

\*\*\*\*\*  
\*\*\*\*\*

BIRD ID#: 117.4  
ISSUE TITLE: Parameterize A\_to\_D and D\_to\_A Converters  
REQUESTER: Arpad Muranyi, Mentor Graphics  
Ambrish Varma,  
Feras Al-Hawari,  
Taranjit Kukal Cadence Design Systems  
DATE SUBMITTED: September 29, 2010  
DATE REVISED: October 5, 2010, November 8, 2010, March 23, 2011,  
March 27, 2012  
DATE ACCEPTED BY IBIS OPEN FORUM:

\*\*\*\*\*  
\*\*\*\*\*

STATEMENT OF THE ISSUE:

[External Model]s and [External Circuit]s with analog ports communicate through A\_to\_D and/or D\_to\_A converters with the purely digital signals of \*\* the EDA tool. The current specification only allows hard-coded values as \*\* arguments for these converters in the IBIS file. There are situations, \*\* however, when it would be desirable to parameterize the arguments of these \*\* converters. For example, an IBIS model could be made much more compact \*\* with parameterized converters than having to use multiple copies of the \*\* otherwise identical [Model]s through [Model Selector].

\*\*\*\*\*

STATEMENT OF THE RESOLVED SPECIFICATIONS:

A new subparameter called Converter\_Parameters shall be added to the \*\* [External Model] and [External Circuit] keywords to provide a mechanism \*\* to declare parameters used as arguments for the A\_to\_D and/or D\_to\_A \*\*\*\* converters. The Converter\_Parameters of the [External Circuit] or \*\*\*\* [External Model] keywords shall be initialized with a constant numeric \*\*\*\* literal value and/or the root name of a parameter tree provided in the \*\*\*\* .ibs file or the root name of a parameter tree provided in a separate \*\*\*\* file, followed by an open and close parentheses enclosing the name of a \*\*\*\* parameter which is declared in that tree. When the constant numeric \*\*\*\* literal and the root name of a parameter tree are both present in such \*\*\*\* an assignment, the constant numeric literal value shall serve as a \*\*\*\* default value for the assignment in case the assignment using the \*\*\*\* reserved word fails.

\*\*\*\* If the Converter\_Parameters subparameter is part of an [External Model] \*\*\*\* keyword, and the corresponding [Model] contains an [Algorithmic Model] \*\*\*\* keyword, the EDA tool will first search for a parameter tree in the .ami file \*\*\*\* that is defined in this [Algorithmic Model] keyword. If the parameter tree is \*\*\*\* not found in that .ami file, or if the [Algorithmic Model] keyword doesn't \*\*\*\* exist in that [Model], or if the Converter\_Parameters subparameter is part \*\*\*\* of an [External Circuit] keyword, the EDA tool will search for a parameter \*\*\*\* tree in the current .ibs file. If a match is not found, the EDA tool will \*\*\*\* next look for a match in an external .par file. If the matching parameter \*\*\*\* tree is located in an external .par file, the .par file must be located in the \*\*\*\* same directory as the .ibs file. The file names of .par files must follow \*\*\*\* the rules for file names given in Section 3, GENERAL SYNTAX RULES AND \*\*\*\* GUIDELINES.

\*\*\*\*\* Multiple [External Model] or [External Circuit] Parameters may be listed  
 \*\*\*\*\* on the same line with one assignment, in which case all of the  
 \*\*\*\*\* parameters on that line will be assigned the same value.

\*\*\*\*\* To implement this concept, the IBIS specification shall be changed  
 \*\*\*\*\* as outlined below. The page number references are with respect to  
 \*\*\*\*\* the official IBIS v5.0 specification.

\*\* On pg. 105 and 125 replace:

| Sub-Params: Language, Corner, Parameters, Ports, D\_to\_A, A\_to\_D

\*\* with the following line:

|\*\* Sub-Params: Language, Corner, Parameters, Converter\_Parameters, Ports,  
 |\*\* D\_to\_A, A\_to\_D

\*\*\* On pg. 107 add the following lines after the "Parameters:" section:

```
|
| Converter_Parameters:
|
|** This optional subparameter lists and initializes parameter
|** names to be used as arguments for the A_to_D and/or D_to_A
|*** converter(s) of the [External Model] keyword under which it
|** appears. The list of Converter_Parameters may span several
|** lines by using the word Converter_Parameters at the start of
|** each line. Any A_to_D or D_to_A argument which is entered
|** as a parameter must be declared and initialized with the
|** Converter_Parameters subparameter.
|**
|** Converter_Parameters are locally scoped under each
|*** [External Model] keyword, i. e. the same converter parameter
|*** under two different [External Model]s will have independent
|** values.
|**
|** The Converter_Parameters subparameter may contain one or more
|** parameter names, which must be followed by an equal sign and
|***** a constant numeric literal and/or the root name of a parameter
|***** tree that is followed by an open and close parentheses
|***** enclosing the name of a parameter in that tree. The parameter
|***** tree may be placed at the end of the .ibs file after the last
|***** [END] keyword, or an external parameter file. If the
|***** Converter_Parameters subparameter is part of an [External Model]
|***** keyword, and the corresponding [Model] contains an [Algorithmic
Model]
|***** keyword, the EDA tool will first search for a parameter tree
|***** in the .ami file that is defined in this [Algorithmic Model]
|***** keyword. If the parameter tree is not found in that .ami file,
|***** or if the [Algorithmic Model] keyword doesn't exist in that
|***** [Model], or if the Converter_Parameters subparameter is part of an
|***** [External Circuit] keyword, the EDA tool will search for a
|***** parameter tree in the current .ibs file. If a match is not
|***** found, the EDA tool will next look for a match in an external
|***** .par file. If the matching parameter tree is located in an
|***** external .par file, the .par file must be located in the same
|***** directory as the .ibs file. The file names of .par files must
```

BIRD117\_4.txt

```
|*****
|***** follow the rules for file names given in Section 3, GENERAL
|***** SYNTAX RULES AND GUIDELINES.
|*****
|***** When TreeRootName(ParameterName) and a constant numeric literal
|***** are both present in an assignment they must be separated by at
|***** least one white space. In this case, the EDA tool should attempt
|***** to make the assignment using TreeRootName(ParameterName) first.
|***** If that fails (for example due to not finding the parameter tree
|***** in any of the files or the or the parameter name in that tree)
|***** the constant numeric literal shall be used for the assignment.
|***** When multiple converter parameters are listed on a single line
|***** with one assignment, all of the parameters on that line will be
|***** assigned the same value.
|**
|**
|** The EDA tool may provide additional means to the user to
|** make assignments to Converter_Parameters. This may include
|** the option to override the values provided in the .ibs file,
|** or to allow the user to make selections for multi-valued
|***** parameters in the parameter tree.
|
```

\*\*\* On pg. 126 add the following lines after the "Parameters:" section:

```
| Converter_Parameters:
|
|** This optional subparameter lists and initializes parameter
|** names to be used as arguments in the A_to_D and/or D_to_A
|*** converter(s) of the [External Circuit] keyword under which it
|** appears. The list of Converter_Parameters may span several
|** lines by using the word Converter_Parameters at the start of
|** each line. Any A_to_D or D_to_A argument which is entered
|** as a parameter must be declared and initialized with the
|** Converter_Parameters subparameter.
|**
|** Converter_Parameters are locally scoped under each
|*** [External Circuit] keyword, i. e. the same converter parameter
|*** under two different [External Circuit]s will have independent
|** values.
|**
|** The Converter_Parameters subparameter may contain one or more
|** parameter names, which must be followed by an equal sign and
|***** a constant numeric literal and/or the root name of a parameter
|***** tree that is followed by an open and close parentheses
|***** enclosing the name of a parameter in that tree. The parameter
|***** tree may be placed at the end of the .ibs file after the last
|***** [END] keyword, or an external parameter file. If the
|***** Converter_Parameters subparameter is part of an [External Model]
|***** keyword, and the corresponding [Model] contains an [Algorithmic
Model]
|***** keyword, the EDA tool will first search for a parameter tree
|***** in the .ami file that is defined in this [Algorithmic Model]
|***** keyword. If the parameter tree is not found in that .ami file,
|***** or if the [Algorithmic Model] keyword doesn't exist in that
|***** [Model], or if the Converter_Parameters subparameter is part of an
|***** [External Circuit] keyword, the EDA tool will search for a
|***** parameter tree in the current .ibs file. If a match is not
|***** found, the EDA tool will next look for a match in an external
|***** .par file. If the matching parameter tree is located in an
|***** external .par file, the .par file must be located in the same
```

```
|*****
|***** directory as the .ibs file. The file names of .par files must
|***** follow the rules for file names given in Section 3, GENERAL
|***** SYNTAX RULES AND GUIDELINES.
|*****
|***** When TreeRootName(ParameterName) and a constant numeric literal
|***** are both present in an assignment they must be separated by at
|***** least one white space. In this case, the EDA tool should attempt
|***** to make the assignment using TreeRootName(ParameterName) first.
|***** If that fails (for example due to not finding the parameter tree
|***** in any of the files or the or the parameter name in that tree)
|***** the constant numeric literal shall be used for the assignment.
|***** When multiple converter parameters are listed on a single line
|***** with one assignment, all of the parameters on that line will be
|***** assigned the same value.
|*
|**
|** The EDA tool may provide additional means to the user to
|** make assignments to Converter_Parameters. This may include
|** the option to override the values provided in the .ibs file,
|***** or to allow the user to make selections for multi-valued
|***** parameters in the parameter tree.
```

\*\* On pg. 109 and pg. 127 insert after this paragraph:

```
|
| The vlow and vhigh entries accept analog voltage values which
| must correspond to the digital off and on states, where the
| vhigh value must be greater than the vlow value. For example,
| a 3.3 V ground-referenced buffer would list vlow as 0 V and
| vhigh as 3.3 V. The trise and tfall entries are times, must
| be positive and define input ramp rise and fall times between
| 0 and 100 percent.
```

the following lines:

```
|
|** Any or all of these entries may be defined by parameter names,
|** which must be declared and initialized by one or more
|** Converter_Parameters subparameter.
```

On pg. 110 and pg. 128 insert after this paragraph:

```
|
| The vlow and vhigh entries list the low and high analog
| threshold voltage values. The reported digital state on
| D_receive will be '0' if the measured voltage is lower than
| the vlow value, '1' if above the vhigh value, and 'X'
| otherwise.
```

the following lines:

```
|
|** Any or all of these entries may be defined by parameter names,
|** which must be declared and initialized by one or more
|** Converter_Parameters subparameter.
```

\*\*\*\*\* Modify the example provided in BIRD 116 for pg. 119 as follows:

|-----

```

| Example [External Model] using ISS:
|-----
|
[Model] ExBufferISS
Model_type I/O
Vinh = 2.0
Vinl = 0.8
|
| Other model subparameters are optional
|
|          typ      min      max
[Voltage Range]  3.3      3.0      3.6
|
[Ramp]
dV/dt_r          1.57/0.36n  1.44/0.57n  1.73/0.28n
dV/dt_f          1.57/0.35n  1.46/0.44n  1.68/0.28n
|
[External Model]
Language ISS
|
| Corner corner_name file_name      circuit_name (.subckt name)
Corner     Typ          buffer_typ.spi  buffer_io_typ
Corner     Min          buffer_min.spi  buffer_io_min
Corner     Max          buffer_max.spi  buffer_io_max
|
| List of parameters
Parameters sp_file_name = TreeRootName(TstoneFile) "MySparameterFile.s4p"
Parameters Cl_value
Parameters Rl_value = TreeRootName(R1)
|
| List of converter parameters
Converter_Parameters MyVlow  = 0.0
Converter_Parameters MyHigh  = 3.3
Converter_Parameters MyVinl  = TreeRootName(Vinl)
Converter_Parameters MyVinh  = TreeRootName(Vinh)
Converter_Parameters MyTfall = TreeRootName(Trf) 1.0p
Converter_Parameters MyTrise = TreeRootName(Trf)
|
| Ports List of port names (in same order as in ISS)
Ports A_signal my_drive my_enable my_receive my_ref
Ports A_puref A_pdref A_pceref A_gceref A_extref
|
| D_to_A d_port  port1      port2      vlow vhigh trise tfall corner_name
D_to_A   D_drive my_drive   my_ref    MyVlow  MyVhigh MyTfall MyTrise Typ
D_to_A   D_enable my_enable A_gceref  0.0     3.3     0.5n    0.3n    Typ
|
| A_to_D d_port  port1      port2      vlow vhigh corner_name
A_to_D   D_receive my_receive my_ref    MyVinl MyVinh Typ
|
| Note: A_signal might also be used instead of a user-defined interface port
| for measurements taken at the die pads
|
[End External Model]
|

```

\*\*\*\*\* Modify the example provided in BIRD 116 for pg. 129 as follows:

```

|-----|
| Example [External Circuit] using ISS:
|-----|
|
[External Circuit] BUFF-ISS
Language ISS
|
| Corner corner_name file_name      circuit_name (.subckt name)
Corner    Typ          buffer_typ.spi  bufferb_io_typ
Corner    Min          buffer_min.spi  bufferb_io_min
Corner    Max          buffer_max.spi  bufferb_io_max
|
| List of parameters
Parameters sp_file_name = TreeRootName(TstoneFile) "MySparameterFile.s4p"
Parameters Cl_value
Parameters Rl_value = TreeRootName(R1)
|
Converter_Parameters MyVlow  = 0.0
Converter_Parameters MyHigh  = 3.3
Converter_Parameters MyVinl  = TreeRootName(Vinl)
Converter_Parameters MyVinh  = TreeRootName(Vinh)
Converter_Parameters MyTfall = TreeRootName(Trf) 1.0p
Converter_Parameters MyTrise = TreeRootName(Trf)
|
| Ports List of port names (in same order as in ISS)
Ports A_signal int_in int_en int_out A_control
Ports A_puref A_pdref A_pcref A_gcref
|
| D_to_A d_port  port1 port2      vlow vhigh trise tfall corner_name
D_to_A   D_drive int_in my_gcref MyVlow MyVhigh MyTfall MyTrise Typ
D_to_A   D_enable int_en my_gnd   0.0  3.3   0.5n  0.3n  Typ
D_to_A   D_enable int_en my_gnd   0.0  3.0   0.6n  0.3n  Min
D_to_A   D_enable int_en my_gnd   0.0  3.6   0.4n  0.3n  Max
|
| A_to_D d_port      port1  port2      vlow vhigh corner_name
A_to_D   D_receive  int_out my_gcref MyVinl MyVinh Typ
|
| Note, the A_signal port might also be used and int_out not defined in
| a modified .subckt.
|
[End External Circuit]
|
|**
**...
**...
[END]

(TreeRootName
  (Description "Converter_Parameters illustration")

  (Reserved_Parameters
    (... (... ) (... ) (... ) )
  ) | End Reserved_Parameters

  (Model_Specific
    (TstoneFile (Usage In)(List "Typ.s4p" "Min.s4p" "Max.s4p" "SSS.s4p"
"FFF.s4p")(Type String)
    (Vinh (Usage In)(List 0.8 0.7 0.9)(Type Float)
    (Vinl (Usage In)(List 0.2 0.1 0.3)(Type Float)

```

BIRD117\_4.txt

```
(R1 (Usage In)(Range 50 45 55)(Type Float)
  (Trf (Usage In)(Value 10.0e-12)(Type Float))
) | End Model_Specific
)
```

\*\*\*\*\*

ANALYSIS PATH/DATA THAT LED TO SPECIFICATION

\*\*\*\* BIRD 117.1 was flawed because it allowed [External Circuit] to reference  
\*\*\*\* .ami files for parameter assignments with the "AMIfile()" reserved word.  
\*\*\* These references cannot be resolved, because  
\*\*\* [External Circuit] is on the same scoping level as [Model] and therefore  
\*\*\* it is not associated with any [Algorithmic Model] keywords which reside  
\*\*\* under the [Model] keyword. For this reason [External Circuit] doesn't  
\*\*\* have a way to know what .ami file the "AMIfile()" syntax should read.  
\*\*\*\* BIRD 117.2 removed the possibility to use the reserved  
\*\*\* word "AMIfile()" for parameter assignments under [External Circuit] to  
\*\*\* eliminate that problem.

\*\*\*\* In BIRD 117.3 a slight modification was made to the rules of the  
\*\*\*\* reserved word AMIfile(). The modification makes provisions for the  
\*\*\*\* usage of a default value in case the assignment using the reserved  
\*\*\*\* word fails for some reason. These changes are marked by four asterisks  
\*\*\*\* at the beginning of each line.

\*\*\*\*\* In BIRD 117.4 the concept of AMIfile() was generalized so that instead  
\*\*\*\*\* of referencing an .ami parameter file with the reserved word AMIfile(),  
\*\*\*\*\* any parameter tree can be referenced by the root name of a parameter  
\*\*\*\*\* tree. The parameter tree may reside in the .ibs file, an .ami file,  
\*\*\*\*\* or a .par file. The search rules are patterned after the search rules  
\*\*\*\*\* defined for the package models in IBIS.

\*\*\*\*\*

ANY OTHER BACKGROUND INFORMATION:

Notes with respect to BIRD 117.4:

We may want to define somewhere in the specification that if the parameter tree is located in the .ibs file, it should be placed after the last [End] statement.

We might also want to consider to have a separate and independent section in the IBIS specification to describe the parameter tree format (independently from IBIS-AMI).

\*\*\*\*\*