Introduction to the IBIS Macro Model Library

Arpad Muranyi
Signal Integrity Engineering

March 10, 2006
IBIS macro modeling concept

IBIS File

[External Circuit] or [External Model] call macro model templates

Macro model templates call building blocks from standard library

Building blocks are written using the analog only features of the *-AMS languages, and can be substituted with native SPICE elements in SPICE tools if necessary

*Other brands and names are the property of their respective owners
Contents of the library

- Three resistors
  - R, VCR, CCR
- Three capacitors
  - C, VCC, CCC
- Four inductors
  - L, VCL, CCL, K
- 22 voltage and 22 current sources, including
  - Delay, Min, Max, Abs, Sum, Mult, Div, PWL
- An ideal T-line
- Eight IBIS buffers
  - Input, Output, IO, 3-state, Opensource, IO_opensource, Opensink, IO_opensink
Philosophy of the test suite

- Assumption:
  - HSPICE* with Verilog-A option installed, and/or
  - SMASH* installed (for the VHDL-A(MS) examples)

- All examples are ready to go
  - No editing required, just simulate and look at the waveforms
  - The examples are simple, just enough to show the concept
  - No attempt was made to show all possibilities
  - The data files contain very short data tables so the waveforms may not all be smooth and rounded
  - Some lines are commented out in the VHDL-A(MS) building blocks to allow them work in the evaluation version of SMASH (Seduction)

- A PERL script has been developed by Paul Fernando (NCSU) to extract and reformat the data from IBIS files so that the library building blocks can read it
Parameter data file format

--- C_comp parameters
---
C_comp
5.00e-12
kC_comp_pc
0.25
kC_comp_pu
0.25
kC_comp_pd
0.25
kC_comp_gc
0.25
--- Vectors of the IV curve tables
---
Ipc_data
0.08
0.00
0.00
0.00
Vpc_data
-5.00
-1.00
5.00
10.00
Ipu_data
0.10
0.00
-0.10
-0.20
...
Test suite architecture

- **Top level SPICE files**
  - Include the simulation control statements
  - Contain simple stimulus source(s), load(s) and a call to the macro model netlist (i.e. template)
  - Equivalent of the SI simulator tool’s IBIS environment with an IBIS file using an [External Model] or [External Circuit] statement

- **“Macro model template”-s**
  - Contain Verilog-A(MS) or VHDL-A(MS) netlists to instantiate the building blocks from the library
  - Show how to pass parameter values into the instances
  - Equivalent of a “complicated buffer” macro model

- **Model library file**
  - Contains the various building blocks of “primitives”
File system of the Verilog-A(MS) distribution for HSPICE

- Installation directory of your choice
  - One .sp and .va file per library building block
  - The .sp file contains the simulation control and stimulus statements and the call to the .va netlist (or the “template” file)

- Macro_lib subdirectory
  - Contains two mandatory files and the library file
    - constants.vams
    - disciplines.vams
    - IBIS_macro_library.va
  - plus a small collection of parameter data files (*.dat)
    - these files contain IBIS buffer parameters, such as C_comp, I-V and V-t tables, or
    - appropriate data tables for the PWL sources
File system of the Verilog-A(MS) distribution for HSPICE
File system of the Verilog-A(MS) distribution for HSPICE
File system of the VHDL-A(MS) distribution for SMASH

- Installation directory of your choice
  - One .nsx and .pat file per library building block
  - The .pat (pattern) file contains the simulation control statements
  - The .nsx file contains both the top level SPICE and the VHDL-AMS netlist of the “template”

- Macro_lib subdirectory
  - Contains the library and a function file
    - IBIS_macro_library.vhd
    - MacroLib_functions.vhd
  - plus a collection of parameter data files (*.txt)
    - these files contain IBIS buffer parameters, such as C_comp, I-V and V-t tables, or
    - appropriate data tables for the PWL sources

*Other brands and names are the property of their respective owners
File system of the VHDL-A(MS) distribution for SMASH

<table>
<thead>
<tr>
<th>Name</th>
<th>Size</th>
<th>Type</th>
<th>Date Modified</th>
</tr>
</thead>
<tbody>
<tr>
<td>Macro_Lib</td>
<td></td>
<td>File Folder</td>
<td>2/2/2006 1:05 PM</td>
</tr>
<tr>
<td>J3ES_STATE_test.nxs</td>
<td>2 MB</td>
<td>NSK File</td>
<td>2/1/2006 1:05 PM</td>
</tr>
<tr>
<td>J3ES_STATE_test.pat</td>
<td>1 KB</td>
<td>PAT File</td>
<td>2/1/2006 1:15 PM</td>
</tr>
<tr>
<td>J3ES_C_test.nxs</td>
<td>1 KB</td>
<td>NSK File</td>
<td>2/1/2006 2:05 PM</td>
</tr>
<tr>
<td>J3ES_C_test.pat</td>
<td>1 KB</td>
<td>PAT File</td>
<td>2/1/2006 2:05 PM</td>
</tr>
</tbody>
</table>

*Other brands and names are the property of their respective owners*
File system of the VHDL-A(AMS) distribution for SMASH

![Folder tree and file list for SMASH Macro Lib directory](image)

*Other brands and names are the property of their respective owners*
The “Compiled” subdirectory

- The Macro_lib directory of the VHDL-A(MS) distribution includes a directory called “Compiled”
  - This contains a compiled version of the library and all of its functions as a convenience to speed up testing
  - You don’t have to use it, but it can save you time

- The “Copy_lib.bat” file copy the compilation into the work area of each library building block example
  - (Copying is faster than compiling it 63 times)

- The “Exclude_files.txt” file will prevent the duplication of files which do not need to be copied

- This is a SMASH specific step, other tools may deal with the WORK and user library hierarchy in a different manner
File system of the VHDL-A(MS) distribution for SMASH

<table>
<thead>
<tr>
<th>Name</th>
<th>Size</th>
<th>Type</th>
<th>Date Modified</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBE1_EQUIivating_BITMAP_OPENINGS.M</td>
<td>117 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
<tr>
<td>IBE1_EQUIivating_BITMAP_OPENINGS.S</td>
<td>118 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
<tr>
<td>IBE1_EQUIivating_BITMAP_OPENINGS.M</td>
<td>111 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
<tr>
<td>IBE1_EQUIivating_BITMAP_OPENINGS.S</td>
<td>111 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
<tr>
<td>IBE1_EQUIivating_BITMAP_OPENINGS.M</td>
<td>117 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
<tr>
<td>IBE1_EQUIivating_BITMAP_OPENINGS.S</td>
<td>118 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
<tr>
<td>IBE1_EQUIivating_BITMAP_OPENINGS.M</td>
<td>111 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
<tr>
<td>IBE1_EQUIivating_BITMAP_OPENINGS.S</td>
<td>111 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
<tr>
<td>IBE1_EQUIivating_BITMAP_OUTPUT.M</td>
<td>187 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
<tr>
<td>IBE1_EQUIivating_BITMAP_OUTPUT.S</td>
<td>181 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
<tr>
<td>IBE1_EQUIivating_BITMAP_OUTPUT.M</td>
<td>181 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
<tr>
<td>IDEAL-BIES.C</td>
<td>4 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
<tr>
<td>IDEAL-BIES.CCC</td>
<td>5 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
<tr>
<td>IDEAL-BIES.CCSS</td>
<td>4 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
<tr>
<td>IDEAL-BIES.CCSS_6BS</td>
<td>4 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
<tr>
<td>IDEAL-BIES.CCSS_DELAY</td>
<td>4 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
<tr>
<td>IDEAL-BIES.CCSS_DIV</td>
<td>6 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
<tr>
<td>IDEAL-BIES.CCSS_MAX</td>
<td>5 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
<tr>
<td>IDEAL-BIES.CCSS_MIN</td>
<td>5 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
<tr>
<td>IDEAL-BIES.CCSS_MULTI</td>
<td>5 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
<tr>
<td>IDEAL-BIES.CCSS_PWM</td>
<td>6 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
<tr>
<td>IDEAL-BIES.CCSS_SUM</td>
<td>5 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
<tr>
<td>IDEAL-BIES.CCSS_CLK</td>
<td>5 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
<tr>
<td>IDEAL-BIES.CCSS_CK</td>
<td>5 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
<tr>
<td>IDEAL-BIES.CCSS_CV</td>
<td>4 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
<tr>
<td>IDEAL-BIES.CCSS_CVABS</td>
<td>4 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
<tr>
<td>IDEAL-BIES.CCSS_CVDR</td>
<td>5 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
<tr>
<td>IDEAL-BIES.CCSS_CVDS</td>
<td>6 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
<tr>
<td>IDEAL-BIES.CCSS_CVMAX</td>
<td>5 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
<tr>
<td>IDEAL-BIES.CCSS_CVMUT</td>
<td>5 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
<tr>
<td>IDEAL-BIES.CCSS_CVWM</td>
<td>5 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
<tr>
<td>IDEAL-BIES.CCSS_CPW</td>
<td>6 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
<tr>
<td>IDEAL-BIES.CCSS_SM</td>
<td>5 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
<tr>
<td>IDEAL-BIES.CCSS_LV</td>
<td>4 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
<tr>
<td>IDEAL-BIES.CCSS_LV</td>
<td>4 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
<tr>
<td>IDEAL-BIES.CCSS_LV</td>
<td>4 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
<tr>
<td>IDEAL-BIES.CCSS_LV</td>
<td>4 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
<tr>
<td>IDEAL-BIES.CCSS_LV</td>
<td>4 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
<tr>
<td>IDEAL-BIES.CCSS_LV</td>
<td>4 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
<tr>
<td>IDEAL-BIES.CCSS_LV</td>
<td>4 KB</td>
<td>YIF File</td>
<td>2/1/2005 11:20 PM</td>
</tr>
</tbody>
</table>

*Other brands and names are the property of their respective owners*
Resistor example - simulation code

```vhdl
-- Resistor example

library IEEE, MacroLib;
use IEEE.ELECTRICAL_SYSTEMS.all;

entity IBIS_R_VHDL is
  port (terminal Node_p, Node_n : electrical);
end entity IBIS_R_VHDL;

architecture Simple_test of IBIS_R_VHDL is
begin
  IBIS_R1 : entity MacroLib.IBIS_R(Ideal)
    generic map ( Rval => 5.0, Scale => 2.0 )
    port map ( p => Node_p, n => Node_n );
end architecture Simple_test;
```

**macro model netlist**

**control statements**

**library call**

**parameters**

**nodes**
Resistor example - library code

```vhdl
library IEEE;
use IEEE.electrical_systems.all;

entity IBIS_R is
  generic (Rval : real := 1.0;
            Scale : real := 1.0);
  port (terminal p, n : electrical);
end entity IBIS_R;

architecture ideal of IBIS_R is
  quantity Vout across Iout through p to n;
  begin
    Vout <= Scale * Rval * Iout;
  end architecture ideal;
```

```verilog
module IBIS_R (p, n);
  electrical p, n;
  branch (p, n) Out;
  parameter real Rval = 1.0;
  parameter real Scale = 1.0;
  analog begin
    V(Out) <+ Scale * Rval * I(Out);
  end
endmodule
```

**VHDL-A(MS)**

**Verilog-A(MS)**

*Other brands and names are the property of their respective owners*
Resistor example - waveforms
VCC example - waveforms
IBIS_IO buffer example - code

```vhd
library IEEE, std;
use IEEE.STD_LOGIC_1164.all;

entity IBIS_IO_WHDL is
  port (Input, Output, Enable : electrical);
end entity IBIS_IO_WHDL;
architecture Simple_test of IBIS_IO_WHDL is
begin
  IBIS_001 : entity std.IBIS_001(I) port map (Input => Power, Output => Ground, Enable => Enable);
end architecture Simple_test;
```

- **macro model netlist**
- **library call**
- **parameters**
- **nodes**
- **control statements**
IBIS_IO buffer example - waveforms
IBIS_IO using real IBIS data
Verilog-A(MS) waveforms of IBIS_IO
IBIS_IO using real IBIS data
The results are identical!

SMASH VHDL-A(MS) and HSPICE Verilog-A(MS) overlay
A differential pre/de-emphasis buffer
- a circuit netlist serves as the macro model, instantiating
  - four Verilog-A or VHDL-AMS IBIS I/O buffer models,
  - an inverter,
  - two ideal delays, and
  - eight current sources to scale the Boost buffer’s current

Diagram borrowed from M. Mirmak
Pre-emphasis buffer - test bench

Test Verilog-A "Macro Model Template" in HSPICE
********************************************************************************
.TRAN 5.0ps 150.0ns
.OPTIONS POST=1 POST_VERSION=9007 PROBE
.hd1 ".\PreDeMacro.va"
********************************************************************************
.PROBE TRAN
+ P1s = V(P1s)
+ OutP = V(Out_p)
+ OutN = V(Out_n)
********************************************************************************
Vvcc  Vcc  0  DC= 5.0
* This source represents a 111000111000 pattern
Vp1s  P1s  0  PULSE  (1.0  0.0  1.0ns  1.0ps  1.0ps  30.0ns  60.0ns)
********************************************************************************
X1  P1s  Out_p  Out_n  Vcc  Vcc  0  0  Vcc  PreDeMacro
*  In  Out_p  Out_n  PC  PU  PD  GC  En
*
Rld1  Out_p  Vtt  R= 50.0
Rld2  Out_n  Vtt  R= 50.0
********************************************************************************
.END
********************************************************************************
module PreDeMacro (InD, IOp, IOn, PCref, PUref, PDref, GCref, EnD);
  input  InD, EnD;
  electrical InD, EnD;

  inout  IOp, IOn, PCref, PUref, PDref, GCref;
  electrical IOp, IOn, PCref, PUref, PDref, GCref;

  electrical InNM, InPB, InNB, Dref;
  electrical PUrefPB, PDrefPB, PCrefPB, GCrefPB;
  electrical PUrefNB, PDrefNB, PCrefNB, GCrefNB;
  electrical RcvPM, RcvNM, RcvPB, RcvNB;

  parameter real BitDelay = 10.0e-9;
  parameter real ScaleBoost = 0.5;

  // These lines show how to pass IV and Vt tables into the IBIS_IO model
  //-------------------------------------------------------------------------------------
  `include "lab_1_io50v_t.dat"

  IBIS_IO #('IO_data) PosM (PUref, PDref, IOp, InD, EnD, RcvPM, PCref, GCref);
  IBIS_IO #('IO_data) NegM (PUref, PDref, IOn, InNM, EnN, RcvNB, PCref, GCref);
  IBIS_IO #('IO_data, .kL_p(ScaleBoost), .kL_p(ScaleBoost), .kL_p(ScaleBoost), .kL_p(ScaleBoost))
    PosB (PUref, PDref, IOp, InPB, EnD, RcvPB, PCref, GCref);
  IBIS_IO #('IO_data, .kL_p(ScaleBoost), .kL_p(ScaleBoost), .kL_p(ScaleBoost), .kL_p(ScaleBoost))
    NegB (PUref, PDref, IOn, InNB, EnD, RcvNB, PCref, GCref);

  endmodule
Pre-emphasis buffer - waveforms
Wrap up

● Links to the two test suites including the most current versions of the macro model library


● Please try it out and provide feedback, that is the only way this effort can be made useful!

● Lots of capabilities and features could still be added to the library, but we need to know what is needed, and what is practical, etc…

● In case you need help to find SMASH…