Agilent Technologies

Advanced Design System 2011.01

Feburary 2011
IBIS Models

Acknowledgments

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: 7z.dll: GNU LGPL + unRAR restriction, 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.

Qt Version 4.6.3 - Qt Notice: The Qt code was modified. Used by permission. Qt copyright: Qt Version 4.6.3, Copyright (c) 2010 by Nokia Corporation. All Rights Reserved. Qt License: 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 Agilent Inc. at brian_buchanan@agilent.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.

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 Agilent Technologies and is known as "Agilent 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.

Warranty The material contained in this document is provided "as is", and is subject to being changed, without notice, in future editions. Further, to the maximum extent permitted by applicable law, Agilent disclaims all warranties, either express or implied, with regard to this documentation and any information contained herein, including but not limited to the implied warranties of merchantability and fitness for a particular purpose. Agilent shall not be liable for errors or for incidental or consequential damages in connection with the furnishing, use, or performance of this document or of any information contained herein. Should Agilent and the user have a separate written agreement with warranty terms covering the material in this document that conflict with these terms, the warranty terms in the separate agreement shall control.

Technology Licenses The hardware and/or software described in this document are furnished under a license and may be used or copied only in accordance with the terms of such license. 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 Agilent. 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 Agilent only.

Restricted Rights Legend U.S. Government Restricted Rights. Software and technical data rights granted to the federal government include only those rights customarily provided to end user customers. Agilent provides this customary commercial license in Software and technical data pursuant to FAR 12.211 (Technical Data) and 12.212 (Computer Software) and, for the Department of Defense, DFARS 252.227-7015 (Technical Data - Commercial Items) and DFARS 227.7202-3 (Rights in Commercial Computer Software or Computer Software Documentation).

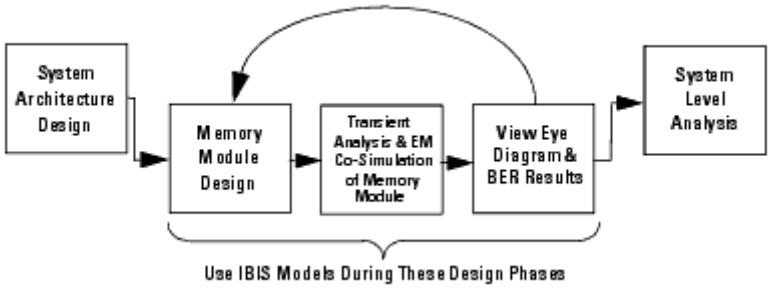
About IBIS Models

IBIS (Input/Output Buffer Information Specification) is a behavioral modeling specification for characterizing the inputs and outputs of integrated circuits. The non-linear elements in the models are based on I-V tables, so they simulate faster than netlists that include detailed transistor models such as BSIM or VBIC. The IBIS approach allows semiconductor vendors to hide their circuit and protect intellectual property (IP), because, like a datasheet, only behavioral information is revealed.

Notes

- IP hiding can also be achieved by asking the vendor to encrypt the netlist using the ADS IP Encoder (formerly a separate product E8894A but now combined into W2200 ADS Core). If a netlist with the ADS encryption key is not available, but a netlist with the Synopsys HSPICE encryption key is, then the model can be incorporated into ADS using cosimulation with HSPICE (formerly E8828A but now combined into W2302 ADS Transient Convolution Element). These approaches are more accurate than using I-V tables, at the cost of increased simulation run time.
- The IBIS specification has been formally ratified as ANSI/EIA-656-A (see http://www.eigroup.org/ibis/).

Use Model



The primary use for IBIS models is designing chip-to-chip serial links such as those on memory modules and boards. You can also use these models to design equalized backplane and cable assemblies. The models are designed for use in creating driver and receiver circuits at speeds below ~5 gigabits/sec. Above ~5 gigabits/sec (encrypted or unencrypted) netlists with accurate transistor models are recommended (see note above).

Support for IBIS Models in ADS

ADS IBIS Models provides easy and reliable access to IBIS technology that is largely compliant with the current standard (see limitations below) and accurate in its simulation results.

This feature provides the following:

- an interface between ADS and the standard IBIS parser (currently supporting version 4.2)

- ADS components for each high-level IBIS model type
- a generic IBIS component
- a schematic palette of these built-in IBIS components
- an IBIS model in the ADS simulator

The table of supported keywords by ADS release is available at http://www.home.agilent.com/upload/cmc_upload/All/ibis_supported_features_10-20-2009.pdf .

Implementation Limitations

Presently, IBIS Models are not compliant with the following specification parameters:

- [Pin Mapping]
- Series models: [Series Pin Mapping], [Series Switch Groups], [On], [Off], [R/L/R1/C/Lc/Rc Series], [Series Current], [Series MOSFET]
- [Model Spec] (the sub-parameters Vinl and Vinh are handled)
- [Receiver Thresholds]
- internal power supply: [Pullup/Pulldown/POWER Clamp/GND Clamp/External Reference or Voltage Reference]
- Board description files (*.ebd)
- Package files (*.pkg): [Package Model], [Alternate Package Models], [Define Package Model]
- External models and circuits: [External Model], [External Circuit], [Node Declarations], [Circuit Call]
- [Test Data]
- [External Reference]

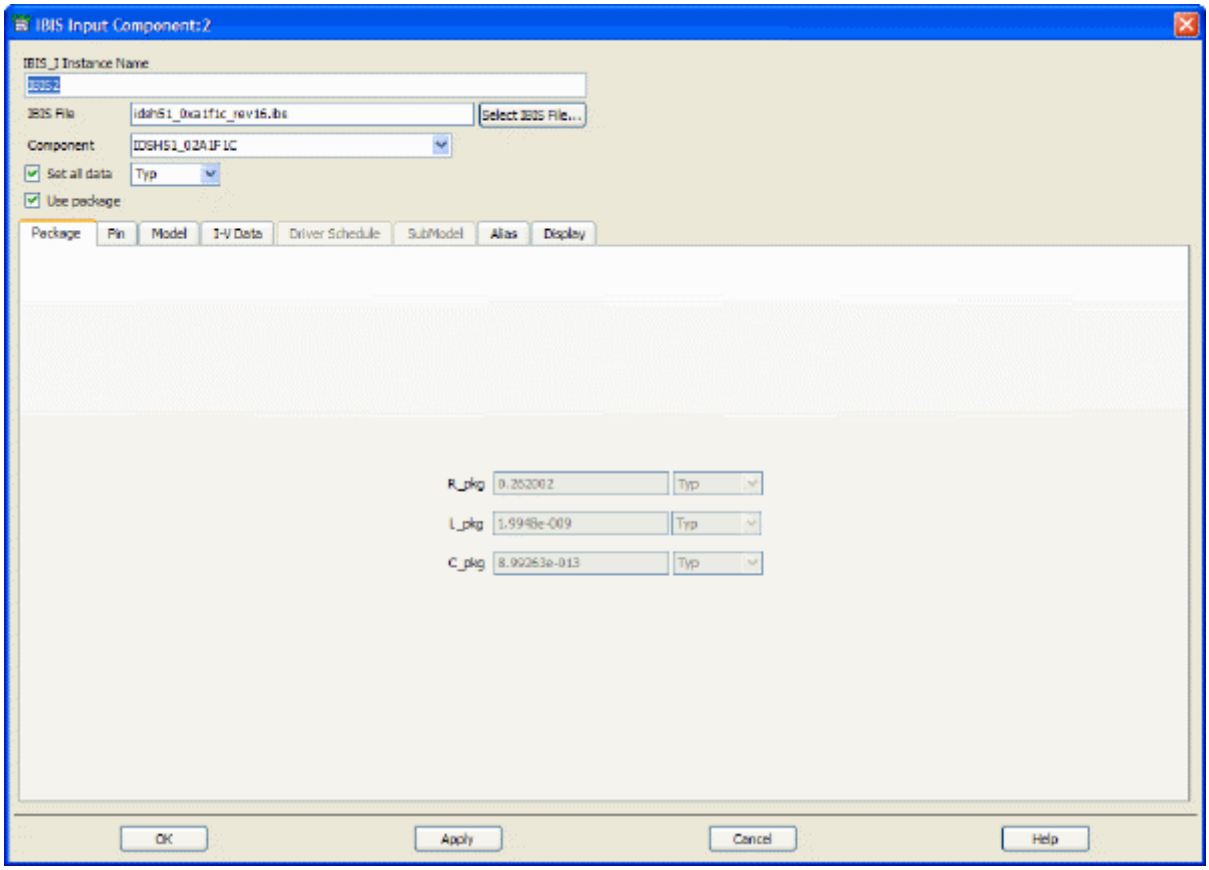
Presently, Interconnect Modeling (ICM) is not supported. Neither keyword form nor file form packages are supported.

List of Models

- *IBIS (Generic Model)* (ibis)
- *IBIS_3S (3-State)* (ibis)
- *IBIS_D3S (Differential 3-State)* (ibis)
- *IBIS_DI (Differential Input)* (ibis)
- *IBIS_DIO (Differential Input-Output)* (ibis)
- *IBIS_DIO_OPENSINK (Differential IO Open Sink)* (ibis)
- *IBIS_DIO_OPENSOURCE (Differential IO Open Source)* (ibis)
- *IBIS_DO (Differential Output)* (ibis)
- *IBIS_DOPENSINK (Differential Open Sink)* (ibis)
- *IBIS_DOPENSOURCE (Differential Open Source)* (ibis)
- *IBIS_DT (Differential Terminator)* (ibis)
- *IBIS_I (Input)* (ibis)

- *IBIS_IO (Input-Output)* (ibis)
- *IBIS_IO_OPENSINK (IO Open Sink)* (ibis)
- *IBIS_IO_OPENSOURCE (IO Open Source)* (ibis)
- *IBIS_O (Output)* (ibis)
- *IBIS_OPENSINK (Open Sink)* (ibis)
- *IBIS_OPENSOURCE (Open Source)* (ibis)
- *IBIS_T (Terminator)* (ibis)
- *IBIS_3S_ECL (3-State_ECL)* (ibis)
- *IBIS_D3S_ECL (Differential 3-State_ECL)* (ibis)
- *IBIS_DI_ECL (Differential Input_ECL)* (ibis)
- *IBIS_DIO_ECL (Differential Input-Output_ECL)* (ibis)
- *IBIS_DO_ECL (Differential Output_ECL)* (ibis)
- *IBIS_I_ECL (Input_ECL)* (ibis)
- *IBIS_IO_ECL (Input-Output_ECL)* (ibis)
- *IBIS_O_ECL (Output_ECL)* (ibis)

IBIS Models Main Dialog



This feature provides IBIS file choice and high-level parameter configuration through its main dialog box and detailed parameter configuration through six tabs.

IBIS File

Click **Select IBIS File** to browse to the .ibs file containing your model. This field is initially

blank and not directly editable. With the exception of the Display tab, all other tabs and fields are inactive and blank until content is available.

Component

The component name to use from the IBIS file. Default component is the first choice.

Set all data

Select this checkbox to set all the model's fields as typical (*Typ*), minimum (*Min*), maximum (*Max*), or one of two predefined combinations: fast (*Fast*) or slow (*Slow*). You can also enter a user-defined label by selecting the blank line in the drop-down (see details in Sweeping IBIS Parameters). Selecting this checkbox disables other *Typ/Min/Max* fields and allows *Typ/Min/Max/Fast/Slow/<user_label>* values. Selecting Fast and Slow sets some fields to Min and others to Max in the model. This parameter is selected (On) by default.
To configure a particular set of parameters:

1. Click on the associated tab.
2. Configure the parameters.
3. Click **Apply**, then **OK**.

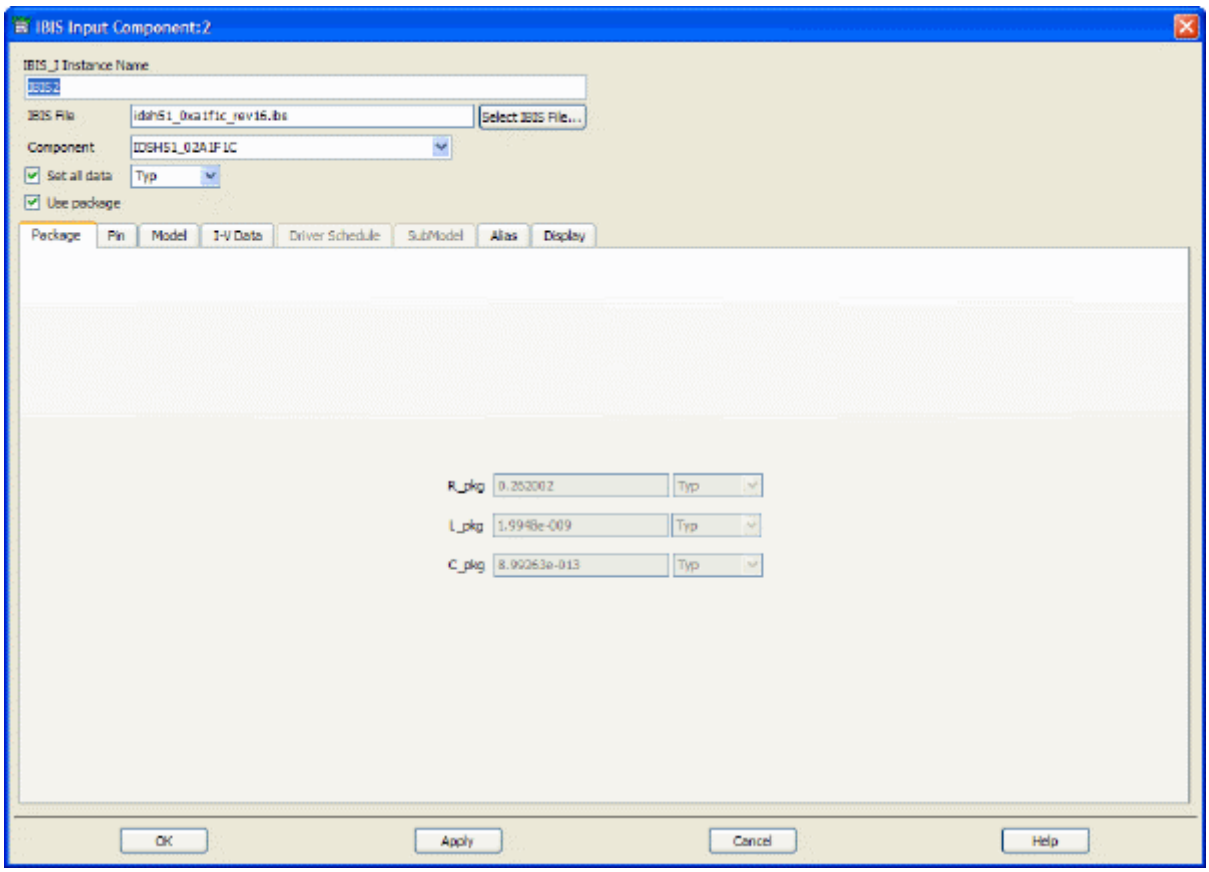
Selecting the "Fast" corner

With the **Set all data** checkbox (On), choose *Fast*. This will automatically select a predefined combination of the values from the IBIS file. The rules for the *Fast* setting are as follows:
(a) the *max* values are selected for all the I-V data (Pullup, Pulldown, Power Clamp a Ground Clamp) and for the waveform data (Ramp, Rising Waveform and Falling Waveform), and
(b) the *min* values are selected for all R, L, C, delay and transit time data.

Selecting the "Slow" corner

With the **Set all data** checkbox (On), choose *Slow*. This will automatically select a predefined combination of the values from the IBIS file. The rules for the *Slow* setting are as follows:
(a) the *min* values are selected for all the I-V data (Pullup, Pulldown, Power Clamp a Ground Clamp) and for the waveform data (Ramp, Rising Waveform and Falling Waveform), and
(b) the *max* values are selected for all R, L, C, delay and transit time data.

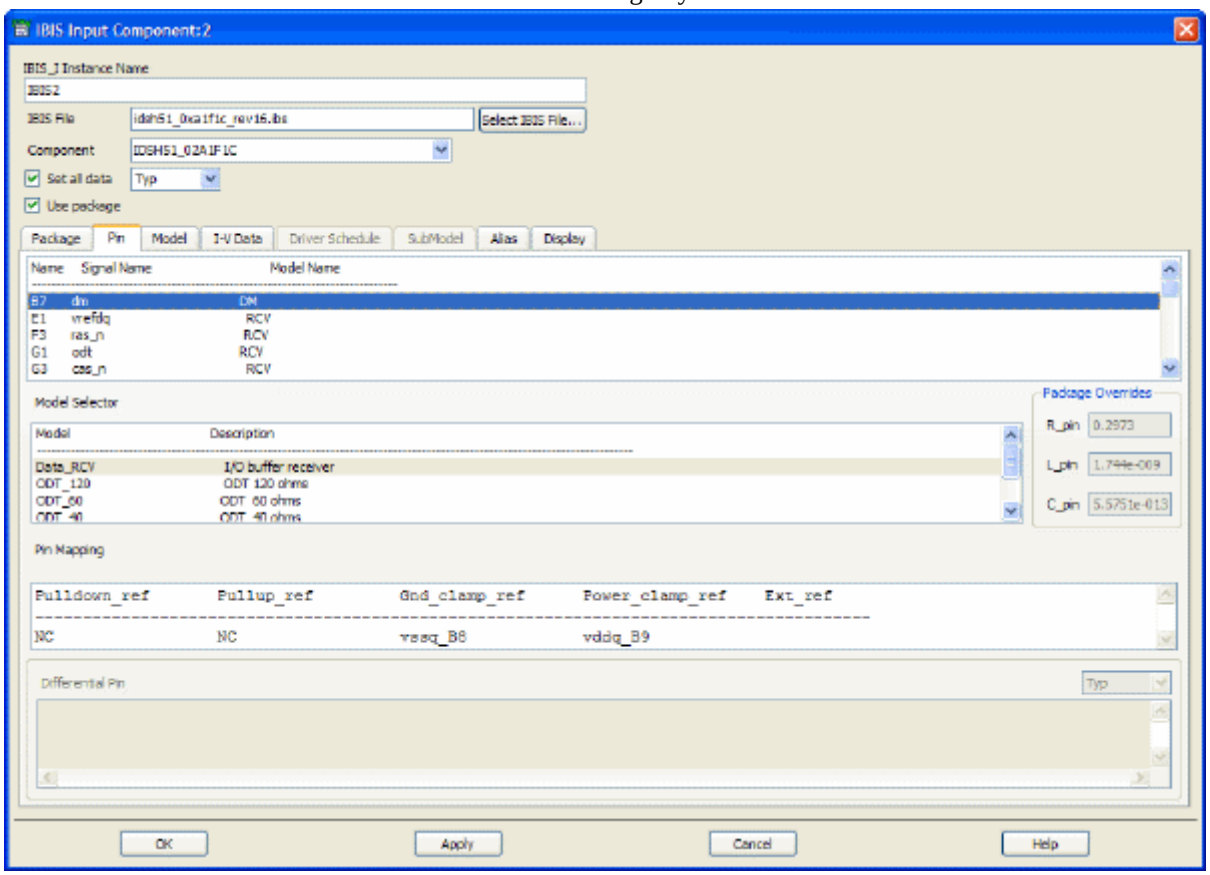
Package Tab



R/L/C_pkg

Data type to be extracted from the IBIS file for R/L/C_pkg under the [Package] keyword if *Set all data* is Off.

Pin Tab



Name

Pin number of an IC, or the non-inverting pin number for a differential buffer. Default value is the first choice.

Model Selector

Current model selected. Disabled and empty if irrelevant for selected pin. Default is first choice.

Package Overrides

Values are informational. If not present in the IBIS file, values are blank. If *Use package* is deselected (Off), these fields are blank and disabled. If present and *Use package* is selected, these values will override those shown on the Package tab.

Pin Mapping

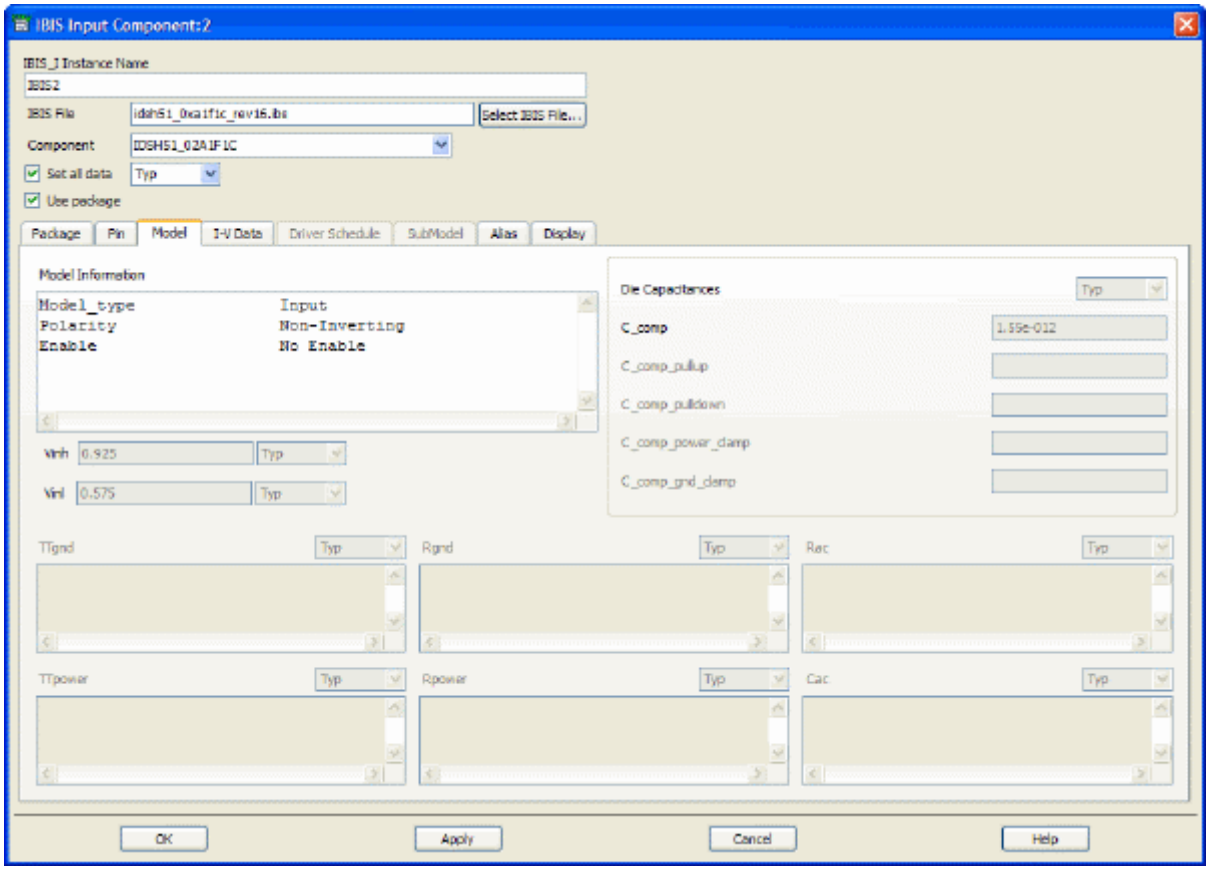
Lists pin references in the IBIS file. Disabled and empty if not included in IBIS file.

Differential Pin

Lists values for the differential pin, if present in the IBIS file. Disabled and empty if not included in IBIS file.

> **Note**
> For information on pin configuration, see *Pin Definitions* (ibis).

Model Tab



Model Information

General information read from the IBIS file.

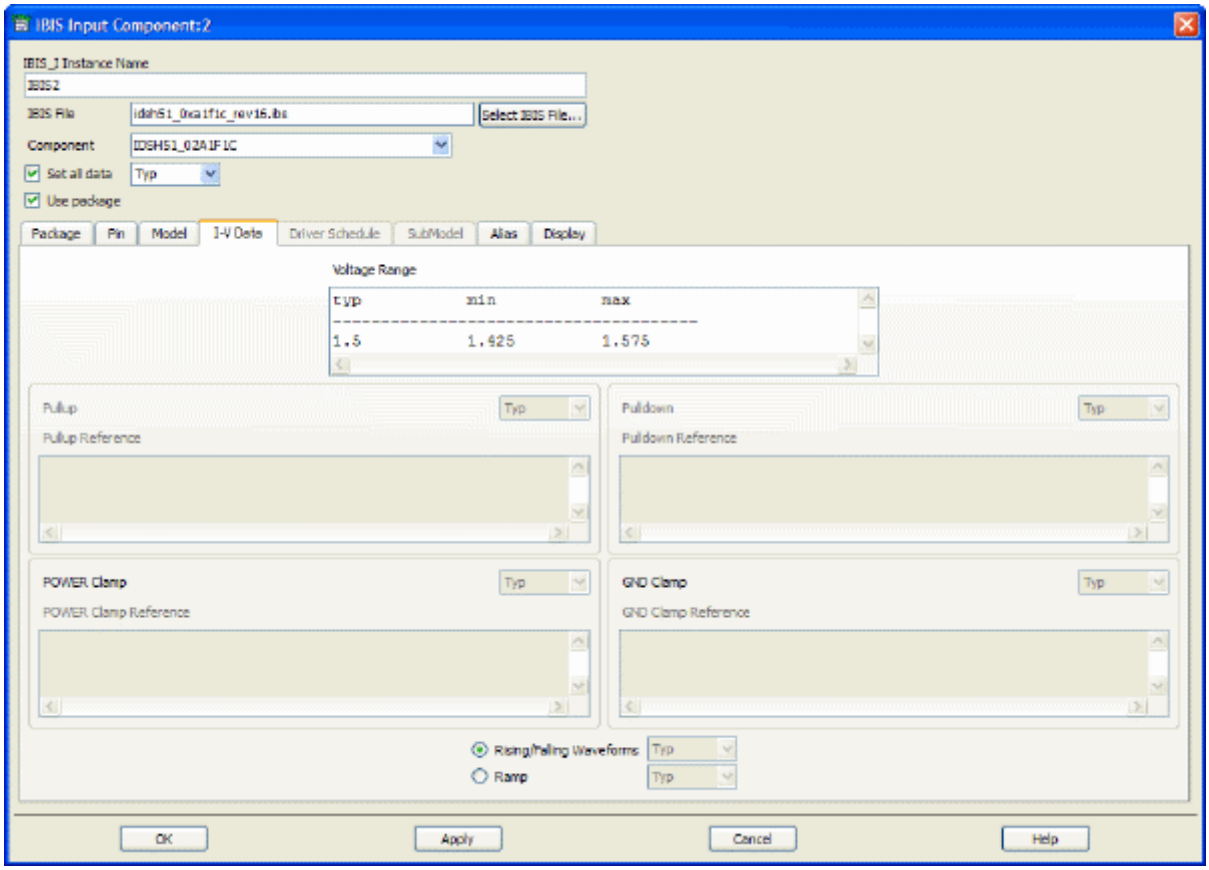
Die Capacitances

These fields are enabled/disabled and populated as appropriate. Typ/Min/Max can be specified if *Set all data* is deselected (Off). Values are informational.

TTgnd, TTpower, Rgnd, Rpower, Rac, Cac

These fields are enabled/disabled and populated as appropriate. Typ/Min/Max can be specified if *Set all data* is deselected (Off). Values are informational.

I-V Data Tab



Voltage Range

Values are informational. Always disabled and empty if not included in IBIS file.

Pullup, Pulldown, POWER Clamp, GND Clamp

Typ/Min/Max can be specified if *Set all data* is deselected (Off).

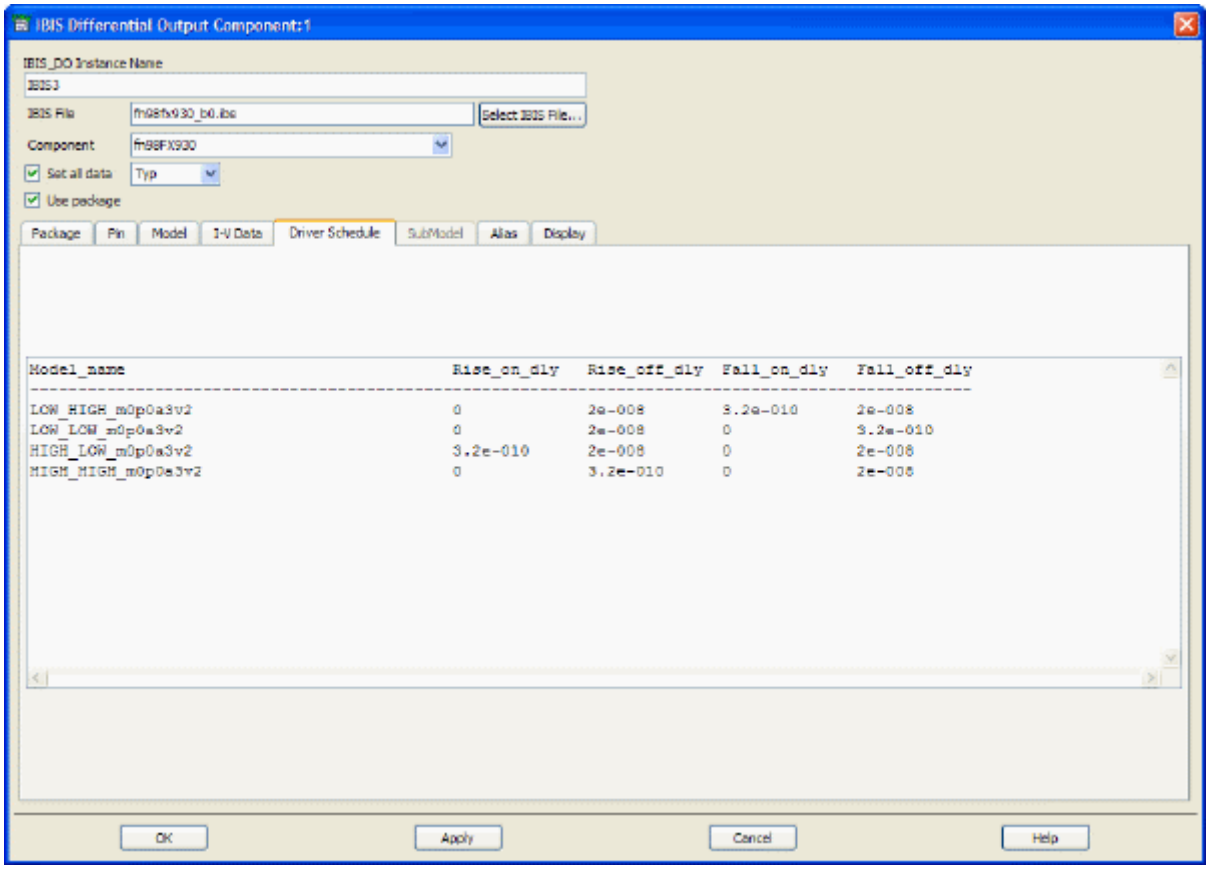
Pullup Ref, Pulldown Ref, POWER Clamp Ref, GND Clamp Ref

Values are informational.

Rising/Falling Waveform, Ramp

Typ/Min/Max can be specified if *Set all data* is deselected (Off). This parameter determines which values will be used during simulation. Rising/Falling waveform data is more accurate than Ramp data.

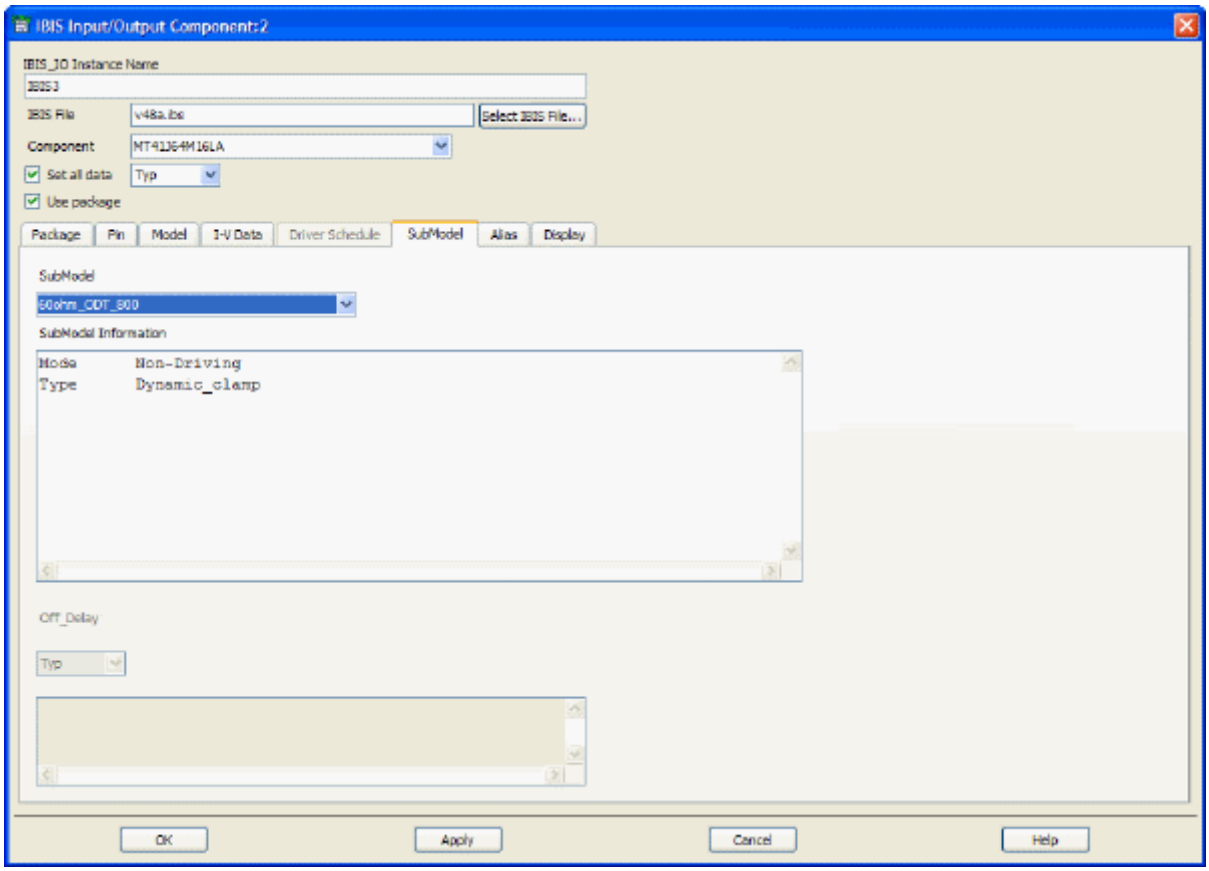
Driver Schedule Tab



Driver Schedule

Entire tab is disabled if irrelevant for the selected component/pin/model. Table content is informational.

SubModel Tab



Entire tab is disabled if irrelevant for the selected component/pin/model.

SubModel

This field allows the selection of any of the submodels specified by [Add Submodel] keywords in the current model.

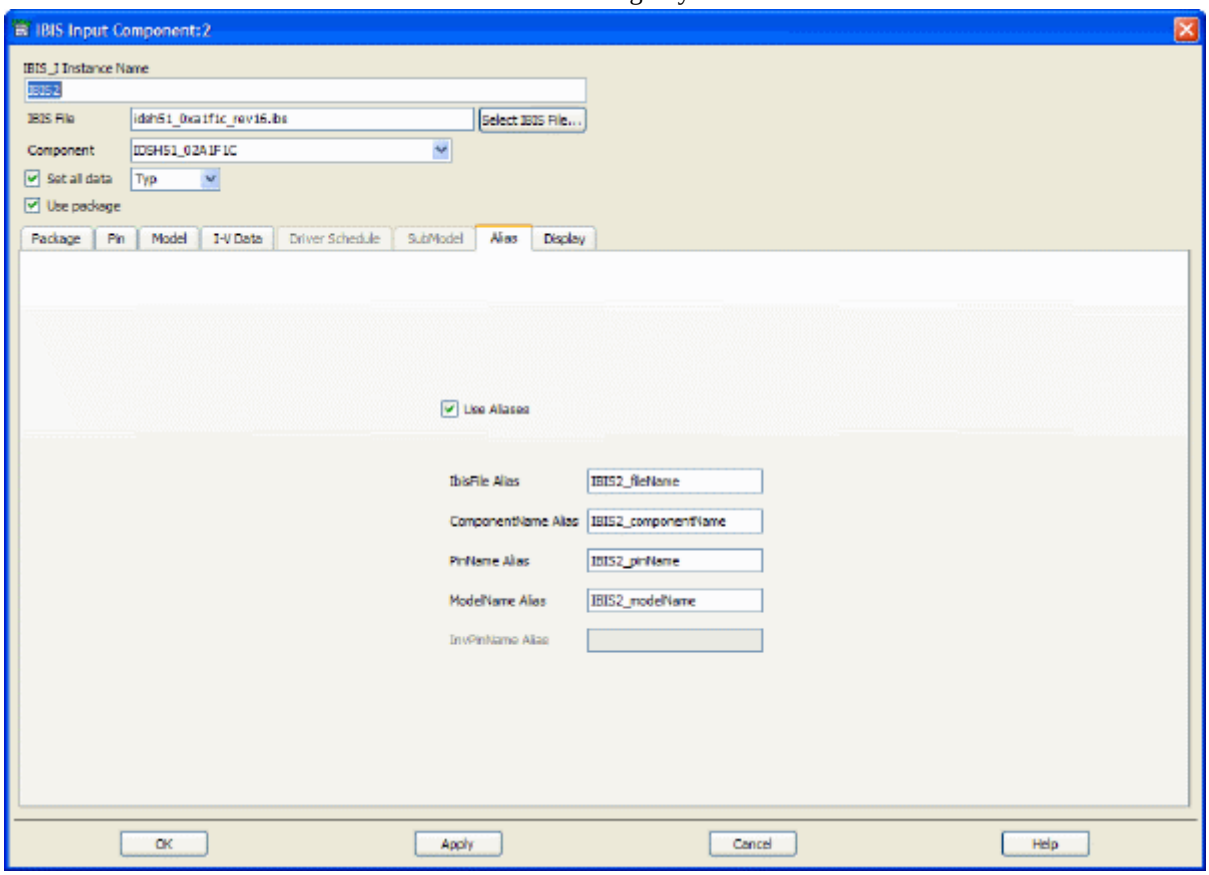
Submodel Information

Values are informational.

Off_Delay

Typ/Min/Max/<*user_label*> can be specified if *Set all data* is deselected (Off). The selection applies to all submodels, if relevant.

Alias Tab



Use the Alias tab to set up advanced sweeps of the IBIS files and models. Such sweeps work only with the *BatchSimController* available in the *Simulation-Batch* palette. It is required that all the aliases are set simultaneously in this tab (except for the *InvPinName Alias* in the case of non-differential buffers), and defined as external variables properly linked to the controller. The batch simulation must be set to run individual simulations in separate processes. For details about using batch simulation to sweep IBIS files, see *Sweeping IBIS Files* (cktsimbatch).

Use Aliases

This checkbox activates/deactivates the fields in the tab. If the buffer is single-ended, all fields become activated except *InvPinName Alias*. Otherwise, all the fields become active. If this box is selected (On), all the available fields must be filled in using user-defined variables.

IbisFile Alias

Provide the variable name for specifying values for the IbisFile parameter. This is a required parameter that must be synchronously swept with the *ModelName* parameter.

ComponentName Alias

Provide the variable name for specifying values for the *ComponentName* parameter. While the *ComponentName* parameter is not required, it is needed to determine the R_pkg, L_pkg and C_pkg values. To encourage a proper setup this alias must be specified if *Use Aliases* is selected (On).

PinName Alias

Provide the variable name for specifying values for the *PinName* parameter. While the *PinName* parameter is not required, it is needed to determine the R_pin, L_pin and C_pin values. To encourage a proper setup this alias must be specified if *Use Aliases* is selected (On).

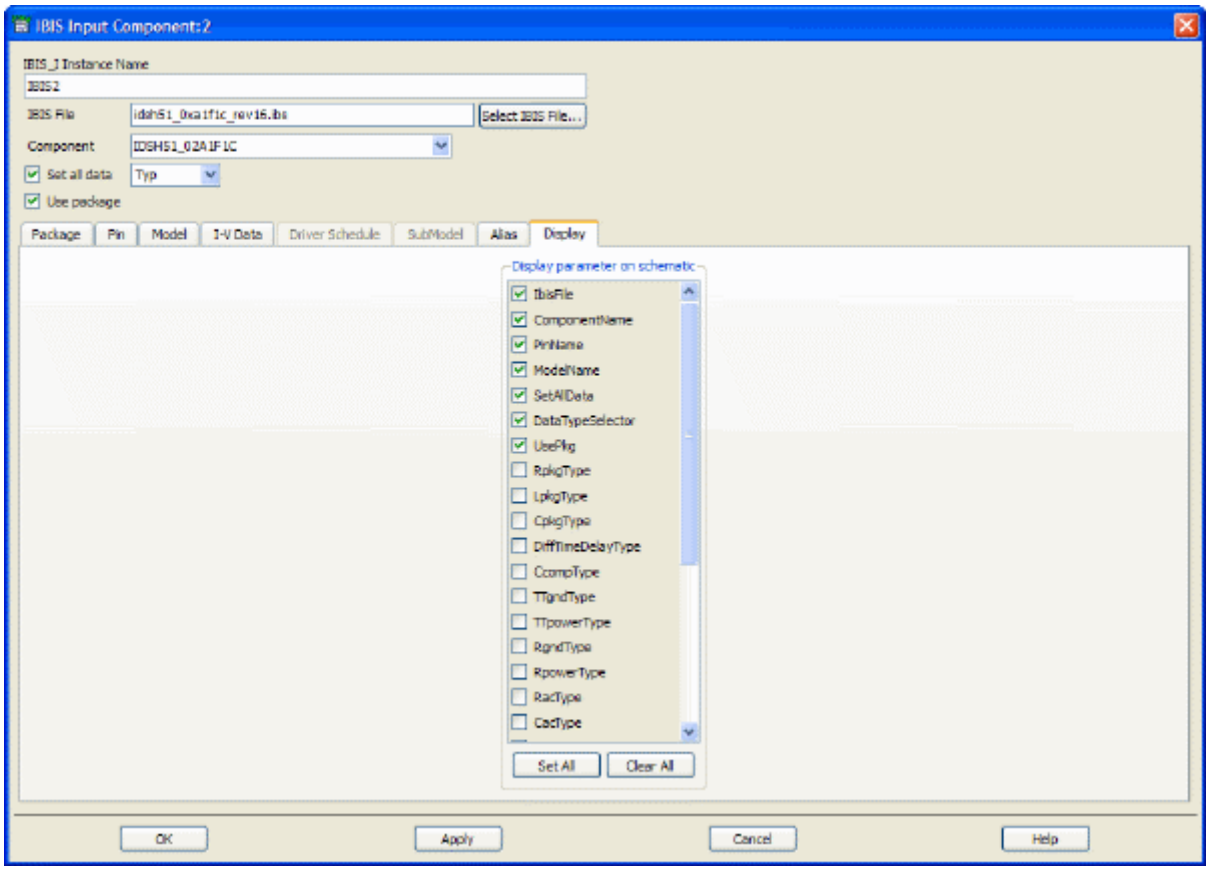
ModelName Alias

Provide the variable name for specifying values for the *ModelName* parameter. *ModelName* is a required parameter and its values (strings) must be synchronously swept with the *IbisFile* parameter.

InvPinName Alias

Provide the variable name for specifying values for the *InvPinName* parameter. This is a required parameter if the buffer is differential. Otherwise, this field is disabled.

Display Tab



Display parameter on schematic

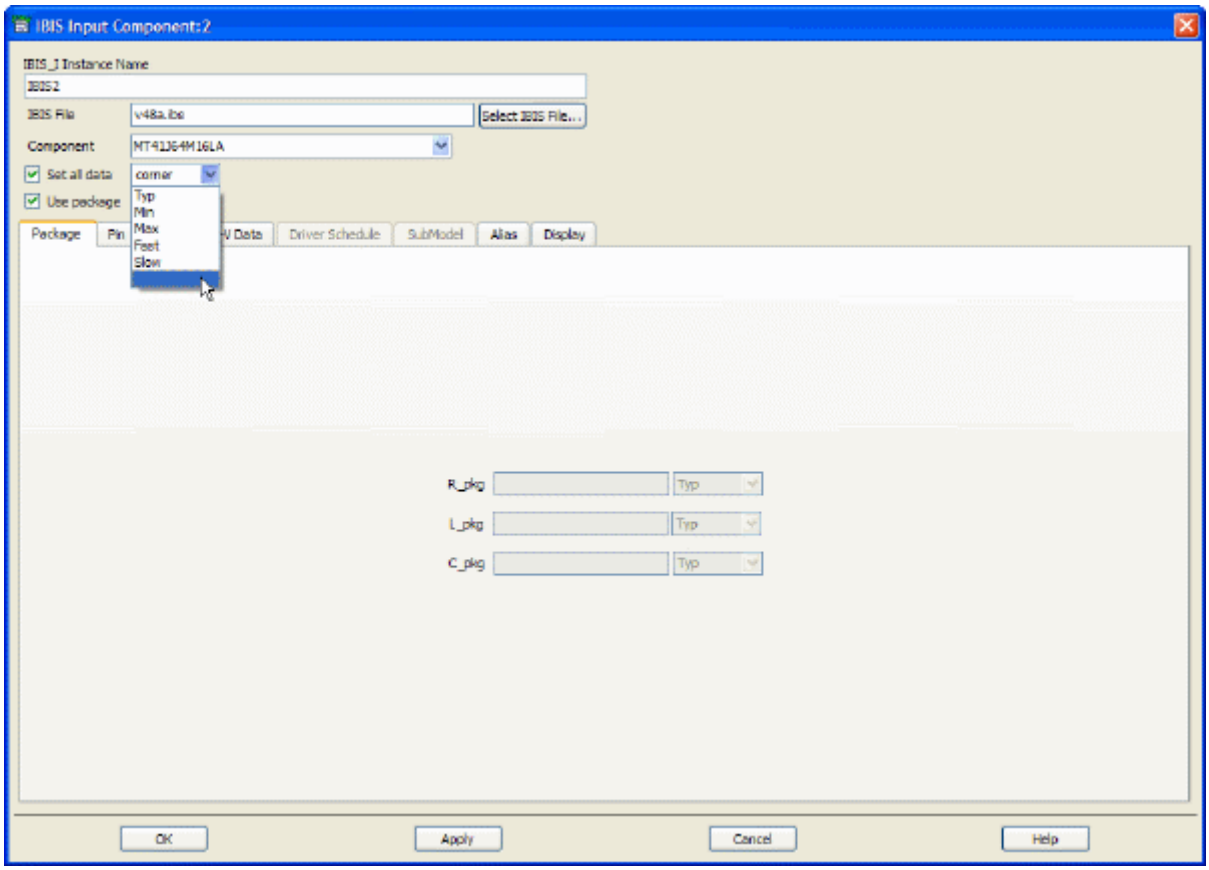
To display parameter values on the schematic, select the associated checkbox.

> **Note**
> The following parameters are not editable from the schematic page: File, Component, Pin, Inverting Pin, and ModelName.

Sweeping IBIS Parameters

In addition to advanced sweeping of IBIS files and models using the Alias Tab, most of the IBIS parameters, such as the corner *Type* parameters, can be swept as any other parameter in ADS. The only parameters that cannot be swept in that fashion are *SetAllData* and *UsePkg*.

Specifying the variables to be passed as the values to the various *Type* parameters of the IBIS component can be done either by on-screen editing or directly in the dialog box. For the latter you must select the last (blank) field in the drop-down list, then type in the variable name, as shown in the following figure. The variable can assume the values of only 1, 2, or 3 for all the fields except *Set all data* (for which the allowed values are 1, 2, 3, 4, or 5. For details, see the table of *Model Parameters* (ibis)).



The advanced sweep of IBIS files and models may involve topology changes and thus must be performed by using the Batch Simulation controller described in *Using Batch Simulation* (cktsimbatch).

IBIS Model Reference

Overview

IBIS Models provides built-in components representing each high-level type of IBIS model.

Most IBIS models have a specific pin configuration for wiring within a circuit schematic. Each model has a component dialog box that displays only those pin/model/[Diff Pin] combinations that match the component's model type.

In addition, this feature provides a generic IBIS component that furnishes a mechanism for selecting any type of supported IBIS model. The generic component has no pins and will not run in the simulator. The generic IBIS model uses the same dialog box as the other IBIS components, but it displays all supported pin/model combinations available in the IBIS file. Once you specify enough information to determine the type of IBIS model and apply the settings, the generic component transforms into the appropriate IBIS component.

Definitions

buffer state	refers to the die (pad) voltage, either low or high. High-Z refers to the high impedance state.
digital output	refers to the voltage at the node DigO (or Digital Output) which can be either 0 or 1 depending on the buffer state and polarity.
disable event	refers to the voltage at the node E (or Enable) and represents a change of the buffer state from enabled to disabled. Enable/disable events take place when the voltage at node E crosses the same trigger thresholds as specified by the TriggerLevel parameter value M. If the parameter Enable is set to Active-High, the enable event occurs when the buffer state is disabled and the voltage at node E becomes greater than the value of M. If the parameter Enable is set to Active-High, the disable event happens when the buffer is enabled and the voltage at node E becomes smaller than the value of (1-M). The opposite applies when the parameter Enable is set to Active-Low. If the enable state is undefined (for example, at the beginning of the simulation) the initial enable state is determined as disabled or enabled depending on whether VE ≤ 0.5 or VE > 0.5, respectively.
enable event	refers to the voltage at the node E and describes a change of the buffer state from disabled to enabled.
enable state	refers to the voltage at the node E. The buffer can be enabled or disabled depending whether the IBIS model parameter Enable is set to Active-High or Active-Low, to interpret the actual voltage for each state.
falling transition	refers to the die voltage going from high to low.
rising transition	refers to the die voltage going from low to high.
trigger event	refers to the voltage at the node T (Trigger or Digital Input) and represents a change of trigger state. The trigger events happen when the voltage at node T crosses the trigger threshold specified by the value of TriggerLevel parameter M. There are two types of trigger events: (1) when the trigger state is low and the voltage at node T becomes greater than the value of M (0.5 ≤ M < 1), and (2) when the trigger state is high and the voltage at node T becomes smaller than the value of (1-M). The following descriptions typically refer to non-inverting buffers. It applies to the inverting buffers by interchanging the trigger states.

Enabling or Disabling a Buffer

One of the ways to enable or disable the buffer is by connecting a voltage source of 1 V or grounding the E pin as shown in figure 1 and figure 2 respectively.

Figure 1

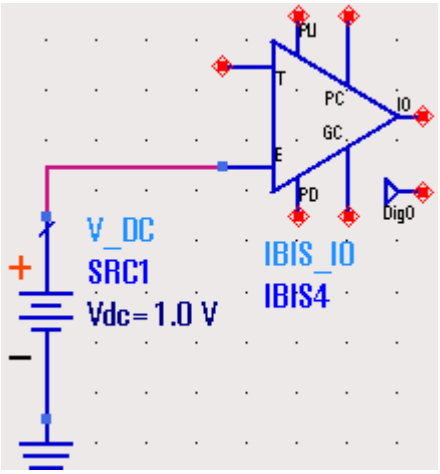
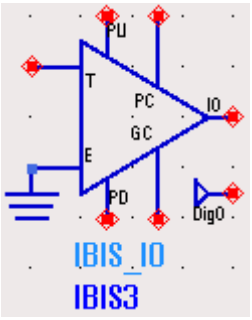


Figure2



The buffer is enabled (acts as an Output buffer, if the Enable subparameter in the IBIS file is Active-High for figure 1, or is Active-Low for figure 2). Otherwise, the buffer is disabled (acts as an Input buffer for “IO”, or a high-Z buffer for “3S”).

Model Parameters

Depending on the model type, IBIS models will have some or all of the following parameters:

Name	Description	Values	Default	Notes
IbisFile	IBIS file name			User selectable Required
ComponentName	IC identifier			User selectable Not required
PinName	Pin number of an IC, or the non-inverting pin number for a differential buffer			User selectable Not required In case when the selected pin is one of the pins in the keyword [Diff Pin] this is set to the first (non-inverting) pin.
InvPinName	Inverting pin number for a differential buffer			Required for all differential buffers. Automatically set if the selected [Pin] is one of the pins in [Diff Pin].

ModelName	IBIS file model name			Required Follows ComponentName and PinName selections. Redundant if both are specified and [Model Selector] is not used.
SetAllData	Flag to use Data type set by the DataTypeSelector for all data and ignore individual parameter settings	yes, no	yes	User selectable Not required
DataTypeSelector	A global setting of data type to be extracted from the IBIS file. For information on fast and slow setting, refer to *Selecting the Fast corner* (ibis) and *Selecting the Slow corner* (ibis) respectively	1 - typ 2 - min 3- max 4 - fast 5 -slow	1	User selectable
UsePkg	Flag to ignore the package description in the IBIS file (as set by R/L/C_pkg or R/L/C_pin) and exclude the package components from the equivalent circuit.	yes, no	yes	User selectable Not required
RpkgType	Data type to be extracted from the IBIS file for R_pkg under the [Package] keyword.	1 - typ 2 - min 3 - max	1	User selectable Not required Ignored if SetAllData=yes or if UsePkg=no Ignored if R_pin is specified in the IBIS file under the [Pin] keyword for the selected PinName (the value of R_pin overrides the value of R_pkg) To avoid simulation errors, all values listed in the IBIS file for typ, min, and max must be real numbers.
LpkgType	Data type to be extracted from the IBIS file for L_pkg under the [Package] keyword.	1 - typ 2 - min 3 - max	1	User selectable Not required Ignored if SetAllData=yes or if UsePkg=no Ignored if L_pin is specified in the IBIS file under the [Pin] keyword for the selected PinName (the value of L_pin overrides the value of L_pkg) To avoid simulation errors, all values listed in the IBIS file for typ, min, and max must be real numbers.
CpkgType	Data type to be extracted from the IBIS file for C_pkg under the [Package] keyword.	1 - typ 2 - min 3 - max	1	User selectable Not required Ignored if SetAllData=yes or if UsePkg=no Ignored if C_pin is specified in the IBIS file under the [Pin] keyword for the selected PinName (the value of C_pin overrides the value of C_pkg)

				To avoid simulation errors, all values listed in the IBIS file for typ, min, and max must be real numbers.
DiffTimeDelayType	Data type to be extracted from the IBIS file as the time delay between the inverting and non-inverting pins for differential buffers, specifically the launch delays of the non-inverting pins relative to the inverting pins.	1 - tdelay_typ 2 - tdelay_min 3 - tdelay_max	1	User selectable Not required Ignored if SetAllData=yes
CcompType	Data type to be extracted from the IBIS file for the die capacitance C_comp, or C_comp_* if they are specified and used.	1 - typ 2 - min 3 - max	1	User selectable Not required Ignored if SetAllData=yes
TTgndType	Data type to be extracted from the IBIS file for the transit time for the ground clamp diffusion capacitance.	1 - typ 2 - min 3 - max	1	User selectable Not required Ignored if SetAllData=yes
TTpowerType	Data type to be extracted from the IBIS file for the transit time for the power clamp diffusion capacitance.	1 - typ 2 - min 3 - max	1	User selectable Not required Ignored if SetAllData=yes
RgndType	Data type to be extracted from the IBIS file for the parasitic resistance Rgnd in a terminator buffer.	1 - typ 2 - min 3 - max	1	User selectable Not required Ignored if SetAllData=yes
RpowerType	Data type to be extracted from the IBIS file for the parasitic resistance Rpower in a terminator buffer.	1 - typ	1	User selectable Not required Ignored if SetAllData=yes
RacType	Data type to be extracted from the IBIS file for the AC terminator resistance Rac.	1 - typ 2 - min 3 - max	1	User selectable Not required Ignored if SetAllData=yes
CacType	Data type to be extracted from the IBIS file for the AC terminator capacitance Cac.	1 - typ 2 - min 3 - max	1	User selectable Not required Ignored if SetAllData=yes
PuDataType	Data type to be extracted from the IBIS file as I-V data for the pullup device.	1 - typ 2 - min 3 - max	1	User selectable Not required Ignored if SetAllData=yes
PdDataType	Data type to be extracted from the IBIS file as I-V data for the pulldown device.	1 - typ 2 - min 3 - max	1	User selectable Not required Ignored if SetAllData=yes
PcDataType	Data type to be extracted from the IBIS file as I-V data for the power clamp.	1 - typ 2 - min 3 - max	1	User selectable Not required Ignored if SetAllData=yes
GcDataType	Data type to be extracted from the IBIS file as I-V data for the ground clamp.	1 - typ 2 - min 3 - max	1	User selectable Not required Ignored if SetAllData=yes
WaveformType	Data type to be extracted from the IBIS file for the rising and falling waveforms.	1 - typ 2 - min 3 - max	1	User selectable Not required Ignored if SetAllData=yes
RampType	Ramp data type to be extracted from the IBIS file for the rising and falling transitions.	1 - typ 2 - min 3 - max	1	User selectable Not required Ignored if SetAllData=yes
IgnoreWaveforms	Flag to use the [Ramp] data even if the waveform tables are provided in	yes, no	no	User selectable Not required

	the IBIS file.			
Polarity	Flag to override the IBIS file setting for the [Polarity] keyword.	0 - non-inverting	0	User selectable Not required Must not be netlisted if the IBIS file setting is to be used. The default of "non-inverting" is used if the model polarity is not specified in the IBIS file.
TriggerLevel	The voltage level at the input node T (Digital In) of a non-inverting output buffer triggering the rising transition from a "low" state to "high" state. The one-complement is used for the opposite event.	range: [0.5, 1.0)	0.5	Not required
Off_DelayType	Data type to be extracted from the IBIS file for the sub-parameter turn-off delay from V_trigger_r or V_trigger_f (if specified) for all Submodels of the current model.	1 - typ 2 - min 3 - max	1	User selectable Not required Ignored if SetAllData=yes
VinlType	Data type to be extracted from the IBIS file for the threshold voltage Vinl as provided under the [Model Spec] keyword, or directly for the model.	1 - typ 2 - min 3 - max	1	User selectable Not required Ignored if SetAllData=yes
VinhType	Data type to be extracted from the IBIS file for the threshold voltage Vinh as provided under the [Model Spec] keyword, or directly for the model.	1 - typ 2 - min 3 - max	1	User selectable Not required Ignored if SetAllData=yes
InterpMode	Interpolation mode.	Linear Spline Cubic	Spline	User selectable Not required

Pin Definitions (used in the component symbols)

Pin Name	Description
T	Trigger – voltage between this pin and the ground controls the buffer state.
E	Enable – voltage between this pin and the ground controls whether the buffer is enabled or disabled.
IO	The input pin of a single-ended Input, or Terminator, or disabled I/O (including open-source and open-sink) or tri-state buffer. The output pin of a single-ended Output, or enabled I/O or tri-state buffer (including open-source and open-sink buffers).
IO_NI	The non-inverting input pin of a differential Input, Terminator, or disabled I/O (including open-source and open-sink) or tri-state buffer. The non-inverting output pin of a differential Output, or enabled I/O or tri-state buffer (including open-source and open-sink buffers).
IO_I	The inverting input pin of a differential Input, Terminator, or disabled I/O (including open-source and open-sink) or tri-state buffer. The inverting output pin of a differential Output, or enabled I/O or tri-state buffer (including open-source and open-sink buffers).
DigO	Digital output pin of an Input, or I/O buffer (including open-source and open-sink buffers).
PU	Power supply pin for the pullup (PU) device. Not available if the pullup device is not present. Also used for ECL buffers as the common pullup/pulldown power supply pin.
PD	Power supply pin for the pulldown (PD) device. Not available if the pulldown device is not present.
PC	Power supply pin for the power clamp (PC).
GC	Power supply pin for the ground clamp (GC).

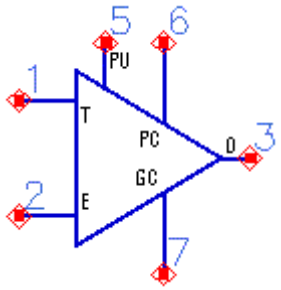
Components

- *IBIS (Generic Model)* (ibis)
- *IBIS_3S_ECL (3-State_ECL)* (ibis)
- *IBIS_3S (3-State)* (ibis)
- *IBIS_D3S_ECL (Differential 3-State_ECL)* (ibis)
- *IBIS_D3S (Differential 3-State)* (ibis)
- *IBIS_DI_ECL (Differential Input_ECL)* (ibis)
- *IBIS_DI (Differential Input)* (ibis)
- *IBIS_DIO_ECL (Differential Input-Output_ECL)* (ibis)
- *IBIS_DIO (Differential Input-Output)* (ibis)
- *IBIS_DIO_OPENSINK (Differential IO Open Sink)* (ibis)
- *IBIS_DIO_OPENSOURCE (Differential IO Open Source)* (ibis)
- *IBIS_DO_ECL (Differential Output_ECL)* (ibis)
- *IBIS_DO (Differential Output)* (ibis)
- *IBIS_DOPENSINK (Differential Open Sink)* (ibis)
- *IBIS_DOPENSOURCE (Differential Open Source)* (ibis)
- *IBIS_DT (Differential Terminator)* (ibis)
- *IBIS_I_ECL (Input_ECL)* (ibis)
- *IBIS_I (Input)* (ibis)
- *IBIS_IO_ECL (Input-Output_ECL)* (ibis)
- *IBIS_IO (Input-Output)* (ibis)
- *IBIS_IO_OPENSINK (IO Open Sink)* (ibis)
- *IBIS_IO_OPENSOURCE (IO Open Source)* (ibis)
- *IBIS_O_ECL (Output_ECL)* (ibis)
- *IBIS_O (Output)* (ibis)

- *IBIS_OPENSINK (Open Sink)* (ibis)
- *IBIS_OPENSOURCE (Open Source)* (ibis)
- *IBIS_T (Terminator)* (ibis)

IBIS_3S_ECL (3-State_ECL)

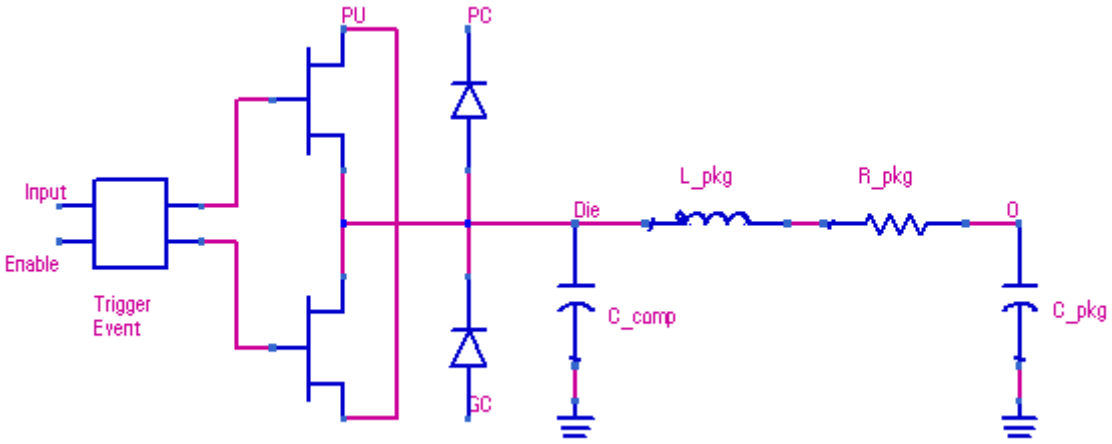
Symbol



Parameters

See *Model Parameters* (ibis).

Equivalent Circuit



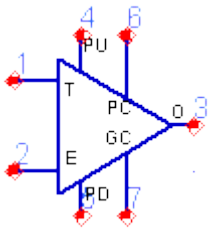
Notes/Equations

1. This buffer behaves like *IBIS_O_ECL (Output_ECL)* (ibis) if the buffer is enabled.
2. If the buffer is disabled the buffer state becomes high-Z.

3. The transitions between the enabled and disabled states follow the enable/disable events.
4. Enable = ActiveHigh
   Time = 0
   Disable State if VE < 0.5
   Enable State if VE ≥ 0.5
   Time > 0
   Disabling trigger if VE < 1 - TriggerLevel
   Enabling trigger if VE ≥ TriggerLevel
5. Enable = ActiveLow
   Time = 0
   Enable State if VE ≤ 0.5
   Disable State if VE > 0.5
   Time > 0
   Enabling trigger if VE ≤ 1 - TriggerLevel
   Disabling trigger if VE > TriggerLevel

IBIS_3S (3-State)

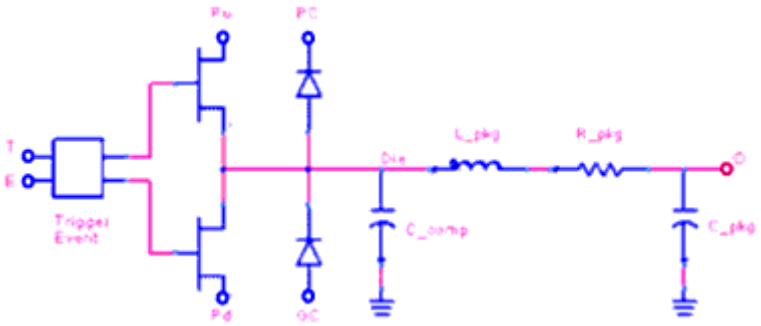
Symbol



Parameters

See *Model Parameters* (ibis).

Equivalent Circuit



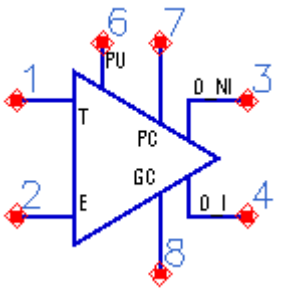
Notes/Equations

1. This buffer behaves like *IBIS_O (Output)* (ibis) if the buffer is enabled.

1. If the buffer is disabled the buffer state becomes high-Z.
2. The transitions between the enabled and disabled states follow the enable/disable events.
3. Enable = ActiveHigh
   Time = 0
   Disable State if VE < 0.5
   Enable State if VE ≥ 0.5
   Time > 0
   Disabling trigger if VE < 1 - TriggerLevel
   Enabling trigger if VE ≥ TriggerLevel
4. Enable = ActiveLow
   Time = 0
   Enable State if VE ≤ 0.5
   Disable State if VE > 0.5
   Time > 0
   Enabling trigger if VE ≤ 1 - TriggerLevel
   Disabling trigger if VE > TriggerLevel

IBIS_D3S_ECL (Differential 3-State_ECL)

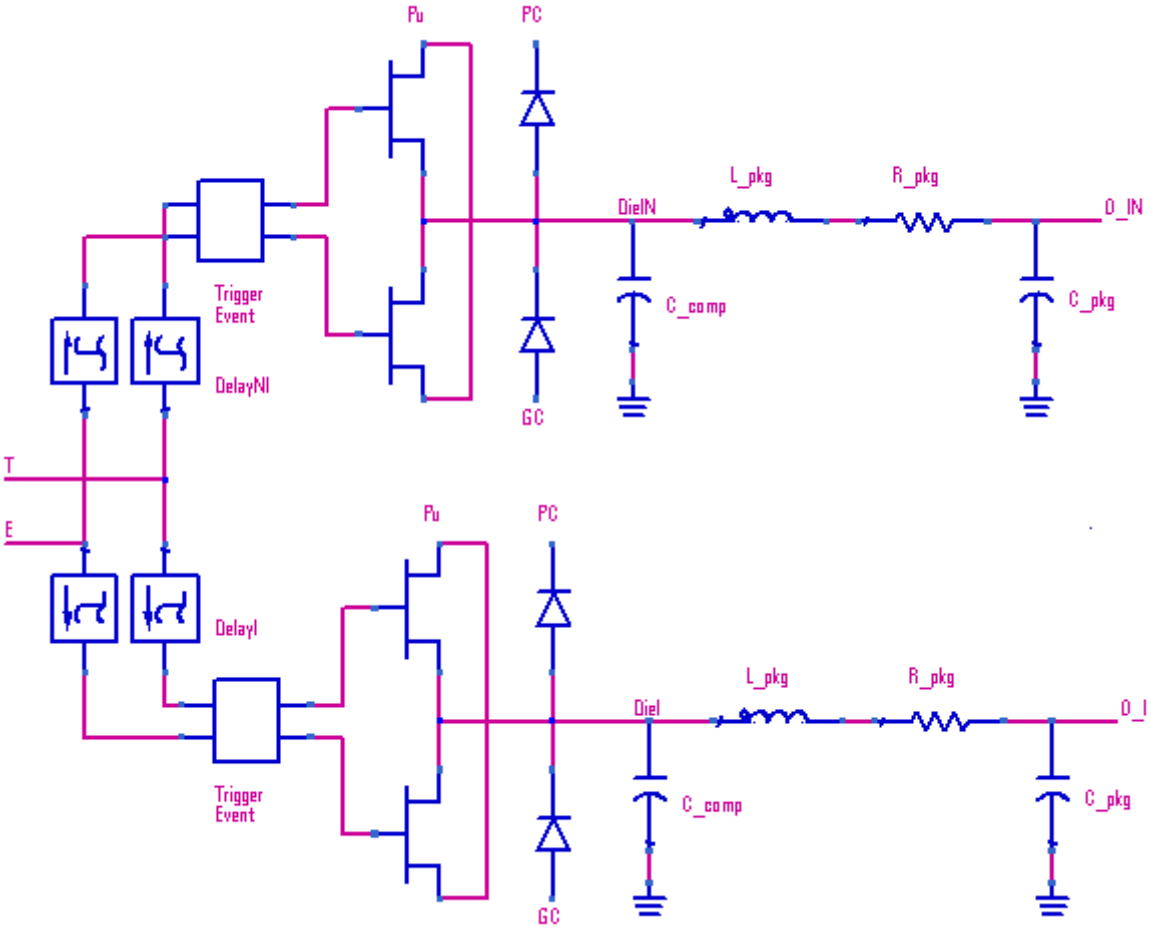
Symbol



Parameters

See *Model Parameters* (ibis).

Equivalent Circuit

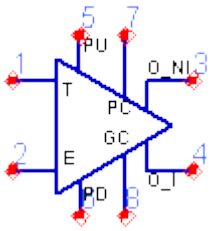


Notes/Equations

1. This buffer consists of two *IBIS_3S_ECL (3-State_ECL)* (ibis) buffers, one non-inverting and one inverting.
2. The trigger event can be delayed for one of the buffers.
3. This delay is controlled by the value of the subparameter tdelay_typ, tdelay_min, tdelay_max (whichever is selected) under the keyword [Diff Pin] in the IBIS file.
4. Which buffer gets a delayed trigger event depends on the sign of tdelay:
   If tdelay ≥ 0
   DelayNI = tdelay DelayI = 0
   If tdelay < 0
   DelayNI = 0 DelayI = -tdelay

IBIS_D3S (Differential 3-State)

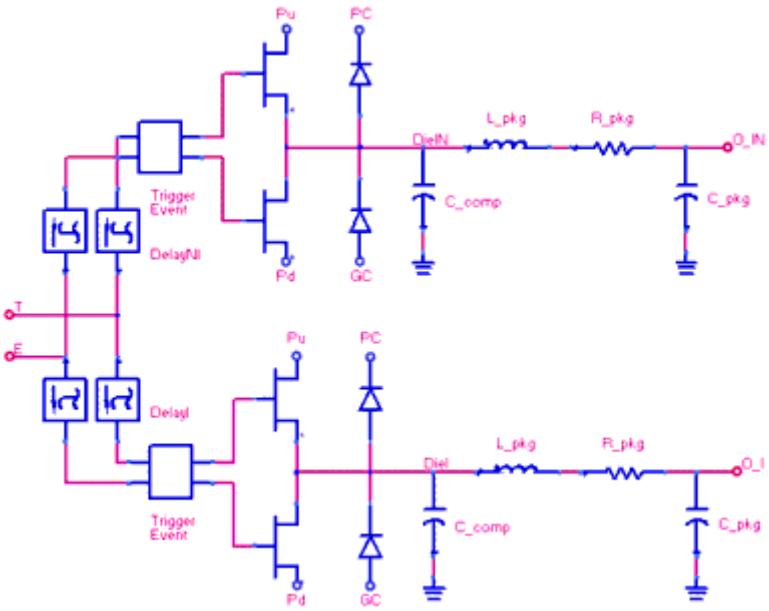
Symbol



Parameters

See *Model Parameters* (ibis).

Equivalent Circuit



Notes/Equations

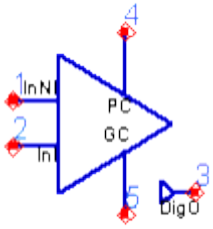
1. This buffer consists of two *IBIS_3S (3-State)* (ibis) buffers, one non-inverting and one inverting.
2. The trigger event can be delayed for one of the buffers.
3. This delay is controlled by the value of the subparameter tdelay_typ, tdelay_min,

tdelay_max (whichever is selected) under the keyword [Diff Pin] in the IBIS file.

4. Which buffer gets a delayed trigger event depends on the sign of tdelay:
   If tdelay ≥ 0
   DelayNI = tdelay DelayI = 0
   If tdelay < 0
   DelayNI = 0 DelayI = -tdelay

IBIS_DI_ECL (Differential Input_ECL)

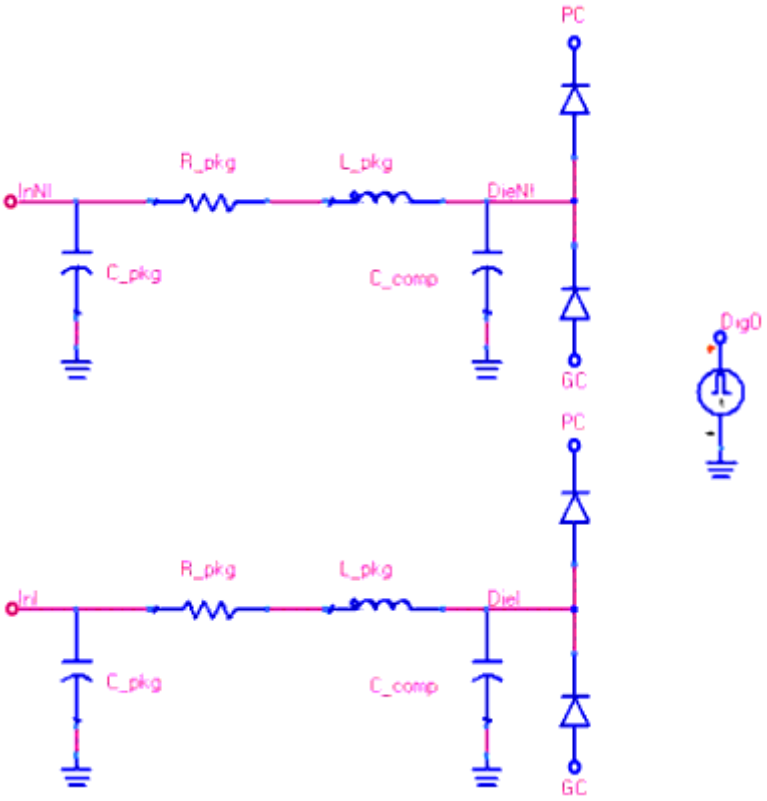
Symbol



Parameters

See *Model Parameters* (ibis).

Equivalent Circuit



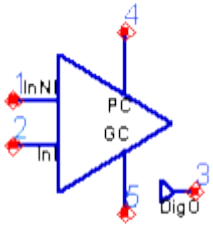
Notes/Equations

1. This buffer consists of two *IBIS_I_ECL (Input_ECL)* (ibis) buffers.
2. It can function as a driver: it supports digital output which can assume the values of 0 or 1 depending on the voltage difference between the nodes InNI and InI as compared to the IBIS model parameter vdiff specified under the [Diff Pin] keyword.
3. There are no separate thresholds and the absolute value of vdiff is used, as follows:
   digital output = 0 if $V_{InNI} - V_{InI} < vdiff$

   digital output = 1 if $V_{InNI} - V_{InI} > vdiff$
4. The only differnce between input_ECL and input buffer is the default value of Vinh and Vinl.

IBIS_DI (Differential Input)

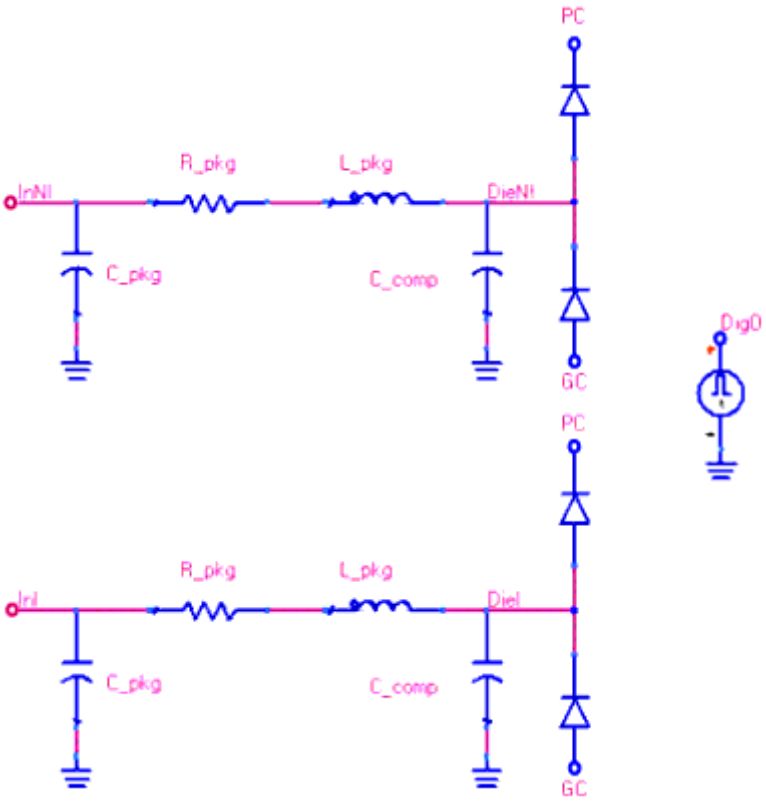
Symbol



Parameters

See *Model Parameters* (ibis).

Equivalent Circuit



Notes/Equations

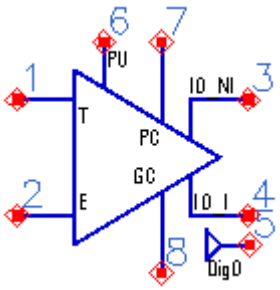
1. This buffer consists of two *IBIS_I (Input)* (ibis) buffers.
2. It can function as a driver: it supports digital output which can assume the values of 0 or 1 depending on the voltage difference between the nodes InNI and InI as compared to the IBIS model parameter vdiff specified under the [Diff Pin] keyword.
3. There are no separate thresholds and the absolute value of vdiff is used, as follows:
   digital output = 0 if $V_{InNI} - V_{InI} < vdiff$

   digital output = 1 if $V - V > vdiff$

InNI InI

IBIS_DIO_ECL (Differential Input/Output_ECL)

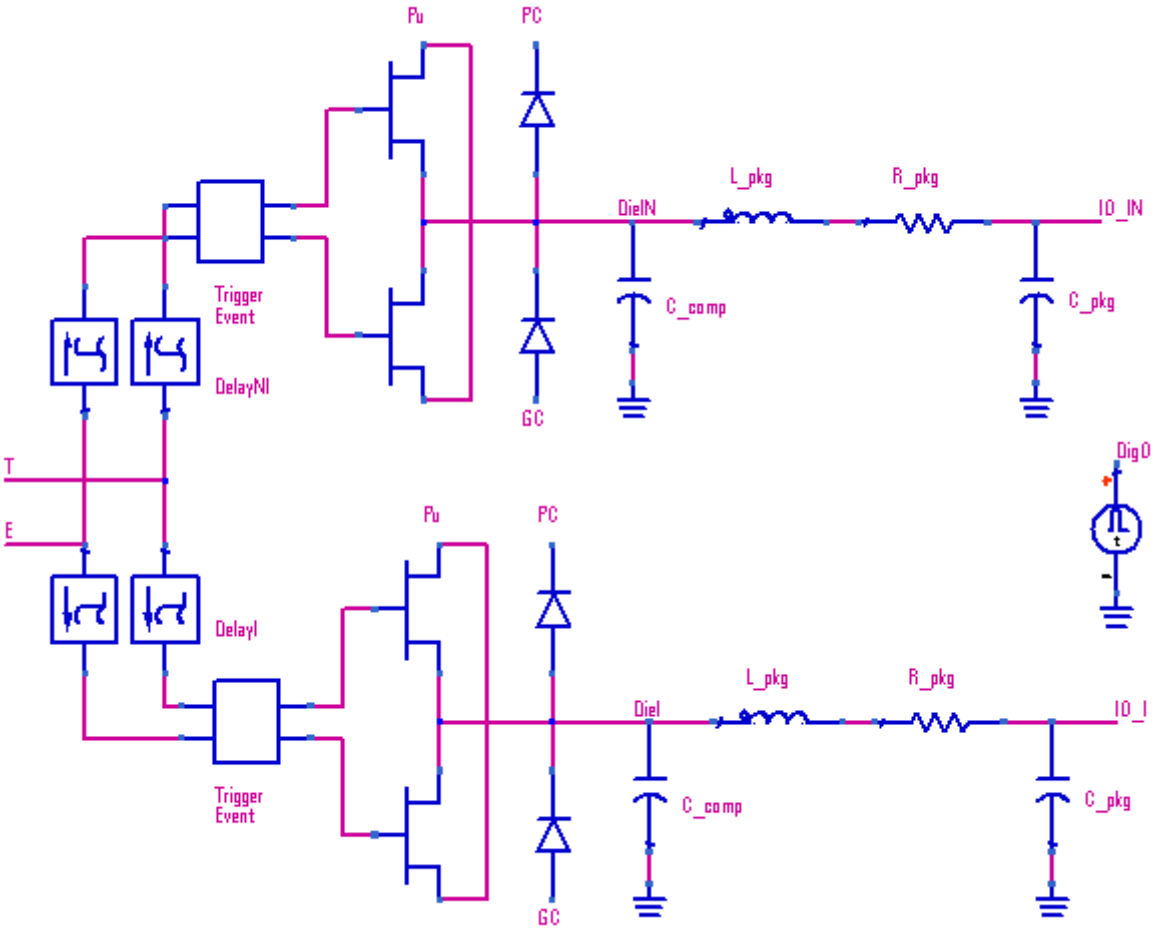
Symbol



Parameters

See *Model Parameters* (ibis).

Equivalent Circuit

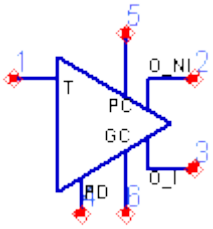


Notes/Equations

1. This buffer consists of two *IBIS_IO_ECL (Input/Output_ECL)* (ibis) buffers, one non-inverting and one inverting.
2. Depending on the enable state, this model acts as a *IBIS_DO_ECL (Differential Output_ECL)* (ibis) buffer or a *IBIS_DI_ECL (Differential Input_ECL)* (ibis) buffer.

IBIS_DIO_OPENSINK (Differential I/O Open Sink)

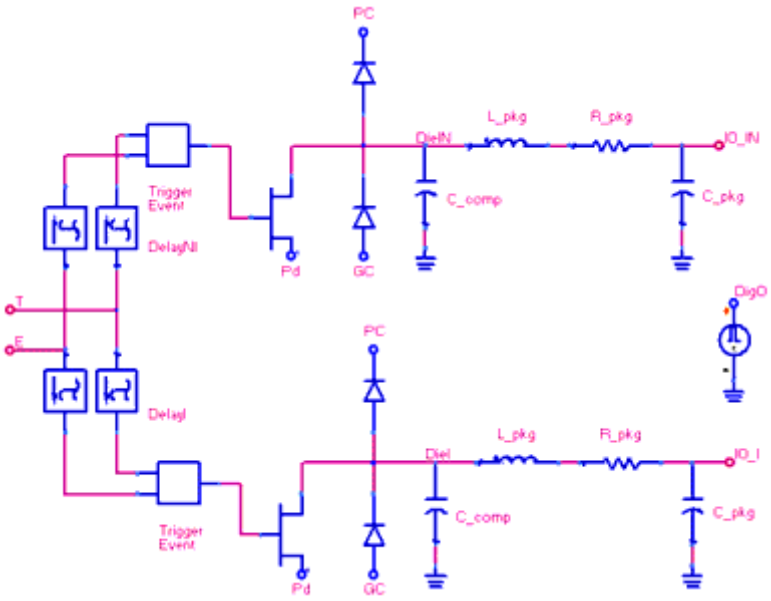
Symbol



Parameters

See *Model Parameters* (ibis).

Equivalent Circuit

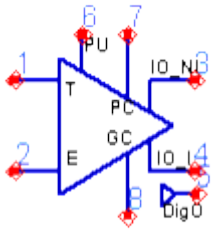


Notes/Equations

1. This buffer consists of two *IBIS_IO_OPENSINK (I/O Open Sink)* (ibis) buffers, one non-inverting and one inverting.
2. The functionality follows that of the *IBIS_DOPENSINK (Differential Open Sink)* (ibis) buffer or the *IBIS_DI (Differential Input)* (ibis) buffer depending on the enable state.

IBIS_DIO_OPENSOURCE (Differential I/O Open Source)

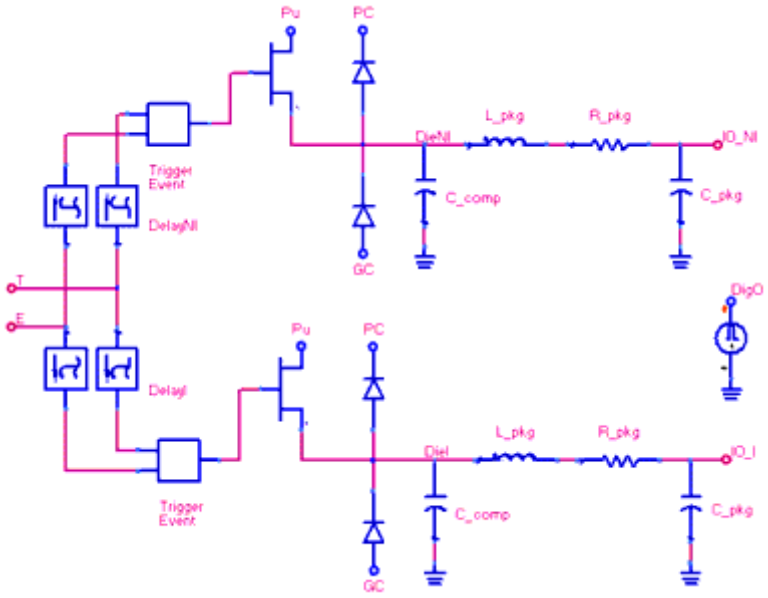
Symbol



Parameters

See *Model Parameters* (ibis).

Equivalent Circuit

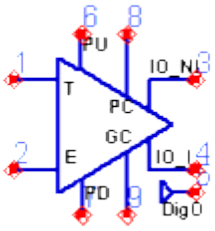


Notes/Equations

1. This buffer consists of two *IBIS_IO_OPENSOURCE (I/O Open Source)* (ibis) buffers, one non-inverting and one inverting.
2. The functionality follows that of the *IBIS_DOPENSOURCE (Differential Open Source)* (ibis) buffer or the *IBIS_DI (Differential Input)* (ibis) buffer depending on the enable state.

IBIS_DIO (Differential Input/Output)

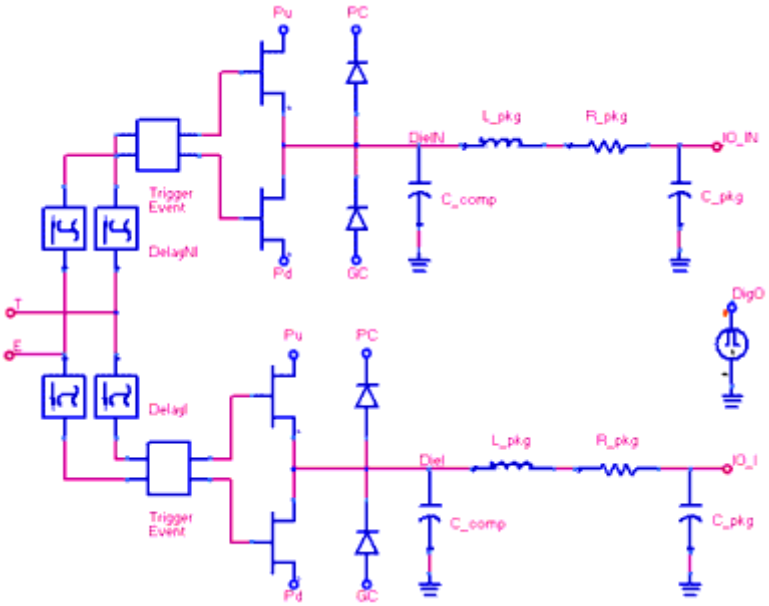
Symbol



Parameters

See *Model Parameters* (ibis).

Equivalent Circuit

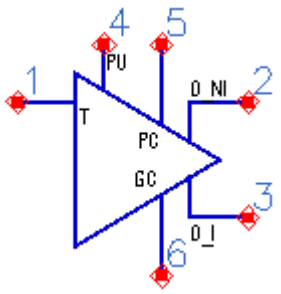


Notes/Equations

1. This buffer consists of two *IBIS_IO (Input/Output)* (ibis) buffers, one non-inverting and one inverting.
2. Depending on the enable state, this model acts as a *IBIS_DO (Differential Output)* (ibis) buffer or a *IBIS_DI (Differential Input)* (ibis) buffer. For more information, refer to *Enabling/Disabling a Buffer* (ibis).

IBIS_DO_ECL (Differential Output_ECL)

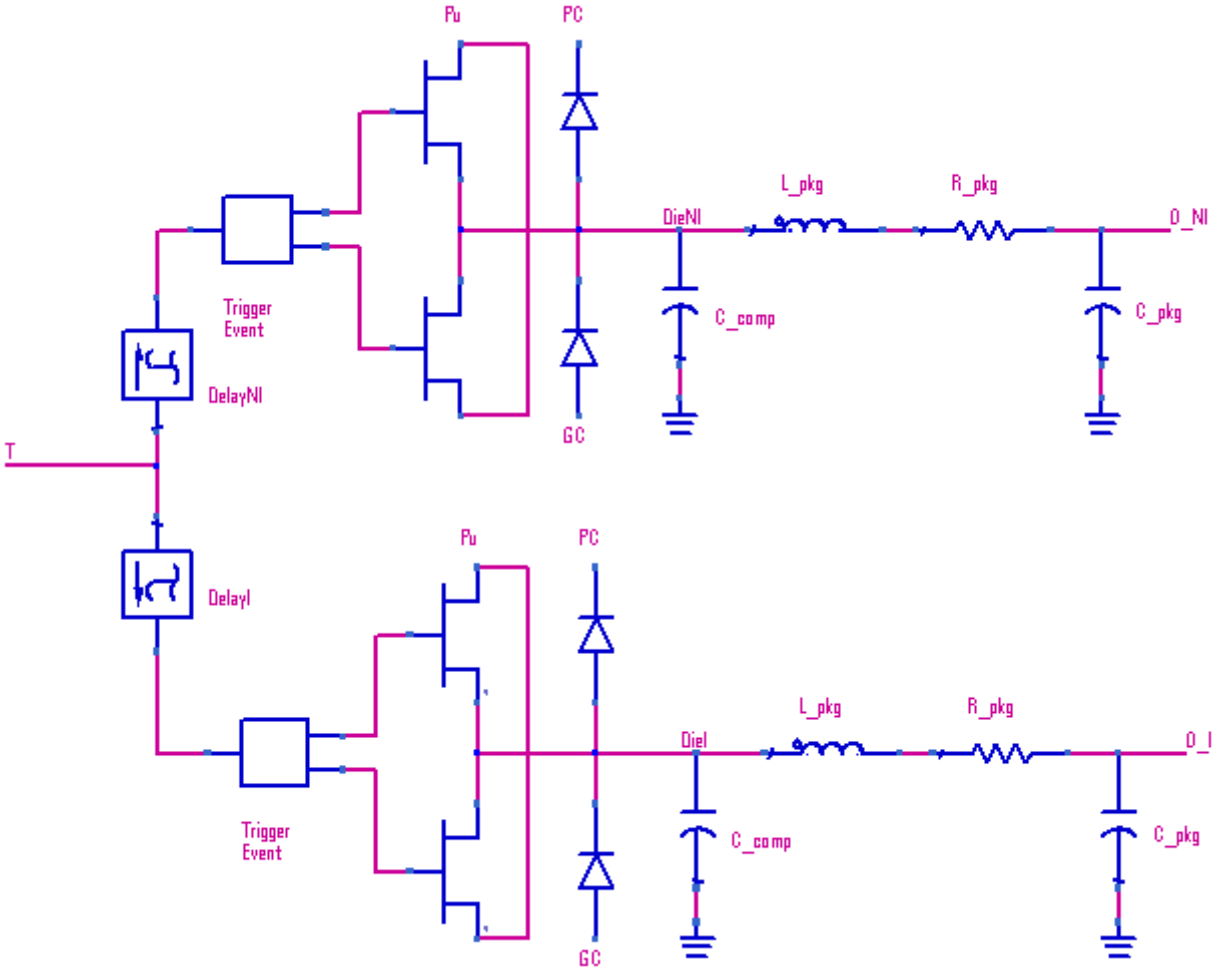
Symbol



Parameters

See *Model Parameters* (ibis).

Equivalent Circuit

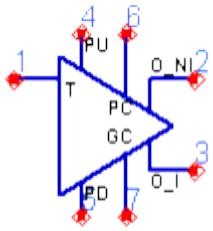


Notes/Equations

1. This buffer consists of two *IBIS_O_ECL (Output_ECL)* (ibis) buffers, one non-inverting and one inverting.
2. The trigger event can be delayed for one of the buffers.
3. This delay is controlled by the value of the subparameter tdelay_typ, tdelay_min, tdelay_max (whichever is selected) under the keyword [Diff Pin] in the IBIS file.
4. Which buffer gets a delayed trigger event depends on the sign of tdelay:
   If tdelay ≥ 0
   DelayNI = tdelay DelayI = 0
   If tdelay < 0
   DelayNI = 0 DelayI = -tdelay

IBIS_DO (Differential Output)

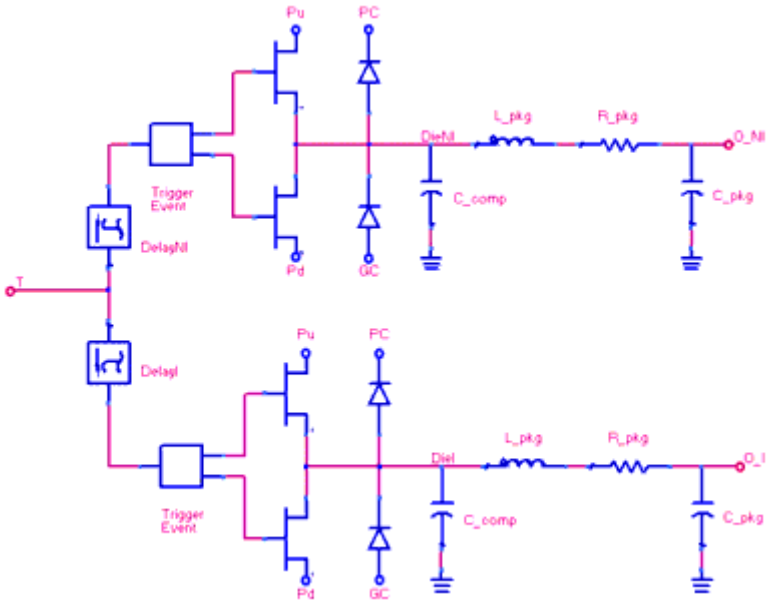
Symbol



Parameters

See *Model Parameters* (ibis).

Equivalent Circuit

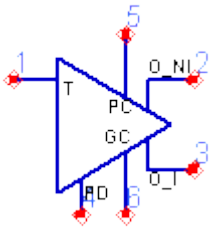


Notes/Equations

1. This buffer consists of two *IBIS_O (Output)* (ibis) buffers, one non-inverting and one inverting.
2. The trigger event can be delayed for one of the buffers.
3. This delay is controlled by the value of the subparameter tdelay_typ, tdelay_min, tdelay_max (whichever is selected) under the keyword [Diff Pin] in the IBIS file.
4. Which buffer gets a delayed trigger event depends on the sign of tdelay:
   If tdelay ≥ 0
   DelayNI = tdelay DelayI = 0
   If tdelay < 0
   DelayNI = 0 DelayI = -tdelay

IBIS_DOPENSINK (Differential Open Sink)

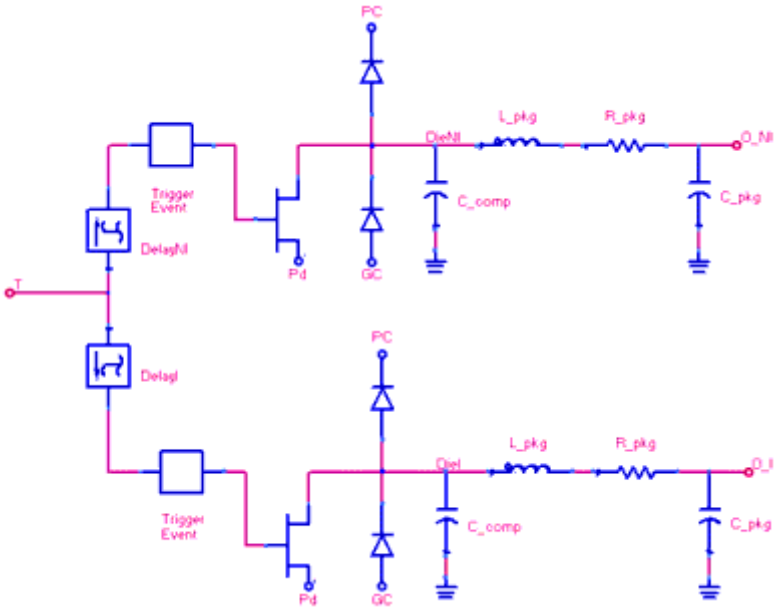
Symbol



Parameters

See *Model Parameters* (ibis).

Equivalent Circuit

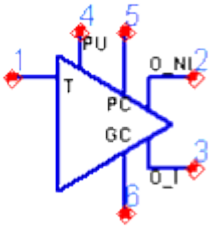


Notes/Equations

1. This buffer consists of two *IBIS_OPENSINK (Open Sink)* (ibis) buffers, one non-inverting and one inverting.
2. The trigger event can be delayed for one of the buffers.
3. This delay is controlled by the value of the subparameter tdelay_typ, tdelay_min, tdelay_max (whichever is selected) under the keyword [Diff Pin] in the IBIS file.
4. Which buffer gets a delayed trigger event depends on the sign of tdelay:
   If tdelay ≥ 0
   DelayNI = tdelay DelayI = 0
   If tdelay < 0
   DelayNI = 0 DelayI = -tdelay

IBIS_DOPENSOURCE (Differential Open Source)

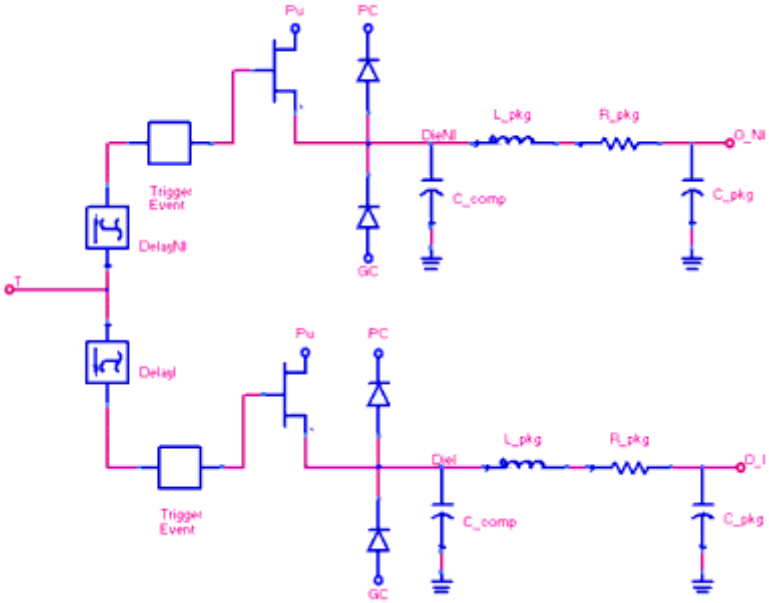
Symbol



Parameters

See *Model Parameters* (ibis).

Equivalent Circuit

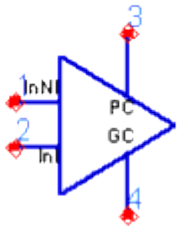


Notes/Equations

1. This buffer consists of two *IBIS_OPENSOURCE (Open Source)* (ibis) buffers, one non-inverting and one inverting.
2. The trigger event can be delayed for one of the buffers.
3. This delay is controlled by the value of the subparameter tdelay_typ, tdelay_min, tdelay_max (whichever is selected) under the keyword [Diff Pin] in the IBIS file.
4. Which buffer gets a delayed trigger event depends on the sign of tdelay:
   If tdelay ≥ 0
   DelayNI = tdelay DelayI = 0
   If tdelay < 0
   DelayNI = 0 DelayI = -tdelay

IBIS_DT (Differential Terminator)

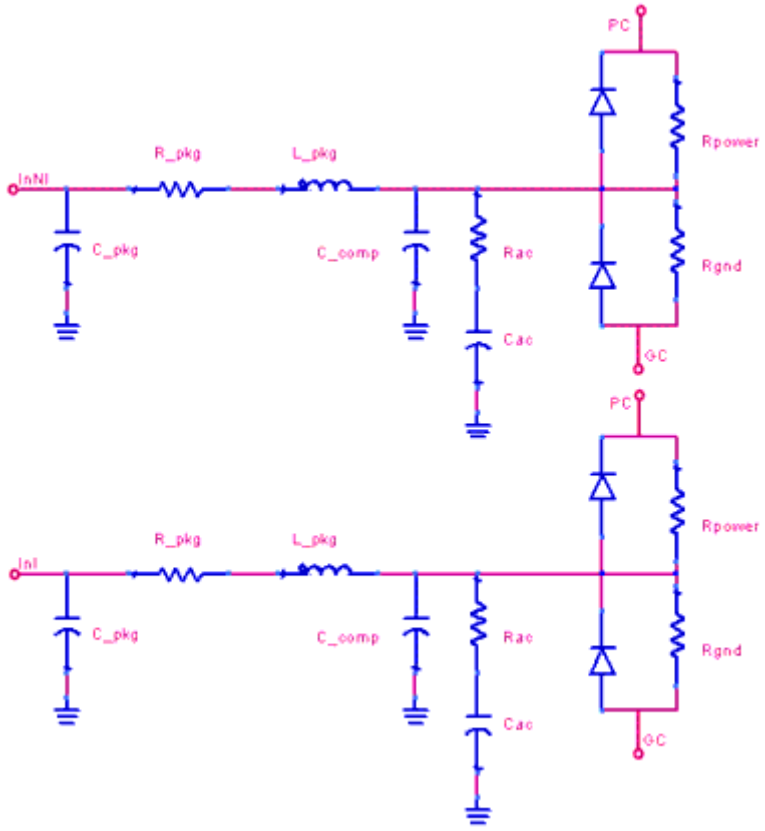
Symbol



Parameters

See *Model Parameters* (ibis).

Equivalent Circuit

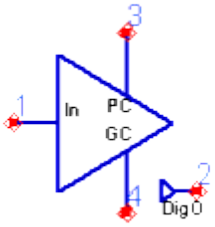


Notes/Equations

1. This buffer consists of two *IBIS_T (Terminator)* (ibis) buffers.
2. This buffer is similar to the *IBIS_DI (Differential Input)* (ibis) buffer without the digital output and with additional parasitic components.

IBIS_I_ECL (Input_ECL)

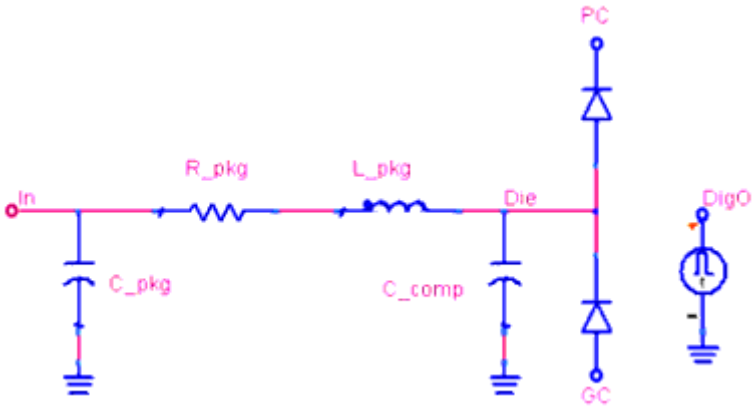
Symbol



Parameters

See *Model Parameters* (ibis).

Equivalent Circuit



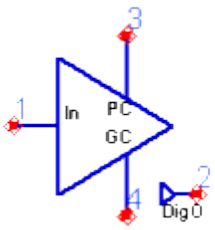
Notes/Equations

1. This buffer can function as a driver.
2. It supports digital output which can assume the values of 0 or 1 depending on the voltage at the node In as compared to the IBIS model parameters Vinl and Vinh, and polarity.
3. The Polarity subparameter of the [Model] keyword reverses digital output for

inverting buffers.
4. The only difference between Input_ECL and Input buffer is the default value of Vinh and Vinl.

IBIS_I (Input)

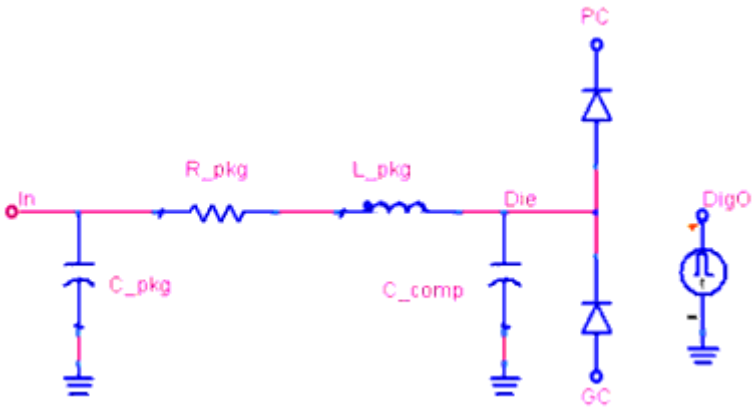
Symbol



Parameters

See *Model Parameters* (ibis).

Equivalent Circuit

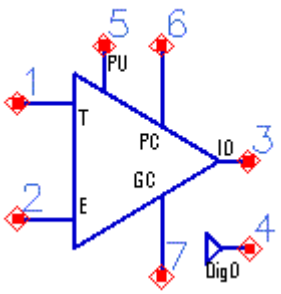


Notes/Equations

1. This buffer can function as a driver.
2. It supports digital output which can assume the values of 0 or 1 depending on the voltage at the node In as compared to the IBIS model parameters Vinl and Vinh, and polarity.
3. The Polarity subparameter of the [Model] keyword reverses digital output for inverting buffers.

IBIS_IO_ECL (Input/Output_ECL)

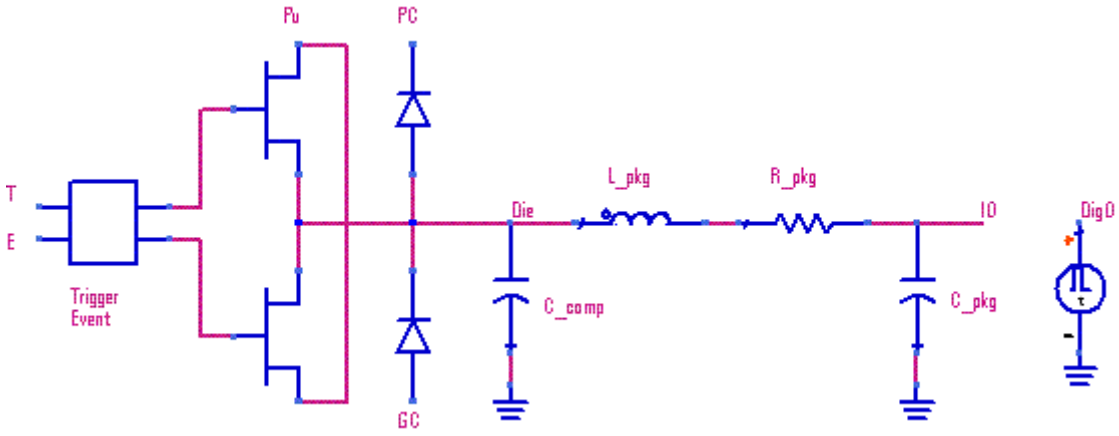
Symbol



Parameters

See *Model Parameters* (ibis).

Equivalent Circuit



Notes/Equations

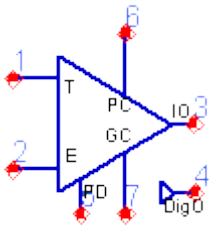
1. This is the most commonly used buffer and it functions as either the *IBIS_O_ECL (Output_ECL)* (ibis) buffer or the *IBIS_I_ECL (Input_ECL)* (ibis) buffer depending on the enable state.
2. If the buffer is disabled, it behaves as an *IBIS_I_ECL (Input_ECL)* (ibis) buffer with input node IO (it is In for the Input buffer).
3. In the Input mode, the buffer supports digital output which can assume the values of

0 or 1 depending on the voltage at the node IO as compared to the IBIS model parameters Vinl and Vinh, and polarity.

4. When the buffer is enabled it functions as an *IBIS_O_ECL (Output_ECL)* (ibis) buffer. However the voltage source Dig0 is still active.
5. This buffer is a combination of the *IBIS_3S_ECL (3-State_ECL)* (ibis) and *IBIS_I_ECL (Input_ECL)* (ibis) buffers.

IBIS_IO_OPENSINK (I/O Open Sink)

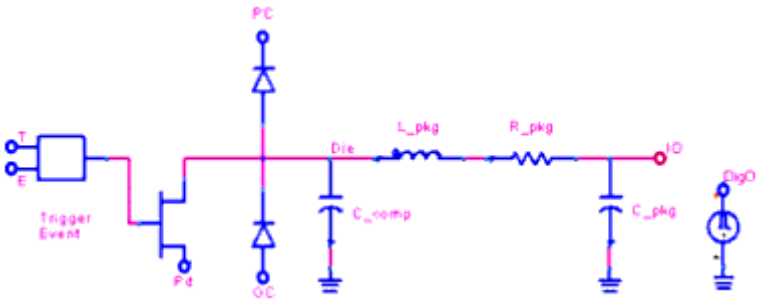
Symbol



Parameters

See *Model Parameters* (ibis).

Equivalent Circuit

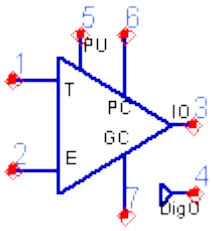


Notes/Equations

1. This buffer does not have the pullup device. Otherwise, all rules of the *IBIS_IO (Input/Output)* (ibis) buffer apply.

IBIS_IO_OPENSOURCE (I/O Open Source)

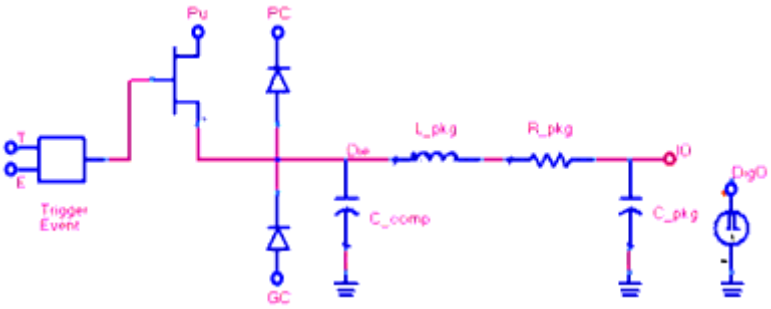
Symbol



Parameters

See *Model Parameters* (ibis).

Equivalent Circuit

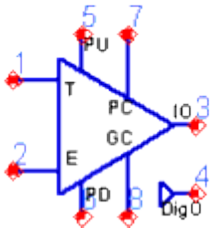


Notes/Equations

1. This buffer does not have the pulldown device. Otherwise, all rules of the *IBIS_IO (Input/Output)* (ibis) buffer apply.

IBIS_IO (Input/Output)

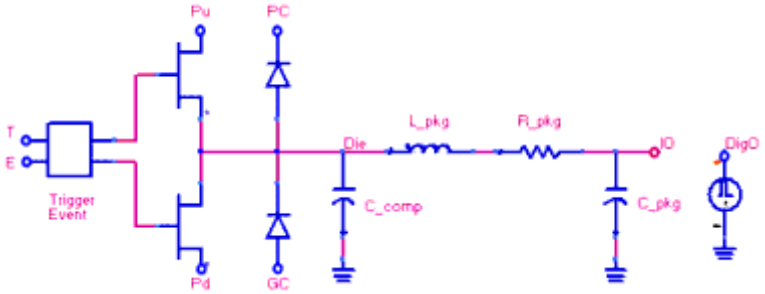
Symbol



Parameters

See *Model Parameters* (ibis).

Equivalent Circuit

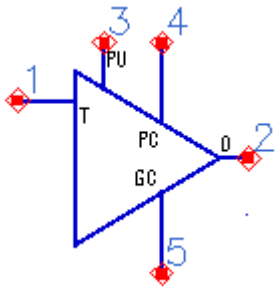


Notes/Equations

1. This is the most commonly used buffer and it functions as either the *IBIS_O (Output)* (ibis) buffer or the *IBIS_I (Input)* (ibis) buffer depending on the enable state.
2. If the buffer is disabled, it behaves as an *IBIS_I (Input)* (ibis) buffer with input node IO (it is In for the Input buffer).
3. In the Input mode, the buffer supports digital output which can assume the values of 0 or 1 depending on the voltage at the node IO as compared to the IBIS model parameters Vinl and Vinh, and polarity.
4. When the buffer is enabled it functions as an *IBIS_O (Output)* (ibis) buffer. However the voltage source Dig0 is still active.
5. This buffer is a combination of the *IBIS_3S (3-State)* (ibis) and *IBIS_I (Input)* (ibis) buffers.

IBIS_O_ECL (Output_ECL)

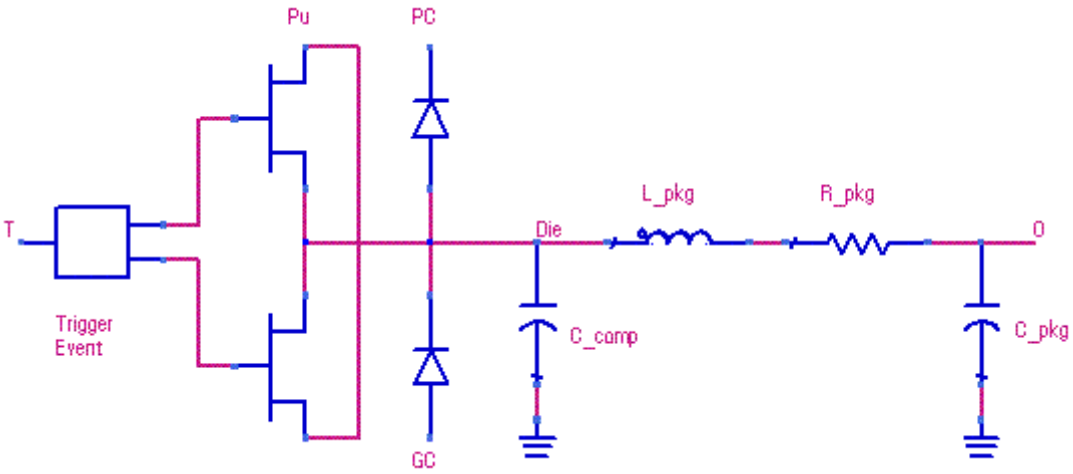
Symbol



Parameters

See *Model Parameters* (ibis).

Equivalent Circuit



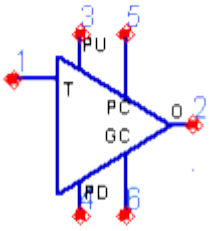
Notes/Equations

1. The trigger event results in a rising or falling trigger depending on the voltage at the node T as compared to the parameter TriggerLevel and polarity,
2. Polarity = Non-Inverting
   Time = 0
   Low State if VT < 0.5
   High State if VT ≥ 0.5
   Time > 0
   Falling trigger if VT < 1 - TriggerLevel
   Rising trigger if VT ≥ TriggerLevel
3. Polarity= Inverting
   Time = 0
   High State if VT ≤ 0.5
   Low State if VT > 0.5

Time > 0
Rising trigger if VT ≤ 1 - TriggerLevel
Falling trigger if VT > TriggerLevel

IBIS_O (Output)

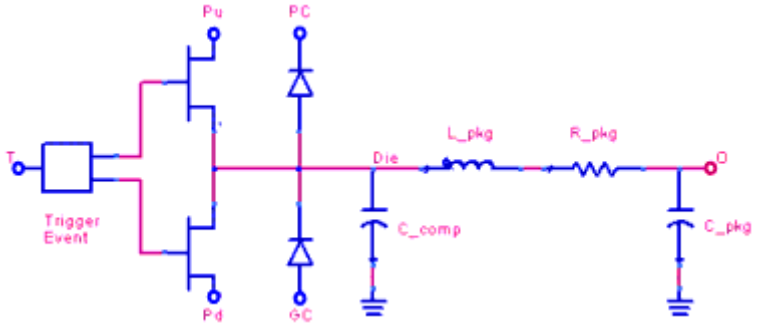
Symbol



Parameters

See *Model Parameters* (ibis).

Equivalent Circuit



Notes/Equations

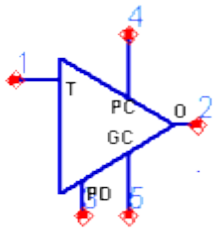
1. The trigger event results in a rising or falling trigger depending on the voltage at the node T as compared to the parameter TriggerLevel and polarity,
2. Polarity = Non-Inverting
   Time = 0
   Low State if VT < 0.5
   High State if VT ≥ 0.5
   Time > 0
   Falling trigger if VT < 1 - TriggerLevel

Rising trigger if VT ≥ TriggerLevel

3. Polarity= Inverting
   Time = 0
   High State if VT ≤ 0.5
   Low State if VT > 0.5
   Time > 0
   Rising trigger if VT ≤ 1 - TriggerLevel
   Falling trigger if VT > TriggerLevel

IBIS_OPENSINK (Open Sink)

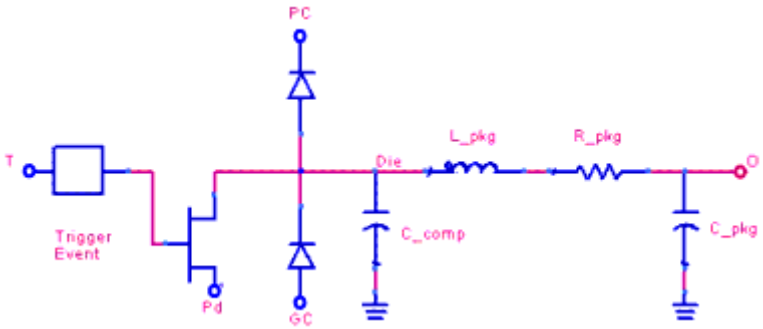
Symbol



Parameters

See *Model Parameters* (ibis).

Equivalent Circuit

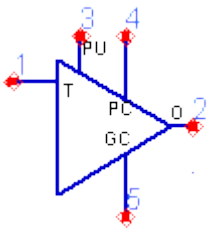


Notes/Equations

1. This buffer does not have the pullup device. Otherwise, all rules of the *IBIS_O (Output)* (ibis) buffer apply.

IBIS_OPENSOURCE (Open Source)

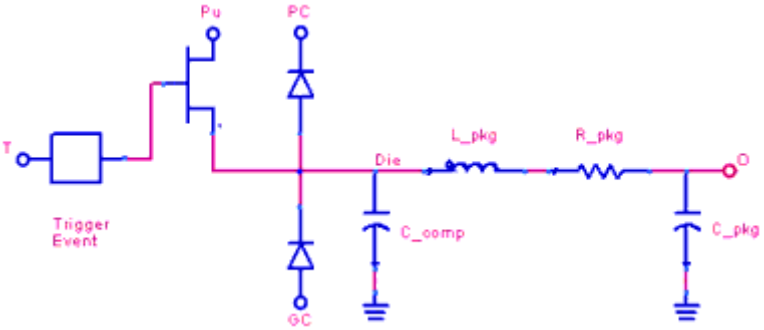
Symbol



Parameters

See *Model Parameters* (ibis).

Equivalent Circuit

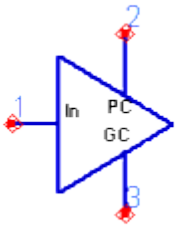


Notes/Equations

1. This buffer does not have the pulldown device. Otherwise, all rules of the *IBIS_O (Output)* (ibis) buffer apply.

IBIS_T (Terminator)

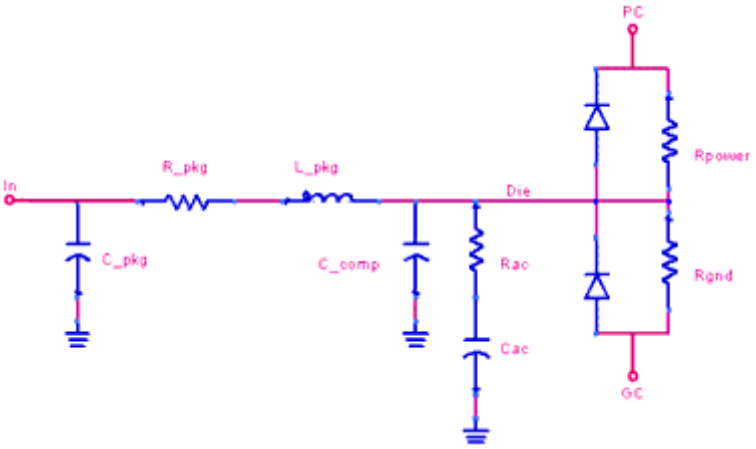
Symbol



Parameters

See *Model Parameters* (ibis).

Equivalent Circuit

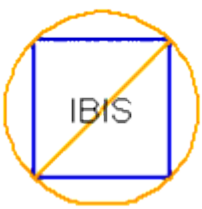


Notes/Equations

1. This buffer is similar to the *IBIS_I (Input)* (ibis) buffer without the digital output and with additional parasitic components.

IBIS (Generic Model)

Symbol



Parameters

See *Model Parameters* (ibis).

Notes/Equations

1. Provides a mechanism for selecting any type of IBIS model.
2. Because the generic component can be used to specify any one of the IBIS model types, it has no pins and cannot be simulated.
3. This component shows all supported pin/model combinations available for the selected component in the IBIS file.
4. When you specify enough information to determine the type of IBIS model and apply the settings, the generic model transforms into the appropriate specialized component.