ADS
2015.01

USB3.1 Compliance Test Bench

KEYSIGHT TECHNOLOGIES

Notice

Restricted Rights Legend

If software is for use in the performance of a U.S. Government prime contract or subcontract, Software is delivered and licensed as "Commercial computer software" as defined in DFAR 252.227-7014 (June 1995), or as a "commercial item" as defined in FAR 2.101(a) or as "Restricted computer software" as defined in FAR 52.227-19 (June 1987) or any equivalent agency regulation or contract clause.

Use, duplication or disclosure of Software is subject to Keysight Technologies' standard commercial license terms, and non-DOD Departments and Agencies of the U.S. Government will receive no greater than Restricted Rights as defined in FAR 52.227-19(c)(1-2) (June 1987). U.S. Government users will receive no greater than Limited Rights as defined in FAR 52.227-14 (June 1987) or DFAR 252.227-7015 (b)(2) (November 1995), as applicable in any technical data.

Acknowledgments

Layout Boolean Engine by Klaas Holwerda, v1.7 http://boolean.klaasholwerda.nl/bool.html

FreeType Project, Copyright (c) 1996-1999 by David Turner, Robert Wilhelm, and Werner Lemberg.

QuestAgent search engine (c) 2000-2002, JObjects.

MATLAB is a registered trademark of The MathWorks, Inc. in the United States and/or other countries.

SuperLU_MT version 2.0

SuperLU Copyright: Copyright © 2003, The Regents of the University of California, through Lawrence Berkeley National Laboratory (subject to receipt of any required approvals from U.S. Dept. of Energy). All rights reserved.

SuperLU Disclaimer: THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

7-zip

7-Zip Copyright: Copyright (C) 1999-2009 Igor Pavlov.

Licenses for files are:

1. 7z.dll: GNU LGPL + unRAR restriction.
2. All other files: GNU LGPL.

7-zip License: This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License , or (at your option) any later version. This library is distributed in the hope that it will be useful,but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details. You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA.

unRAR copyright: The decompression engine for RAR archives was developed using source code of unRAR program.All copyrights to original unRAR code are owned by Alexander Roshal.

unRAR License: The unRAR sources cannot be used to re-create the RAR compression algorithm, which is proprietary. Distribution of modified unRAR sources in separate form or as a part of other software is permitted, provided that it is clearly stated in the documentation and source comments that the code may not be used to develop a RAR (WinRAR) compatible archiver.

7-zip Availability: http://www.7-zip.org/

AMD Version 2.2

AMD Notice: The AMD code was modified. Used by permission.

AMD copyright: AMD Version 2.2, Copyright © 2007 by Timothy A. Davis, Patrick R. Amestoy, and Iain S. Duff. All Rights Reserved.

AMD License: Your use or distribution of AMD or any modified version of AMD implies that you agree to this License. This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License , or (at your option) any later version. This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details. You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA Permission is hereby granted to use or copy this program under the terms of the GNU LGPL, provided that the Copyright, this License, and the Availability of the original version is retained on all copies.User documentation of any code that uses this code or any modified version of this code must cite the Copyright, this License, the Availability note, and "Used by permission." Permission to modify the code and to distribute modified code is granted, provided the Copyright, this License, and the Availability note are retained, and a notice that the code was modified is included.

AMD Availability: http://www.cise.ufl.edu/research/sparse/amd

UMFPACK 5.0.2

UMFPACK Notice: The UMFPACK code was modified. Used by permission.

UMFPACK Copyright: UMFPACK Copyright © 1995-2006 by Timothy A. Davis. All Rights Reserved.

UMFPACK License: Your use or distribution of UMFPACK or any modified version of UMFPACK implies that you agree to this License. This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version. This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details. You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA Permission is hereby granted to use or copy this program under the terms of the GNU LGPL, provided that the Copyright, this License, and the Availability of the original version is retained on all copies. User documentation of any code that uses this code or any modified version of this code must cite the Copyright, this License, the Availability note, and "Used by permission." Permission to modify the code and to distribute modified code is granted, provided the Copyright, this License, and the Availability note are retained, and a notice that the code was modified is included.

UMFPACK Availability: http://www.cise.ufl.edu/research/sparse/umfpack UMFPACK (including versions 2.2.1 and earlier, in FORTRAN) is available at http://www.cise.ufl.edu/research/sparse. MA38 is available in the Harwell Subroutine Library. This version of UMFPACK includes a modified form of COLAMD Version 2.0, originally released on Jan. 31, 2000, also available at http://www.cise.ufl.edu/research/sparse. COLAMD V2.0 is also incorporated as a built-in function in MATLAB version 6.1, by The MathWorks, Inc. http://www.mathworks.com. COLAMD V1.0 appears as a column-preordering in SuperLU (SuperLU is available at http://www.netlib.org). UMFPACK v4.0 is a built-in routine in MATLAB 6.5. UMFPACK v4.3 is a built-in routine in MATLAB 7.1.

Errata

The ADS product may contain references to "HP" or "HPEESOF" such as in file names and directory names. The business entity formerly known as "HP EEsof" is now part of Keysight Technologies and is known as " Keysight EEsof". To avoid broken functionality and to maintain backward compatibility for our customers, we did not change all the names and labels that contain "HP" or "HPEESOF" references.

Qt Version 4.8.4

Qt Notice: The Qt code was modified. Used by permission.

Qt Availability: http://www.qtsoftware.com/downloads

Patches Applied to Qt can be found in the installation at: $HPEESOF_DIR/prod/licenses/thirdparty/qt/patches.

You may also contact Brian Buchanan at Keysight Inc. at brian_buchanan@keysight.com for more information .

The HiSIM_HV source code, and all copyrights, trade secrets or other intellectual property rights in and to the source code, is owned by Hiroshima University and/or STARC.

HDF5

HDF5 Notice: The HDF5 code was modified. Used by permission.

HDF5 License:

Copyright Notice and License Terms for HDF5 (Hierarchical Data Format 5) Software Library and Utilities

----

HDF5 (Hierarchical Data Format 5) Software Library and Utilities

NCSA HDF5 (Hierarchical Data Format 5) Software Library and Utilities

libpng

libpng versions 0.97, January 1998, through 1.0.6, March 20, 2000, are Copyright (c) 1998, 1999 Glenn Randers-Pehrson, and are distributed according to the same disclaimer and license as libpng-0.96, with the following individuals added to the list of Contributing Authors: Tom Lane, Glenn Randers-Pehrson, Willem van Schaik

libpng versions 0.89, June 1996, through 0.96, May 1997, are Copyright (c) 1996, 1997 Andreas Dilger Distributed according to the same disclaimer and license as libpng-0.88, with the following individuals added to the list of Contributing Authors: John Bowler, Kevin Bracey, Sam Bushell, Magnus Holmgren, Greg Roelofs, Tom Tanner

libpng versions 0.5, May 1995, through 0.88, January 1996, are Copyright (c) 1995, 1996 Guy Eric Schalnat, Group 42, Inc.
For the purposes of this copyright and license, "Contributing Authors" is defined as the following set of individuals: Andreas Dilger, Dave Martindale, Guy Eric Schalnat, Paul Schmidt, Tim Wegner

The PNG Reference Library is supplied "AS IS". The Contributing Authors and Group 42, Inc. disclaim all warranties, expressed or implied, including, without limitation, the warranties of merchantability and of fitness for any purpose. The Contributing Authors and Group 42, Inc. assume no liability for direct, indirect, incidental, special, exemplary, or consequential damages, which may result from the use of the PNG Reference Library, even if advised of the possibility of such damage.
Permission is hereby granted to use, copy, modify, and distribute this source code, or portions hereof, for any purpose, without fee, subject to the following restrictions:

1. The origin of this source code must not be misrepresented.
2. Altered versions must be plainly marked as such and must not be misrepresented as being the original source.
3. This Copyright notice may not be removed or altered from any source or altered source distribution.

The Contributing Authors and Group 42, Inc. specifically permit, without fee, and encourage the use of this source code as a component to supporting the PNG file format in commercial products. If you use this source code in a product, acknowledgment is not required but would be appreciated.

OpenSSL

The OpenSSL toolkit stays under a dual license, i.e. both the conditions of the OpenSSL License and the original SSLeay license apply to the toolkit. See below for the actual license texts. Actually both licenses are BSD-style Open Source licenses. In case of any license issues related to OpenSSL please contact openssl-core@openssl.org.

OpenSSL License

==
Copyright (c) 1998-2011 The OpenSSL Project. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Original SSLeay License

3. All advertising materials mentioning features or use of this software must display the following acknowledgement: "This product includes cryptographic software written by Eric Young (eay@cryptsoft.com)" The word 'cryptographic' can be left out if the rouines from the library being used are not cryptographic related.
4. If you include any Windows specific code (or a derivative thereof) from the apps directory (application code) you must include an acknowledgement: "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"

THIS SOFTWARE IS PROVIDED BY ERIC YOUNG ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The licence and distribution terms for any publically available version or derivative of this code cannot be changed. i.e. this code cannot simply be copied and put under another distribution license \[including the GNU Public Licence.]

Growl GNTP support:
[The "BSD licence"] Copyright (c) 2009-2010 Yasuhiro Matsumoto
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR "AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Cuda
Cuda Redistributable Software - 1.8. Attachment A

In connection with Section 1.2.1.1 of this Agreement, the following files may be redistributed with software applications developed by Licensee, including certain variations of these files that have version number or architecture specific information embedded in the file name - as an example only, for release version 6.0 of the 64-bit Windows software, the file cudart64_60.dll is redistributable.

Component : CUDA Runtime
Windows : cudart.dll, cudart_static.lib
MacOS : libcudart.dylib, libcudart_static.a
Linux : libcudart.so, libcudart_static.a
Android : libcudart.so, libcudart_static.a

Component : CUDA FFT Library
Windows : cufft.dll
MacOS : libcufft.dylib
Linux : libcufft.so
Android : libcufft.so

Component : CUDA BLAS Library
Windows : cublas.dll
MacOS : libcublas.dylib
Linux : libcublas.so
Android : libcublas.so

Component : CUDA Sparse Matrix Library
Windows : cusparse.dll
MacOs : libcusparse.dylib
Linux : libcusparse.so
Android : libcusparse.so

Component : CUDA Random Number Generation Library
Windows : curand.dll
MacOs : libcurand.dylib
Linux : libcurand.so
Android : libcurand.so

Component : NVIDIA Performance Primitives Library
Windows : nppc.dll, nppi.dll, npps.dll
MacOs : libnppc.dylib, libnppi.dylib, libnpps.dylib
Linux : libnppc.so, libnppi.so, libnpps.so
Android : libnppc.so, libnppi.so, libnpps.so

Component : NVIDIA Optimizing Compiler Library
Windows : nvvm.dll
MacOs : libnvvm.dylib
Linux : libnvvm.so

Component : NVIDIA Common Device Math Functions Library
Windows : libdevice.compute_20.bc, libdevice.compute_30.bc, libdevice.compute_35.bc
MacOs : libdevice.compute_20.bc, libdevice.compute_30.bc, libdevice.compute_35.bc
Linux : libdevice.compute_20.bc, libdevice.compute_30.bc, libdevice.compute_35.bc

Component : CUDA Occupancy Calculation Header Library
All : cuda_occupancy.h

Read more at: http://docs.nvidia.com/cuda/eula/index.html#ixzz30CrknWfU

Table of Contents

USB 3.1 Compliance Test Bench

This section describes the following topics:

- Installing USB 3.1 Compliance Test Bench
- Difference between USB 3.0 and USB 3.1
- USB 3.1 Compliance Test Bench Simulation Setups
- Running USB 3.1 Compliance Tests on Infiniium Offline

Installing USB 3.1 Compliance Test Bench

Installing USB 3.1 Compliance Test Bench

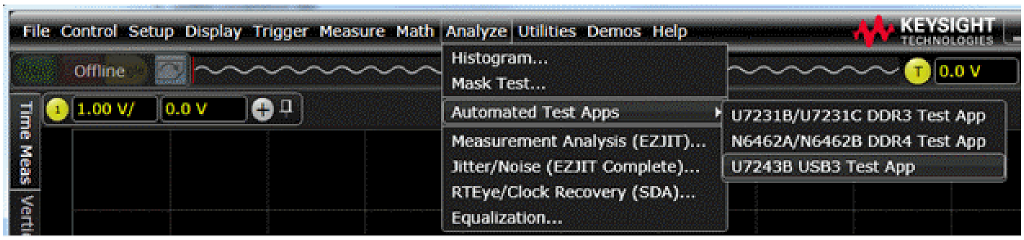
This section provides information on prerequisites and steps to install the USB 3.1 Compliance Test Bench (CTB).

Prerequisites

Before using the USB 3.1 CTB, ensure that the following softwares are installed:

- Infiniium Offline (Version 05.20.0010)
- USB 3.1 Compliance App (Version 2.00)
- ADS 2015.01

After installing the USB 3.1 CTB, launch the Infiniium Offline software to ensure the USB 3.1 Test App is available under **Analyze > Automated Test Apps**.



Install Instructions

To install the USB 3.1 CTB:

1. Download the *USB3p1CTB.deb* package.
2. Select **DesignGuide > Add DesignGuide** from the ADS Main window.
   The Add DesignGuide dialog box is displayed.
3. Click **Add Global DesignGuide**.
4. Browse and select the *USB3p1CTB.deb* package.
5. Click **Open**.
   The USB 3.1 Compliance Test Bench will be added.
6. Restart ADS.
7. Open a Schematic view and select **DesignGuide**.
   The USB 3.1 Compliance Test Bench will be listed under the DesignGuide menu with the name 'USB 3.1 Compliance Test Bench'.

Difference between USB 3.0 and USB 3.1

Difference between USB 3.0 and USB 3.1

The USB 3.1 standard also referred as USB 3 Gen2 (or Superspeed+) increases the data signal rate to 10 Gbps, double that of USB 3.0 (referred to as USB 3 Gen1 or SuperSpeed). It reduces the line encoding overhead to just 3%, by changing the encoding scheme to 128b/132b.

NOTE The USB 3.1 standard is also backward compatible with USB 3.0 and USB 2.0.

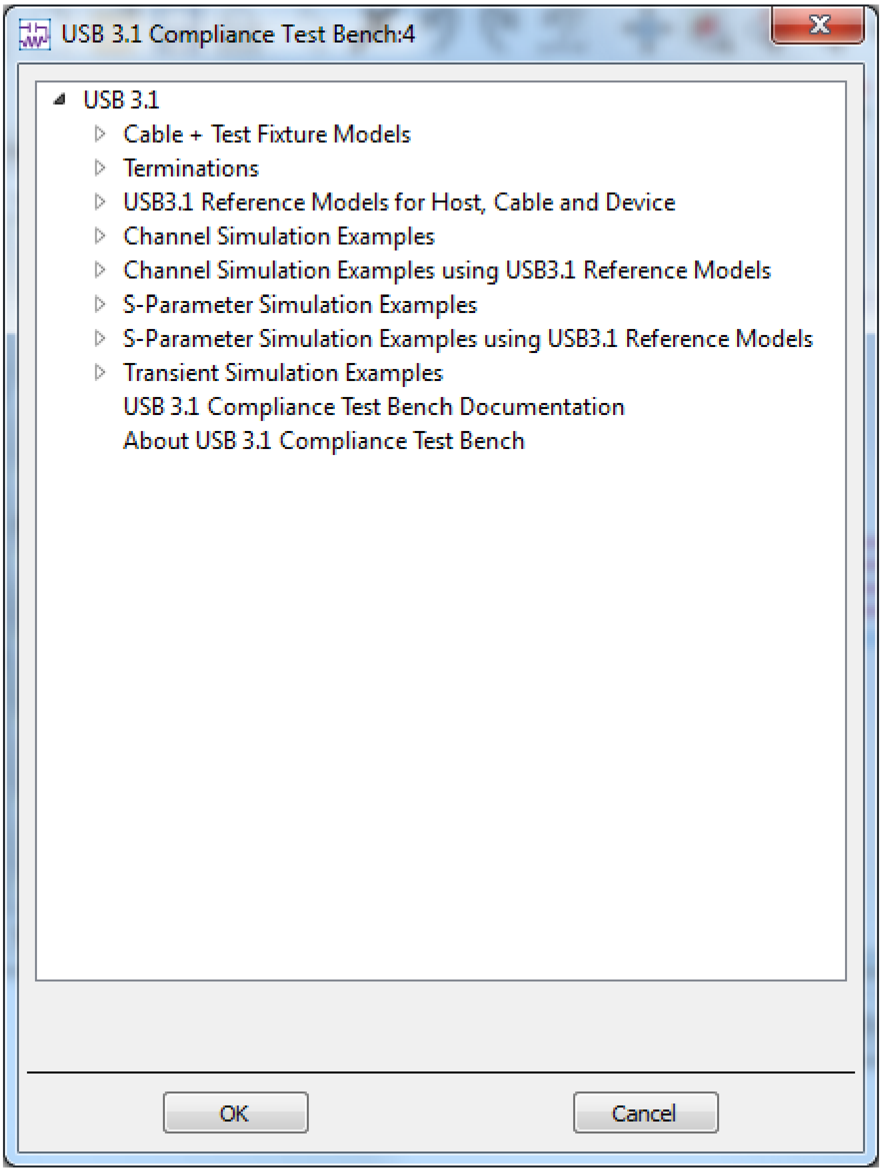
The following table lists the differences between the two generations of USB 3 standards.

	USB 3.0 (Gen1/SuperSpeed)	**USB 3.1 (Gen2/SuperSpeed+)**
Data Rate	5 Gb/s	10 Gb/s
Coding	8b/10b	128b/132b
Target Channel	3 meter (-17 dB @ 2.5 GHz)	1 meter (-20 dB @ 5 GHz)
CDR	JTF BW 4.9Mhz	JTF BW 7.5Mhz
SSC	Slew rate test	New df/dt requirement: 1250 (max) ppm/µs
De-emphasis	Post: -3dB (Required)	Pre: 2.7dB (Informative) Post: -3dB (Informative)
RX Ref EQ	CTLE	CTLE + 1 tap DFE
Eye Height, TJ	100mV, 132ps(.66UI)	70mV, 71.3ps(.714UI)

USB 3.1 Compliance Test Bench Simulation Setups

USB 3.1 Compliance Test Bench Simulation Setups

The USB 3.1 Compliance Test Bench provides a variety of tests, which helps to understand the various aspects of the USB digital standard. It provides you the ability to create designs using the included models or your own models. You can refer to the included examples when developing the designs. This Compliance Test Bench provides the following Models and Examples:



Models

Find below the list of models:

- Cable + Test Fixture Models: Includes models for different channels, cables, receptacles, connectors etc. which can be used to design USB links from Transmitter to Receiver. Some of these have been used in the examples mentioned below.
- Terminations: Includes source and load terminations.
- USB 3.1 Reference Models for Host, Cable and Device: Includes reference models for host, cable and device which can be used to design USB links from Transmitter to Receiver. These have been downloaded from the USB-IF website.

Examples

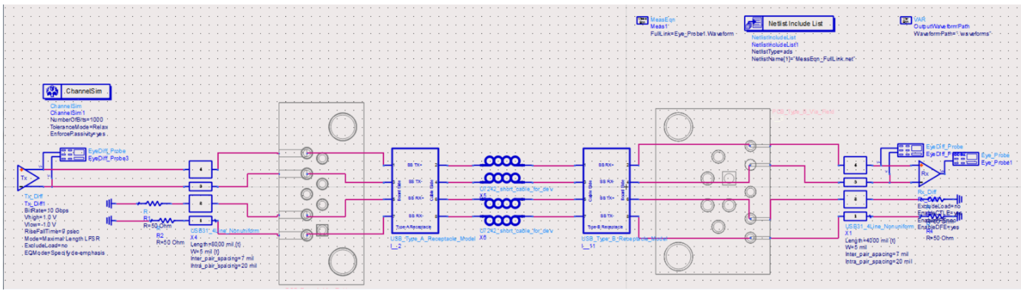
Find below the list of examples included in the USB 3.1 Compliance Test Bench:

- Channel Simulation Examples
- Channel Simulation Examples using USB 3.1 Reference Models
- S-Parameter Simulation Examples
- S-Parameter Simulation Examples using USB 3.1 Reference Model
- Transient Simulation Examples

Channel Simulation Examples

Full Link with Tx/Rx Models

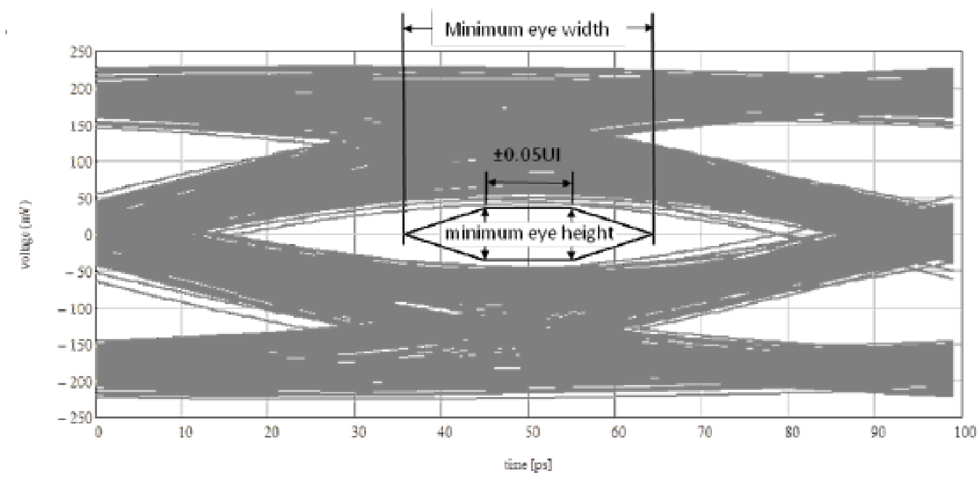
This design displays a typical USB connection from Transmitter to Receiver. The signal from a Differential Transmitter flows through the PCB traces, the Via field and the Receptacle (this part represents the Host) to reach the Cable. From here it flows through the Receptacle, Via Field and PCB Traces (this part represents the Device) to reach the Receiver. Eye Probes placed in the circuit display the signal leaving the Transmitter, the signal reaching the Receiver and the signal after the Receiver Equalization.



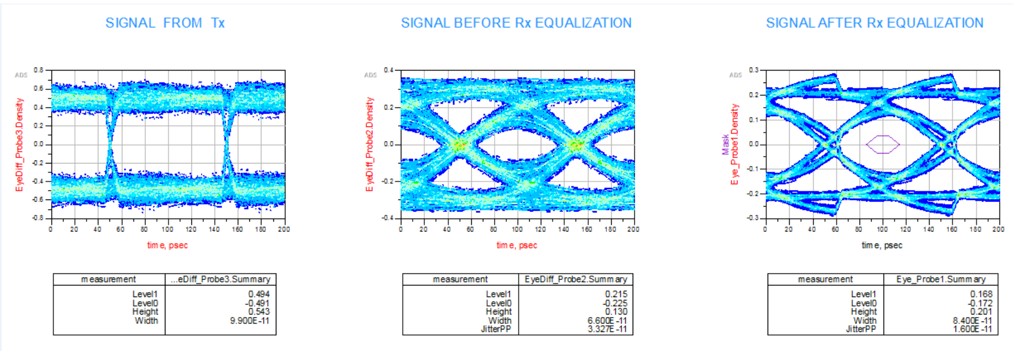
The Transmitter transmits PRBS data at a rate of 10Gbps and uses 128b132b encoding as per USB 3.1 standard.
The Receiver Equalization consists of CTLE + DFE. This is in addition to de-emphasis applied at the Transmitter end.

The channel consists of PCB traces, Via field and Receptacle for both the host and device, along with a short cable.

NOTE The number of bits being simulated has been kept at 1000 to reduce the size of the Compliance Test Bench. You should change it to 1000000 or more before running a practical simulation.



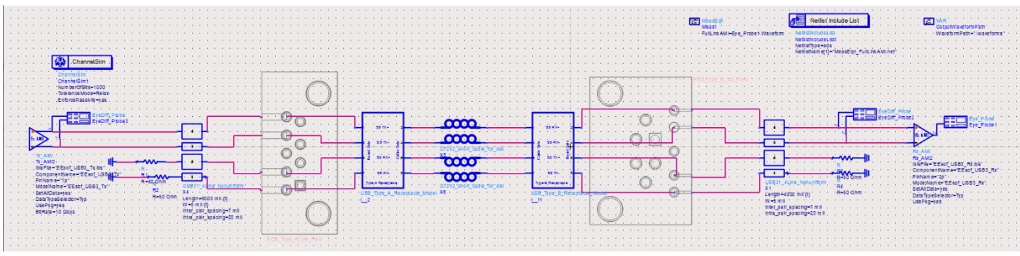
Gen 2 eye mask



The waveform generated after receiver equalization is saved in the data/waveforms directory with the name *FullLink.h5*.

Full Link with IBIS AMI models

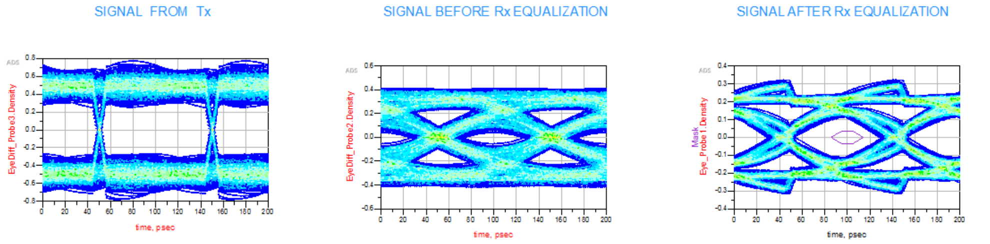
This design shows a typical USB connection from Transmitter to Receiver. The difference from example above is that both the Transmitter and Receiver use IBIS AMI models.



The Transmitter AMI model provides an option to enter the pre-shoot value in addition to de-emphasis. The Receiver AMI model provides an option to enter CTLE and DFE parameters along with the CDR parameters as well.

NOTE The number of bits being simulated has been kept at 1000 to reduce the size of the Compliance Test Bench. You should change it to 1000000 or more before running a practical simulation.

The resulting eye along with measurement summary is shown below. The third eye diagram also plots the Eye Mask for USB 3.1.



measurement	..eDiff_Probe3.Summary
Level1	0.494
Level0	-0.493
Height	0.543
Width	9.700E-11

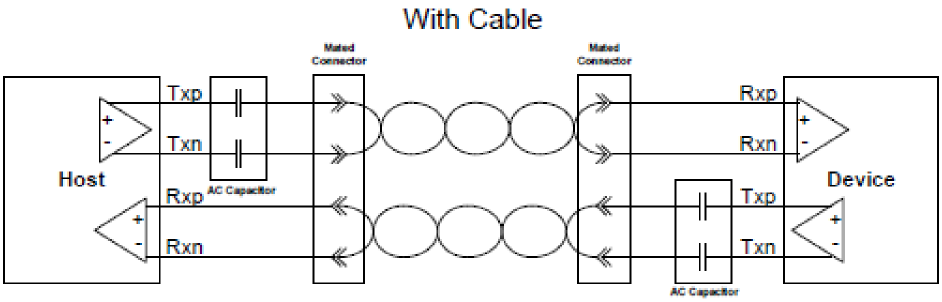
measurement	EyeDiff_Probe2.Summary
Level1	0.235
Level0	-0.234
Height	0.079
Width	5.100E-11
JitterPP	4.187E-11

measurement	Eye_Probe1.Summary
Level1	0.179
Level0	-0.179
Height	0.189
Width	7.500E-11
JitterPP	2.400E-11

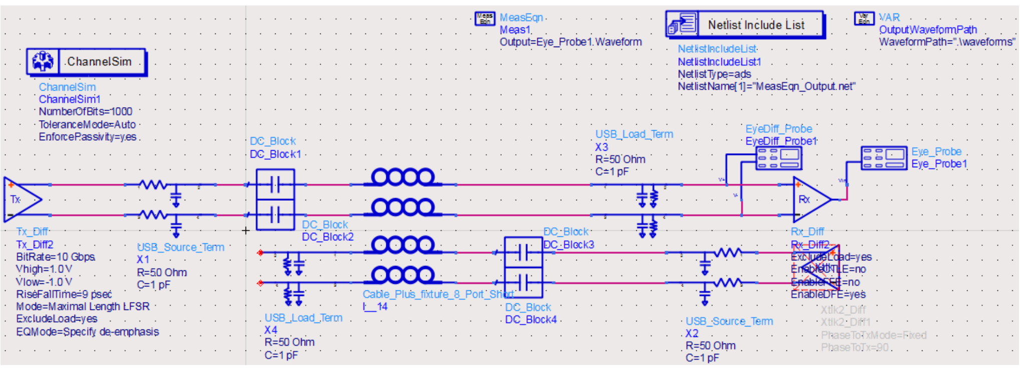
The waveform generated after receiver equalization is saved in the data/waveforms directory with the name *FullLinkAMI.h5*.

Cable plus fixture with Crosstalk

This design implements a channel simulation of a cable+fixture model, as shown in the USB 3.1 specifications.



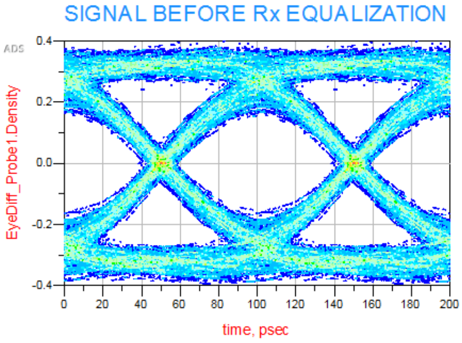
It can also optionally include the effect of crosstalk, which is disabled by default. To understand the effect of crosstalk, enable the Xtlk2_Diff component in the design.



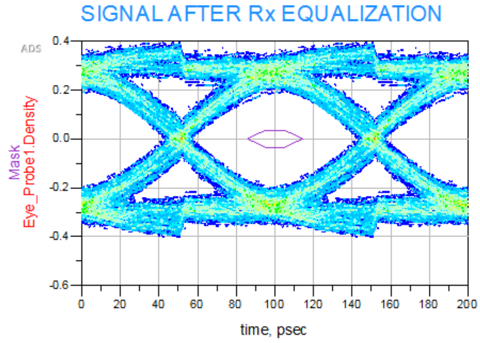
The Transmitter transmits PRBS data at a rate of 10Gbps and uses 128b132b encoding as per USB 3.1 standard .
The Receiver Equalization consists of DFE. This is in addition to de-emphasis applied at the Transmitter end.

NOTE The number of bits being simulated has been kept at 1000 to reduce the size of the Compliance Test Bench. You should change it to 1000000 or more before running a practical simulation.

The resulting eye along with measurement summary is shown below. The second eye diagram also plots the Eye Mask for USB 3.1, as specified in the USB 3.1 specifications.



measurement	EyeDiff_Probe1.Summary
Level1	0.269
Level0	-0.273
Height	0.263
Width	8.250E-11

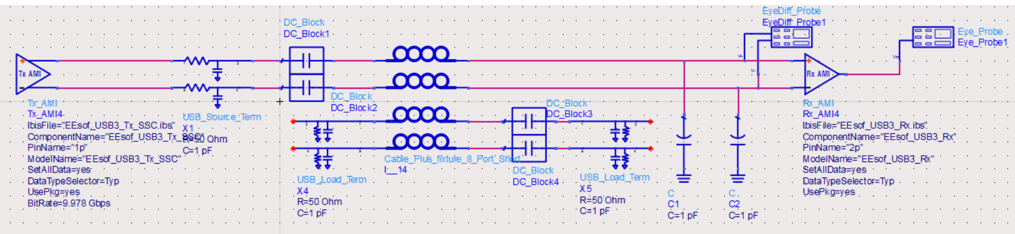


measurement	Eye_Probe1.Summary
Level1	0.268
Level0	-0.270
Height	0.321
Width	8.200E-11
JitterPP	1.714E-11

The waveform generated after receiver equalization is saved in the data/waveforms directory with the name *Output.h5*.

Cable plus fixture with IBIS AMI models with SSC

This design shows a channel simulation of a cable+fixture model using IBIS AMI models for the Transmitter and Receiver. The difference from example #3 above is that both Transmitter and Receiver use IBIS AMI models, and there is no CrossTalk.

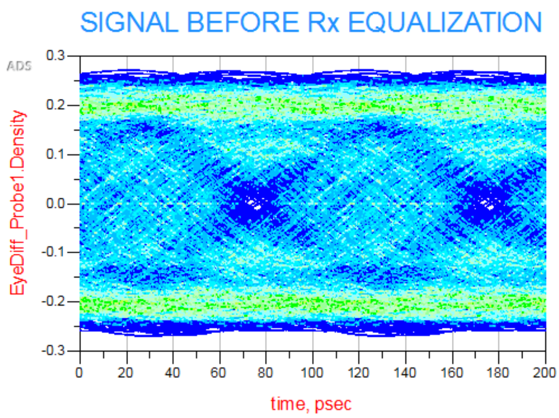


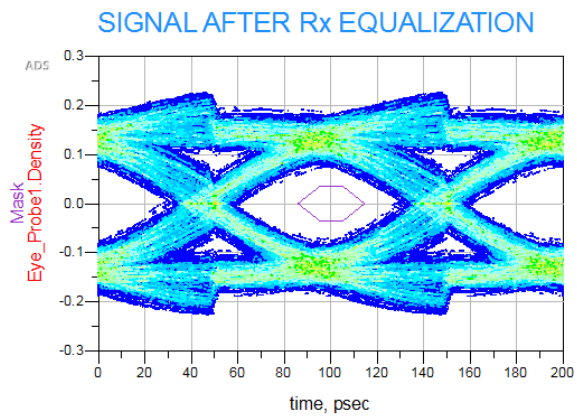
The Transmitter AMI model provides an option to enter the pre-shoot value in addition to de-emphasis. You can also enter Spread Spectrum Clock (SSC) parameters in this model. The Receiver AMI model provides an option to enter CTLE and DFE parameters along with the CDR parameters as well.

The bitrate is slightly less than 10Gbps to accommodate for SSC.

NOTE The number of bits being simulated has been kept at 1000 to reduce the size of the Compliance Test Bench. You should change it to 1000000 or more before running a practical simulation.

The resulting eye along with measurement summary is shown below. The second eye diagram also plots the Eye Mask for USB 3.1





measurement	EyeDiff_Probe1.Summary
Level1	0.000
Level0	0.000
Height	0.000
Width	0.000
JitterPP	0.000

measurement	Eye_Probe1.Summary
Level1	0.128
Level0	-0.128
Height	0.120
Width	6.164E-11
JitterPP	3.254E-11

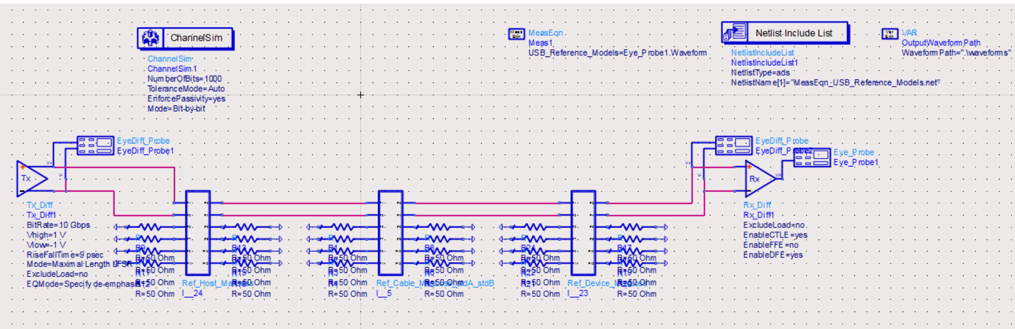
The waveform generated after receiver equalization is saved in the data/waveforms directory with the name *OutputAMI.h5*.

Channel Simulation Examples using USB 3.1 Reference Models

Reference Full LInk with Tx/Rx Models

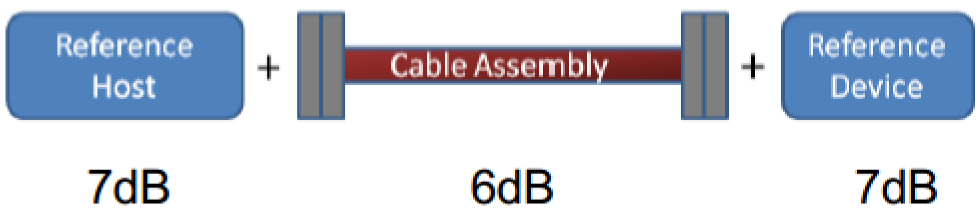
This design shows a typical USB connection from Transmitter to Receiver using the USB 3.1 standard reference models downloaded from the USB-IF website.

The host, cable and device models depicting maximum loss have been used in this design. Other reference models provided in the CTB can also be used in this design. Eye Probes placed in the circuit display the signal leaving the Transmitter, the signal reaching the Receiver and the signal after the Receiver Equalization.



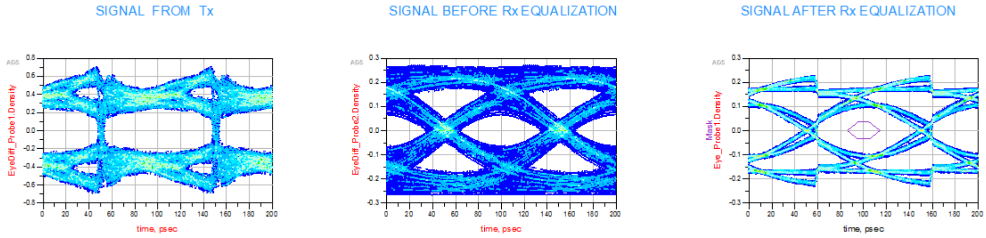
The Transmitter transmits PRBS data at a rate of 10Gbps and uses 128b132b encoding as per USB 3.1 standard. The Receiver Equalization consists of CTLE + DFE. This is in addition to de-emphasis applied at the Transmitter end.

The channel consists of a host, cable and device, each having maximum loss, as defined by the USB reference models.



NOTE The number of bits being simulated has been kept at 1000 to reduce the size of the Compliance Test Bench. You should change it to 1000000 or more before running a practical simulation.

The resulting eye along with measurement summary is shown below. The third eye diagram also plots the Eye Mask for USB 3.1, as specified in the USB 3.1 specs doc.



measurement	...eDiff_Probe1.Summary
Level1	0.371
Level0	-0.371
Height	0.440
Width	9.500E-11

measurement	EyeDiff_Probe2.Summary
Level1	0.166
Level0	-0.163
Height	0.099
Width	6.650E-11

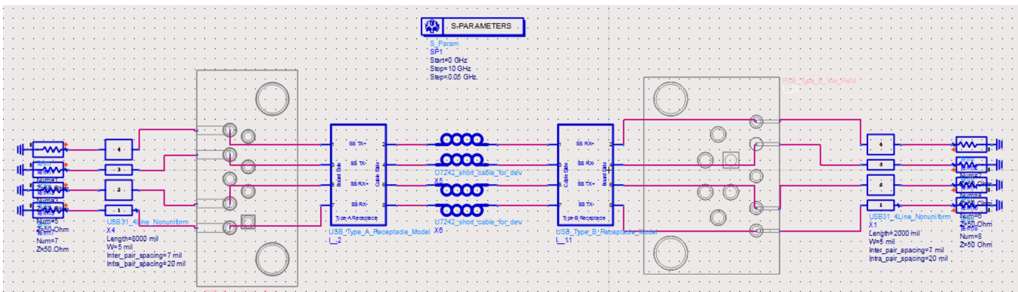
measurement	Eye_Probe1.Summary
Level1	0.138
Level0	-0.137
Height	0.178
Width	8.700E-11
JitterPP	1.300E-11

The waveform generated after receiver equalization is saved in the data/waveforms directory with the name *USB_Reference_Models.h5*.

S-Parameter Simulation Examples

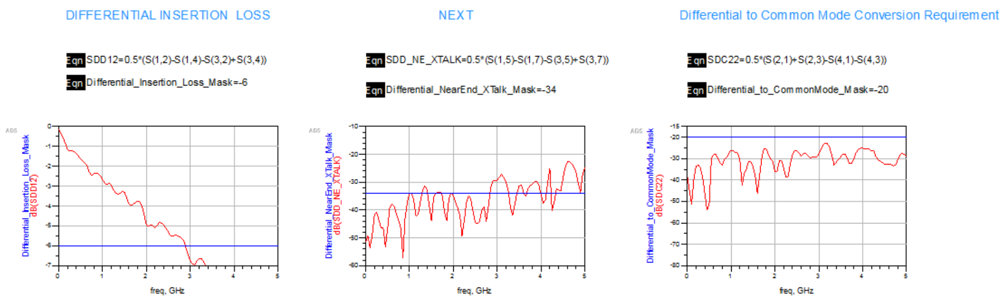
Full Link S-Parameter Modeling

This design shows the S-Parameter simulation of a typical USB connection. The results have been compared with the corresponding design targets.



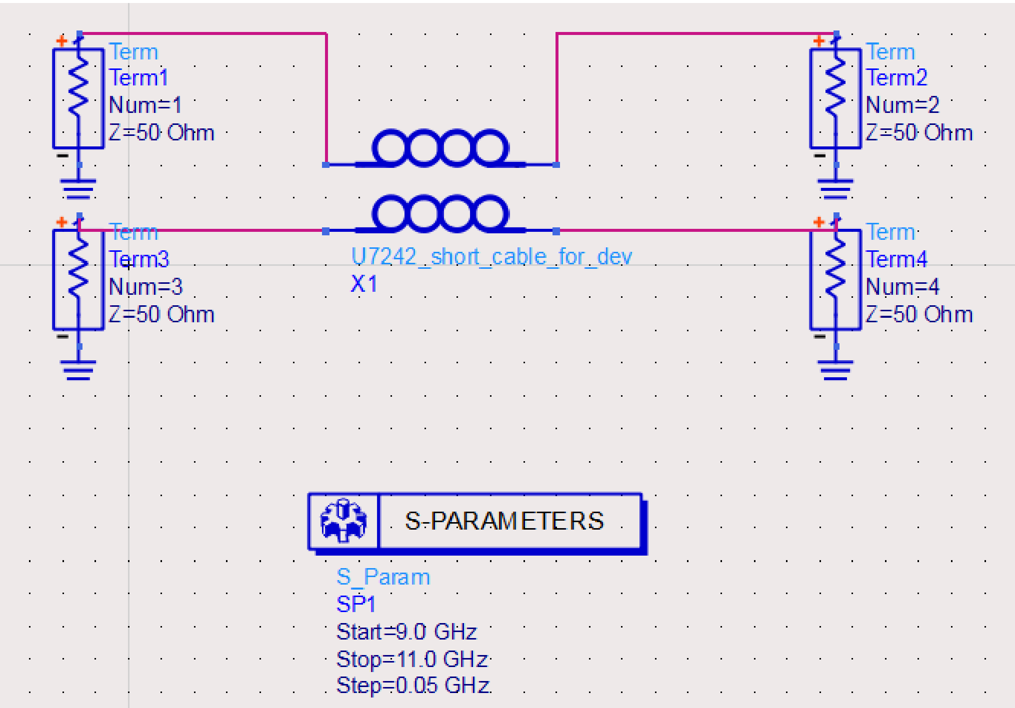
The Three Design Targets that are plotted in the results, along with their masks are -

1. Differential Insertion Loss
2. Near End CrossTalk(NEXT)
3. Differential to Common Mode Conversion Requirement

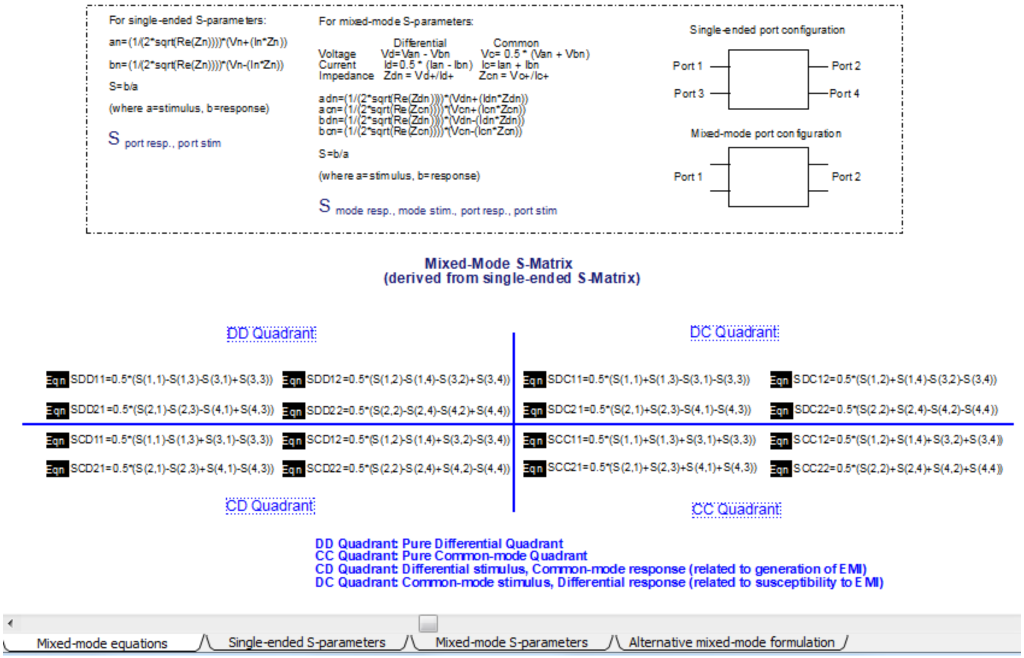


Mixed Mode S-Parameter Template

This design shows the relation between Single-Ended S-Parameters and Mixed Mode S-Parameters. You can run the simulation to see how mixed mode S-Parameters can be derived from Single-Ended S-Parameters.



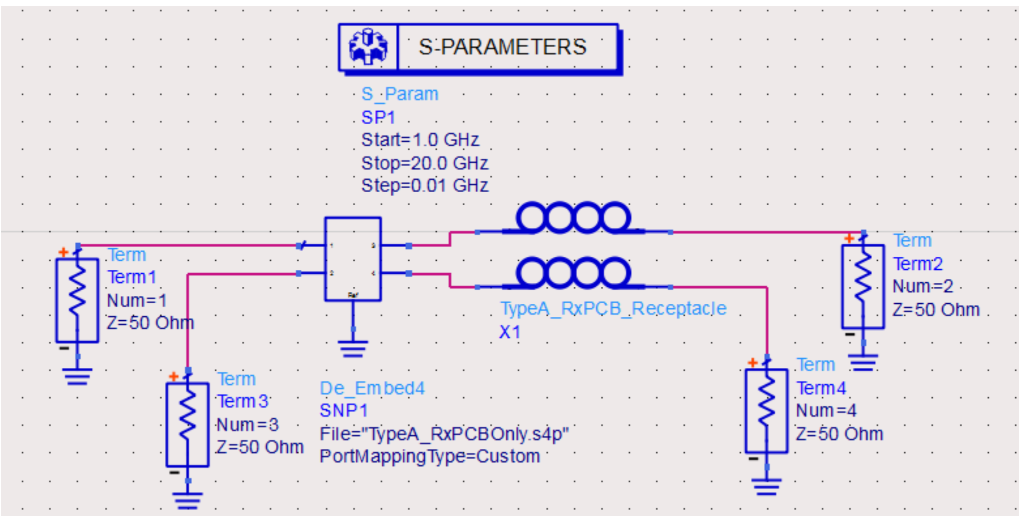
The results window for this design displays the Single Ended S-Parameters, Mixed Mode S-Parameters and also the formula to convert Single Ended S-parameters to Mixed Mode S-Parameters.



Frequency Domain De-embedding

This design shows the process of de-embedding. When you have a composite measurement of a DUT/fixture combination , you can isolate the performance of the fixture and use de-embedding to extract or de-embed the fixture from the measurements.

In this example, the effect of the Receiver PCB is removed using the ADS de-embed component, to get the performance of only the Receptacle.

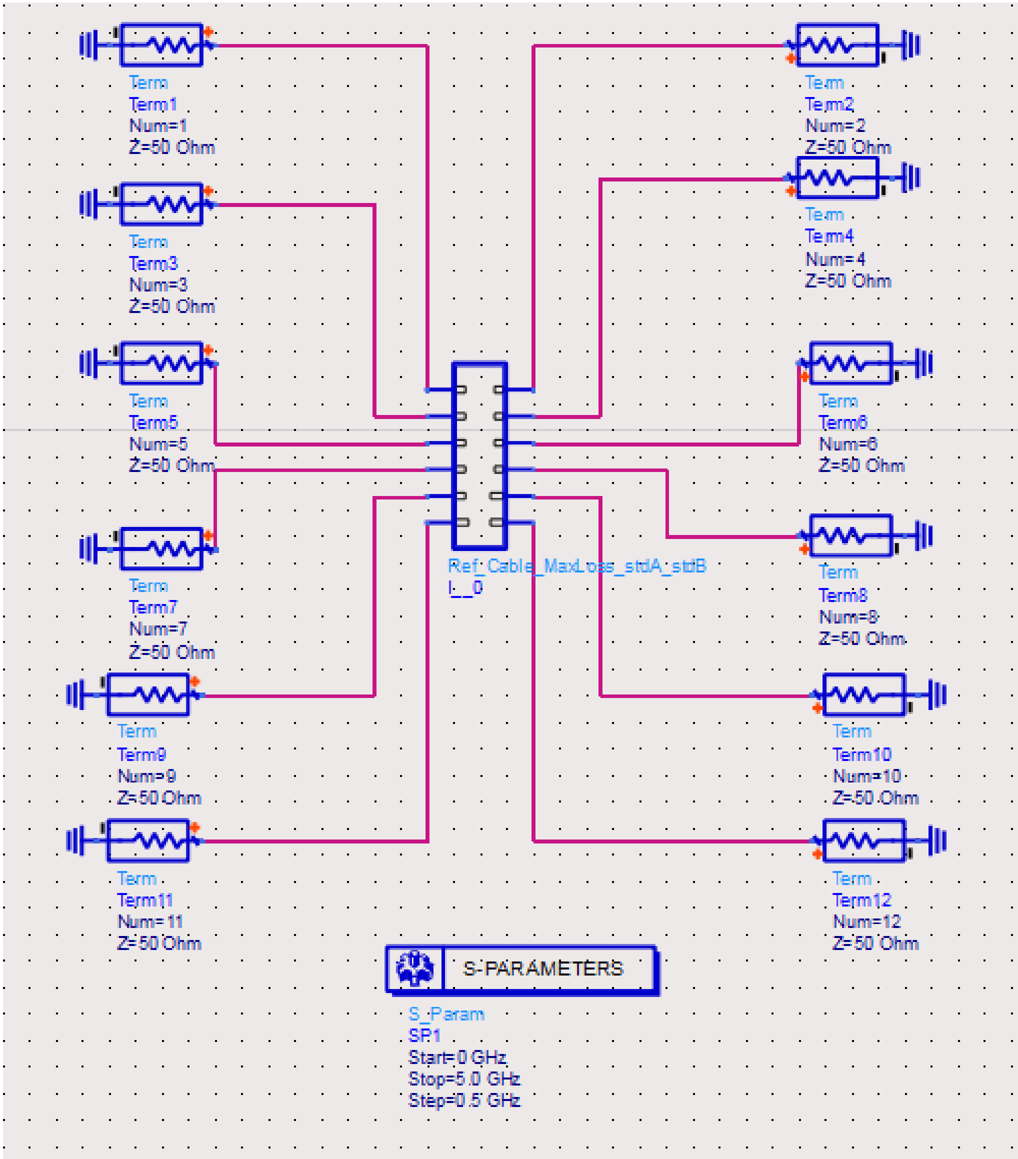


S-Parameter Simulation Examples using USB 3.1 Reference Model

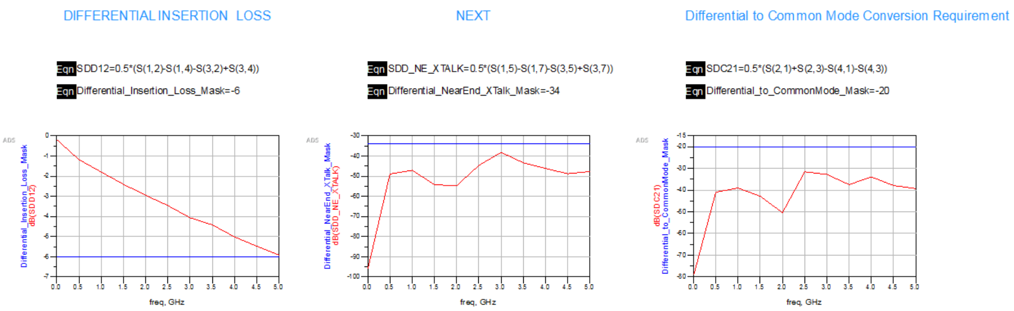
USB S-Parameter Mask Template

This design compares the Differential Insertion Loss, Near End CrossTalk and Differential to Common Mode Conversion Requirement of the USB 3.1 Standard Reference Cable model(downloaded from the USB-IF website), with the design targets as mentioned in the USB 3.1 specifications.

The reference cable model with maximum loss(as defined by USB 3.1 specifications) is used in this design.

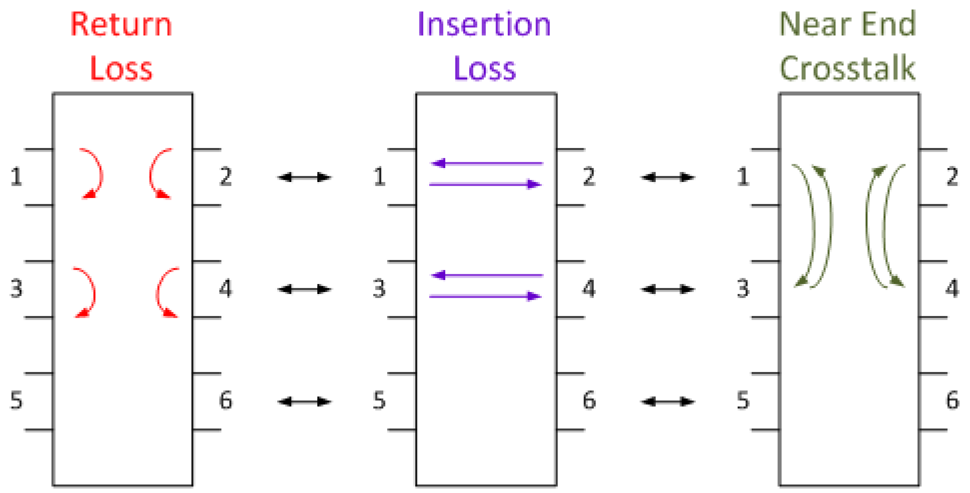


The results show that the model with maximum loss meet the specifications set by the USB 3.1 standard.

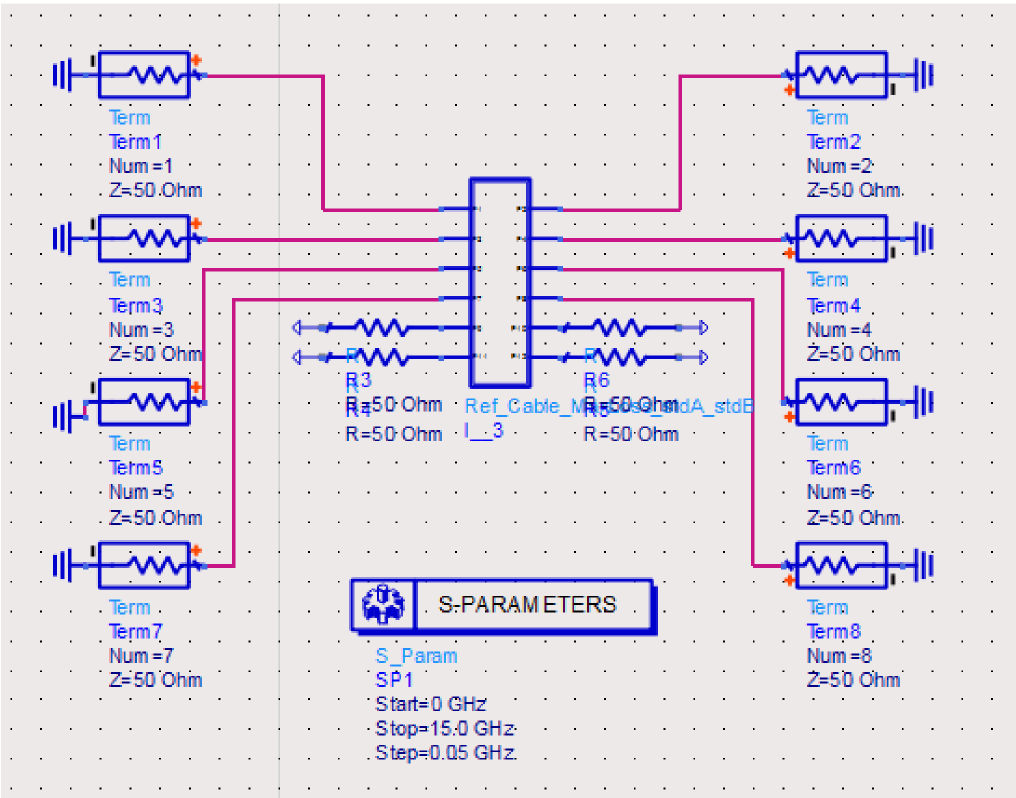


Differential S-Parameter for Cable/Host/Device

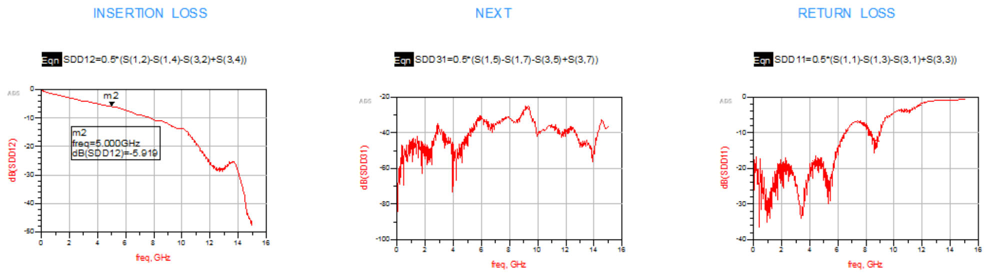
These three design finds the Insertion Loss(at Nyquist Frequency), Near End CrossTalk and Return Loss of the USB 3.1 standard reference Cable, Host and Device models (downloaded from the USB-IF website).



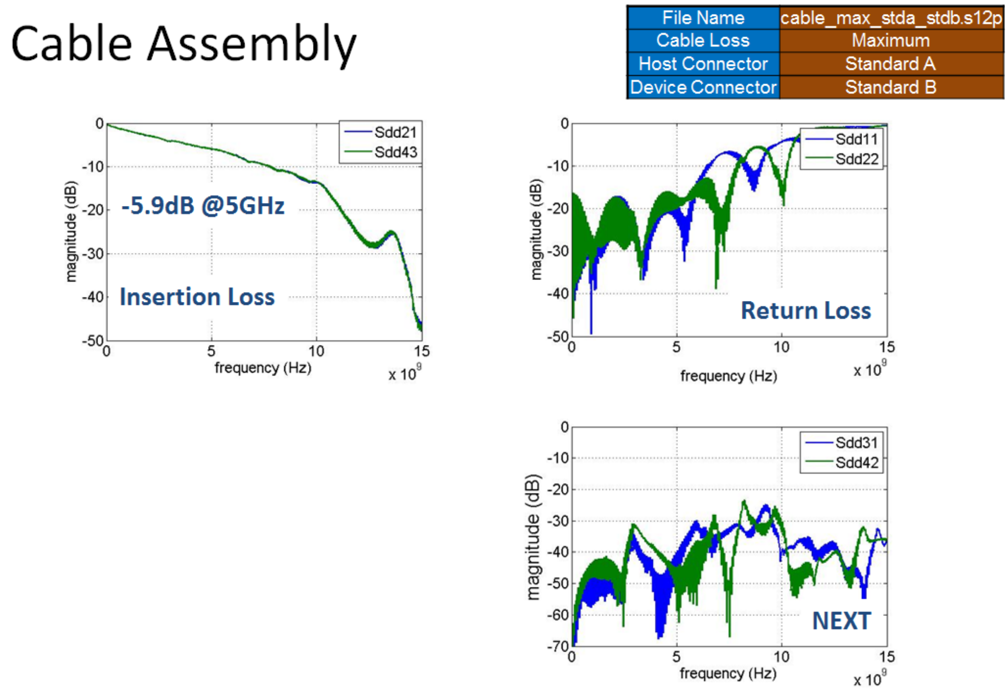
These values are in agreement with the frequency response shown in the presentation which is downloaded along with the USB 3.1 reference models. The reference cable design results are compared below.



ADS Results –



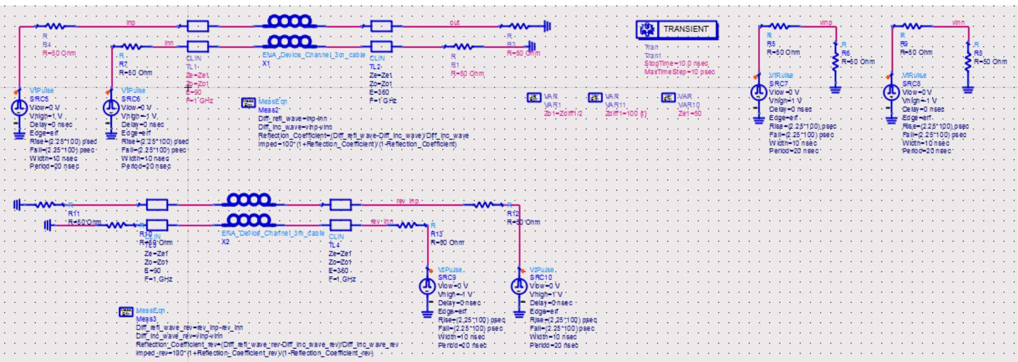
USB-IF Results -



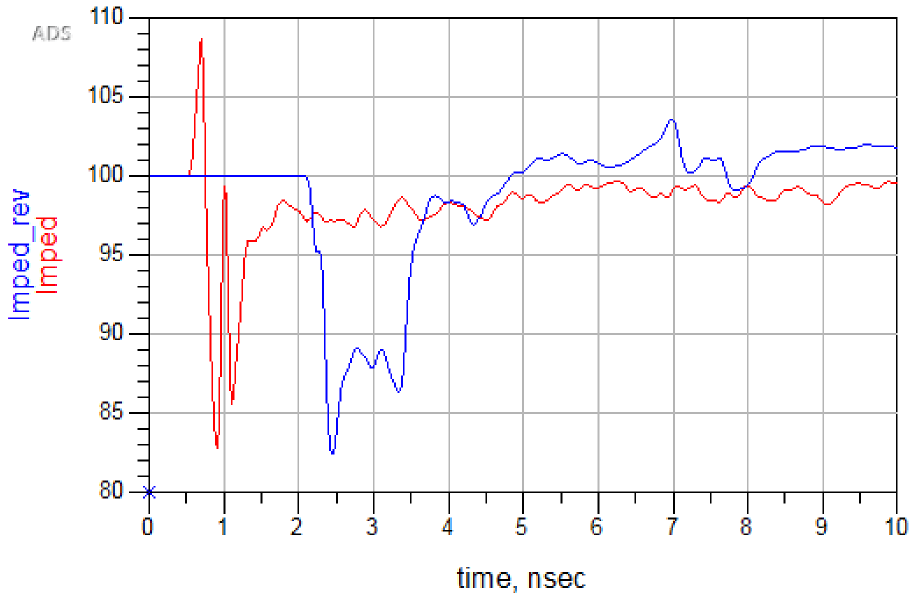
Transient Simulation Examples

TDR Simulation

This design shows the process of Time-Domain Reflectometry. Time-domain Reflectometry or TDR is a measurement technique used to determine the characteristics of electrical lines by observing reflected waveforms.



The result shows the impedance and the reverse impedance as seen by the signal.



References

- For Universal Serial Bus Specification 3.1, refer to USB_3_1_r1.0.pdf.
- For USB 3.1 Reference Channel Models, refer to Channel Model Usage - Heck - 2014-01-10.pptx.

Running USB 3.1 Compliance Tests on Infiniium Offline

Running USB 3.1 Compliance Tests on Infiniium Offline

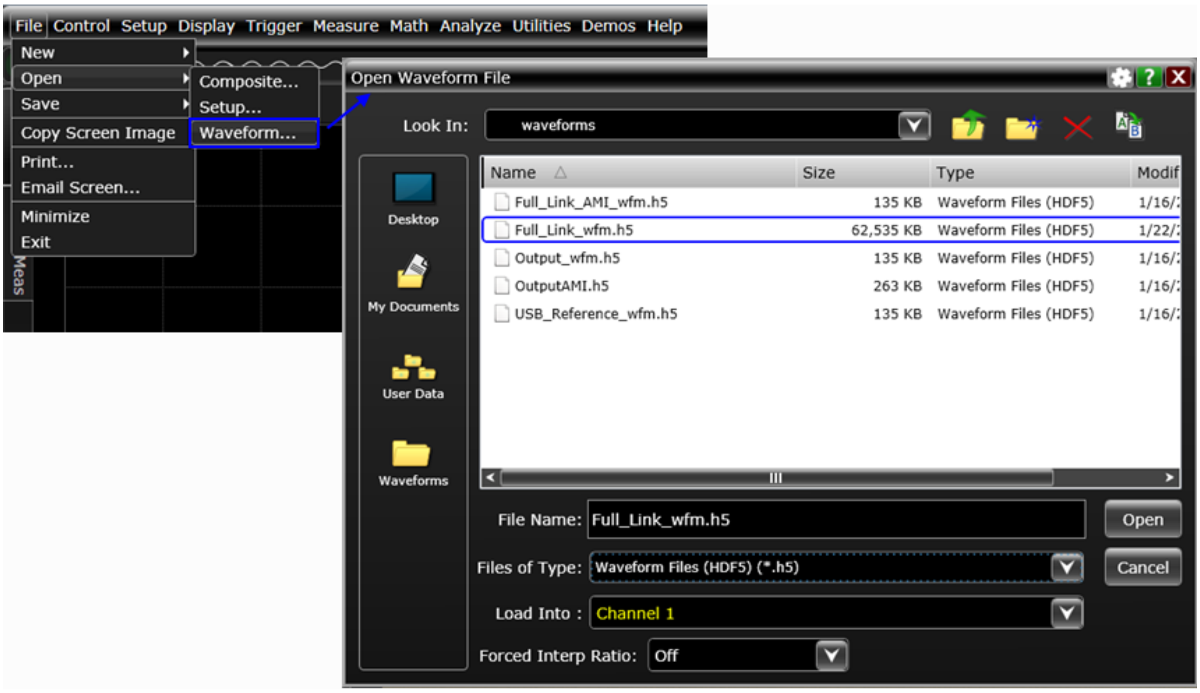
In the USB 3.1 Compliance Test Bench Simulation Setups section, all Channel Simulation examples generate signal waveforms in .h5 format. Using the Keysight Infiniium Offline software you can run the compliance test on these waveforms.

The waveform *Full_Link_wfm.h5* is used in this tutorial.

NOTE Ensure that the waveform is generated with number of bits simulated in the Channel Simulation Controller equal to or more than 1000000.

Version 2.0 of the Infiniium USB application supports only .wfm format for waveforms. So the *.h5* waveforms have to be converted to *.wfm* format before running the compliance tests.
To open the Waveform in Infiniium Offline software,

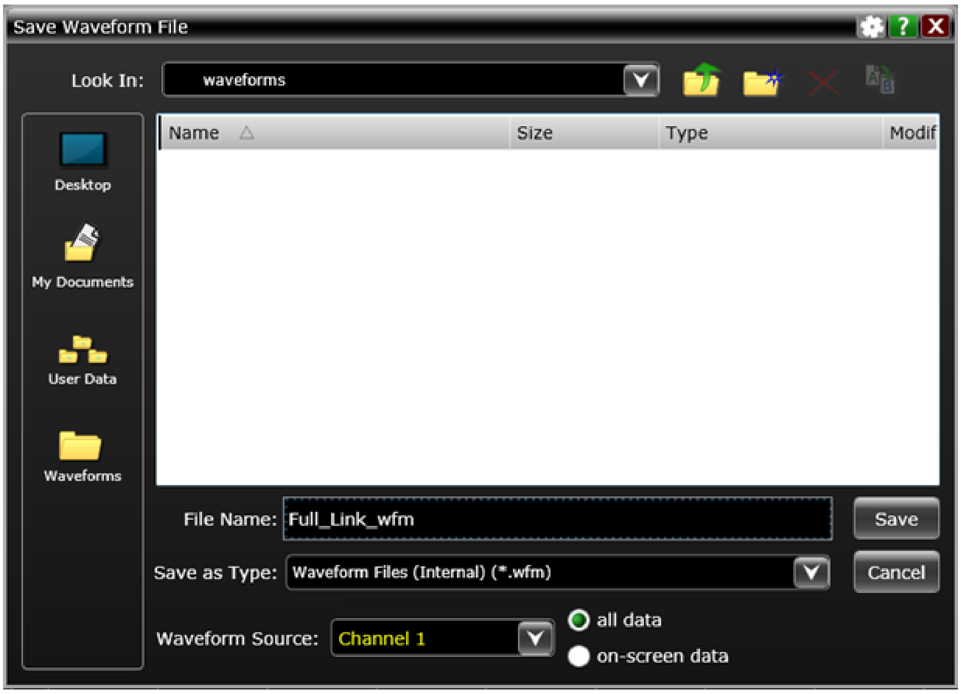
1. Start Infiniium offline.
2. Click **File** > **Open** > **Waveform...**.
   The Open Waveform File dialog box is displayed.



3. Click **Open**.
   The Waveform file is displayed.

4. Click **File** > **Save** > **Waveform** to save this in .wfm format.

   The Save Waveform File dialog box is displayed.

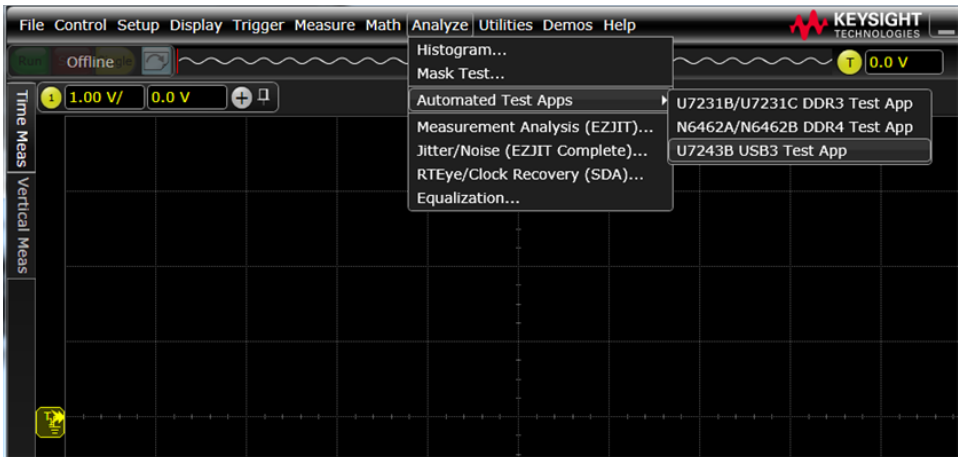


   Make sure to select 'all data' while saving the file.

5. Click **Save**.

To run the Compliance tests:

1. Click **Analyze** > **Automated Test Apps** > **U7243B USB3 Test App** from the Infiniium Offline software to open the USB application.

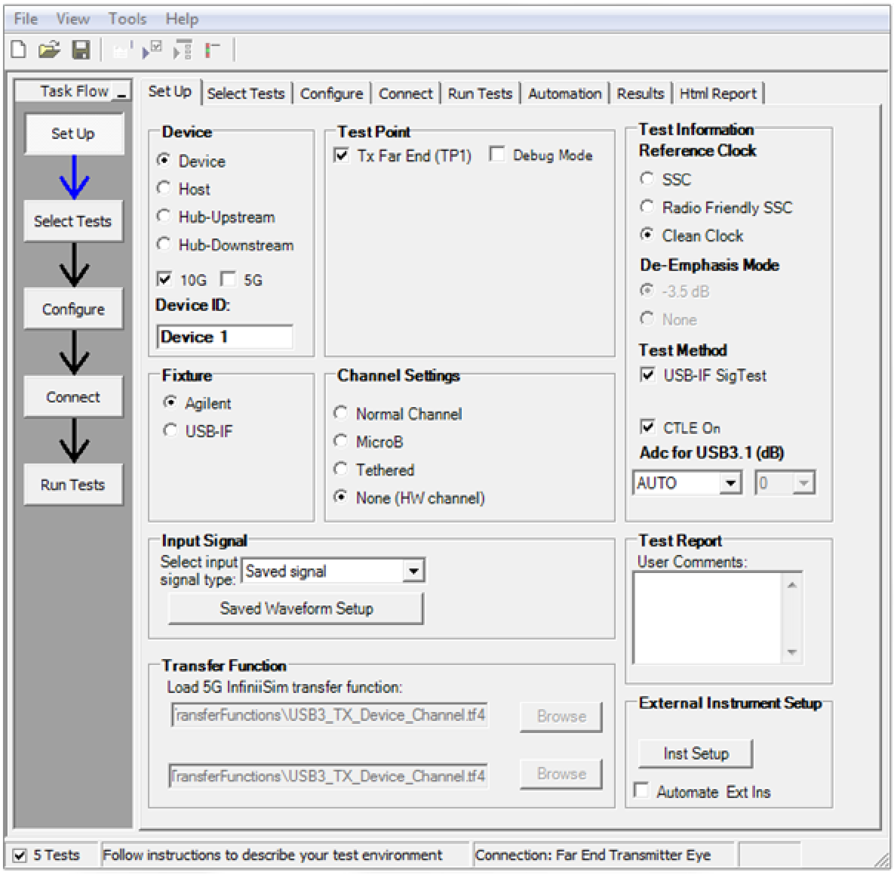


2. Under the **Setup** tab:

   a. Select **10G** Device

   b. Select **Clean Clock** Reference Clock

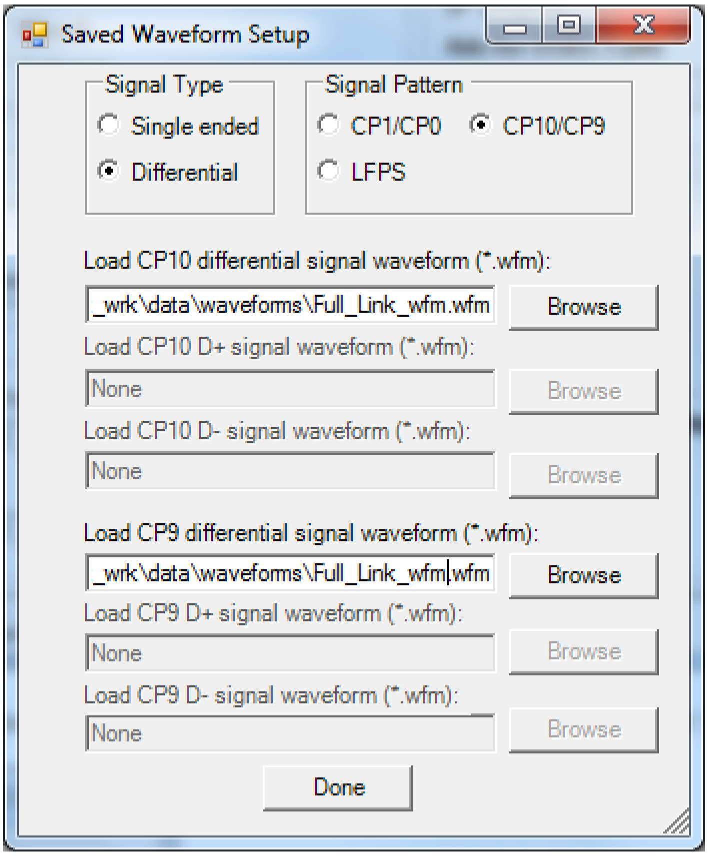
   c. Select **Saved Signal** from the Input Signal drop down.



3. Under the **Saved Waveform Setup** tab.

   a. Select Signal Type as **Differential.**

   b. Select Signal pattern as **CP10/CP9.**

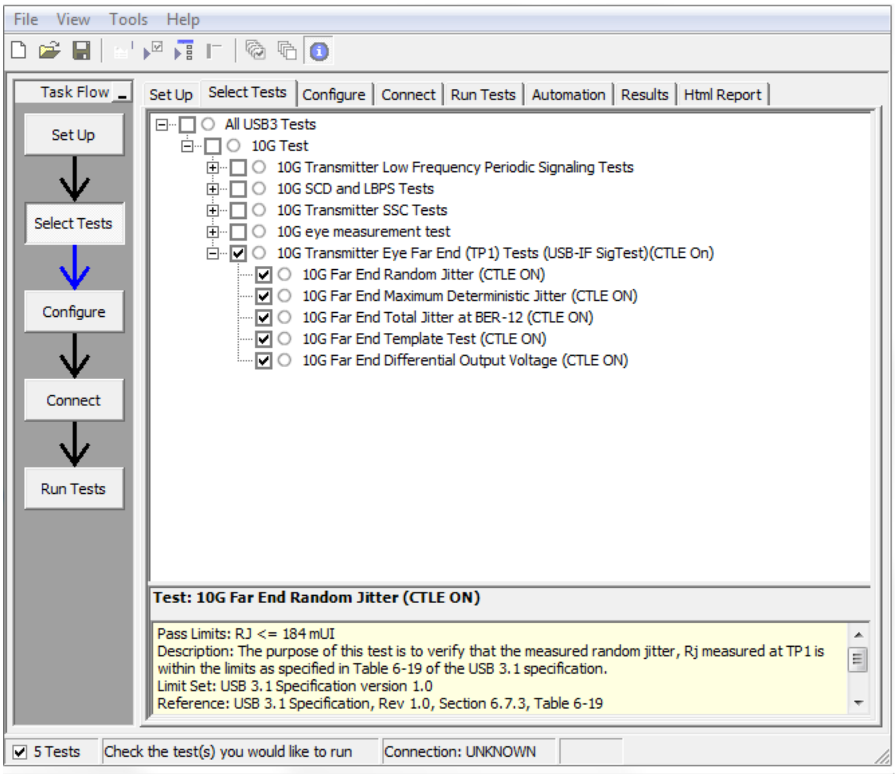


   c. Click **Browse** and select the *Full_Link_wfm.wfm.*

   d. Click **Done**.

4. Under the **Select Tests** tab

   a. Select all the TP1 tests.



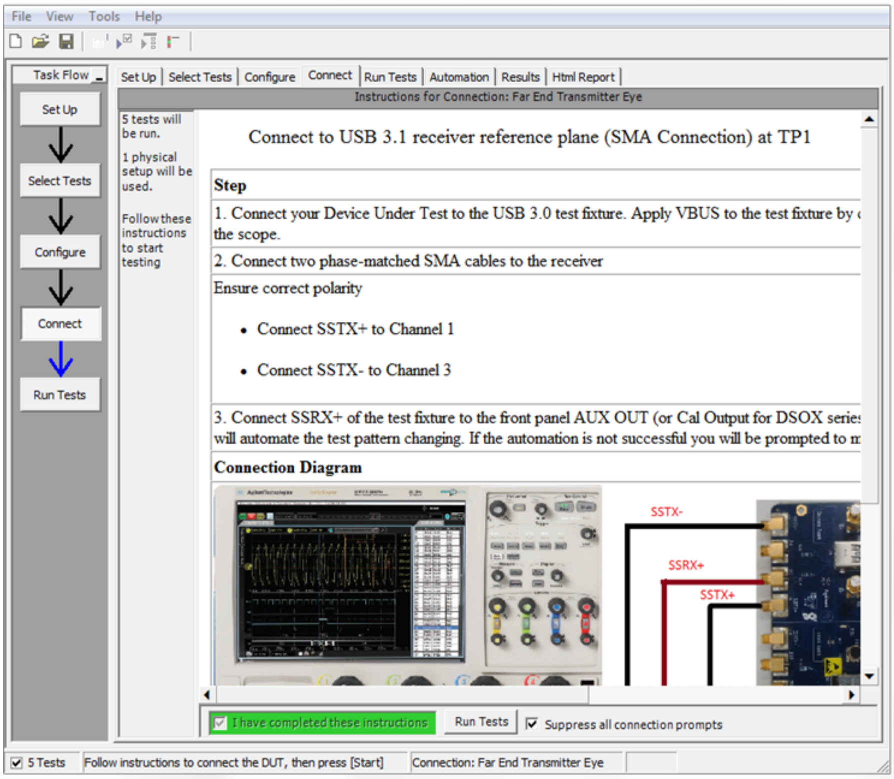
NOTE

The remaining tests in the application are not supported currently due to known issues in the software. They will be supported in a future release of this Compliance Test Bench and USB application.

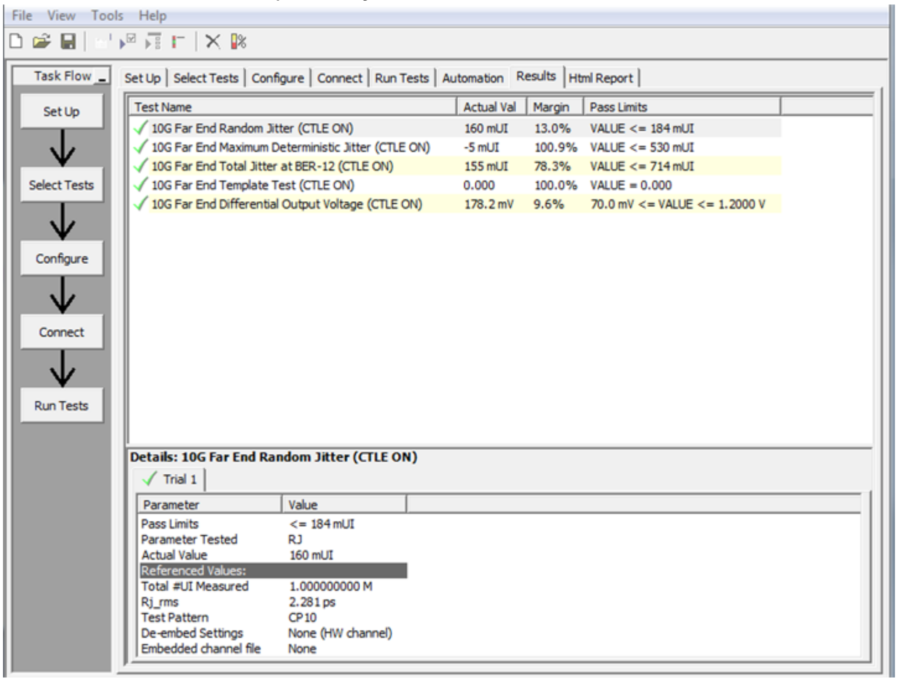
5. Under the **Connect** tab.

   a. Check **I have completed the instructions.**

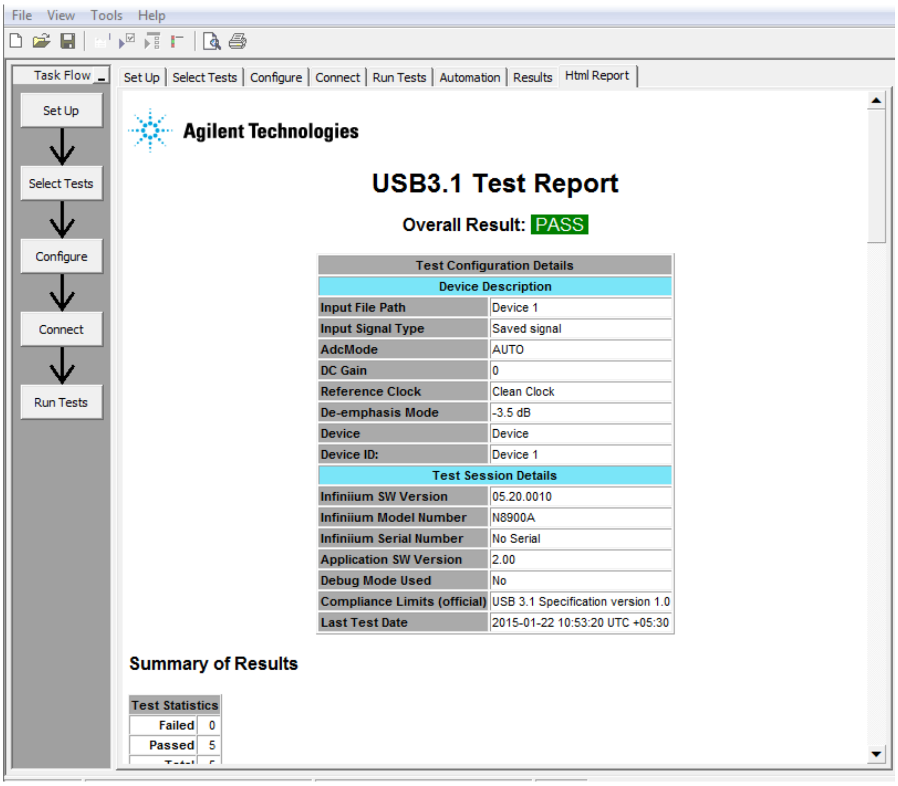
   b. Click **Run Tests**.



Once the tests are completed, you can view the test results under the **Results** tab.



You can also view the HTML report under the **HTML Report** tab.



References

- For Universal Serial Bus Specification 3.1, refer to USB_3_1_r1.0.pdf.
- For USB 3.1 Reference Channel Models, refer to Channel Model Usage - Heck - 2014-01-10.pptx.