cādence™

Cadence Design Systems, Inc.

IBIS-AMI Toolkit

Revision: 3.1

Cadence Design Systems, Inc. Toolkit License Agreement

Copyright 2010 Cadence Design Systems, Inc. All rights reserved worldwide.

The information and computer programs ("Licensed Material") contained herein are protected by copyright law and international treaties.

Cadence grants Recipient of the Licensed Material a nonexclusive right to use, copy, modify, and distribute the Licensed Material.   Should Recipient distribute any portion of the Licensed Material, with or without modifications, Recipient must retain the above copyright notice and all terms and conditions associated with this Licensed Material, including the disclaimer below. In no event shall Recipient use the Licensed Material for benchmarking purposes against Cadence's products.

The Licensed Material is provided to Recipient to use at Recipient's own risk. The Licensed Material may not be compatible with current or future versions of Cadence products. THE LICENSED MATERIAL IS PROVIDED "AS IS" AND WITH NO WARRANTIES, INCLUDING WITHOUT LIMITATION ANY EXPRESS WARRANTIES OR IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR USE.

IN NO EVENT SHALL CADENCE BE LIABLE TO RECIPIENT OR ANY THIRD PARTY FOR ANY INCIDENTAL, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES, OR ANY OTHER DAMAGES WHATSOEVER (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF BUSINESS PROFITS, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION, OR OTHER PECUNIARY LOSS) ARISING OUT OF THE USE OR INABILITY TO USE LICENSED MATERIAL, WHETHER OR NOT THE POSSIBILITY OR CAUSE OF SUCH DAMAGES WAS KNOWN TO CADENCE.

Table of Contents

1. INTRODUCTION

This toolkit was created to aid the model developer (IP Vendor) to quickly build, compile and test IBIS models utilizing the Algorithmic Modeling Interface (AMI). Users of this toolkit are expected to have gone through the introductory material for algorithmic modeling that is available on IBIS website (IBIS_Home). IBIS-AMI models include compiled, executable, behavioral models which encapsulate signal processing functions. They are used to encode transmitter (TX) and decode receiver (RX) signals enabling them to travel long distance over printed circuit boards and backplanes (channel). A good introduction to algorithmic modeling is contained in the initial presentation material:

AMI_BIRD_Intro_082407_v7.pdf

(presented by IBIS ATM subcommittee to IBIS Open Forum on August 24th 2007)

Additional material on the **Algorithmic Modeling Interface (AMI)** can be found in the Help and Contacts section.

Delivered models include a Shared Object (SO) library and/or Dynamically Linked Library (DLL) that can be executed (called) from within an EDA tool. This toolkit includes an executable "tester" program which allows the model's behavior to be examined outside the confines of any specific EDA tool.

The Cadence Design Systems, Inc. "tester", called **CDNS_AMI_TESTER** is supported for both LINUX and Windows.

LINUX – Red Hat 3.2.3-53 using gcc version 3.2.3 20030502

Windows – XP using MS Visual Studio .NET 2005

2. HELP and CONTACTS

The toolkit is provided "AS IS" free of charge.

(Please read the **Toolkit License Agreement** at the beginning of this document.)

Any issues, difficulties, or suggestions with the Toolkit should be reported via email to

Cadence Design Systems, Inc. using the link:

IBIS-AMI@cadence.com

Discussions about this and other toolkits can be followed at:

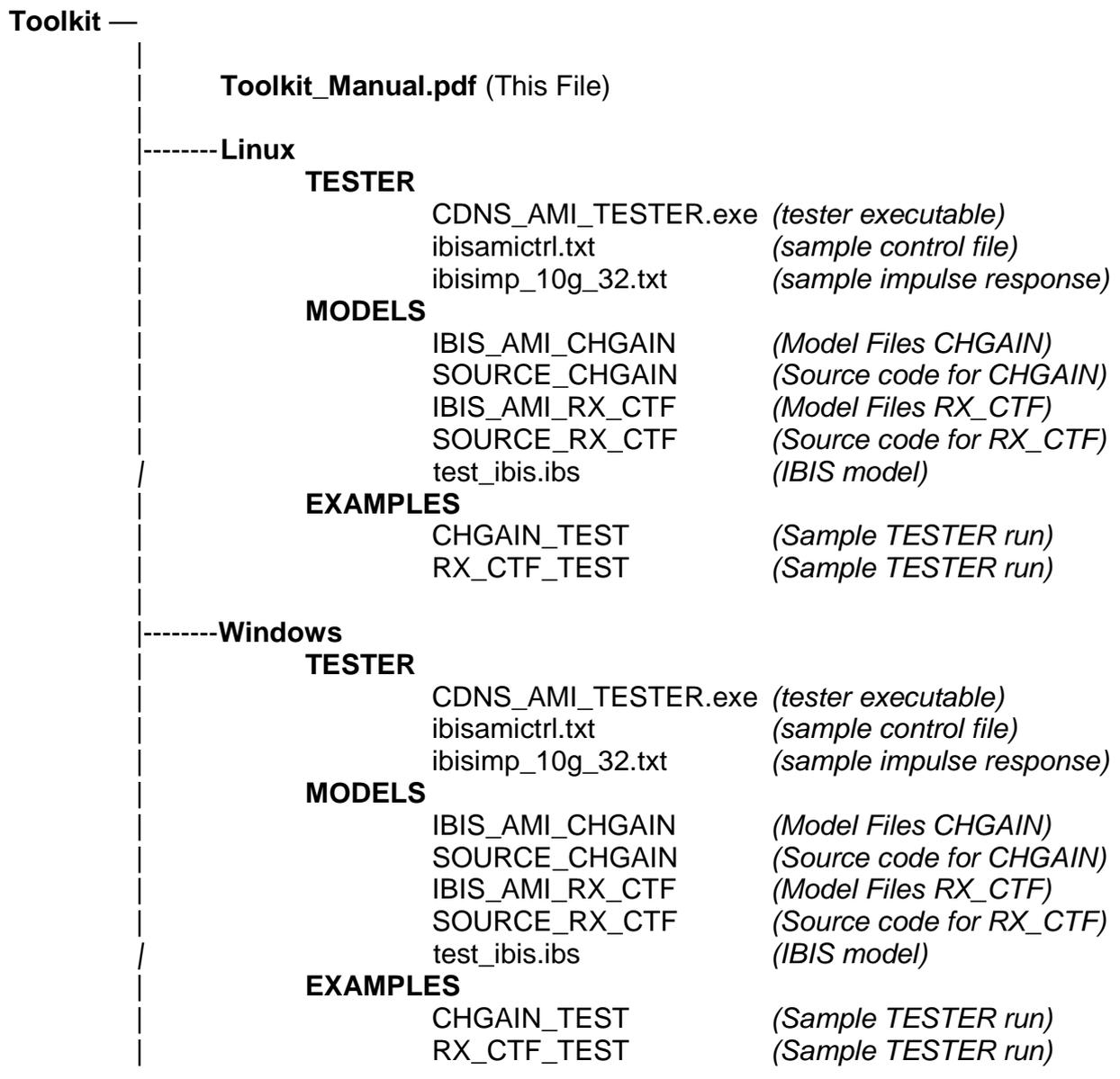www.freelists.org/archive/ibis-ami-toolkit

For more background information on Algorithmic Modeling, visit the archive for the

IBIS ATM Task Group at:

www.vhdl.org/pub/ibis/macromodel_wip/index.html

3. TOOLKIT CONTENTS

Cadence Design Systems, Inc. has created this toolkit and tester program to aid in the development and use of IBIS - Algorithmic Model Interface (AMI) features introduced with the IBIS 5.0 specification. The enclosed kit consists of this reference manual and an executable test program for both the Windows and Linux operating systems as well as sample models and tester output. The Toolkit contains two sub-directories, one for each supported operating system. Within each operating system sub-directory, TESTER executables, sample MODELS, and EXAMPLE runs are included. The Toolkit is organized as follows:

Toolkit —
```
        |
        |       Toolkit_Manual.pdf (This File)
        |
        |--------Linux
        |          TESTER
        |                  CDNS_AMI_TESTER.exe  (tester executable)
        |                  ibisamictrl.txt          (sample control file)
        |                  ibisimp_10g_32.txt       (sample impulse response)
        |          MODELS
        |                  IBIS_AMI_CHGAIN          (Model Files CHGAIN)
        |                  SOURCE_CHGAIN            (Source code for CHGAIN)
        |                  IBIS_AMI_RX_CTF          (Model Files RX_CTF)
        |                  SOURCE_RX_CTF            (Source code for RX_CTF)
        /                  test_ibis.ibs            (IBIS model)
        |          EXAMPLES
        |                  CHGAIN_TEST              (Sample TESTER run)
        |                  RX_CTF_TEST              (Sample TESTER run)
        |
        |--------Windows
        |          TESTER
        |                  CDNS_AMI_TESTER.exe  (tester executable)
        |                  ibisamictrl.txt          (sample control file)
        |                  ibisimp_10g_32.txt       (sample impulse response)
        |          MODELS
        |                  IBIS_AMI_CHGAIN          (Model Files CHGAIN)
        |                  SOURCE_CHGAIN            (Source code for CHGAIN)
        |                  IBIS_AMI_RX_CTF          (Model Files RX_CTF)
        |                  SOURCE_RX_CTF            (Source code for RX_CTF)
        /                  test_ibis.ibs            (IBIS model)
        |          EXAMPLES
        |                  CHGAIN_TEST              (Sample TESTER run)
        |                  RX_CTF_TEST              (Sample TESTER run)
        —
```

TESTER - This folder contains the "tester" executable (.exe) program.
There is also a sample impulse response and a control file in this folder.

MODELS - This folder contains models delivered with the toolkit. Each model consists of three files with the same base name and the following file extensions.
　　xxx.**ami** - Parameter file with all possible option choices
　　xxx.**amix** - Current setting for each parameter
　　xxx.**dll** (so) - Executable model file
Additionally, in this directory each model has its own folder containing the source files for the sample models provided.

EXAMPLE - This folder contains example runs of the supplied IBIS AMI models. It illustrates all inputs, outputs and supporting files.

MODELS INCLUDED IN THIS TOOLKIT

----

1. **RX_CTF (Continuous Time Filter)**

Model Name: IBIS_AMI_RX_CTF.dll

　　Description: The RX model does not have clock recovery and therefore does not write a clk output. It has fixed coefficients. The INIT function in the model does not modify the impulse response. A waveform passed to the GetWave call will undergo equalization. It has been tested for a 5 GBPS data rate.

2. **CHGAIN (Gain Filter)**

Model Name: IBIS_AMI_CHGAIN.dll

　　Description: The CHGAIN filter is a simple model that multiplies the input wave by a gain factor (The default gain factor is 2).

4. FILE DESCRIPTIONS

Input files:

<*Model_Name*>.dll   (Model)

<*Model_Name*>.so   (Alternative model for Linux)

<*Model_Name*>.ami   (Model Parameter File)

<*Model_Name*>.amix (Current Model Parameter Settings) (optional)

ibisamictrl.txt   (Control File - provides user control of data rate, ignore bits, simulation length, output file names etc. If not present, tester uses default values.) (optional)

ibisimp_10g_32.txt (Impulse Response File - must be referenced in control file)

bits_file.txt   (Stimulus Bit stream File - If not defined, the tester will generate a random bit stream.) (optional)

test_ibis.ibs   (Provided for informational purposes only in this toolkit to demonstrate how AMI models are referenced through the IBIS model. Can be used in Licensed Cadence Allegro PCB SI Products that supports AMI models) (For informational purposes only)

NOTE: <*Model_Name*> is unique for each model and must match for all files in the set defining the device (i.e. "IBIS_AMI_RX_CTF" or "IBIS_AMI_CHGAIN").

Output files:

After a successful run of the TESTER, the following files will be generated.

NOTE: Each output produces a pair of files containing identical data. One *<File_Name>*.txt is an ASCII text file containing (time, voltage) ordered pairs defining the output wave. The second *<File_Name>*.sim defines the same waveform but is formatted for viewing with Sigwave (the waveform viewer included with Cadence Design Systems, Inc.'s Allegro tool suite.)

impin.txt   (Input Impulse Response)

impin.sim

impout.txt   (Output Impulse Response from *INIT*)

impout.sim

stimin.txt   (Stimulus Input [bits] to *GetWave*)

stimin.sim

wavein.txt   (Stimulus Input [wave] to *GetWave*)

wavein.sim

waveout.txt (Output [wave] from *GetWave*)

waveout.sim

clks.txt   (Output of the Clock Data Recovery [CDR] in *GetWave*)

clks.sim

Control File (ibisamictrl.txt):

SAMPLE:

```
(ibisamictrl
; simulation control
    (data_rate              5e9)                    ; data rate for simulation
    (stimulus_bits_cnt      1100)                   ; no of bits to be simulated
    (waveform_start_time    120e-9)                 ; output the waveform after 120ns
; impulse response file
    (imp_file               ibisimp_10g_32.txt)
; bit stream file
;   (bits_file              bits_file.txt)          ;OPTIONAL
; model files
;   (dll_name               .../IBIS…dll)           ;OPTIONAL
;   (dll_parameter_file_name .../IBIS…amix)         ;OPTIONAL
; output files
    (stim_file              stimin.txt)
    (wave_in_file           wavein.txt)
    (wave_out_file          waveout.txt)
)
```

Where:

data_rate: The data rate at which the AMI model is to be simulated.
Data Rate is defined in scientific notation (i.e. 6 GBPS = 6e9)
Default: 6.25e9

stimulus_bits_cnt: Number of bits in this simulation run. If the stimulus_bits_cnt is less than 1000, the TESTER runs a 1000 bits simulation.
Default: 5000

waveform_start_time: Initial time that the tester ignores in the waveform.
Default: 120e-9

imp_file: The impulse response file. This file should have 2 columns, the first should be time, the second should be voltage.
Default: ibisimp_10g_32.txt (sample file)

bits_file: Bits file provided by the user. If no bits file is provided, the tester will generate a random bit stream.

dll_name: Name of the Shared Object Library (dll). If using Linux, complete path may be required. Putting dll_name in the ibisamictrl.txt file is optional. The dll_name can also be supplied as an argument to the TESTER.

dll_parameter_file_name:
Name of the Parameter file that contains the values of the parameters for the Shared Object (dll). This is required if the dll_name is used in the ibisamictrl.txt. If the dll_name is supplied as an argument to the TESTER, the parameter file name is not required in the ibisamictrl.txt file.

stim_file: Wave file going in. The waveform is a generated by the TESTER program using either the bits_file provided by the user, or if not, random bits generated by the TESTER.

wave_in_file: Wave file going in the Getwave call of the AMI model.
Default: wavein.txt

wave_out_file: Wave file coming out of the Getwave call of the AMI model.
Default: waveout.txt

5. COMPILING MODELS

Linux

To compile the executable model:

    a) gcc -c *<model_name>*.c

    b) gcc -shared -o *<model_name>*.dll *<model_name>*.o

Windows

To compile the executable model:

    a) Open the project file in the directory Visual_Studio_Project_Files with Visual Studio .NET.

    b) Make sure the source files are present

    c) Build Solution to create dll

    Version 3.0 of the toolkit was built using Visual Studio 2005

6. TESTER OPERATION

COMMAND LINE:

(for Linux)
Model as arguement:

> %CDNS_AMI_TESTER *<AMI_Model.dll>*

> where: *<AMI_Model.dll>* is the name of the AMI model.
> NOTE: "FULL" path may be required for proper execution.

Model defined in Control file:

> %CDNS_AMI_TESTER

(for Windows)

Model as arguement:

> >CDNS_AMI_TESTER.exe *<AMI_Model.dll>*

> where: *<AMI_Model.dll>* is the name of the AMI model.

Model defined in Control file:

> >CDNS_AMI_TESTER.exe

The **ibisamictrl.txt** file must be in the directory from where the test is run.

If there is a ***<model>*.amix** file, the parameters are read from it. Otherwise, default parameters are used.

The dll and the parameter file names can be defined in the **ibisamictrl.txt** file. The control file has precedence over the command line argument.

7. RESULTS

NOTE: Each output produces a pair of files containing identical data. One *<File_Name>*.txt is an ASCII text file containing (time, voltage) ordered pairs defining the output wave. The second *<File_Name>*.sim defines the same waveform but is formatted for viewing with Sigwave (the waveform viewer included with Cadence Design Systems, Inc.'s Allegro tool suite.)

Sample waveforms:

Model: CHGAIN

Wavein & Waveout