TLK10232 HSPICE Model Guide

Rev. 1.0 – April 2013

This document describes the HSPICE models for the TLK10232 device. This release contains models for the HSRX, HSTX, LSRX and LSTX pins. Also included are a variety of stimulus patterns and s-parameter models of varying lengths of FR4. A Perl script exists to generate the correct stimulus for arbitrary bit patterns.

WARNING: Because the package substrate design was not finished at the time of this release, these models do NOT contain package models.

1. Component Models

The transistor, resistor and capacitor models are located in the subdirectory "models". This release contains models for typical (typical.inc), strong (strong.inc) and weak (weak.inc) process parameters.

The supply voltage and temperature are set in the top-level testbench.

2. Circuit Models

The models for the various I/O cells are located in the subdirectory "include". This release contains the following models:

File	Description	Applicable Pins
tlk10232_hstx.inc	High-speed side transmitter	HSTXAP, HSTXAN, HSTXBP, HSTXBN
tlk10232_hsrx.inc	High-speed side receiver	HSRXAP, HSRXAN, HSRXBP, HSRXBN
tlk10232_lstx.inc	Low-speed side transmitter	LSTXA[3:0]P/N, LSTXB[3:0]P/N
tlk10232_lsrx.inc	Low-speed side receiver	LSRXA[3:0]P/N, LSRXB[3:0]P/N

2.1. Scale factor

The TLK10232 models require a scale of 1e-6. If these models must be combined with other instances requiring a scale of 1 it is required to add the following line to the top-level testbench:

```
.option hier_scale=1 scale=1
```

In addition, the parameter "S=1e-6" must be added to all instantiations of the TLK10232 models. All instances without the "S=1e-6" will use the global scale of 1 set by the "scale=1" in the .option command while the TLK10232 models will use the required scale of 1e-6.

2.2. Model tlk10232_hstx.inc

The model, tlk10232_hstx.inc, includes the transmitter and package for the high-speed side of the TLK10232. It contains the following ports:

Port	Description
txp_ball	Positive input
txn_ball	Negative input
pc_0	Pre-cursor data input 0
pc_1	Pre-cursor data input 1
pc_2	Pre-cursor data input 2
pc_3	Pre-cursor data input 3
cu_0	Main-cursor data input 0
cu_1	Main-cursor data input 1
cu_2	Main-cursor data input 2
cu_3	Main-cursor data input 3
p1_0	Post-cursor1 data input 0
p1_1	Post-cursor1 data input 1
p1_2	Post-cursor1 data input 2
p1_3	Post-cursor1 data input 3
p2_0	Post-cursor2 data input 0
p2_1	Post-cursor2 data input 1
p2_2	Post-cursor2 data input 2
p2_3	Post-cursor2 data input 3
ph0	4-phase clock with period=4UI
ph1	4-phase clock with period=4UI
ph2	4-phase clock with period=4UI
ph3	4-phase clock with period=4UI
Vdda	Analog supply (1.0V +/- 5%)
Vddr	Analog Supply (1.5 or 1.8V +/- 5%)
Vssa	Analog Ground

The outputs of this model (txp_ball and txn_ball) represent the differential high-speed signal as it exits the TLK10232 package. An ac-coupling capacitor is assumed between these signals and a 100ohm differential load.

The remaining signals are inputs to the driver model. Because of speed limitations, the data is applied to the transmitter 4UI at a time with respect to the 4-phase clock (ph[3:0]). In order properly align the inputs with the 4-phase clock, a script has been written to generate 4-phase clock, ph[3:0] as well as the inputs, cu_[3:0], pc_[3:0], p1_[3:0] and p1_[3:0]. Sample patterns for PRBS7, PRBS23 and PRBS31 containing 1000UI worth of input data are contained in the “hs_patterns” directory. These patterns do not repeat after 1000UI, though, so they would need to be regenerated if a longer simulation time was desired. See Section 4 below for details on the script used to generate these patterns.

The TLK10232 high-speed transmitter has five controls that are included in this model. The user can control the swing, main-cursor reduction, pre-cursor and two post-cursor settings with the following parameters. These parameters must be defined in the top-level testbench and have a value of 0 or 1. They are internally scaled by the supply voltage to drive the actual circuit.

Parameter	Description
hs_swing_0_reg3p12	Bit 0 of the 4-bit SWING control defined by MDIO register 3.15:12
hs_swing_1_reg3p13	Bit 1 of the 4-bit SWING control defined by MDIO register 3.15:12
hs_swing_2_reg3p14	Bit 2 of the 4-bit SWING control defined by MDIO register 3.15:12
hs_swing_3_reg3p15	Bit 3 of the 4-bit SWING control defined by MDIO register 3.15:12
hs_twcrf_0_reg4p0	Bit 0 of the 5-bit TWCRF control defined by MDIO register 4.4:0
hs_twcrf_1_reg4p1	Bit 1 of the 5-bit TWCRF control defined by MDIO register 4.4:0
hs_twcrf_2_reg4p2	Bit 2 of the 5-bit TWCRF control defined by MDIO register 4.4:0
hs_twcrf_3_reg4p3	Bit 3 of the 5-bit TWCRF control defined by MDIO register 4.4:0
hs_twcrf_4_reg4p4	Bit 4 of the 5-bit TWCRF control defined by MDIO register 4.4:0
hs_twpre_0_reg5p4	Bit 0 of the 4-bit TWPRE control defined by MDIO register 5.7:4
hs_twpre_1_reg5p5	Bit 1 of the 4-bit TWPRE control defined by MDIO register 5.7:4
hs_twpre_2_reg5p6	Bit 2 of the 4-bit TWPRE control defined by MDIO register 5.7:4
hs_twpre_3_reg5p7	Bit 3 of the 4-bit TWPRE control defined by MDIO register 5.7:4
hs_twpost1_0_reg5p8	Bit 0 of the 5-bit TWPOST1 control defined by MDIO register 5.12:8
hs_twpost1_1_reg5p9	Bit 1 of the 5-bit TWPOST1 control defined by MDIO register 5.12:8
hs_twpost1_2_reg5p10	Bit 2 of the 5-bit TWPOST1 control defined by MDIO register 5.12:8
hs_twpost1_3_reg5p11	Bit 3 of the 5-bit TWPOST1 control defined by MDIO register 5.12:8
hs_twpost1_4_reg5p12	Bit 4 of the 5-bit TWPOST1 control defined by MDIO register 5.12:8
hs_twpost2_0_reg5p0	Bit 0 of the 4-bit TWPOST2 control defined by MDIO register 5.3:0
hs_twpost2_1_reg5p1	Bit 1 of the 4-bit TWPOST2 control defined by MDIO register 5.3:0
hs_twpost2_2_reg5p2	Bit 2 of the 4-bit TWPOST2 control defined by MDIO register 5.3:0
hs_twpost2_3_reg5p3	Bit 3 of the 4-bit TWPOST2 control defined by MDIO register 5.3:0

2.3. Model tlk10232_hsrx.inc

The model, tlk10232_hsrx.inc, includes the receive package and termination networks for the high-speed side of the TLK10232. It contains the following ports:

Port	Description
rxp_ball	Positive input
rxn_ball	Negative input
vdda	Analog supply (1.0V +/- 5%)
vssa	Analog Ground

There are no configuration parameters for this model and it assumes that the transmitter is ac-coupled.

2.4. Model tlk10232_lstx.inc

The model, tlk10232_lstx.inc, includes the transmitter and package for the low-speed side of the TLK10232. It contains the following ports:

Port	Description
txp_ball	Positive input
txn_ball	Negative input
tx_even	Even data input (First bit transmitted)
tx_odd	Odd data input
vdda	Analog supply (1.0V +/- 5%)
vssa	Analog Ground

The outputs of this model (txp_ball and txn_ball) represent the differential low-speed signal as it exits the TLK10232 package. An ac-coupling capacitor is assumed between these signals and a 100ohm differential load.

The inputs tx_even and tx_odd define the data to be transmitted. Because of speed limitations, the data is applied to the transmitter 2UI at a time with tx_even being transmitted first. In order properly align the inputs with the transmit clock, a script has been written to generate tx_even and tx_odd. Sample patterns for PRBS7, PRBS23 and PRBS31 containing 1000UI worth of input data are contained in the "ls_patterns" directory. These patterns do not repeat after 1000UI, though, so they would need to be regenerated if a longer simulation time was desired. See Section 5 below for details on the script used to generate these patterns.

The TLK10232 low-speed transmitter has two controls that are included in this model. The user can control the swing and pre-emphasis settings with the following parameters. These parameters must be defined in the top-level testbench and have a value of 0 or 1. They are internally scaled by the supply voltage to drive the actual circuit.

Parameter	Description
ls_swing_0_reg7p12	Bit 0 of the 3-bit SWING control defined by MDIO register 7.14:12
ls_swing_1_reg7p13	Bit 1 of the 3-bit SWING control defined by MDIO register 7.14:12
ls_swing_2_reg7p14	Bit 2 of the 3-bit SWING control defined by MDIO register 7.14:12
ls_de_0_reg7p4	Bit 0 of the 4-bit DE control defined by MDIO register 7.7:4
ls_de_1_reg7p5	Bit 1 of the 4-bit DE control defined by MDIO register 7.7:4
ls_de_2_reg7p6	Bit 2 of the 4-bit DE control defined by MDIO register 7.7:4
ls_de_3_reg7p7	Bit 3 of the 4-bit DE control defined by MDIO register 7.7:4

2.5. Model tlk10232_lsrx.inc

The model, tlk10232_lsrx.inc, includes the receive package and termination networks for the low-speed side of the TLK10232. It contains the following ports:

Port	Description
rxp_ball	Positive input
rxn_ball	Negative input
vdda	Analog supply (1.0V +/- 5%)
vssa	Analog Ground

There are no configuration parameters for this model and it assumes that the transmitter is ac-coupled.

3. FR4 channel models

A set of s-parameter models for varying lengths of FR4 trace are included in this release to allow the user to explore the effect of trace length before running simulations with a model of their board. This release contains the following FR4 models:

Trace Length	Model file
2 in	diffstripline_w4_s30_h10p2_t1p2_fr4_2in.s4p
4 in	diffstripline_w4_s30_h10p2_t1p2_fr4_4in.s4p
6 in	diffstripline_w4_s30_h10p2_t1p2_fr4_6in.s4p
8 in	diffstripline_w4_s30_h10p2_t1p2_fr4_8in.s4p
12 in	diffstripline_w4_s30_h10p2_t1p2_fr4_12in.s4p
16 in	diffstripline_w4_s30_h10p2_t1p2_fr4_16in.s4p
18 in	diffstripline_w4_s30_h10p2_t1p2_fr4_18in.s4p
20 in	diffstripline_w4_s30_h10p2_t1p2_fr4_20in.s4p
24 in	diffstripline_w4_s30_h10p2_t1p2_fr4_24in.s4p
28 in	diffstripline_w4_s30_h10p2_t1p2_fr4_28in.s4p
32 in	diffstripline_w4_s30_h10p2_t1p2_fr4_32in.s4p
36 in	diffstripline_w4_s30_h10p2_t1p2_fr4_36in.s4p

4. Input stimulus for high-speed models

The input stimulus are located in the “hs_patterns” subdirectory. This release contains the following stimulus files:

Stimulus file	Length	Description
ethernet_idle.hsp	1000 UI	Repeating pattern of “00111110101001000101”
prbs7.hsp	1000 UI	2^7-1 PRBS starting will seed of all 1’s
prbs23.hsp	1000 UI	2^23-1 PRBS starting will seed of all 1’s
prbs31.hsp	1000 UI	2^31-1 PRBS starting will seed of all 1’s

4.1. Script to generate stimulus for high-speed models

A PERL script to convert arbitrary bit patterns into a stimulus file for the high-speed transmitter called gen_hs_pattern.pl is also included in the "hs_patterns" subdirectory. This script has the following required and optional arguments:

Required Arguments	
Argument	Description
-p pattern	Specifies the pattern to generate. If pattern is "prbs7", "prbs23", or "prbs31", the appropriate PRBS pattern will be generated. Otherwise, pattern must be the name of the file containing the desired pattern
-o output_file	The script will write the stimulus to the file output_file
Optional Arguments	
Argument	Description
-n pattern_length	By default, the script generates 1000UI worth of stimulus. If this argument is used, the script will generate pattern_length worth of stimulus. Remember that the stimulus does not repeat after this length so pattern_length must be longer than the expected simulation time.
-s seed	By default, the script uses a seed of all 1's to generate the PRBS patterns. While this is probably acceptable for PBRS7 due to the short length, it may be undesirable for the other two PRBS patterns. This argument can be used to allow a short simulation over a different portion of a longer PRBS sequence by changing the starting point in the sequence for stimulus generation. The seed value must be a string of 1's and/or 0's that is exactly equal in length to the PRBS pattern requested. For example, a seed 31 characters long must be supplied to generate a PRBS31 pattern.

4.2. Restrictions on the pattern file for the –p argument

When generating the stimulus from a bit pattern file using the –p argument, the pattern file must only contain 0's, 1's and white-space (space, tab, new-lines). There are no restrictions on the number of bits per line. All white-space and line breaks are removed while reading the pattern. If the specified pattern is less than the requested number of bits in the stimulus file (1000 or user-specified pattern length using –n) the pattern will repeat.

5. Input stimulus for low-speed models

The input stimulus are located in the "ls_patterns" subdirectory. This release contains the following stimulus files:

Stimulus file	Length	Description
ethernet_idle.hsp	1000 UI	Repeating pattern of "00111110101001000101"
prbs7.hsp	1000 UI	2^7-1 PRBS starting will seed of all 1's
prbs23.hsp	1000 UI	2^23-1 PRBS starting will seed of all 1's
prbs31.hsp	1000 UI	2^31-1 PRBS starting will seed of all 1's

5.1. Script to generate stimulus for low-speed models

A PERL script to convert arbitrary bit patterns into a stimulus file for the low-speed transmitter called gen_ls_pattern.pl is also included in the "ls_patterns" subdirectory. This script has the following required and optional arguments:

Required Arguments	
Argument	Description
-p pattern	Specifies the pattern to generate. If pattern is "prbs7", "prbs23", or "prbs31", the appropriate PRBS pattern will be generated. Otherwise, pattern must be the name of the file containing the desired pattern
-o output_file	The script will write the stimulus to the file output_file
Optional Arguments	
Argument	Description
-n pattern_length	By default, the script generates 1000UI worth of stimulus. If this argument is used, the script will generate pattern_length worth of stimulus. Remember that the stimulus does not repeat after this length so pattern_length must be longer than the expected simulation time.
-s seed	By default, the script uses a seed of all 1's to generate the PRBS patterns. While this is probably acceptable for PBRS7 due to the short length, it may be undesirable for the other two PRBS patterns. This argument can be used to allow a short simulation over a different portion of a longer PRBS sequence by changing the starting point in the sequence for stimulus generation. The seed value must be a string of 1's and/or 0's that is exactly equal in length to the PRBS pattern requested. For example, a seed 31 characters long must be supplied to generate a PRBS31 pattern.

5.2. Restrictions on the pattern file for the –p argument

When generating the stimulus from a bit pattern file using the –p argument, the pattern file must only contain 0's, 1's and white-space (space, tab, new-lines). There are no restrictions on the number of bits per line. All white-space and line breaks are removed while reading the pattern. If the specified pattern is less than the requested number of bits in the stimulus file (1000 or user-specified pattern length using –n) the pattern will repeat.

6. Sample testbenches

This release contains a sample testbench for the high-speed and low-speed models. Each testbench contains the transmitter model driving the receiver model through ac-coupling capacitors. This release contains the following sample testbenches:

Testbench	Description
ls_prbs7_4infr4.hsp	Low-speed testbench with PRBS7 stimulus
hs_prbs7_4infr4.hsp	High-speed testbench with PRBS7 stimulus

7. Release History

7.1. Version 1.0, July 16, 2010

- Initial release of models for HSTX, HSRX, LSTX and LSRX models
- Initial release of typical, weak and strong device models
- Initial release of stimulus patterns and sample testbenches
- Models require a scale of 1e-6
- Models do NOT contain package information. The package design was not complete at the time of release.