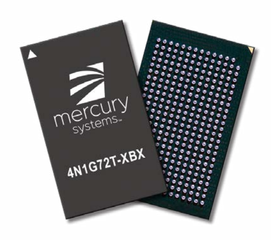
How to Implement EMD Model Using Hyperlynx for the Mercury System 8GB DDR4 Memory Module

4N1G72T-XB2X

Author: Truong, Khang

PURPOSE

The purpose of this application note is to provide a brief background on signal integrity, Electrical Board Description (EBD) and Electrical Module Description (EMD) models and how to use an EMD model to simulate signal integrity in Mercury 8GB DDR4 module 4N1G72T-XB2X based on DDR4 die using Hyperlynx.



TRANSMISSION LINE

In low-frequency systems, the resistance of wires or printed circuit board (PCB) traces connecting circuit components is insignificant enough to be negligible in most situations. But as signal frequency increases, these interconnects require special consideration since they stop behaving as ordinary wires but rather as transmission lines. A transmission line is a pair of conductors that must be analyzed according to the characteristics of high-frequency signal propagation.

Two basic PCB transmission lines are microstrip and stripline. A microstrip consists of a trace located on the outer layer of the board and a ground plane, which provides the return path for the signal, separated by PCB dielectric. A stripline consists of a trace on the inner layer of the board and two conducting planes running in parallel and sandwiching the trace[1].

A good rule of thumb to determine whether to incorporate transmission-line effects into analysis is: if the trace length is greater than one-fourth of the signal wavelength, transmission line effects become significant. The one-fourth threshold for a signal at 1 GHz frequency is about 50 mm, and at 10 GHz about 5 mm. With such low thresholds, almost all applications of microelectronics require transmission line analysis.

If the geometry of the transmission line is the same along the length, the line impedance is uniform, and the line is considered a controlled impedance trace. With controlled impedance, signal integrity is maintained. However, in the case of non-uniform impedance, signal reflections, crosstalk, jitter, and electromagnetic noise are much more noticeable.

An electrical wave travelling down a transmission line gets reflected at the point where there is a discontinuity in the line impedance. This can be a via going through the PCB, an abrupt change in the trace geometry such as a sharp turn or a discontinuity in the current return path. At the load, if there is an impedance mismatch, only some of the propagating energy gets absorbed and the rest is directed back at the source in the form of an electrical wave travelling in the opposite direction. The amount of reflected energy is dependent on the magnitude of the mismatch between the source and load impedance. Terminating the signal path by matching impedance allows for better energy dissipation at the load. Furthermore, it minimizes standing waves, a case where the sum of the incident and reflected waves creates a combination wave that has higher peak amplitude[2].

Another intra-system interference that hampers the signal integrity of the circuit is crosstalk. Crosstalk is a disturbance caused by coupling from one trace to another, even if they are not in contact. Two traces running near each other create plates of a capacitor separated by a dielectric. When there is a change in voltage, or electric field, between the two traces acting as capacitor plates, an unwanted current is induced. This effect is also known as parasitic capacitance. It is amplified by the proximity of the two traces. The more in parallel and close they are, the worse the effect. Magnetic, or inductive coupling is similar. When there is a change in the conducting current of a trace, and consequently the magnetic field, voltage is induced on the other trace. Although inductive coupling can occur, capacitive is the primary coupling in high-speed PCBs. Crosstalk degrades the signal integrity of the board by introducing timing delay, overshoot of voltage, and false triggering[3].

Crosstalk and reflections are the main cause of deterministic jitter. Other contributors include clock skew, impedance discontinuity and electromagnetic interference. Most deterministic jitter can be easily detected and minimized (only minimized because it's impossible to eliminate jitter completely) since it is periodic and repetitive. However, it is difficult to do so in the case of random jitter. Random jitter can be the consequence of thermal noise or randomness of charge carriers in materials, resulting in the loss of signal from one port to another. But any signal degradation, including all jitter, can be analyzed by S-parameters.

S-PARAMETERS

S (scattering) parameters are a behavioral representation of electrical networks. Scattering refers to how voltage and current are affected when a discontinuity is presented on a transmission line. One great advantage of S-parameters is it allows a device (or a combination of circuit elements) to be treated as a "black box" with inputs and outputs. This enables modeling a system without having to deal with the complexity of the design. Each S-parameter value is a ratio of the sine wave coming out of the terminating end to the sine wave entering the starting terminal. Consider a system below:

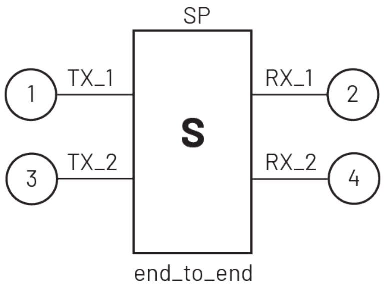


Figure 1: S-parameter[3]

When a stimulus is applied at net 1, the resulting waveform at net 2 will have an S-parameter denoted as S_{21}. Similarly, for S_{31}, S_{41} and S_{11}, the first sub-notation will always be the terminating end. In the case of S_{11}, the response starts and terminates on the same net. The transmission line can be characterized by these S-parameter values.

- S_{11}: Reflection
- S_{21}: Insertion/Transmission loss
- S_{31}: Near-end crosstalk
- S_{41}: Far-end crosstalk[3]

Circuit design has become increasingly complex to accommodate higher speed signals. Traces and wiring in a PCB board are almost all considered to be transmission lines. Analyzing behavioral characteristics of each device through S-parameters individually has then become a daunting task, not to mention each microelectronic vendor having their own method in quantifying S-parameter values. To combat this, industry gravitates towards a common standard: IBIS.

IBIS EBD & IBIS EMD

IBIS(I/O Buffer Information Specification) is an industry standard, fast and accurate method of modeling input/output buffers and high-speed interconnects. IBIS stores the behavioral information of the device characteristics of integrated circuits in a parse-able ASCII format. An IBIS model contains data of voltage and current values in outputs and inputs pins, as well as the voltage and time relationship at the output pins. By not disclosing circuit design such as nodal connections and process parameters, IBIS model protects proprietary information[4].

Originated from Intel, IBISv1.0 was released in April 1993. IBIS EBD was added to IBIS Specification in April 1997, but did not become popular until the invention of DDR2 specification around 2003[4]. An electrical board description file describes the connections between the module pins and its components on the board. IBIS EBD addresses the challenges of simulating a board for system designs.

However, there are limitations: it does not analyze coupled models - no crosstalk analysis for transmission line; it does not support power delivery networks; and most egregiously, it is limited to a "board". There is insufficient connector modelling built into IBIS EBD from its earlier version that assumes one-to-one package pin to I/O buffer connection. In essence, it is difficult for IBIS EBD model to model most multi-die packages such as stacked memories[5].

In 2019, IBIS 7.0 was approved. It increased package modelling capabilities of IBIS by the inclusion of Touchstone and IBIS-ISS format files[6]. The history of Touchstone (SnP) format dates back to the 1990s when a company acquired by Agilent standardized a way to express not just S, but also H, Y, Z parameters in an ASCII text file format. Touchstone format has been accepted as standard for transfer of frequency dependent n-port parameters[7].

IBIS-ISS (IBIS Interconnect SPICE Subcircuit) specification was developed to enable easier data exchange between users of signal/power integrity simulation and layout software tools. IBIS-ISS describes interconnect structures (PCB traces, connectors, cables) and their arrangement as they relate to each other and active devices in a system through an ASCII text file format. The syntax is similar to SPICE's and is a subset of HSPICE, which is a trademark registered by Synopsys[8]

Two years after the approval of IBIS 7.0, in 2021, the Electrical Module Description (EMD) was created. EMD describes electrical interconnectivity between and within modules for multi-chip modules, interposers, connectors, and cables by integrating both Touchstone and IBIS-ISS format. As a result, it solves EBD's package model limitation and can be used to model stacked-die DRAM components.

It allows the EMD model maker the flexibility to include crosstalk and power distribution network requirements in exchange for increased model complexity and simulation speed. EMD is a major improvement and a complete replacement for the antiquated EBD format[6].

The EMD model Mercury provides contains multiple IBIS-ISS and Touchstone files for devices within Mercury 8GB DDR4 module. It allows the end user to seamlessly import and instantiate with Mentor Graphics – Hyperlynx. Users have the option to connect the module to pins on a PCB or simply connect to a memory controller IBIS, then configure the traces with details.

CONTENT OF EMD PACKAGE

Each Touchstone file provided by Mercury contains a certain number of ports, indicated by the number in the file extension. For instance, 4N1G72T_XB2X_CAC1.s77p describe the S-parameter responses for all 77 ports incident by every other port including itself. Therefore, at each frequency, there are 77x77 responses.

Thus, the S-parameters for Command/Address/Control (CAC) and Clock nets are divided into multiple files to reduce file size and include neighboring nets to capture coupling. The S-Parameters for DQ/DQS/DM are split by byte. The S-Parameters are extracted from DC-10GHz.

Below is a table that lists all the files that are necessary for the EMD model to work. All files will need to be in the same folder as the EMD file references the other. The format and brief description of each file are also included.

TABLE 1: EMD PACKAGE FILES

File Name	Format	Description
4n1g72t_xb2x.emd	EMD	EMD model for Mercury 4N1G72T product.
4N1G72T_XB2X_CAC1.s77p	Touchstone sNp	Includes the S-parameters for signals ACT_N, CAS_N_A15, CKE, CS_N, ODT, RAS_N_A16, WE_N_A14 throughout the fly-by-topology.
4N1G72T_XB2X_CAC2.s66p	Touchstone sNp	Includes S-parameters for signals A10, A12, BA0, BA1, BG0, BG1 throughout the fly-by-topology.
4N1G72T_XB2X_CAC3.s66p	Touchstone sNp	Includes S-parameters for signals A0, A1, A3, A4, A5, A6 throughout the fly-by-topology.
4N1G72T_XB2X_CAC4.s77p	Touchstone sNp	Includes S-parameters for signals A2, A7, A8, A9, A11, A13, PARITY throughout the fly-by-topology.
4N1G72T_XB2X_CLK.s22p	Touchstone sNp	Includes S-parameters for signals CLK_C and CLK_T throughout the fly-by-topology.
4N1G72T_XB2X_U1_BYTE.s22p	Touchstone sNp	Includes S-parameters for DM0, DQS0_C, DQS0_T and DQ0 to DQ7 of Z11B die U1.
4N1G72T_XB2X_U2_BYTE.s22p	Touchstone sNp	Includes S-parameters for DM1, DQS2_C, DQS2_T and DQ8 to DQ15 of Z11B die U2.
4N1G72T_XB2X_U3_BYTE.s22p	Touchstone sNp	Includes S-parameters for DM2, DQS2_C, DQS2_T and DQ16 to DQ23 of Z11B die U3.
4N1G72T_XB2X_U4_BYTE.s22p	Touchstone sNp	Includes S-parameters for DM3, DQS3_C, DQS3_T and DQ24 to DQ31 of Z11B die U4.
4N1G72T_XB2X_U5_BYTE.s22p	Touchstone sNp	Includes S-parameters for DM4, DQS4_C, DQS4_T and DQ32 to DQ39 of Z11B die U5.
4N1G72T_XB2X_U6_BYTE.s22p	Touchstone sNp	Includes S-parameters for DM5, DQS5_C, DQS5_T and DQ40 to DQ47 of Z11B die U6.
4N1G72T_XB2X_U7_BYTE.s22p	Touchstone sNp	Includes S-parameters for DM6, DQS6_C, DQS6_T and DQ48 to DQ55 of Z11B die U7.
4N1G72T_XB2X_U8_BYTE.s22p	Touchstone sNp	Includes S-parameters for DM7, DQS7_C, DQS7_T and DQ56 to DQ63 of Z11B die U8.
4N1G72T_XB2X_U9_BYTE.s22p	Touchstone sNp	Includes S-parameters for DM8, DQS8_C, DQS8_T and DQ64 to DQ71 of Z11B die U9.
clk_term.iss	IBIS-ISS	Defines the subcircuit for CLK_C and CLK_T termination network.
resistor_39ohm.ibs	IBIS-ISS	IBIS model for 39 ohm resistor.
z11b_it.ibs	IBIS-ISS	Micron IBIS model for DDR4 DRAM

FILE FORMATTING

All SnP files have the following format:

- Netlist Comment
- Touchstone option line: # Hz S MA R 50
  - **Hz**: Frequency listed below is in Hz.
  - **S**: Values listed are S-parameters.
  - **MA**: Values listed are in format of Magnitude and Angle.
  - **R 50**: Resistance value that was normalized to. In this case, it will always be 50 Ohm.
- Port/Pin assignment
- S parameter values

PORT NAMING CONVENTION

Recommended IBIS Settings for Memory Controller:

- **CAC buses:** Use Stub Series Terminated Logic 1.2V (SSTL-12) models.
- **CLK signals:** Use Differential Stub Series Terminated Logic 1.2V (DSSTL-12) models.
- **DQ and DM signals:** Use Pseudo-Open Drain 1.2V (POD-12) models.
- **DQS signals:** Use Differential Pseudo-Open Drain 1.2V (DPOD-12) models.

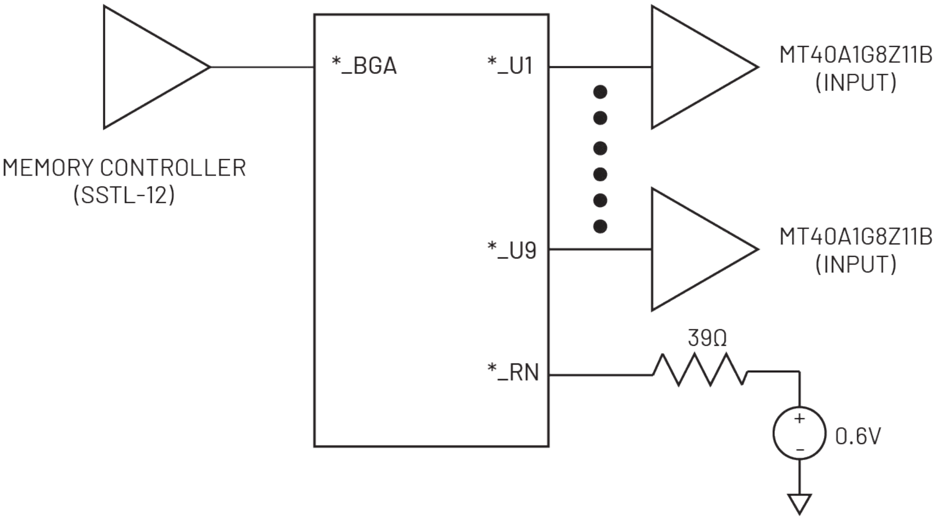


Figure 2: Command/Address/Control (CAC)

TABLE 2: CONNECTIVITY - COMMAND/ADDRESS/CONTROL (CAC)

Naming	Meaning	Example
*_BGA	BGA port	Port[1] = ACT_N_BGA.D4
*_RN	Termination resistor port	Port[2] = ACT_n_RN
_U	DRAM die reference destination port	Port[3] = ACT_N_U1.83
*.##	Pin number	

PORT NAMING CONVENTION (CON'T)

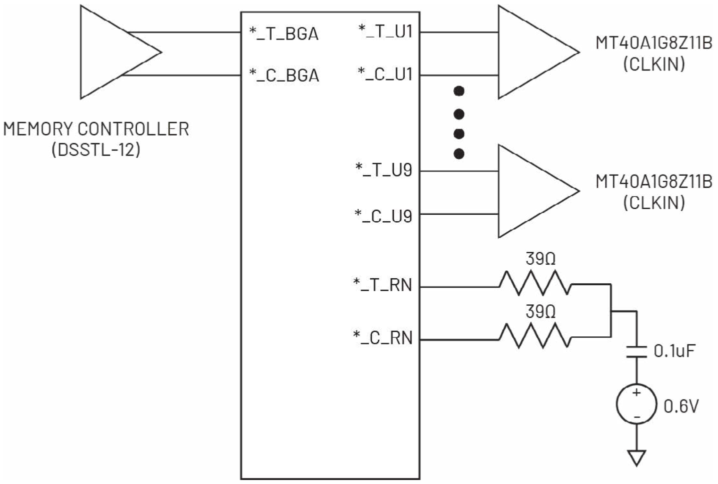


Figure 3: Clock

TABLE 3: CONNECTIVITY - CLOCK (CLK)

Naming	Meaning	Example
*_BGA	BGA port	Port[1] = CLK_T_BGA.D4
*_RN	Termination resistor port	Port[2] = CLK_T_RN
_U	DRAM die reference destination port	Port[3] = CLK_T_U1.70
*.##	Pin number	

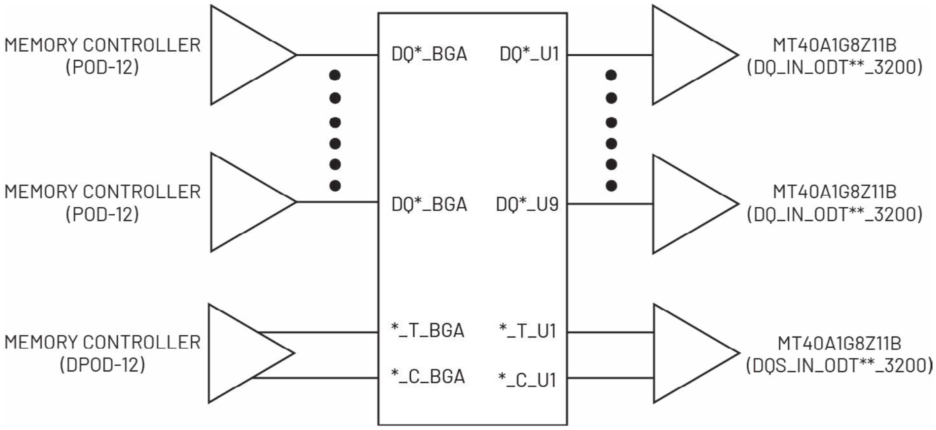


Figure 4: DQ/DQS/DM

TABLE 4: CONNECTIVITY - DQ/DQS/DM

Naming	Meaning	Example
*_BGA	BGA port	Port[1] = DQ12_BGA.H10
_U	DRAM die reference destination port	Port[2] = DQ12_U3.61
*.##	Pin number	

To load the EMD model into Hyperlynx (applicable to version 2.13 and above), perform the following:

1. Navigate to the **Models** dropdown menu
2. Select **Assign Models/Values by Reference Designator (.REF File)...**

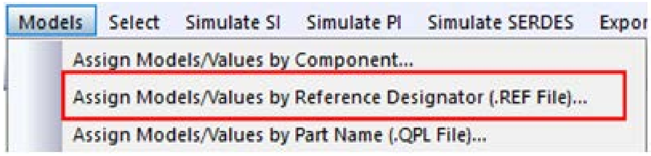
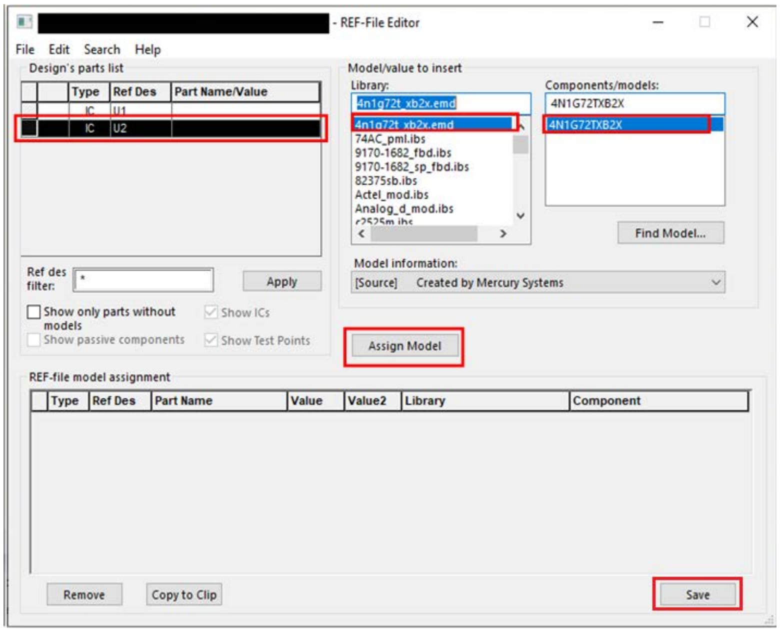

Models
Select
Simulate SI
Simulate PI
Simulate SERDES
Expor
Assign Models/Values by Component...
Assign Models/Values by Reference Designator (.REF File)...
Assign Models/Values by Part Name (.QPL File)...

Figure 5: Model Assignment

3. Select the part that will be the reference designator for Mercury DRAM. The figure below shows **U2** as the reference designator.
4. Select the **Library** and the **Components/Models** fields as in the below.

   NOTE: To have these options, the file path to the EMD model and its dependents needs to be included in Setup/Option/Directories.
5. Click on the **Assign Model** button.
6. Click on **Save**.



[redacted] - REF-File Editor

Design's parts list

Type	Ref Des	Part Name/Value
IC	U1	
IC	U2	

Model/value to insert

Library: 4n1g72t_xb2x.emd

4n1g72t_xb2x.emd
74AC_pml.ibs
9170-1682_fbd.ibs
9170-1682_sp_fbd.ibs
82375sb.ibs
Actel_mod.ibs
Analog_d_mod.ibs
c2525m.ibs

Components/models: 4N1G72TXB2X

4N1G72TXB2X

Model information: [Source] Created by Mercury Systems

Ref des filter: *

☐ Show only parts without models
☑ Show ICs
☐ Show passive components
☑ Show Test Points

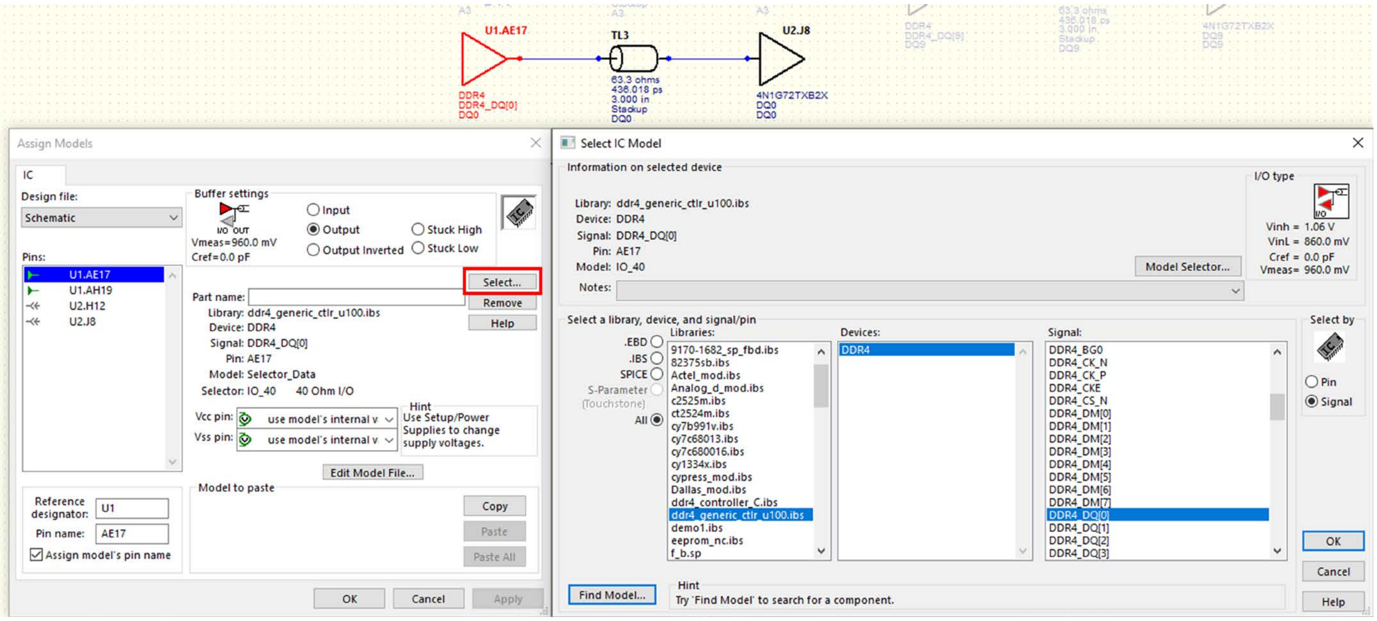
REF-file model assignment

Type	Ref Des	Part Name	Value	Value2	Library	Component

Figure 6: Library and Component/Models Selection

The steps below show how to instantiate and simulate a simple response for DQ0 when a write operation is performed:

1. Create a simple schematic with two I/O and one transmission line components.
2. Select **DQ0** as the signal.
   NOTE: The DDR4 controller IBIS model was supplied by Siemens.



Assign Models

IC

Design file: Schematic

Pins:
- U1.AE17
- U1.AH19
- U2.H12
- U2.J8

Buffer settings
Vmeas=960.0 mV
Cref=0.0 pF
○ Input
● Output
○ Output Inverted
○ Stuck High
○ Stuck Low

Part name:
Library: ddr4_generic_ctlr_u100.ibs
Device: DDR4
Signal: DDR4_DQ[0]
Pin: AE17
Model: Selector_Data
Selector: IO_40   40 Ohm I/O

Vcc pin: use model's internal v
Vss pin: use model's internal v

Hint
Use Setup/Power Supplies to change supply voltages.

Model to paste

Reference designator: U1
Pin name: AE17
☑ Assign model's pin name

Select IC Model

Information on selected device
Library: ddr4_generic_ctlr_u100.ibs
Device: DDR4
Signal: DDR4_DQ[0]
Pin: AE17
Model: IO_40
Notes:

I/O type
Vinh = 1.06 V
VinL = 860.0 mV
Cref = 0.0 pF
Vmeas= 960.0 mV

Select a library, device, and signal/pin
○ .EBD
○ .IBS
○ SPICE
○ S-Parameter (Touchstone)
● All

Libraries:
9170-1682_sp_fbd.ibs
82375sb.ibs
Actel_mod.ibs
Analog_d_mod.ibs
c2525m.ibs
ct2524m.ibs
cy7b991v.ibs
cy7c68013.ibs
cy7c680016.ibs
cy1334x.ibs
cypress_mod.ibs
Dallas_mod.ibs
ddr4_controller_C.ibs
ddr4_generic_ctlr_u100.ibs
demo1.ibs
eeprom_nc.ibs
f_b.sp

Devices:
DDR4

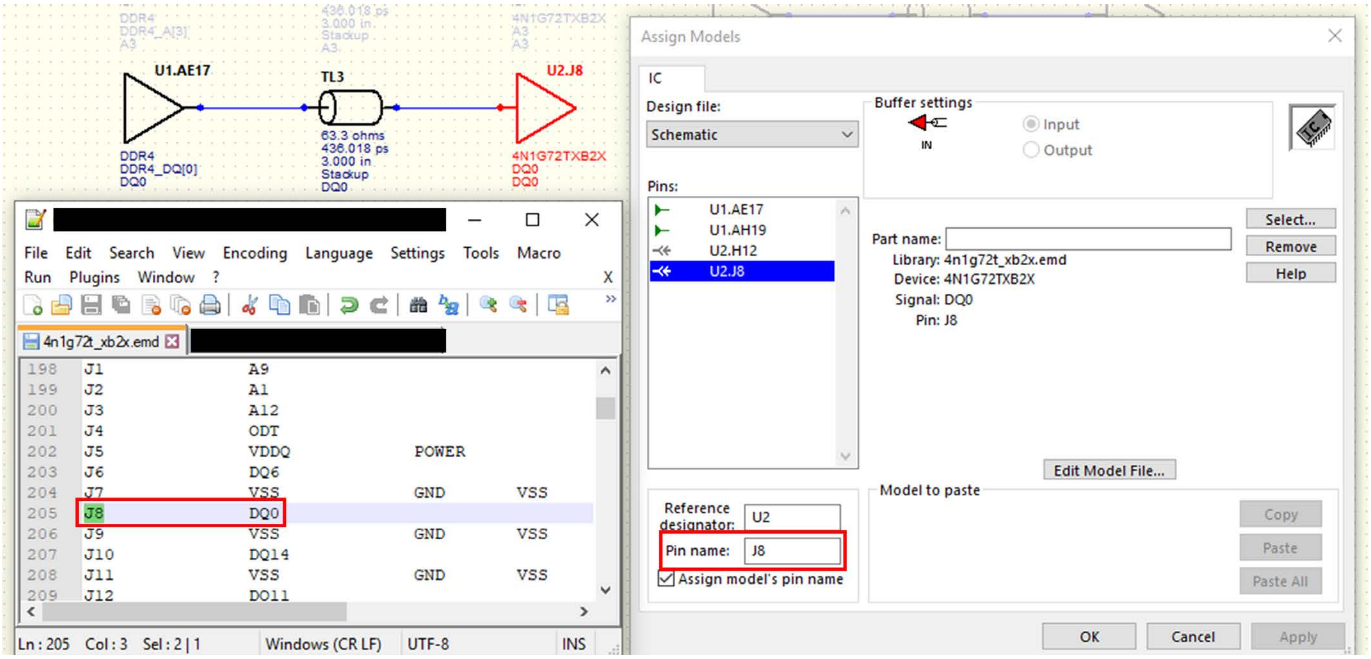
Signal:
DDR4_BG0
DDR4_CK_N
DDR4_CK_P
DDR4_CKE
DDR4_CS_N
DDR4_DM[0]
DDR4_DM[1]
DDR4_DM[2]
DDR4_DM[3]
DDR4_DM[4]
DDR4_DM[5]
DDR4_DM[6]
DDR4_DM[7]
DDR4_DQ[0]
DDR4_DQ[1]
DDR4_DQ[2]
DDR4_DQ[3]

Select by
○ Pin
● Signal

Hint
Try 'Find Model' to search for a component.

Figure 7: Simple Schematic Creation

3. For the other component, assign the pin name as listed in the EMD pin list from file 4n1g72t_xb2x.emd.



[redacted]

4n1g72t_xb2x.emd [redacted]

```
J1          A9
J2          A1
J3          A12
J4          ODT
J5          VDDQ        POWER
J6          DQ6
J7          VSS         GND        VSS
J8          DQ0
J9          VSS         GND        VSS
J10         DQ14
J11         VSS         GND        VSS
J12         DQ11
```

Ln : 205   Col : 3   Sel : 2 | 1   Windows (CR LF)   UTF-8   INS

Assign Models

IC

Design file: Schematic

Pins:
- U1.AE17
- U1.AH19
- U2.H12
- U2.J8

Buffer settings
○ Input
○ Output

Part name:
Library: 4n1g72t_xb2x.emd
Device: 4N1G72TXB2X
Signal: DQ0
Pin: J8

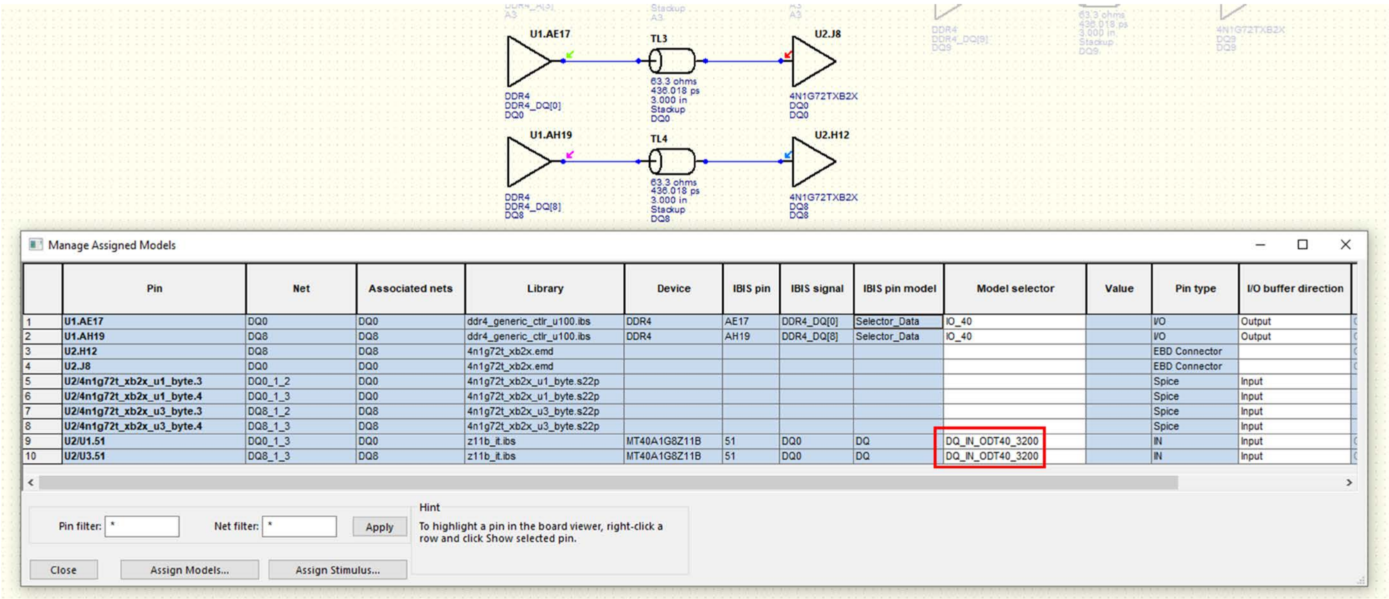
Model to paste

Reference designator: U2
Pin name: J8
☑ Assign model's pin name

Figure 8: Pin Name Assignment

4. After selecting Net DQ0 and DQ8, then select **Models/Manage Assign Models** to bring up the Manage Assign Models window.

   NOTE: The signals can be configured to be either input or output with predefined termination in the Model Selector column.



Manage Assigned Models

	Pin	Net	Associated nets	Library	Device	IBIS pin	IBIS signal	IBIS pin model	Model selector	Value	Pin type	I/O buffer direction
1	U1.AE17	DQ0	DQ0	ddr4_generic_ctlr_u100.ibs	DDR4	AE17	DDR4_DQ[0]	Selector_Data	IO_40		I/O	Output
2	U1.AH19	DQ8	DQ8	ddr4_generic_ctlr_u100.ibs	DDR4	AH19	DDR4_DQ[8]	Selector_Data	IO_40		I/O	Output
3	U2.H12	DQ8	DQ8	4n1g72t_xb2x.emd							EBD Connector	
4	U2.J8	DQ0	DQ0	4n1g72t_xb2x.emd							EBD Connector	
5	U2/4n1g72t_xb2x_u1_byte.3	DQ0_1_2	DQ0	4n1g72t_xb2x_u1_byte.s22p							Spice	Input
6	U2/4n1g72t_xb2x_u1_byte.4	DQ0_1_3	DQ0	4n1g72t_xb2x_u1_byte.s22p							Spice	Input
7	U2/4n1g72t_xb2x_u3_byte.3	DQ8_1_2	DQ8	4n1g72t_xb2x_u3_byte.s22p							Spice	Input
8	U2/4n1g72t_xb2x_u3_byte.4	DQ8_1_3	DQ8	4n1g72t_xb2x_u3_byte.s22p							Spice	Input
9	U2/U1.51	DQ0_1_3	DQ0	z11b_it.ibs	MT40A1G8Z11B	51	DQ0	DQ	DQ_IN_ODT40_3200		IN	Input
10	U2/U3.51	DQ8_1_3	DQ8	z11b_it.ibs	MT40A1G8Z11B	51	DQ0	DQ	DQ_IN_ODT40_3200		IN	Input

Pin filter: *
Net filter: *

Hint
To highlight a pin in the board viewer, right-click a row and click Show selected pin.

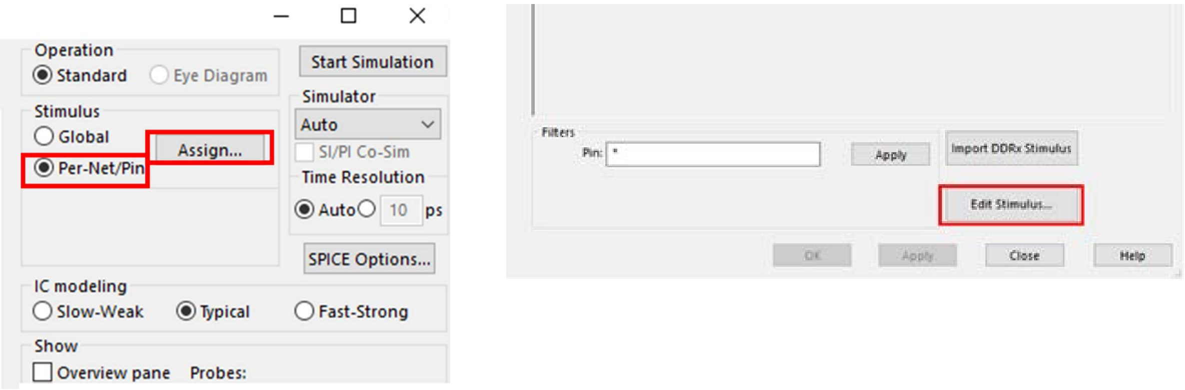
Figure 9: Models/Manage Assign ModelsScreen

5. Select the appropriate button on the toolbar to view the **Digital Oscilloscope**.

6. Click on the **Per-Net Pin** button.

7. Click on **Assign...**.

8. Click on **Edit Stimulus...** in the new **Edit Stimulus** window.



Operation
● Standard ○ Eye Diagram

Stimulus
○ Global
● Per-Net/Pin

Simulator
Auto
□ SI/PI Co-Sim

Time Resolution
● Auto ○ 10 ps

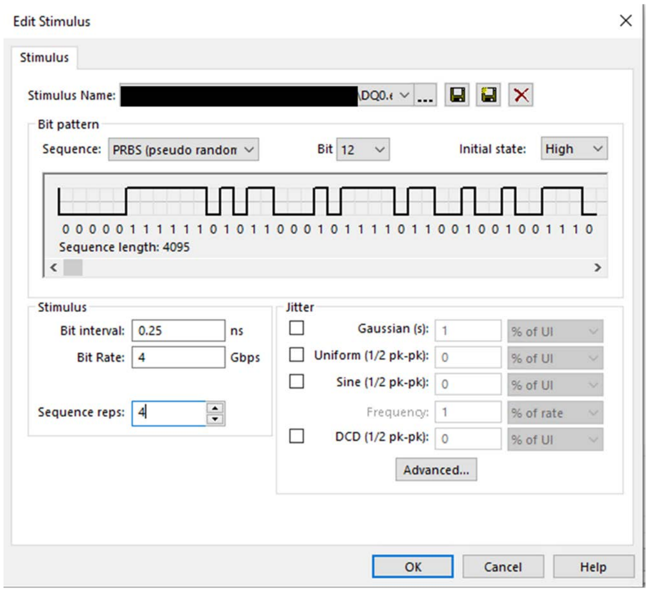
IC modeling
○ Slow-Weak ● Typical ○ Fast-Strong

Show
□ Overview pane Probes:

Filters
Pin: *

Figure 10: Digital Oscilloscope Settings

9. After configuring the stimulus, it can be saved for future uses.


Edit Stimulus

Stimulus

Stimulus Name: [redacted]\DQ0.‹

Bit pattern

Sequence: PRBS (pseudo randon

Bit: 12

Initial state: High

0 0 0 0 0 1 1 1 1 1 1 0 1 0 1 1 0 0 0 1 0 1 1 1 1 0 1 1 0 0 1 0 0 1 0 0 1 1 1 0

Sequence length: 4095

Stimulus

Bit interval: 0.25 ns

Bit Rate: 4 Gbps

Sequence reps: 4

Jitter

□	Gaussian (s):	1	% of UI
□	Uniform (1/2 pk-pk):	0	% of UI
□	Sine (1/2 pk-pk):	0	% of UI
	Frequency:	1	% of rate
□	DCD (1/2 pk-pk):	0	% of UI

Figure 11: Stimulus Screen

SUMMARY

As signal frequency increases and devices become more compact, traces and wires need to be treated as transmission lines. Signal Integrity needs to be preserved through accounting for signal reflection, crosstalk, jitter, and noise by inspecting S-parameters of the devices within the module. Mercury provides the customer with 8GB DDR4 EMD model, which is a new improved standard from the IBIS Forum compared to old EBD model. This EMD model enables users to simulate the response of the module through Mentor Graphics Hyperlynx.

TABLE 5: REFERENCES

Reference Number	Source
1	Transmission Lines. Electromagnetic Field Theory. Purdue Engineering. Purdue University.
2	Atar Mittal, What Is a Transmission Line? Sierra Circuits. ProtoExpress. Feb 2023.
3	How to Handle Crosstalk in PCB Design. Sierra Circuits. ProtoExpress. Nov 2023.
4	IBIS Interconnect SPICE Subcircuit (IBIS-ISS) Specification Version 1.0, IBIS Open Forum, 2011.
5	Mercedes Casamayor. A First Approach to IBIS Models: What They Are and How They Are Generated. Analog Devices. 2004.
6	Syed B. Huq, An Introduction to IBIS (I/O Buffer Information Specification) Modeling. National Semiconductor, June 1998.
7	Touchstone® File Format Specification Version 2.0. Tech America. 2009
8	Randy Wolff. IBIS 7.1 Delivers a Wealth of Improvements for Signal Integrity Engineers. Signal Integrity Journal. May 24, 2022
9	Donald Telian, Understanding and Using S-Parameters for PCB Signal Integrity. Cadence Design System Inc. 2004.

Corporate Headquarters

50 Minuteman Road
Andover, MA 01810 USA
+1 978.967.1401 tel
+1 866.627.6951 tel
+1 978.256.3599 fax

International Headquarters
Mercury International

Avenue Eugène-Lance, 38
PO Box 584
CH-1212 Grand-Lancy 1
Geneva, Switzerland
+41 22 884 51 00 tel

Learn more

Visit: mrcy.com/memory

Contact: secure.memory@mrcy.com

The Mercury Systems logo is a registered trademark of Mercury Systems, Inc. Other marks used herein may be trademarks or registered trademarks of their respective holders. Mercury products identified in this document conform with the specifications and standards described herein. Conformance to any such standards is based solely on Mercury's internal processes and methods. The information contained in this document is subject to change at any time without notice.

MADE IN USA

4608.00E-0424-an-8GBDDR4