NOTICE: This document contains references to Agilent Technologies. Agilent's former Test and Measurement business has become Keysight Technologies. For more information, go to **www.keysight.com.**

ADS
2016.01

DDR3 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.

Portions of the code Copyright (c) 1990-1996
The Regents of the University of California. All rights reserved. Permission is hereby granted, without written agreement and without license or royalty fees, to use, modify and distribute the Ptolemy software and its documentation for any purpose, provided that the above copyright notice and the following two paragraphs appear in all copies of the software and documentation.

In no event shall the University of California be liable to any party for direct, indirect, special, incidental, or consequential damages arising out of the use of this software and its documentation, even if the University of California has been advised of the possibility of such damage.

The University of California specifically disclaims any warranties, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The software provided hereunder is on an "as is" basis and the University of California has no obligation to provide maintenance, support, updates, enhancements, or modifications.

Portions of this product include the SystemC software licensed under Open Source terms, which are available for download at http://systemc.org/. This software is redistributed by Keysight. The Contributors of

the SystemC software provide this software "as is" and offer no warranty of any kind, express or implied, including without limitation warranties or conditions or title and non-infringement, and implied warranties or conditions merchantability and fitness for a particular purpose. Contributors shall not be liable for any damages of any kind including without limitation direct, indirect, special, incidental and consequential damages, such as lost profits. Any provisions that differ from this disclaimer are offered by Keysight only.

Motif is a trademark of the Open Software Foundation.

Netscape is a trademark of Netscape Communications Corporation.

UNIX is a registered trademark of the Open Group.

Microsoft, Windows, Windows NT, Windows 2000, and Windows Internet Explorer are registered trademarks of Microsoft Corporation in the United States and other countries.

Pentium is a registered trademark of Intel Corporation.

Acrobat, PDF, and PostScript are registered trademarks of Adobe Systems Incorporated.

FLEXlm and FLEXnet are registered trademarks of Flexera Software LLC Terms of Use for Flexera Software information can be found at http://www.flexerasoftware.com/company/about/terms.htm

Netscape Portable Runtime (NSPR), Copyright (c) 1998-2003 The Mozilla Organization. A copy of the Mozilla Public License is at http://www.mozilla.org/MPL/

FFTW, The Fastest Fourier Transform in the West, Copyright (c) 1997-1999 Massachusetts Institute of Technology. All rights reserved.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Cadence, Allegro, Assura, Dracula, SKILL, Spectre, and Virtuoso are registered trademarks of Cadence Design Systems, Inc. in the United States and/or other jurisdictions.

Mentor, Mentor Graphics, Board Station, Calibre, and Expedition are registered trademarks of Mentor Graphics Corporation in the United States and other countries.

SystemC is a registered trademark of Open SystemC Initiative, Inc. in the United States and other countries and is used with permission.

Gradient, HeatWave and FireBolt are trademarks of Gradient Design Automation Inc.

The following third-party libraries are used by the NlogN Momentum solver:

Metis 4.0, Copyright © 1998, Regents of the University of Minnesota", http://www.cs.umn.edu/~metis, METIS was written by George Karypis (karypis@cs.umn.edu).

Intel@ Math Kernel Library, http://www.intel.com/software/products/mkl

HSPICE is a registered trademark of Synopsys, Inc. in the United States and/or other countries.

DWG and DXF are registered trademarks of Autodesk, Inc. in the United States and/or other countries.

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 Version 4.8.4, Copyright (C) 2014 Digia Plc and/or its subsidiary(-ies). All Rights Reserved. Contact: http://www.qt-project.org/legal

Qt License: http://qt-project.org/doc/qt-4.8/lgpl.html. Your use or distribution of Qt or any modified version of Qt 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.

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 Copyright: Copyright 2006-2013 by The HDF Group.

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

Copyright 2006-2013 by The HDF Group.

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

Redistribution and use in source and binary forms, with or without modification, are permitted for any purpose (including commercial purposes) 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 materials provided with the distribution.
3. In addition, redistributions of modified forms of the source or binary code must carry prominent notices stating that the original code was changed and the date of the change.
4. All publications or advertising materials mentioning features or use of this software are asked, but not required, to acknowledge that it was developed by The HDF Group and by the National Center for Supercomputing Applications at the University of Illinois at Urbana-Champaign and credit the contributors.
5. Neither the name of The HDF Group, the name of the University, nor the name of any Contributor may be used to endorse or promote products derived from this software without specific prior written permission from The HDF Group, the University, or the Contributor, respectively.

libpng

libpng Copyright: libpng versions 1.2.6, August 15, 2004, through 1.6.3, July 18, 2013, are Copyright (c) 2004, 2006-2013.

libpng License: This copy of the libpng notices is provided for your convenience. In case of any discrepancy between this copy and the notices in the file png.h that is included in the libpng distribution, the latter shall prevail.
COPYRIGHT NOTICE, DISCLAIMER, and LICENSE:
If you modify libpng you may insert additional notices immediately following this sentence.

This code is released under the libpng license.
libpng versions 1.2.6, August 15, 2004, through 1.6.3, July 18, 2013, are Copyright (c) 2004, 2006-2013 Glenn Randers-Pehrson, and are distributed according to the same disclaimer and license as libpng-1.2.5 with the following individual added to the list of Contributing Authors, Cosmin Truta

libpng versions 1.0.7, July 1, 2000, through 1.2.5 - October 3, 2002, are Copyright (c) 2000-2002 Glenn Randers-Pehrson, and are distributed according to the same disclaimer and license as libpng-1.0.6 with the following individuals added to the list of Contributing Authors
Simon-Pierre Cadieux, Eric S. Raymond, Gilles Vollant and with the following additions to the disclaimer:
There is no warranty against interference with your enjoyment of the library or against infringement. There is no warranty that our efforts or the library will fulfill any of your particular purposes or needs. This library is provided with all faults, and the entire risk of satisfactory quality, performance, accuracy, and effort is with the user.

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.

3. All advertising materials mentioning features or use of this software must display the following acknowledgment: "This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (http://www.openssl.org/)"
4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact openssl-core@openssl.org.
5. Products derived from this software may not be called "OpenSSL" nor may "OpenSSL" appear in their names without prior written permission of the OpenSSL Project.
6. Redistributions of any form whatsoever must retain the following acknowledgment: "This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org/)"

THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT "AS IS" AND ANY EXPRESSED 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 OpenSSL PROJECT OR ITS 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.
==========

This product includes cryptographic software written by Eric Young (eay@cryptsoft.com). This product includes software written by Tim Hudson (tjh@cryptsoft.com).

Original SSLeay License

Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com)
All rights reserved.

This package is an SSL implementation written by Eric Young (eay@cryptsoft.com).
The implementation was written so as to conform with Netscapes SSL. This library is free for commercial and non-commercial use as long as the following conditions are aheared to. The following conditions apply to all code found in this distribution, be it the RC4, RSA, lhash, DES, etc., code; not just the SSL code. The SSL documentation included with this distribution is covered by the same copyright terms except that the holder is Tim Hudson (tjh@cryptsoft.com).

Copyright remains Eric Young's, and as such any Copyright notices in the code are not to be removed. If this package is used in a product, Eric Young should be given attribution as the author of the parts of the library used. This can be in the form of a textual message at program startup or in documentation (online or textual) provided with the package.

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 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. 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

DDR3 Compliance Test Bench

This section describes the following topics:

DDR3 Simulation Setup

DDR3 Simulation Setup

There are two simulation setups for DDR3, which are prefixed with `WaveformBridge`:

- DDR3_CA: Denotes command and address bus simulation
- DDR3_DQ_Write: Denotes data bus simulation in ***WRITE*** mode, that is, data transfer from memory controller to DRAM

These simulation setups have five common blocks: Simulation Engine, Basic Variable definition, Equations for Compliance Trace Export, PCB Substrate Definition, and IBIS Alias Parameter Definition, as shown in the following figure:

ddr3_sim.svg

ddr3_basic.svg

ddr3_equ.svg

ddr3_substrate.svg

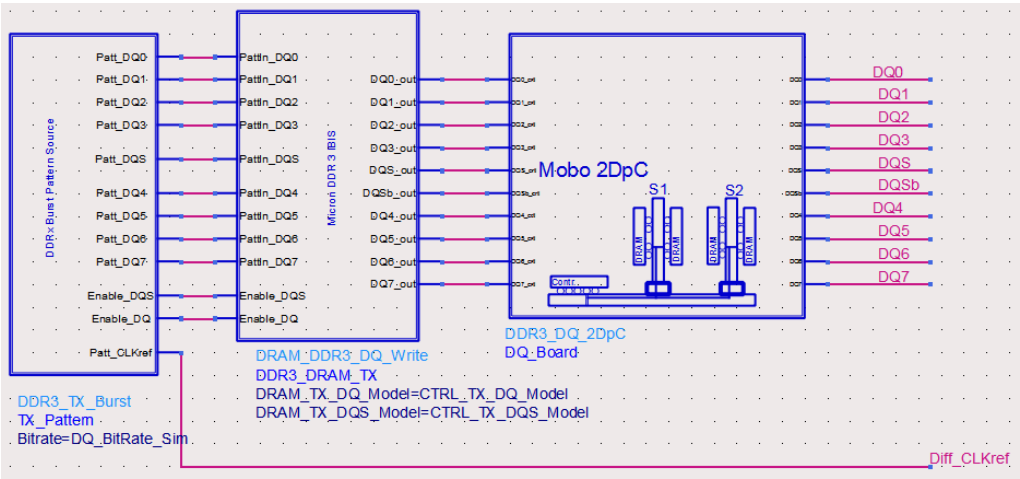
ddr3_allg.svg

A simulation runs in the following sequence:

- Variables in VarEqn blocks are calculated as pre-processing equations
- Transient simulation run from StartTime to StopTime
- Variables in MeasEqn blocks, AEL expression equations in Netlist Include block are calculated as post-processing equations.

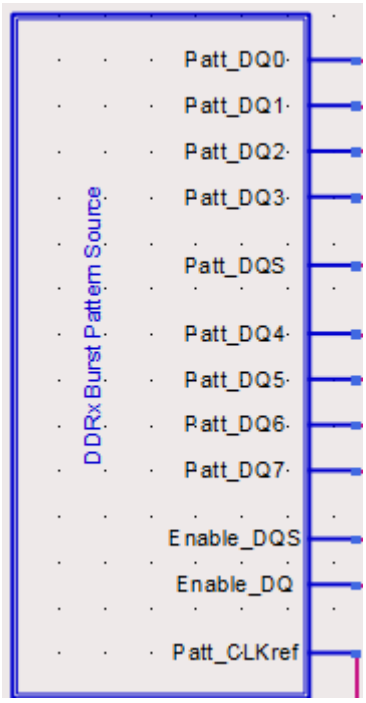
DDR3_DQ_Write schematic

A top-level schematic for DDR3 DQ Write is shown in the following figure. The blocks from left to right are: DQ Burst Pattern Generator, Controller DQ Driver Pins, Mother Board PCB + 2 DIMMs per Channel DDR3 Memory.



Block details used in this schematic are explained in the subsequent sections.

DQ Burst Pattern Generator

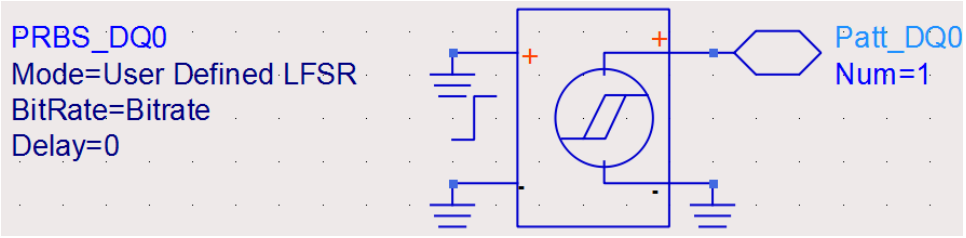


The DQ burst pattern generator produces the following patterns:

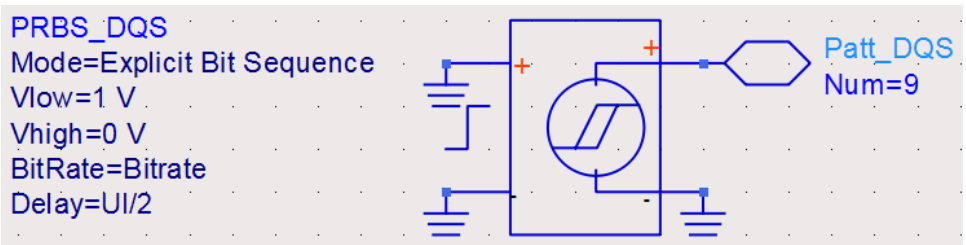
- 8 pseudo-random data (DQ) patterns labeled as Patt_DQ0 to Patt_DQ7,
- 1 repetitive strobe (DQS) pattern labeled as Patt_DQS,
- 1 repetitive clock pattern labeled as Patt_CLKref,
- 2 pulse patterns to enable DQ and DQS bursts

Inside the pattern generator schematic, there are primitive models from ADS Analog Lib:

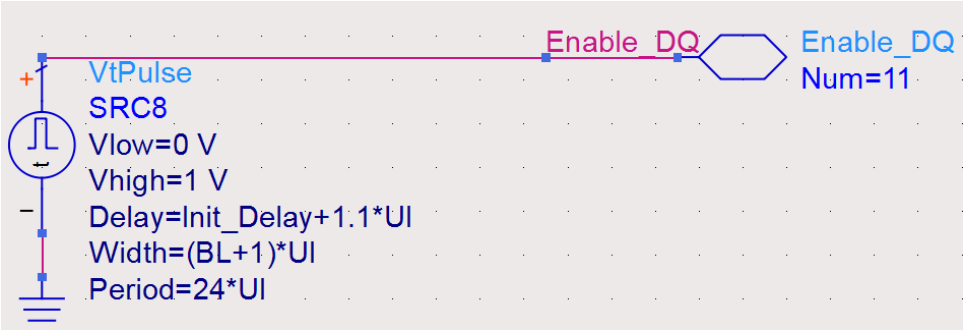
- Pseudo random bit sequence (PRBS) sources with user-defined linear feedback shift register (LFSR) are used to generate the DQ pattern, with each DQ bit having different LFSR taps.



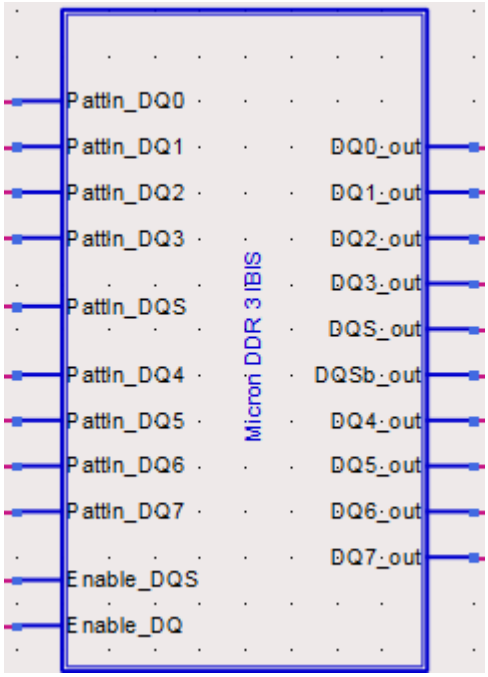
- PRBS source with explicit bit sequence is used to generate the repetitive DQS pattern with preamble



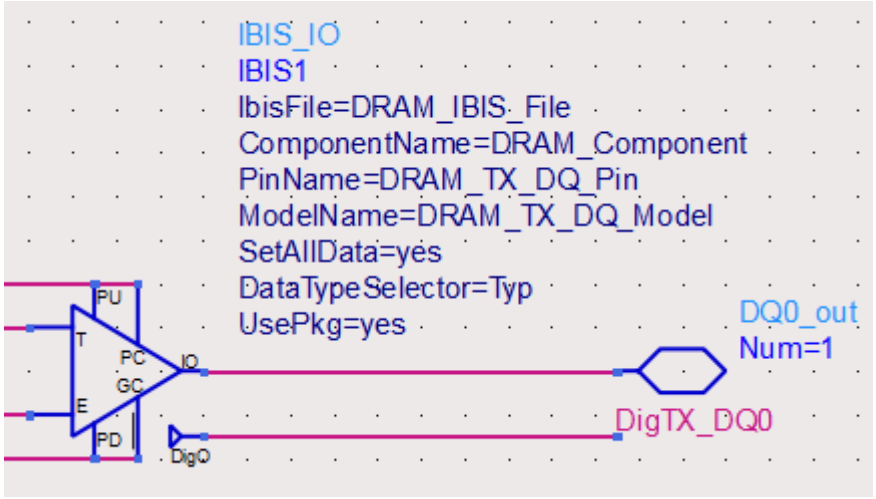
- Pulse source is used to generate a pulse pattern to enable DQ and DQS bursts



DQ Driver



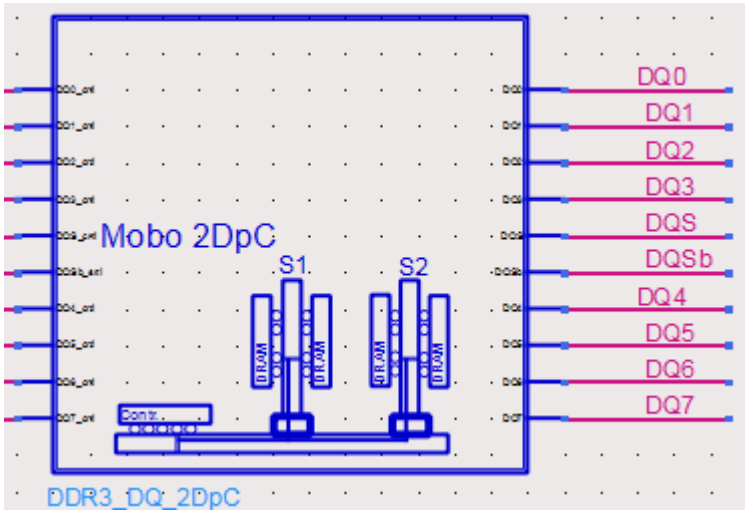
The DQ driver model takes a pulse pattern as an input, and generates a waveform as an output that mimics the driver circuit output at memory controller I/O pad. Inside the drive schematic, there is an IBIS I/O model for each pin, as shown in the following figure:



The IBIS parameters uses alias names defined in top level schematic.

NOTE The IBIS model used in this example is for illustration purpose only. You must use the IBIS model or SPICE model provided by your controller vendor to get good simulation accuracy for your system.

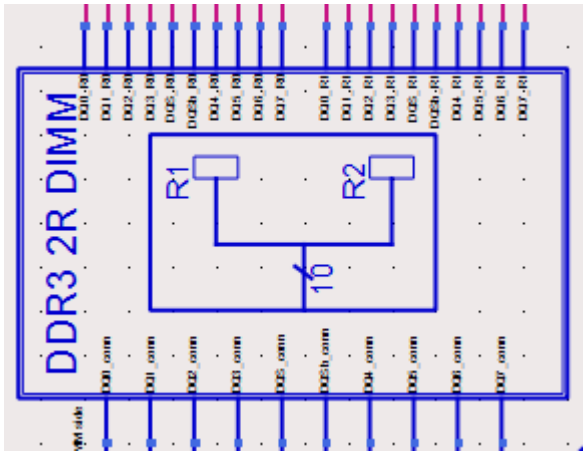
Mother Board (Mobo) PCB with 2 DIMMs per Channel Memory (2DpC)



This sub-circuit contains the motherboard PCB traces, DIMM PCB traces, DIMM sockets, and DRAM receiver models having several levels of sub-hierarchies.

ddr3_2dpc.svg

Slot0 and Slot1 are the 2 DIMM slots, where the DIMM socket is represented by S-Parameter blocks. The PCB traces between Slot 0 and Slot 1 are the DIMM traces, and the PCB traces on the left side of Slot 0 are the motherboard traces. Each DIMM has 2 ranks of memory on it, as shown in the following sub-circuit:



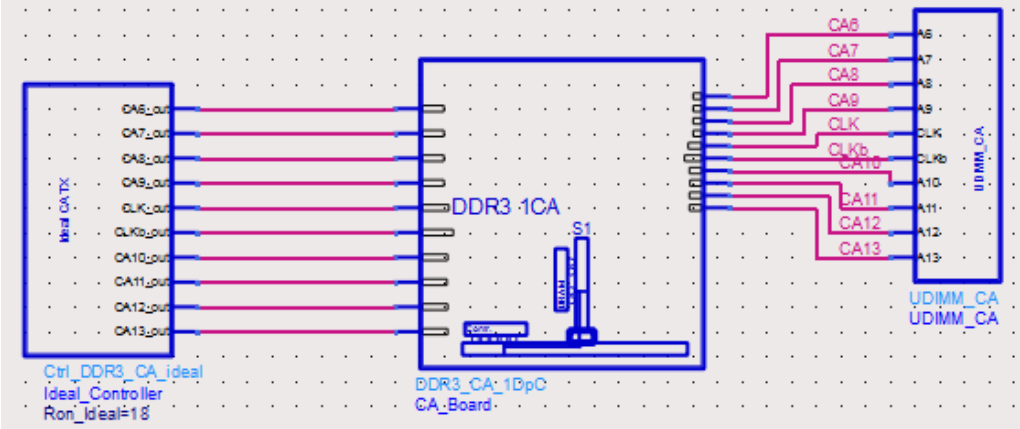
ddr3_2rank_dimm.svg

The 2 blocks on the right-hand side are the DRAM devices in rank-0 and rank-1, respectively.

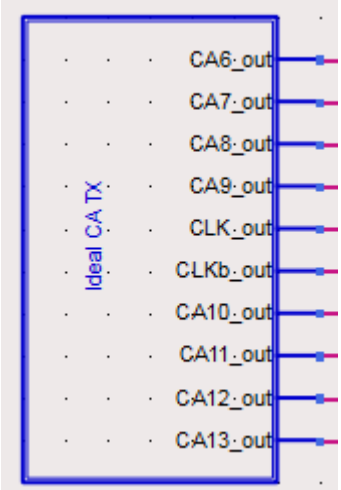
Selecting the correct DIMM and rank for DRAM output is accomplished by the ideal switches in DDR3 2DpC schematic.

DDR3_CA schematic

A top-level schematic for DDR3 CA (command and address bus) simulation is shown in the following figure. The blocks from Left to Right are – CA Pattern Generator › PCB Interconnects › UDIMM Receiver. UDIMM stands for Un-buffered Dual Inline Memory Module. There are other DIMM types, such as RDIMM (Registered DIMM) and LRDIMM (Load Reduced DIMM), which are not discussed here.



CA Pattern Generator

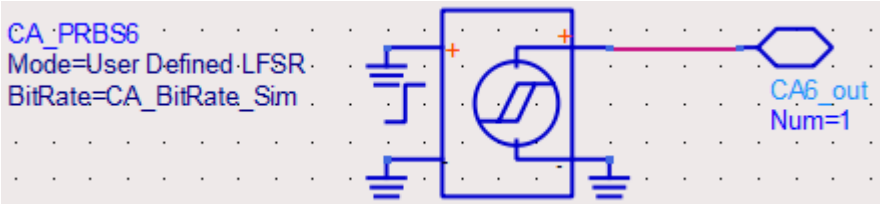


The CA pattern generator produces the following patterns:

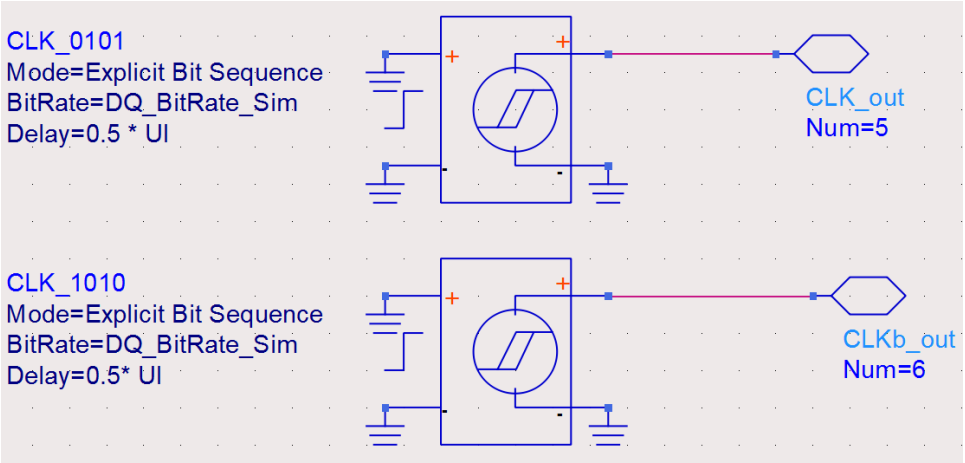
- 8 pseudo-random pulse patterns labeled as CA6_out to CA13_out,
- 1 repetitive clock pattern labeled as CLK_out (+) and CLKb_out(-),

Inside the CA pattern generator schematic, there are primitive models from ADS Analog Lib:

- Pseudo random bit sequence (PRBS) sources with user-defined linear feedback shift register (LFSR) are used to generate the CA pattern, with each CA bit having different LFSR taps

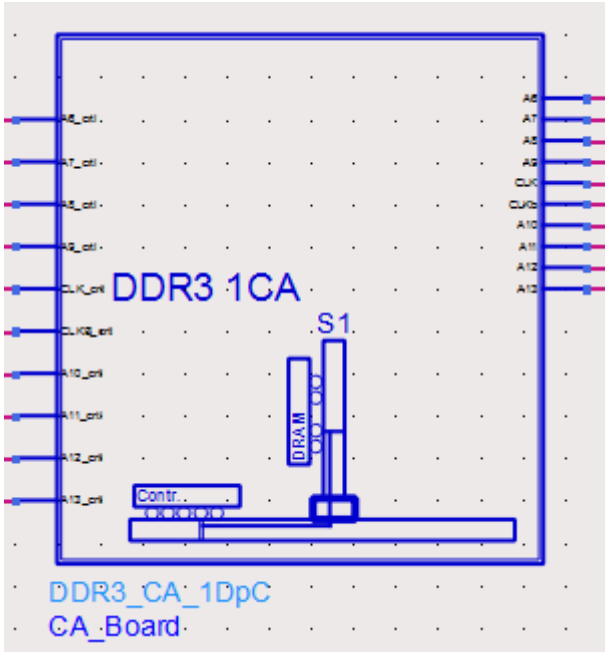


- PRBS source with explicit bit sequence is used to generate the repetitive CLK pattern

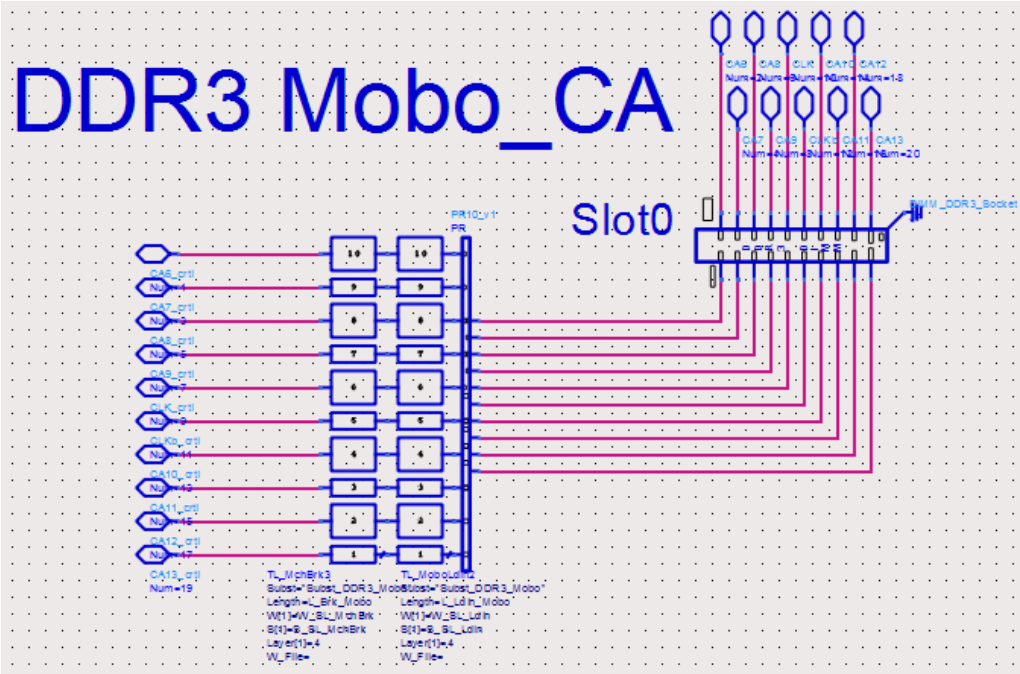


- Unlike DQ/DQS patterns that are either WRITE burst or READ burst, CA pattern is continuous, and transmitted in uni-direction from controller driver pin to DRAM receiver pin

PCB Interconnects

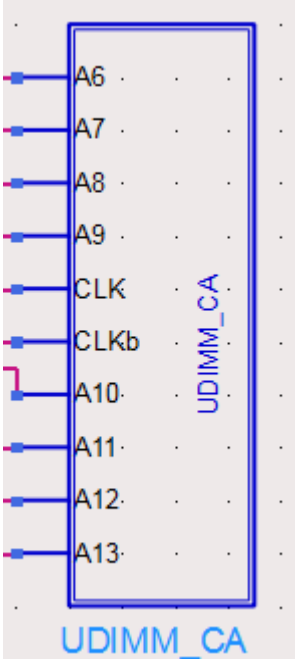


This is a sub-circuit including motherboard PCB traces and DIMM socket S-parameter block, as shown in the following figure:

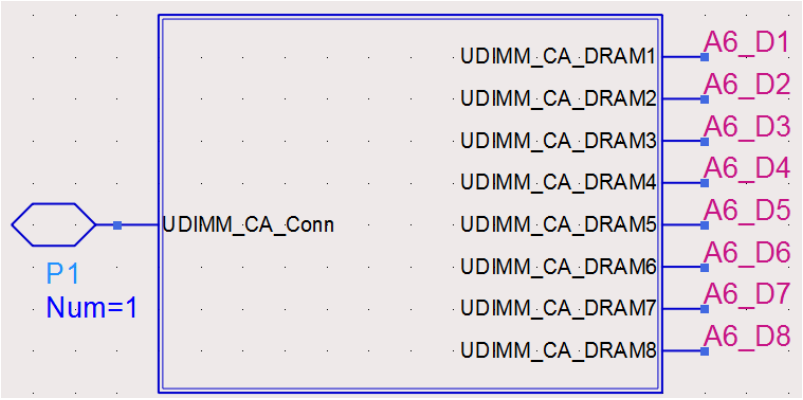


This example is for illustration purpose only. For your own DDR3 system simulation, ensure to modify this example to match the topology of your system design.

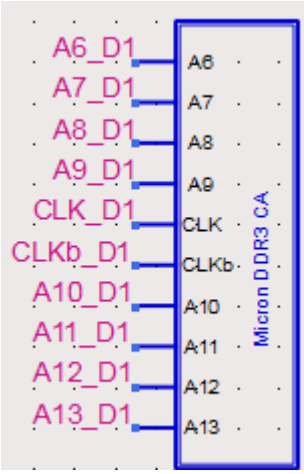
DRAM receivers on UDIMM module



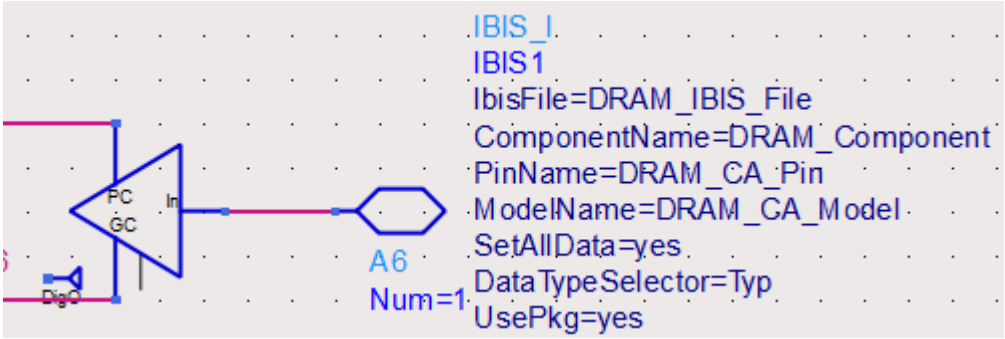
This UDIMM sub-circuit takes the clock and the address signals as an input, and re-distribute each signal to 8 DRAM devices on the UDIMM. For example, address signal A6 is split into 8 paths as shown in the following figure.



These address signals are then connected to the 8 DRAM devices on UDIMM module, as shown in the following figure.



Inside each DRAM device, a receiver pin is terminated by an IBIS receiver model.



Introduction to DDR3 Compliance Test Bench

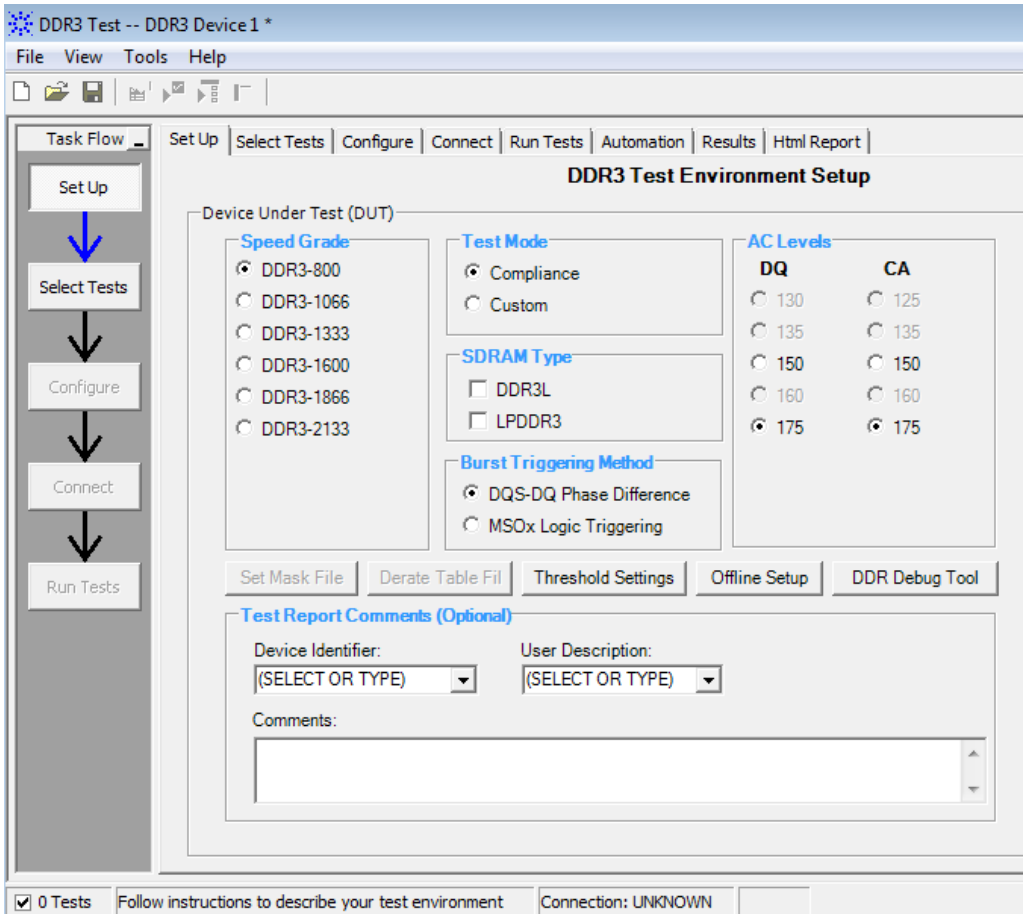
Introduction to DDR3 Compliance Test Bench

DDR3 compliance test bench contains simulation setups for DDR3 (1.5V), and LPDDR3 (1.2V) devices. The workflow for DDR3 and LPDDR3 compliance test is:

- Run ADS simulation to generate waveforms in hdf5 format, with `.h5` file extension
- Use ADS generated waveforms in DDR3 Compliance Test Application to perform tests

About Infiniium Offline DDR3 Compliance Test Application

The Infiniium Offline DDR3 compliance test application provides accurate and detailed verification of DDR3 memory interfaces to ensure compliance with JDEC specification. The offline mode runs on Windows 7 PCs with ADS simulated waveforms or stored waveforms captured in oscilloscope measurement. The DDR3 compliance test application contains tests for different SDRAM types, that is, DDR3 (1.5V), DDR3L(1.35V), and LPDDR3 (1.2V), as shown in the following figure.



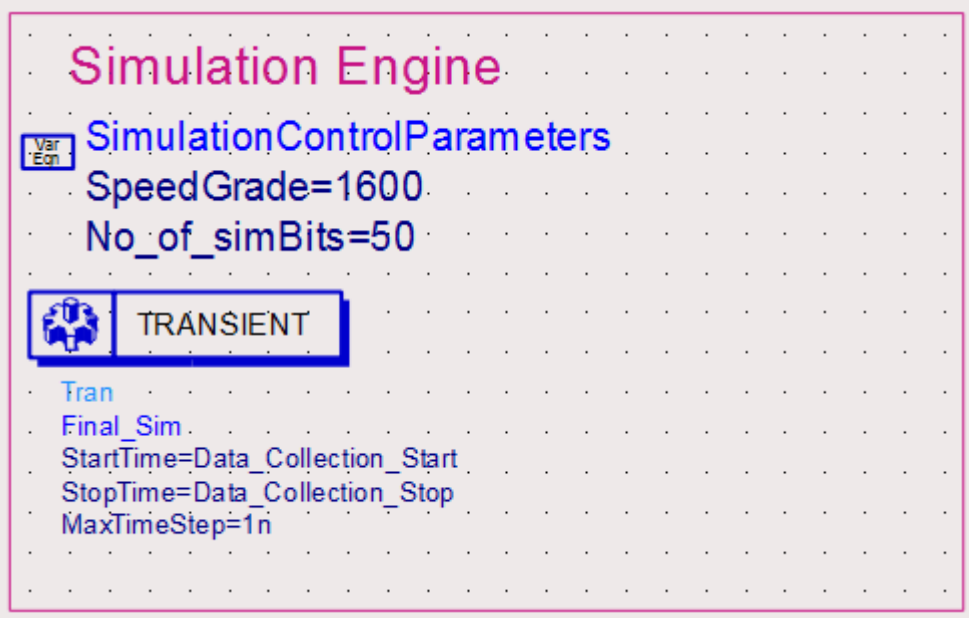
When working with the Infiniium Offline DDR3 compliance application, consider the differences between simulated and measured waveforms:

- Variable time-step vs. fixed time step
- Continuous data-stream vs. burst type signaling
- Simulated DQ + DQS without CLK vs. captured waveforms with DQ, DQS, CLK

Usually simulations work with variable time-steps, while measurements uses a fixed time step. The DDR3 compliance test application on the oscilloscope assumes a fixed time-step of 50 ps and is used for the simulated waveforms. In addition, the oscilloscope application works with bursts of data and have the ability to separate out write, read, and tri-state modes using a Preamble structure at the beginning of the data burst. To generate the required waveforms for offline DDR3 test compliance application, add preamble structure and clock timing to set up the ADS simulation of DQ and DQS.

About ADS Simulation Setup

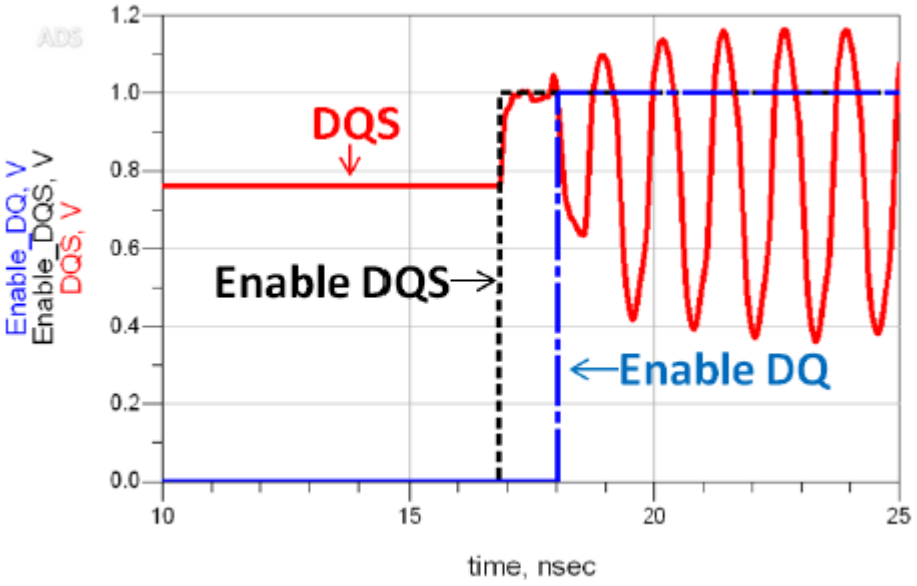
ADS transient simulator is used in all simulation setups. The transient simulation controller rely on the basic settings done by the variables in the `SimulationControlParameters`, as shown in the following figure, where speed grade and the number of simulated bits are set.



Simulation can be done in two configurations, either with arbitrary alignment between Clock and DQS, or with precise alignment between them. If the simulation does not take care of DQS to CLK alignment and the timing is arbitrary, then one should not enable the compliance tests tDQSS, tDSS and tDSH that are used to verify this alignment. The second option is to align CLK to DQS timing manually in simulation through optimization. In operational systems this is done by the Write Leveling training during Power up. Regardless of the CLK to DQS alignment, the DDR3 Compliance Test Application still needs a CLK waveform in order to work properly.

For the fixed time-step, the ADS example does have an interpolation algorithm to generate waveforms with a fixed 50 ps step size. This means it is not required to adjust the simulation controller to a 50 ps time step. In case a different test bench setup is used one may need to make sure that there is a fixed time-step of 50 ps for the saved waveforms.

Simulating burst mode with preamble will result in better coverage for compliance verification. To generate the burst mode with Preamble the ADS simulation example uses the enable input of the IBIS model. This is driven by a pulse source that generates the delayed enable signals for DQS and DQ. In the example workspace the simulation works with a double burst (i.e., 16 bits per Burst) and an 8 bit bubble between two bursts. If this is changed the enable signals need to be adjusted accordingly. In order to have accurate control over the enable signal timing one should leave the maximum time-step at UI/10 or smaller.



Installing DDR3 Compliance Test Bench

Installing DDR3 Compliance Test Bench

This section provides information about the prerequisites and steps for installing DDR3 Compliance Test Bench (CTB).

DDR3 Prerequisites

- The ADS 2015.01 DDR3 Compliance Test Bench is installed with ADS 2015.01.
- Licenses for ADS Core and the ADS Transient Convolution Element or a bundle (such as the W2210BP/BT) that contains these two are required.
- W2350EP/ET DDR3 Compliance Test Bench license is required.
- Additionally, the following oscilloscope software must be downloaded and licensed separately. However, no actual oscilloscope is required: the software runs on an ordinary Windows PC in offline/remote mode:
    - N8900A-001 Infiniium Offline, Transportable License
    - N8900A-002 DSA Package (EZJIT Plus and SDA), Transportable License
    - U7231B-1TP DDR3 and LPDDR3 Compliance Software, Transportable License

Install Instructions

To install DDR3:

1. Launch ADS 2015.01 and open a Schematic view.

   NOTE The DDR3 Advanced Compliance Test Bench is available under the DesignGuide menu.

2. Download the Infiniium Offline Oscilloscope Analysis Software from the Keysight website and follow the on-screen installation instructions.

   NOTE Close all the applications on your PC before installing the software.

3. Restart your PC to complete the installation.
4. Download the DDR3 Compliance Test Application Software from the Keysight website and follow the on-screen installation instructions.
   http://www.keysight.com/main/software.jspx?cc=IN&lc=eng&ckey=2157066&nid=-34333.1094284&id=2157066
5. Restart your PC after completing the installation.

Introduction to DDR3 SDRAM

Introduction to DDR3 SDRAM

DDR3 Compliance Test Bench content was originally generated by Hermann Ruckerbauer of EKH (www.eyeknowhow.de). Keysight Technologies owns all the rights to this content.

About DDR3 SDRAM

DDR3 stands for double data rate 3rd generation. SDRAM stands for synchronous dynamic random access memory. As of October 2014, DDR3 memory still dominates the memory usage in computing, networking, storage and mobile applications. It's a successor to a previous generation DDR2 memory, and a predecessor to the next generation DDR4 memory. DDR3 interface specification (JEDC Standard JESD 79-3) doubles the data rate of DDR2. DDR4 interface specification (JDEC Standard JESD 79-4) doubles the data rate of DDR3. Following table compares different generations of DDR memory technologies. (http://www.virtium.com/resources/quick-reference-topics/ddr123/)

DRAM Technology Comparison	**SDR DRAM**	**DDR SDRAM**	**DDR2 SDRAM**	**DDR3 SDRAM**	**DDR4 SDRAM**
Data rate (Mb/s per pin), Chip speed bin *(*Note 3)*	PC66 PC100 PC133	DDR-200 DDR-266 DDR-333 DDR-400	DDR2-400 DDR2-533 DDR2-667 DDR2-800	DDR3-800 DDR3-1066 DDR3-1333 DDR3-1600	DDR4-2133 DDR4-2400 DDR4-2666 DDR4-3200
Clock (Mhz)	66 100 133	100 133 166 200	200 266 333 400	400 533 666 800	1067 1200 1600
Module ranks (# of chip select lines)	1, 2	1, 2, 4	1, 2, 4	1, 2, 4	1, 2, 4
Module data bus width (I/O organization)	x64, (x72 with ECC)	x16, x32, x64, (x72 with ECC)	x16, x32, x64, (x72 with ECC)	x16, x32, x64, (x72 with ECC)	x16, x32, x64 (x72 with ECC)
			VLP	VLP	VLP
	RDIMM	RDIMM	RDIMM	RDIMM	RDIMM

DRAM Technology Comparison	SDR DRAM	DDR SDRAM	DDR2 SDRAM	DDR3 SDRAM	DDR4 SDRAM
JEDEC Modules and JEDEC Form Factors	UDIMM	UDIMM	UDIMM	UDIMM	UDIMM
	SODIMM	SODIMM	SODIMM	SODIMM	SODIMM
			SO-CDIMM	SO-UDIMM	SODIMM ECC
			SO-RDIMM	SO-RDIMM	Reg SODIMM ECC
		microDIMM	microDIMM	microDIMM	
			mini-DIMM	mini-DIMM	mini-DIMM
			FB-DIMM	LR-DIMM	LR-DIMM
		16b-SODIMM	16b-SODIMM		
			32b-SODIMM	32b-SODIMM	
		32b-DIMM			
Module Densities	up to 256MB	128MB to 2GB	256MB to 4GB	1GB to 32GB	4GB to 64GB
Chip Densities	32Mb to 256Mb	128Mb to 1Gb	256Mb to 2Gb	512Mb to 8Gb	4Gb to 8Gb
Chip Density @ Lowest Cost per Bit	128Mb	256Mb	512Mb	1Gb	4Gb
Chip data bus width (I/O organization)	x4, x8, x16	x4, x8, x16	x4, x8, x16	x4, x8, x16	X4, x8, x16
Voltage (VDD = VDDQ/[V])	3.3V	2.5V	1.8V	1.5V	1.2V
		32% reduction	39% reduction	20% reduction	25% reduction

DRAM Technology Comparison	SDR DRAM	DDR SDRAM	DDR2 SDRAM	DDR3 SDRAM	DDR4 SDRAM
% Power Reduction from previous generation (VDD only) *(*Note 2)*					
Interface	LVTTL	SSTL_2	SSTL_18	SSTL_15	POD_12
DRAM Banks (inside the chip)	2/4	4	4 (8 for 1Gb)	8	8
Prefetch (bits)	1	2	4	8	8
Burst length *(*Note 5)*	1, 2, 4, 8 (page)	2, 4, 8	4, 8	8 (4 burst chop)	8
Bidirectional strobe	None	Single Ended (SE)	SE, Differential optional	Differential only	Differential only
DQ driver strength /calibration	Wide envelope	Narrow envelope	18 Ω ,OCD calibration	34 Ω , ZQ-pin self-calibration	40 Ω, 48 Ω
Termination		only on MoBo	MoBo/ODT values = 50, 75, 150, or “off””	DIMM/Dynamic ODT	Dynamic ODT
Data mask	Yes	Yes	Yes	Yes	Yes
DRAM Package (monolithic)	TSOP-54	TSOP-66, BGA	FBGA only	FBGA only	FBGA only

Notes:

- DDR2 and DDR3 UDIMMs and RDIMMs have a 240-pin, 1.0mm pitch memory sockets.
- DDR3 may be as much as 30% reduction over DDR2 at the same speed, when considering lower IDD currents and other DDR3 architectural changes. DDR3-1600 is at the same power level, as DDR2-800.
- DDR3 has higher CAS Latency than DDR2: DDR3-800 (5-5-5), DDR3-1066 (7-7-7), DDR3-1333 (8-8-8), DDR3-1333 (9-9-9).

- The memory sockets (slots) per channel are memory controller and motherboard dependent. RDIMMs may have more slots than UDIMM. Faster and higher density DIMMs may require less slots per channel.
- “DDR3 Burst Length: 8 (Interleave without any limit, sequential with starting address “000” only), 4 with tCCD = 4 which does not allow seamless read or write [either On the fly using A12 or MRS]“.
- DDR3 Programmable CAS Write Latency (CWL) = 5 (DDR3-800), 6 (DDR3-1066), 7 (DDR3-1333), 8 (DDR3-1600).

LPDDR3 Simulation Setup

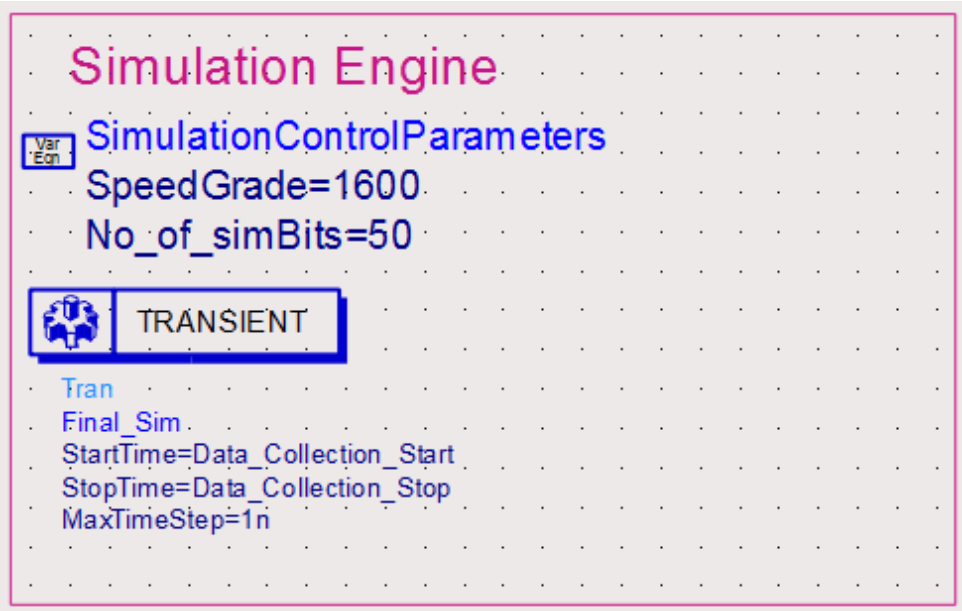
LPDDR3 Simulation Setup

There are four simulation setups for LPDDR3, which are prefixed with `WaveformBridge`:

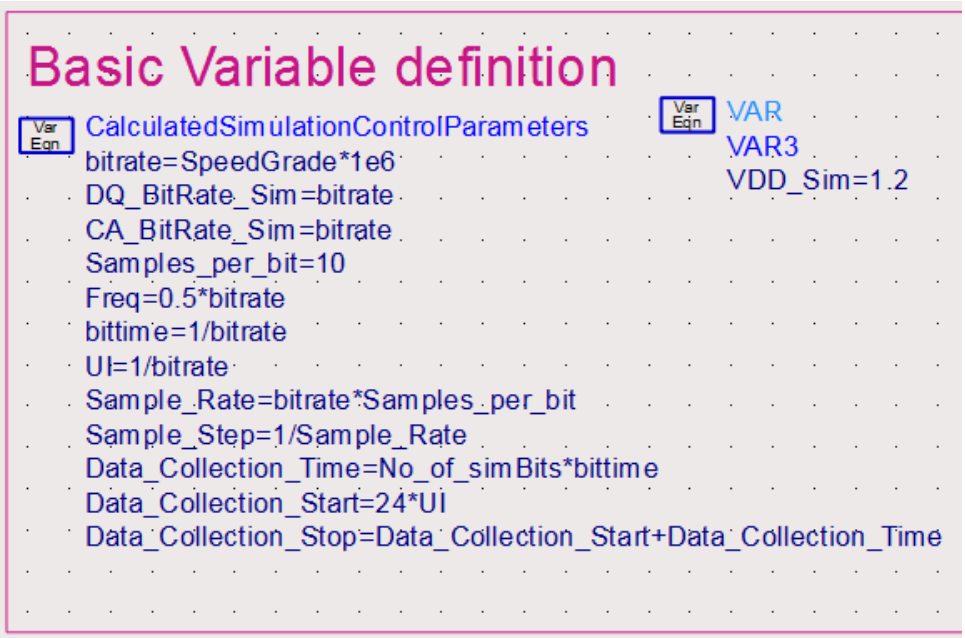
- LPDDR3_DQ_Write: Denotes data bus simulation in ***WRITE*** mode, that is, data transfer from memory controller to DRAM
- LPDDR3_DQ_Read: Denotes data Bus simulation in ***READ*** mode, that is, data transfer from DRAM to memory controller
- LPDDR3_CA: Denotes command and address bus simulation
- LPDDR3_Complete: Denotes a complete set of signals in one simulation, by combining ***CA***, ***DQ_Write*** and ***DQ_Read*** simulation setups into one schematic.

These simulation setups have three common blocks.

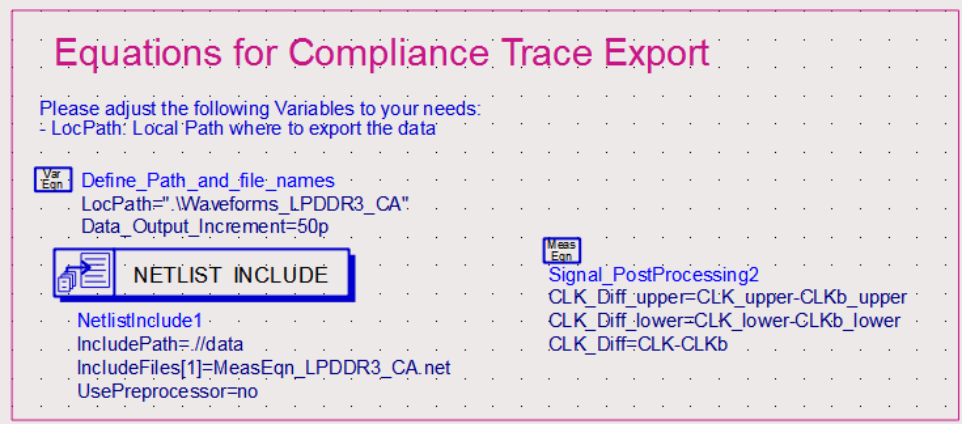
- Simulation Engine



- Basic Variable definition



- Equations for Compliance Trace Export

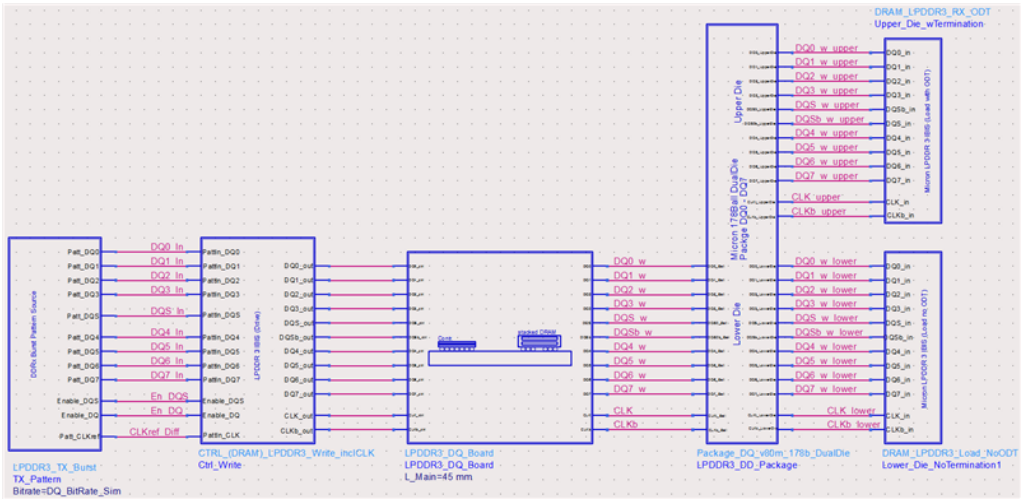


A simulation runs in the following sequence:

- Variables in VarEqn blocks are calculated as pre-processing equations
- Transient simulation runs from StartTime to StopTime
- Variables in MeasEqn blocks, AEL expression equations in Netlist Include block, are calculated as post-processing equations

LPDDR3_DQ_Write schematic

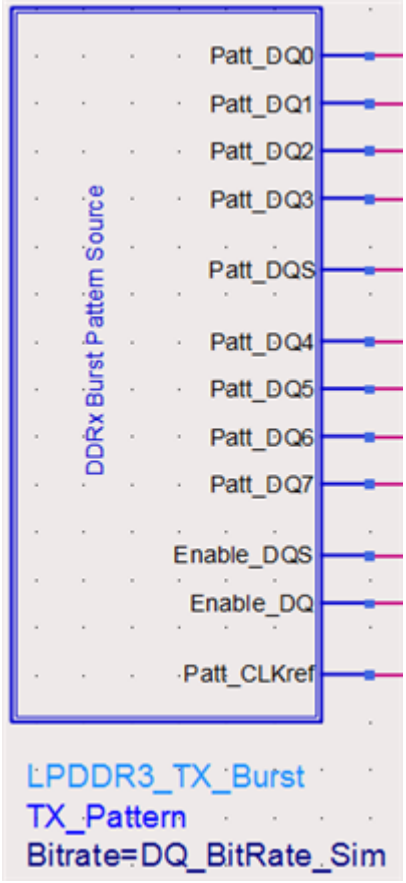
Following is an example of a top-level schematic for LPDDR3 DQ Write. The blocks from left to right are DQ Burst Pattern Generator, Controller DQ Driver Pins, PCB Interconnects, Dual-Die DRAM Package, DRAM DQ Receiver Pins.



While setting your own simulation, procure IBIS simulation models from your CPU and DRAM vendors and build the PCB interconnect model for the PCB stack-up used in your system.

Block details used in this schematic are explained in the subsequent sections.

Pattern Generator

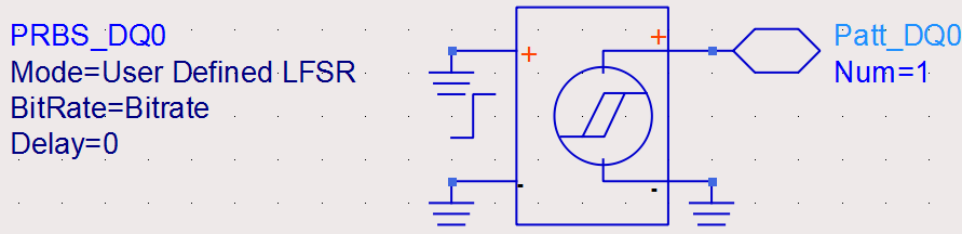


The DQ burst pattern generator produces the following patterns:

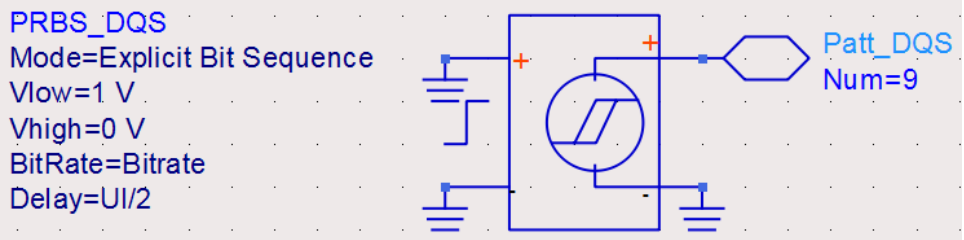
- 8 pseudo-random data (DQ) patterns labeled as Patt_DQ0 to Patt_DQ7,
- 1 repetitive strobe (DQS) pattern labeled as Patt_DQS,
- 1 repetitive clock pattern labeled as Patt_CLKref,
- 2 pulse patterns to enable DQ and DQS bursts

Inside the pattern generator schematic, there are primitive models from ADS Analog Lib:

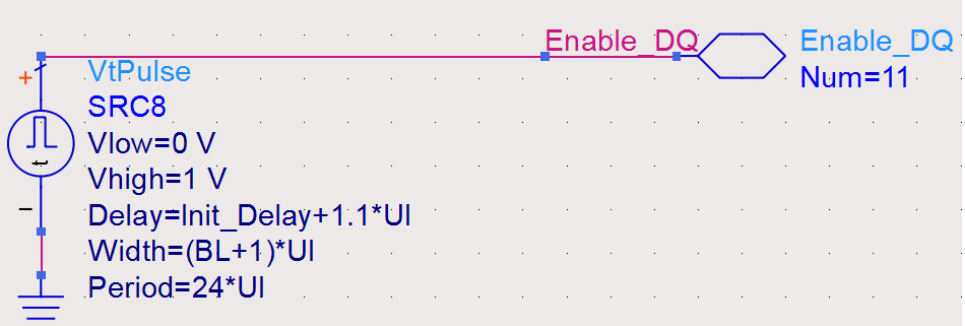
- Pseudo random bit sequence(PRBS) sources with user-defined linear feedback shift register (LFSR) are used to generate the DQ pattern, with each DQ bit having a different set of LFSR taps.



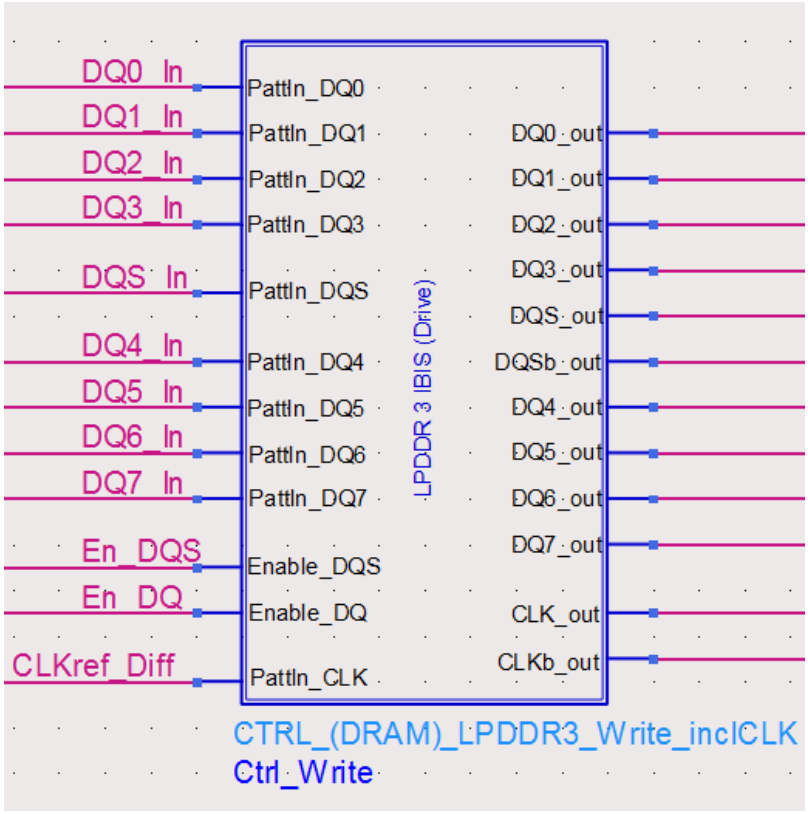
- PRBS source with explicit bit sequence is used to generate the repetitive DQS pattern with preamble



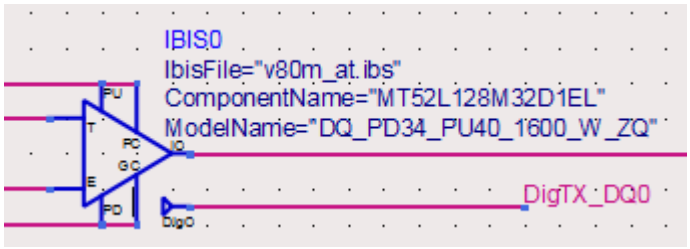
- Pulse source is used to generate a pulse pattern to enable DQ and DQS bursts



Controller DQ Driver



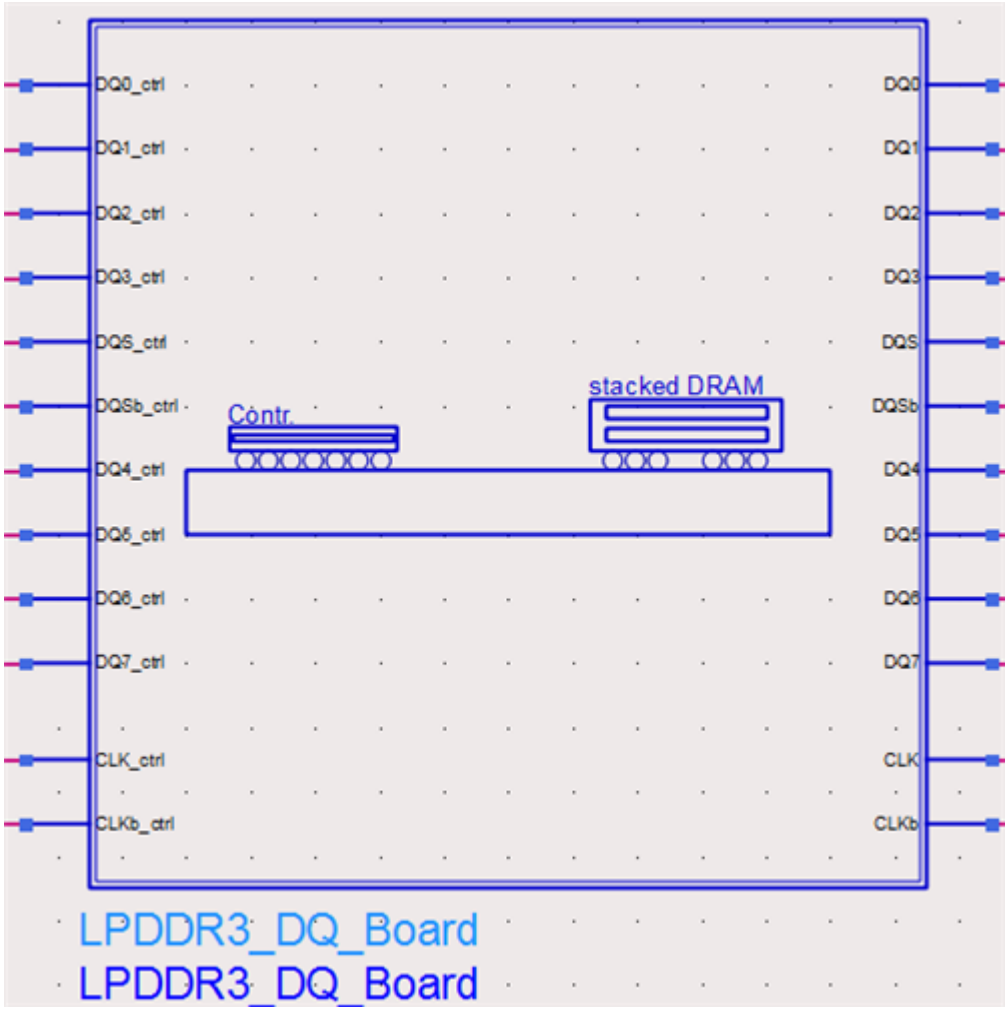
The controller DQ driver model takes pulse pattern as an input, and generates a waveform as an output that mimics the driver circuit output at controller I/O pad. Inside the drive schematic, there is an IBIS I/O model for each pin, as shown in the following figure:



NOTE

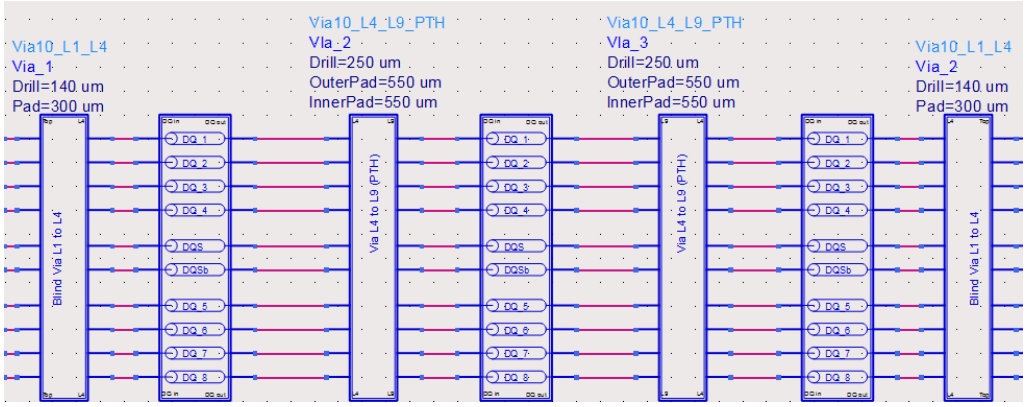
The IBIS model used in this example is for illustration purpose only. You must use the IBIS model or SPICE model provided by your CPU or memory controller vendor to get good simulation accuracy for your system.

PCB Interconnects

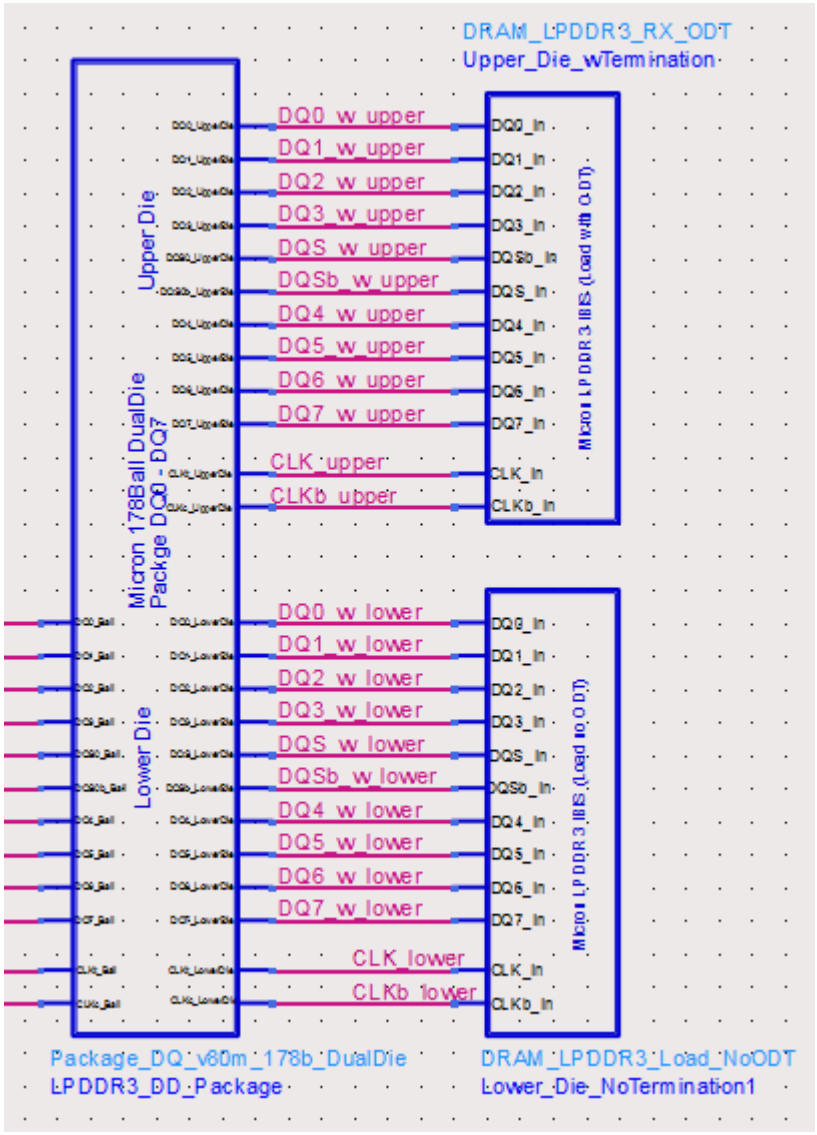


This model is a combination of passive interconnects, such as CPU package, CPU socket, PCB vias and traces, DIMM PCB, and DRAM package. Physical effects such as, signal reflection, attenuation, and crosstalk, are captured in this model in the form of S-parameters or coupled transmission lines. Inside the model is a cascade of sub-circuits built out of ADS Multilayer Interconnect Library.

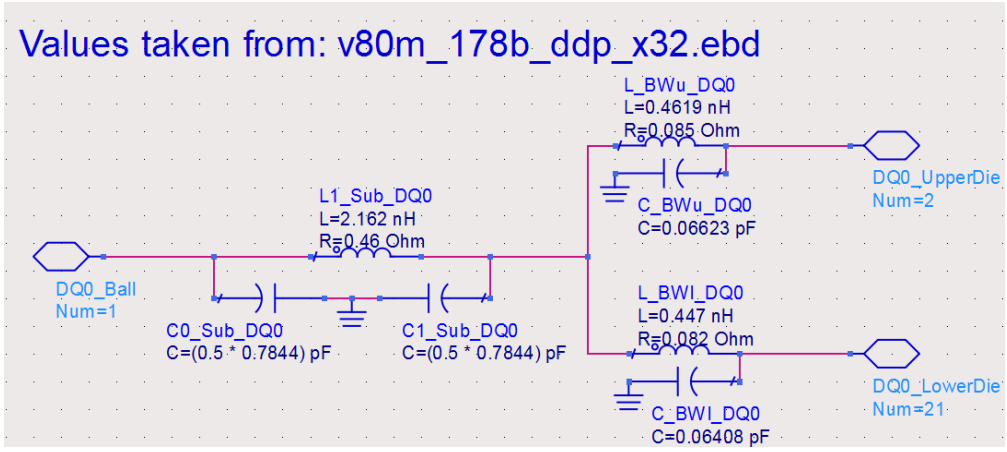
NOTE This model is for illustration purpose only. You must use the actual PCB stack-up and trace dimensions of your system to get accurate simulation results.



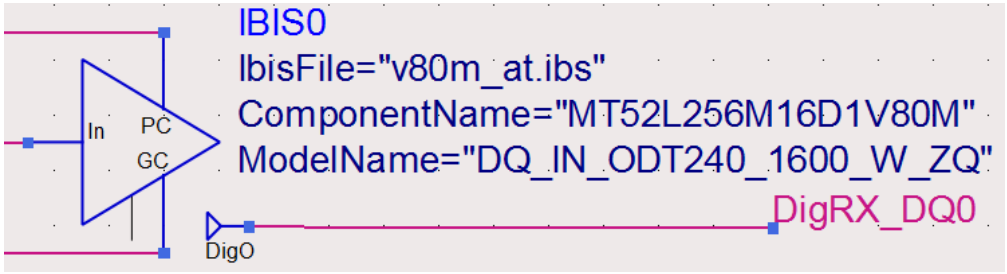
Dual-Die Package and DRAM Receiver



Dual-Die Package is a physical structure housing 2 DRAM dies. Each input pin of the package is connected to 2 output pins for the 2 DRAM dies. The electrical model for each pin connection is a lumped R/L/C circuit, and the cross-talk between pins is not modeled. In this example, the R/L/C values were imported into ADS from an electrical board description (.ebd) file, provided by the package vendor.



Inside DRAM receiver, is a sub-circuit of IBIS models referencing an IBIS file from Micron:

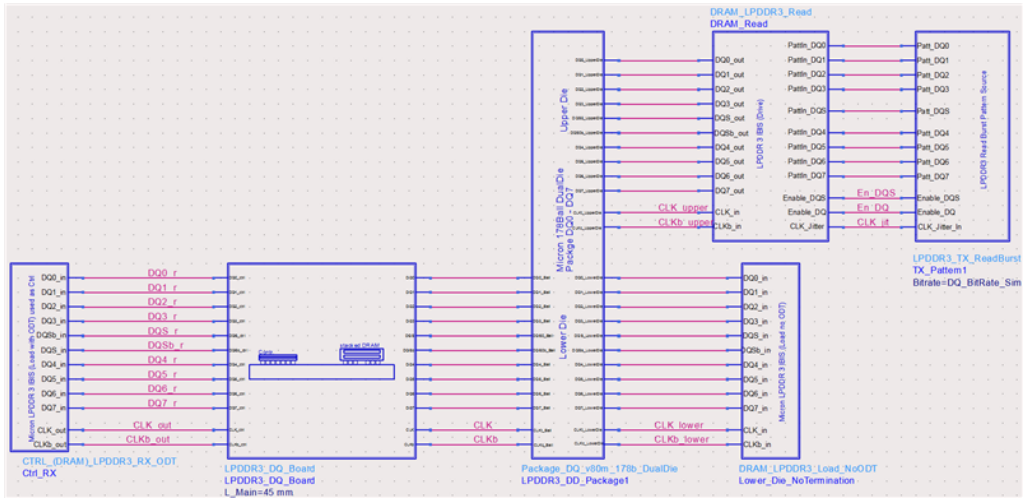


NOTE The IBIS model used in DRAM receiver is for illustration purpose only. You must procure an IBIS file from your DRAM vendor to get accurate simulation results of your system.

LPDDR3_DQ_Read schematic

A top-level schematic for LPDDR3 DQ Read is shown in the following figure. The blocks from Upper-Right to Left are – Pattern Generator > DRAM DQ Driver Pins > Dual-Die DRAM Package > PCB Interconnects > Controller DQ Receiver Pins.

LPDDR3 DQ Read schematic is similar to that of LPDDR3 DQ Write. In DQ Read operation, the pattern generator and DQ drivers are on the DRAM side, and DQ receivers are on the CPU memory controller side.



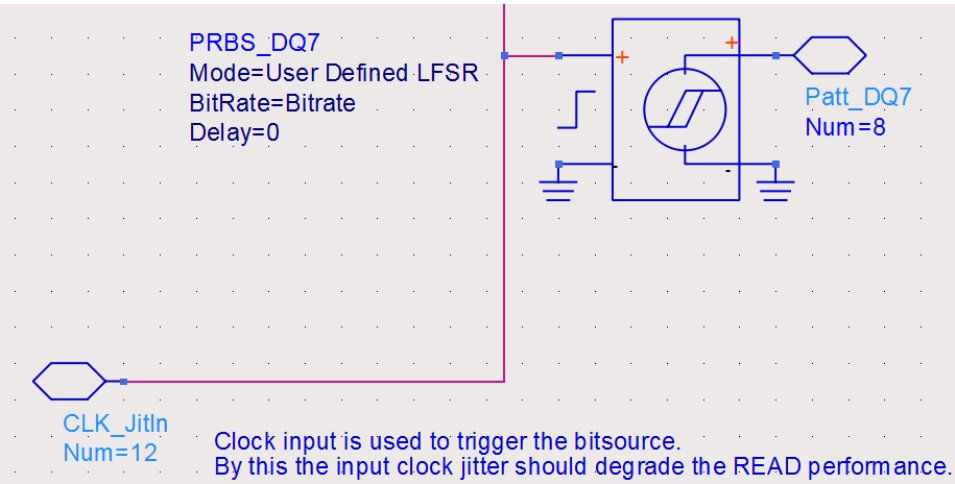
Two features in LPDDR3 DQ Read simulation are as follows:

1. The saved signals are probed on the CPU memory controller side, as highlighted in the figure above. Although, JDEC specifies only the electrical probing and measurements at DRAM package balls, it is prudent to probe signals on the controller package balls as well.

   NOTE In WRITE cycles, DQ and DQS have a phase difference of 90 degrees, i.e., DQ center is aligned with DQS edge. In READ cycles, DQ and DQS are in same phase, i.e., DQ edge is aligned with DQS edge. In order to use DDR3 compliance test application, center-align DQ and DQS signals, to execute receiver input tests for controller receiver pins in READ cycles. This is accomplished by the following MeasEqn post-processing equations in the schematic.

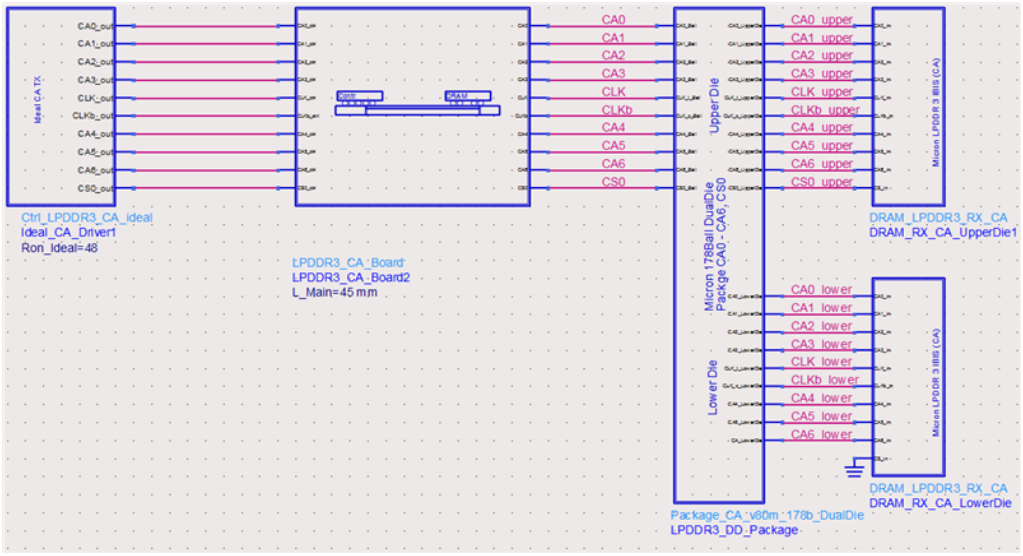
```
Meas
Eqn
Signal_PostProcessing_DQS
Delay_DQS_r=0.5 * UI
time_Axis=indep(DQS_r)
DQS_r_Diff=DQS_r-DQSb_r
DQSb_r_Delayed=vs(DQSb_r, Delay_DQS_r+time_Axis)
DQS_r_Delayed=vs(DQS_r, Delay_DQS_r+time_Axis)
DQS_r_Diff_Delayed=vs(DQS_r-DQSb_r, Delay_DQS_r+time_Axis)
```

2. The actual data transfer from DRAM driver to Controller receiver is triggered by a clock signal transmitted from the controller side, as shown in the following figure for the pattern generator in READ cycle. The clock jitter is included in the READ cycle simulation.

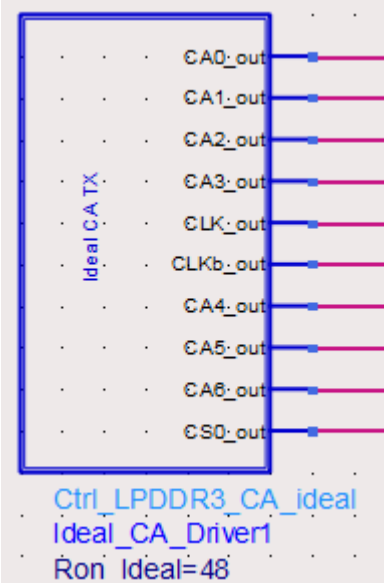


LPDDR3_CA schematic

A top-level schematic for LPDDR3 CA (command and address) simulation is shown in the following figure. The blocks from Left to Right are: CA Pattern Generator, PCB Interconnects, Dual-Die DRAM Package, Controller CA Receiver Pins.



CA Pattern Generator

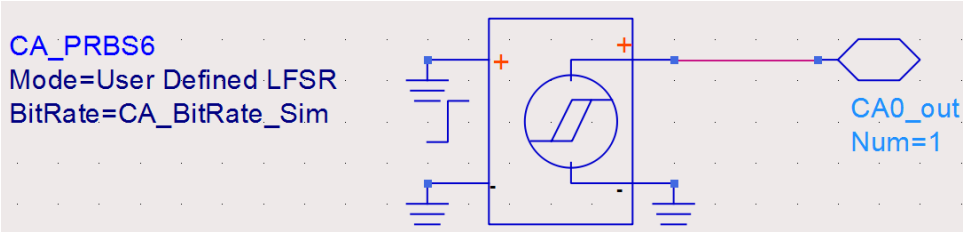


The CA pattern generator produces the following patterns:

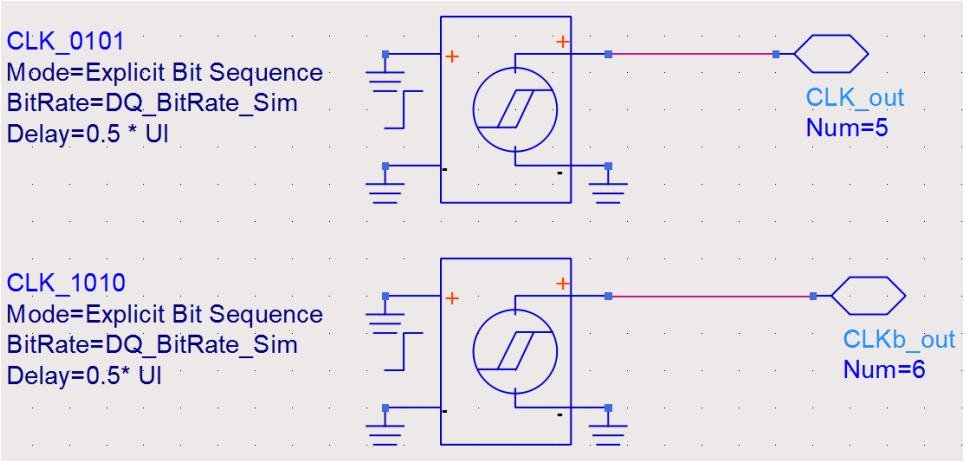
- 7 pseudo-random pulse patterns labeled as CA0_out to CA6_out,
- 1 repetitive clock pattern labeled as CLK_out (+) and CLKb_out(-),
- 1 pseudo-random pattern for Chip Select signal labeled as CS0_out

Inside the CA pattern generator schematic, there are primitive models from ADS Analog Lib:

- Pseudo random bit sequence(PRBS) sources with user-defined linear feedback shift register (LFSR) are used to generate the CA pattern, with each CA bit having different LFSR taps



- PRBS source with explicit bit sequence is used to generate the repetitive CLK pattern



- Unlike DQ/DQS patterns that are either WRITE burst or READ burst, both CA pattern and CLK pattern are continuous, and transmitted in uni-direction from controller driver pin to DRAM receiver pin

PCB Interconnects

It is similar to those described in LPDDR3 Write section.

Dual-Die Package and DRAM Receiver

It is similar to those described in LPDDR3 Write section.

LPDDR3_Complete schematic

A top-level schematic for LPDDR3_Complete simulation is shown in the following figure. It is a combination of LPDDR3_CA, LPDDR3_Write, and LPDDR3_Read schematics. The intent of this simulation is to generate DQ Write, DQ Read, Clock, CA and Control signals in one simulation run. This approach is useful in final design verification.

NOTE

The blocks used in LPDDR3_Complete schematic are identical to those described in previous sections, therefore, are not repeated in this section.

ddr3_comp_sch.svg

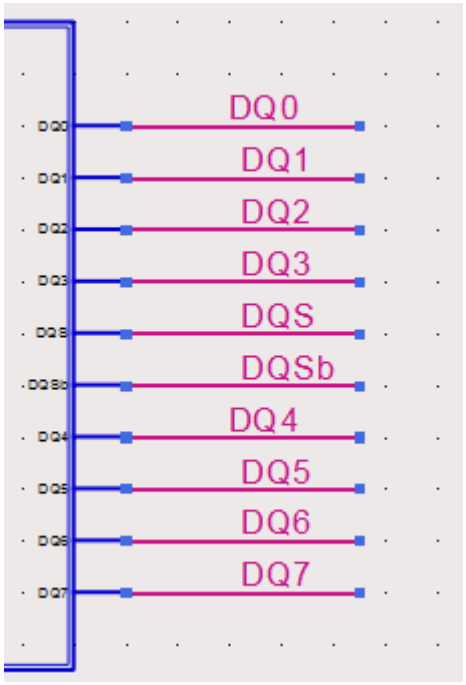
Perform LPDDR3 and DDR3 Compliance Tests using ADS simulated waveforms

Perform LPDDR3 and DDR3 Compliance Tests using ADS simulated waveforms

Each simulation setup contains a Netlist Include block as shown in the following figure:

The IncludeFiles[1] refers an ADS netlist , i.e., MeasEqn_DQ_inc.net, which writes a set of waveforms to *.hdf5* files. For example, the following AEL Expression (aele) statement in *MeasEqn_DQ.net* file, writes a schematic node name *DQ0* (1st argument) into a file named *DQ0.h5* (2nd argument). This is located in a folder specified by LocPath (3rd argument):

```
aele Write_DQ0_HDF5=write_infiniium_ddr3_h5(DQ0, "DQ0", LocPath, "", 1,
Data_Collection_Start[0], Data_Collection_Stop[0],
Data_Output_Increment[0], 50e9)
```

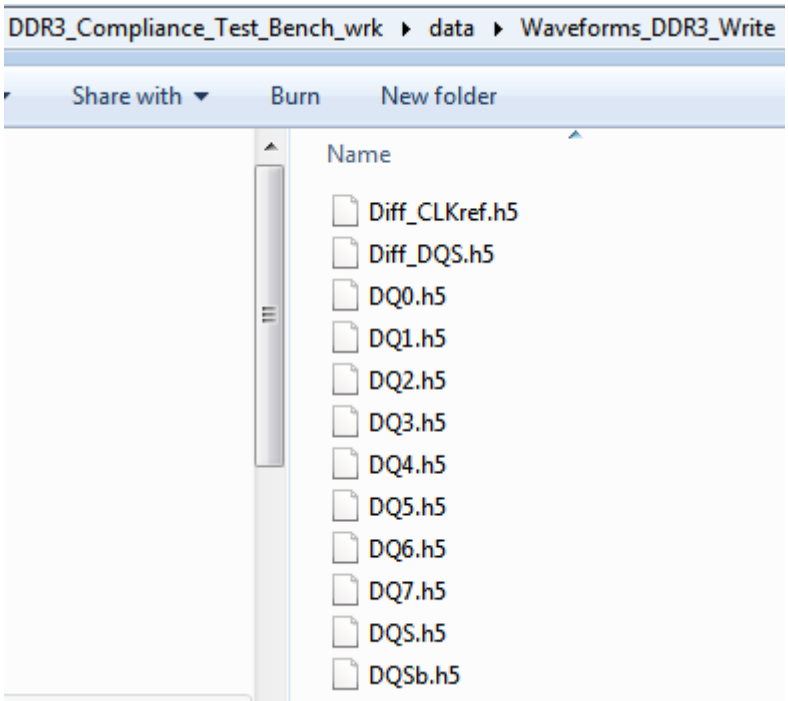


Please note that DQ0~DQ7 are node names defined in the schematic window, as shown above. If your schematic uses node names other than DQ0~DQ7, you need to edit the netlist file (e.g. *MeasEqn_DQ.net*) to reference the actual node names in your schematic.

For example:

```
aele Write_YourName_HDF5=write_infiniium_ddr3_h5(YourNodeName,
"YourHDF5FileName", LocPath, "", 1, Data_Collection_Start[0],
Data_Collection_Stop[0], Data_Output_Increment[0], 50e9)
```

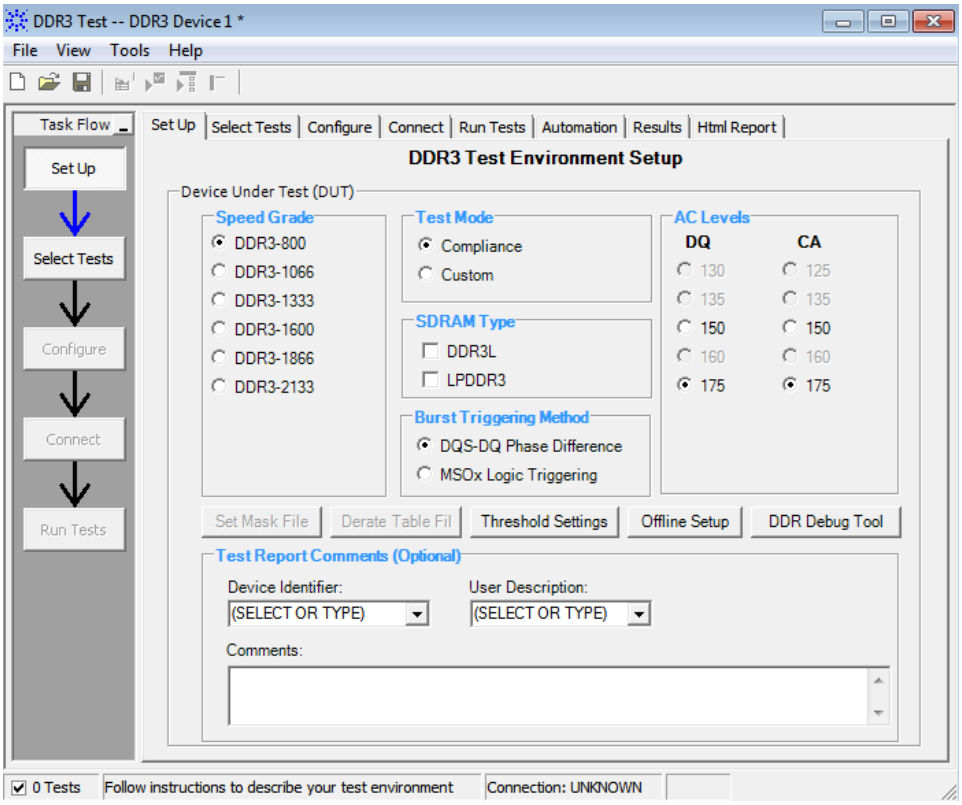
At the end of a simulation, a set of *.h5* files should be written into the folder specified in your schematic. For example:



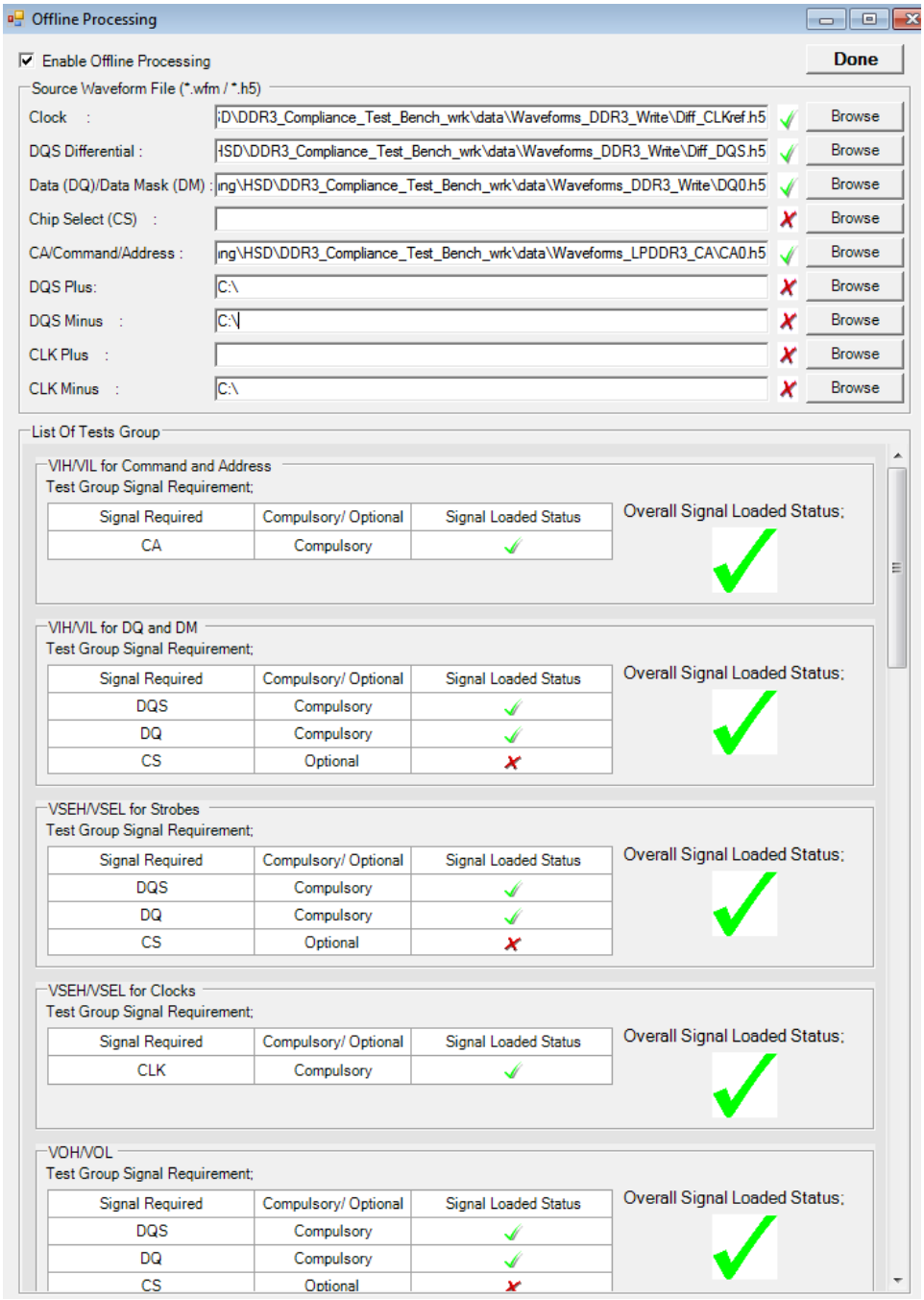
These ADS simulated waveforms are used to run DDR3 compliance tests, as shown in the following figure:

1. Start **Infiniium Offline** and select **Analyze > Automated Test Apps > U7231B/U7231C DDR3 Test App**.

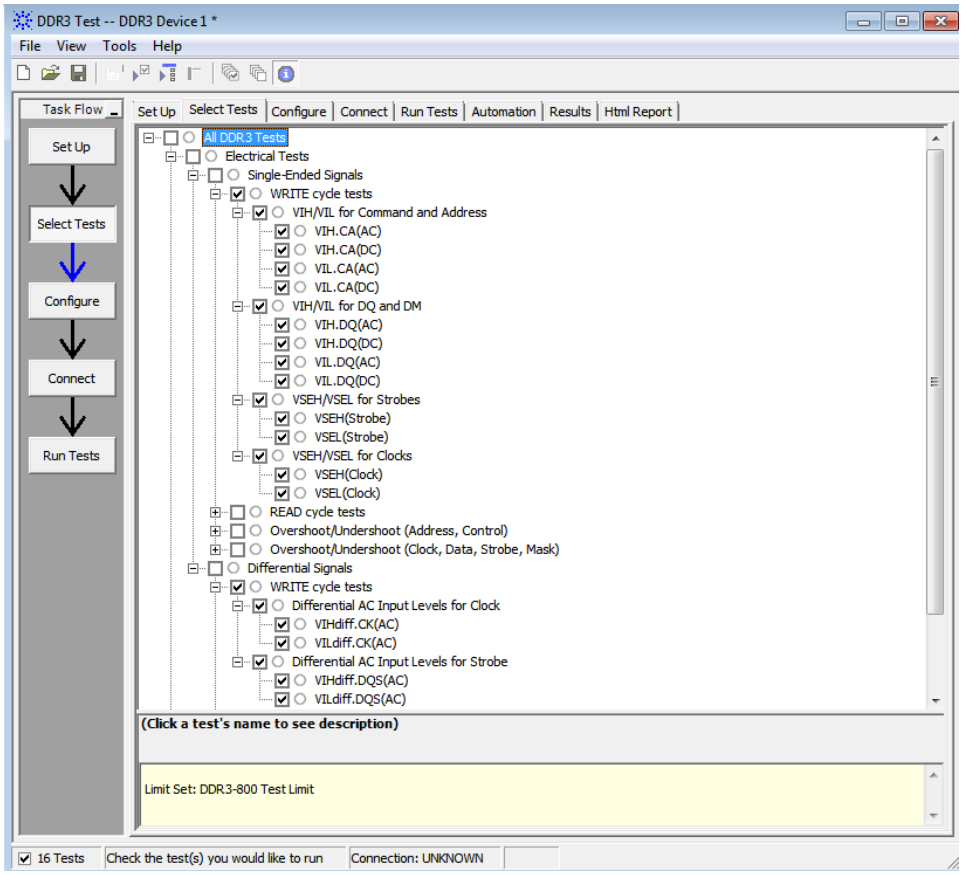
The DDR3 Compliance Test Application is launched.



2. Select **Speed Grade** as DDR3-1600 under the **Device Under Test (DUT)**.
3. Click **Offline Setup**.
4. Select **Enable Offline Processing**, and then click **Browse** to locate the ADS simulated waveforms. This displays the available tests with a green check-mark.

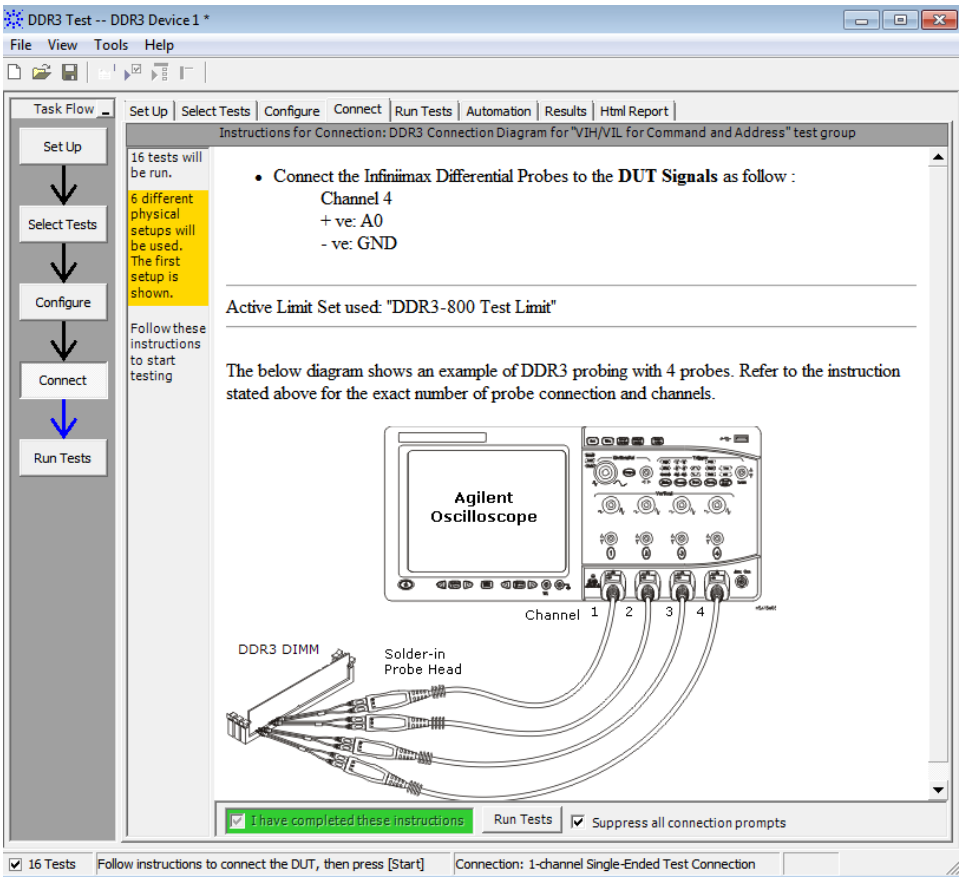


5. Click the **Select Tests** tab and select the test(s) you want to execute:

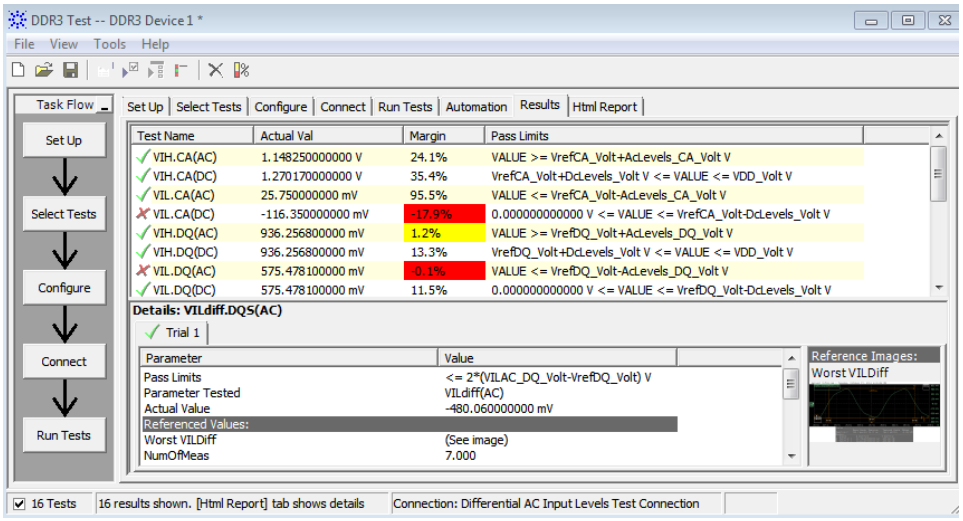


6. Click the **Configure** tab, to view the measurement threshold settings. These threshold values are automatically loaded based on the JDEC specifications.

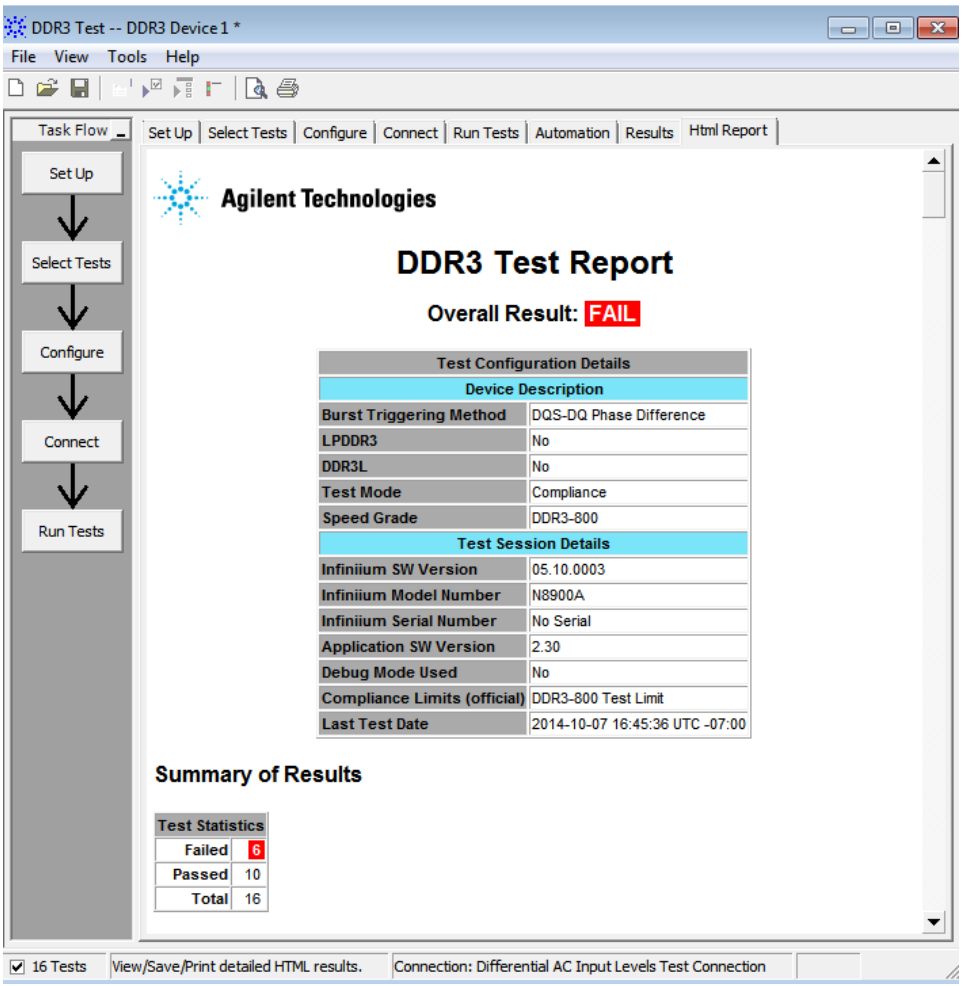
7. Click the **Connect** tab to view how a scope is connected to the DUT. In offline mode, no actual connection is needed. Select both the check boxes docked at the bottom of the screen, and click **Run Tests**.



8. Click the **Results** tab to view the test results listing the actual measurement values and compliance pass limits.



9. Click the **HTML Report** tab to see a summary of tests and an overall status of Pass or Fail.



It is expected to see test failures in the initial iteration of a design. You will receive a test failure alert if the signal being tested is outside the JEDEC pass limits. You will need to adjust the design parameters to fix the failure if such a failure is vital for the overall system operation.