.SUCCESS)
    return "DUT Id Changed...";
else
    return CommandFailed(returnVal);
```

Comments

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

SaveSession()

SaveSession(*string* id, *string* name)

Saves the current session. The name of the session is provided by the client.

Parameters

Name	Type	Direction	Description
id	string	IN	Identifier of the client that is performing the remote function.
name	string	IN	The name of the session being saved.

Return Value

String that suggests the status of the operation after it has been performed.

Example

```
returnVal = remoteObject.SaveSession(clientId, sessionName);  
if ((OP_STATUS)returnVal == OP_STATUS.SUCCESS)  
    return "Session Saved...";  
else  
    return CommandFailed(returnVal);
```

Comments

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. Instead SaveSessionAs can be used.

SaveSessionAs()

SaveSessionAs(*string* id, *string* name)

Saves the current session in a different name every time this method is called. The name of the session is provided by the client.

Parameters

Name	Type	Direction	Description
id	string	IN	Identifier of the client that is performing the remote function.
name	string	IN	The name of the session being saved.

Return Value

String that suggests the status of the operation after it has been performed.

Example

```
returnVal = remoteObject.SaveSessionAs(clientId,sessionName);  
if ((OP_STATUS)returnVal == OP_STATUS.SUCCESS)  
    return "Session Saved...";  
else  
    return CommandFailed(returnval);
```

Comments

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.

RecallSession()

RecallSession(*string* id, *string* name)

Recalls a saved session. The name of the session is provided by the client.

Parameters

Name	Type	Direction	Description
id	string	IN	Identifier of the client that is performing the remote function.
name	string	IN	The name of the session being recalled.

Return Value

String that suggests the status of the operation after it has been performed.

Example

```
returnVal = remoteObject.RecallSession(clientId,sessionName);  
if ((OP_STATUS)returnVal == OP_STATUS.SUCCESS)  
    return "Session Recalled...";  
else  
    return CommandFailed(returnVal);
```

Comments

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

Run()

Run(*string* id)

Runs the setup. Once the server is set up and is configured, it can be run remotely using this function.

Parameters

Name	Type	Direction	Description
id	string	IN	Identifier of the client that is performing the remote function.

Return Value

String that suggests the status of the operation after it has been performed.

Example

```
returnVal = remoteObject.Run(clientId);  
if ((OP_STATUS)returnVal == OP_STATUS.SUCCESS)  
    return "Run started...";  
else  
    return CommandFailed(returnVal);
```

Comments

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

Stop()

Stop(*string* id)

Stops the run operation.

Parameters

Name	Type	Direction	Description
id	string	IN	Identifier of the client that is performing the remote function.

Return Value

String that suggests the status of the operation after it has been performed.

Example

```
returnVal = remoteObject.Stop(clientId);  
if ((OP_STATUS)returnVal == OP_STATUS.SUCCESS)  
    return "Stopped...";  
else  
    return CommandFailed(returnVal);
```

Comments

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

QueryStatus()

QueryStatus(*string* id, *out string[]* status)

This method gives the status of the run as messages. The status messages are generated once the run is started.

Parameters

Name	Type	Direction	Description
id	string	IN	Identifier of the client that is performing the remote function.
status	string array	OUT	The list of status messages generated during run.

Return Value

String that suggests the status of the operation after it has been performed.

Example

```
returnVal = remoteObject.QueryStatus(clientId, out statusMessages);  
if ((OP_STATUS)returnVal == OP_STATUS.SUCCESS)  
    return "Status updated...";  
else  
    return CommandFailed(returnVal);
```

Comments

The status messages are updated periodically after the run begins. The status is an out parameter which is set when the server processes the request.

TransferReport()

TransferReport(*string* id, *string* filePath)

This method transfers the report generated after the run. The report contains the summary of the run. The client has to provide the location where the report is to be saved at the client-end.

Parameters

Name	Type	Direction	Description
id	string	IN	Identifier of the client that is performing the remote function.
filePath	string	IN	The location where the report has to be saved in the client.

Return Value

String that suggests the status of the operation after it has been performed.

Example

```
returnVal = remoteObject.TransferReport(clientId);  
if ((OP_STATUS)returnVal == OP_STATUS.SUCCESS)  
    return "Transferred...";  
else  
    return CommandFailed(returnVal);
```

Comments

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

CheckSessionSaved()

CheckSessionSaved(*string* id, *out bool* saved)

This function is called when a check is to be made to know if the current session is saved.

Parameters

Name	Type	Direction	Description
id	string	IN	Identifier of the client that is performing the remote function.
saved	bool	OUT	Boolean which represents if the current session is saved.

Return Value

Void

Example

```
if (clientId == ClientID)
{
    returnVal = remoteObject.IsSessionSaved(id,out saved);
}
```

Comments

The saved parameter gives the boolean value suggesting if the current session is saved or not. This is used as a check in SaveSession() and SaveSessionAs() Functions.

ApplicationStatus()

ApplicationStatus(string id)

This method gets the status (ready, running, paused) of the server application.

Parameters

Name	Type	Direction	Description
Id	string	IN	Identifier of the client that is performing the remote function.

Return Value

String value that gives the status of the server application.

Example

```
returnVal = remoteObject.ApplicationStatus(clientId);  
return returnVal;
```

Comments

The application can be in one of the following states at a given time:

- Ready – Test configured and ready to start
- Running – Test running
- Paused – Test paused

ErrorCodes

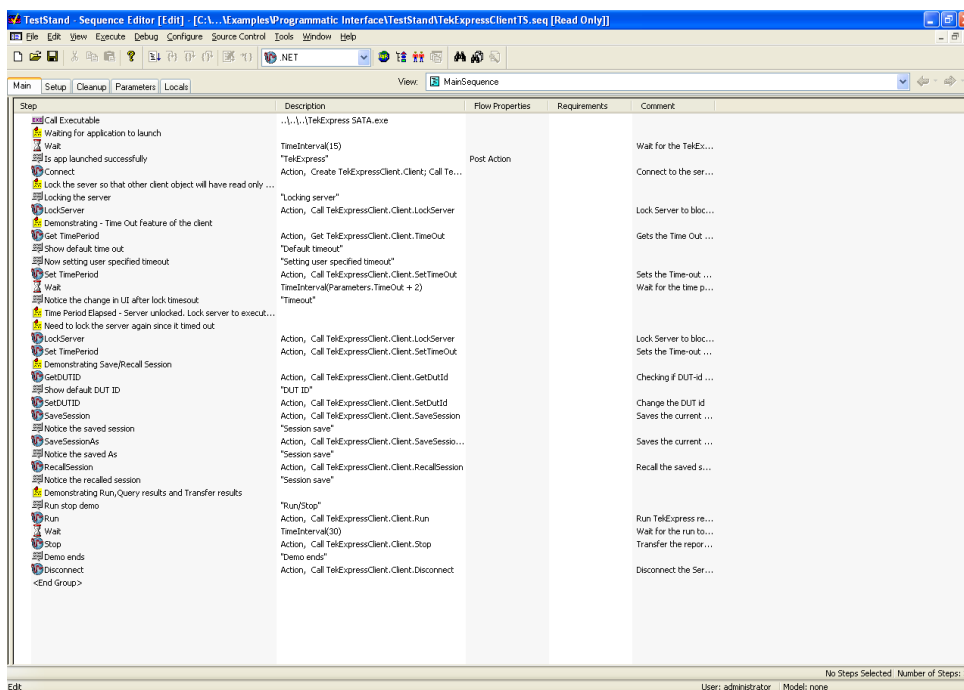
The return value of the remote automations at the server-end is OP_STATUS which is changed to a string value depending on its code and returned to the client.

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 operation cannot be performed.
UNLOCK	4	The server is not locked. Lock the server before performing the operation.
NULL	0	Nothing

Example

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



Instrument Connectivity

Related Topics

[TestStand Runtime Engine Installation](#)

If the instrument(s) are displayed in TekVISA Instrument Manager but not in the TekExpress Instrument Bench, check the following:

- Only those instruments that respond to *i dn? and *opt? queries successfully, are displayed in Instrument Bench.
- Ensure that VXI-11 Server is running on the instruments.

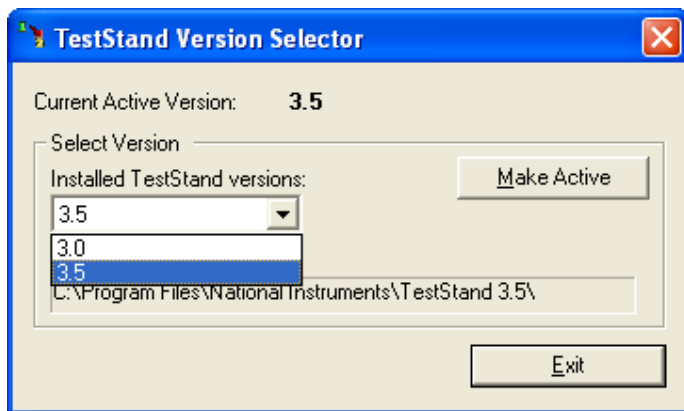
If Instrument initialization fails during test sequence execution, do the following:

It is observed that GPIB communication with instrument over Tek-VISA layer is not initialized if in TekVISA Instrument manager the search criteria is turned-off, even if a valid instrument is connected in the network. It is necessary to turn ON the respective search criteria by opening the TekVISA Instrument manager.

TestStand Run time Engine Installation

Managing multiple versions of TestStand installed on the system.

TekExpress installs TestStand version 3.5 runtime engine. If you have versions other than 3.5, while working with TekExpress, ensure that the version shipped with TekExpress is active. You can do so by clicking **Start > Programs > National Instruments > TestStand 3.5 > TestStand Version Selector**.



Shortcut Keys

The following table lists the short cut keys to access the application:

Table 11: Keyboard shortcut keys

Menu	Shortcut keys
File	Alt + F
New Session	Ctrl + N
Open Session...	Ctrl + O
Save Session...	Ctrl + S
Save Session As	Alt + F + A
Save Report As...	Alt + F + R
Print Preview Report	Alt + F + V
Print Report...	Ctrl + P
Exit	Ctrl + X
View	Alt + V
Log	Ctrl + L
Tools	Alt + T
Instrument Bench	Ctrl + I
Help	Alt + H
TekExpress Help (F1)	Alt + H + H
About TekExpress	Alt + H + A
Activate License	Alt + H + L

Index

A

- About menu, 19
- About TekExpress, 13
- Acquire options, 31
- Acquire parameters, 26
- Acquire Parameters, 31
- Acquire tab, 30
- acquisition table, 31
- acquisitions, 30
- Activating license, 8
- Analyze parameters, 26
- Application
 - exiting, 15
 - resizing, 14
 - starting, 14
- Application overview, 13
- Application summary, 13

B

- Before clicking Run, 10

C

- Client, 45
- Compliance mode, 25
- Configure button, 23
- Controlling the Server, 46

D

- Data storage, 35
- Default directory usage, 6
- Deskew
 - real time oscilloscopes, 21
- Deterministic Jitter test, 40
- Directory structure, 6
- Dongle, 8
- DPOJET, 6
- DUT type, 23

E

- Exit, 16

- Exiting the application, 15

F

- File name extensions, 8
- Filter selection, 36
- Filters, 36

G

- General parameters, 25
- Global Controls, 15

I

- Instrument bench, 17
- Instrument Bench menu, 28
- Instrument discovery, 29
- Instrument initialization, 69
- Interface, 45
- Interface commands
 - ChangeDutId, 57
 - Connect, 49
 - Disconnect, 51
 - GetDutId, 56
 - GetTimeout, 55
 - LockSession, 52
 - Recall Session, 60
 - Run, 61
 - SaveSession, 58
 - SaveSessionAs, 59
 - SetTimeout, 54
 - Status, 63
 - Stop, 62
 - Transfer report, 64
 - UnlockSession, 53
- Interface error codes, 67

K

- Key features, 13

L

- Log file, 17

M

- Mapping My TekExpress
 - folder, 11
- Menus
 - File, 16
 - Help, 19
 - Tools, 17
 - View, 17
- Minimum System Requirements, 6
- My TekExpress, 35
- My TekExpress folder, 10

N

- Non VISA resources, 18
- Nonstandard waveform mask, 37

P

- Parameters to configure, 24
- Prerecorded Waveform, 30
- Probes, 5
- Programmatic interface, 45
- Progress of analysis, 32
- Proxy object
 - client, 48
 - remote, 47

R

- Report
 - preview, 16
 - print, 16
 - save, 16
- Report panel overview, 34
- Required equipment, 39
- Resizing the application, 14
- Retrieved instruments, 18
- Run, 10
 - Run button, 15
- Run button, 15

S

- Safety Summary, iii
- Save
 - report, 16
 - session, 16
- SDLA software, 6
- Selecting connected instruments, 29
- Server, 45
- Session
 - new, 16
 - open, 16
 - save, 16
- Session folder, 35
- Set up diagram, 39
- Shortcut keys, 71
- Show MOI button, 23
- Show schematic button, 24
- Signal Path Compensation (SPC), 11

- SMA Breakout Fixture, 5
- Software version, 19
- Starting the application, 14
- Supported Probes, 5
- System requirements, 6

T

- Technical support, 3
- TekVISA instrument manager, 28
- Test
 - configure, 24
 - report, 34
 - select, 22
- Test limits, 27
- Test Method, 23
- Test parameters, 26
- Test point, 23
- Test point selection, 23
- Test related files, 35

- TestStand client example, 67
- Troubleshooting
 - instrument connectivity, 69
 - TestStand Run-time Engine installation, 69

U

- Untitled session, 10
- USB devices, 22
- User defined filter file, 36
- User defined mode, 25
- Using filters, 36

V

- View scorecard, 34
- Viewing connected instruments, 28