A look at PCB Signal Integrity and the Cadence SPB / OrCAD Tools

The objective: send a signal from a driver to a receiver so that it arrives within known thresholds, what could possibly go wrong? Time was that engineers could design and layout a product and it would "just work" but now, as device performance has increased, it has become more essential to carefully examine the behaviour of drivers and receivers since previous assumptions and "rules of thumb" no longer provide sufficiently reliable results.

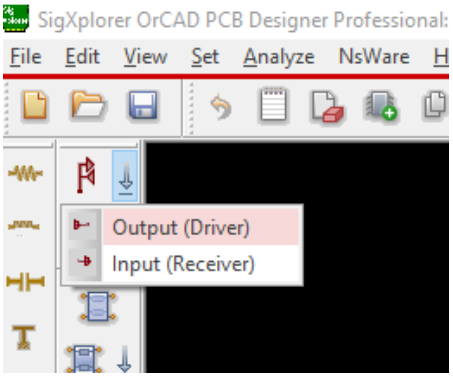
Reflections

Reflections happen when impedances don't match. Relatively, the driver is going to have low impedance, the trace is going to have moderate impedance and the receiver is going to have high impedance, this "simple" mix is likely to result in a reflection from the receiver back to the driver. This may not necessarily be a problem but, since the driver is no longer "in control" of the waveform on the trace, results may not be as reliable as required.

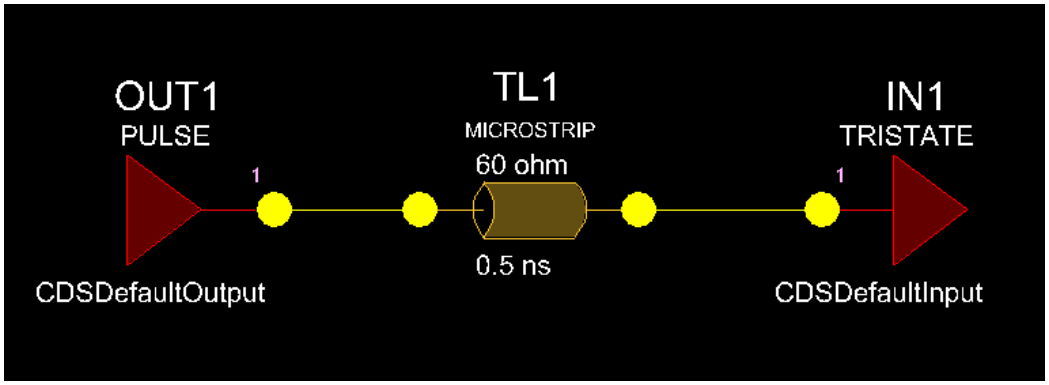
Investigating Reflections

The initial investigations will be for "ideal" components in the simulation environment of Signal Explorer, this can be started from Windows Icon>Cadence PCB Utilities 17.4-2019>SigXplorer 17.4 (you may need to choose a license) this will start with a default file of unamed1.top open, use the File>Save As to save this session to a known name and location. For example, in C:\SPB_Data\SIWork\Reflect.top the TOP file saves the topology of the connectivity and simulation settings.

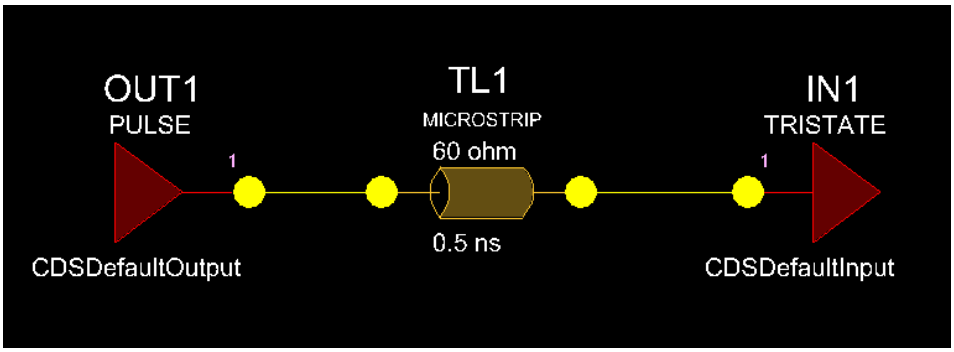
Use the "Single Ended buffer" toolbar icon, select Output (driver) from the list and place the part on the canvas.



Use the Transmission Line icon on the left to place a piece of Microstrip and then add an Input (Receiver)

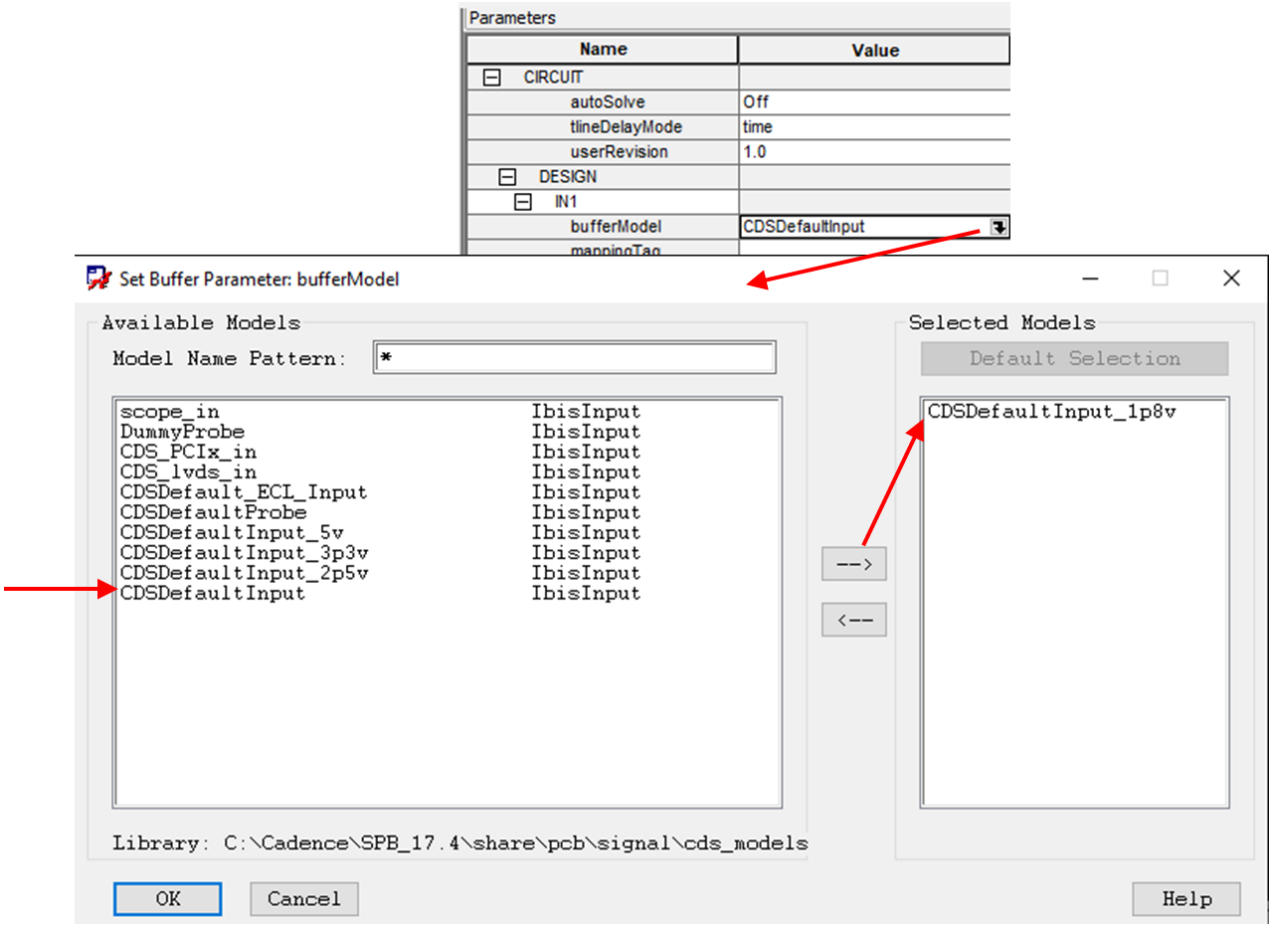


The pins can be “dropped” together to make a connection, use Edit>Cleanup to arrange the canvas items automatically.

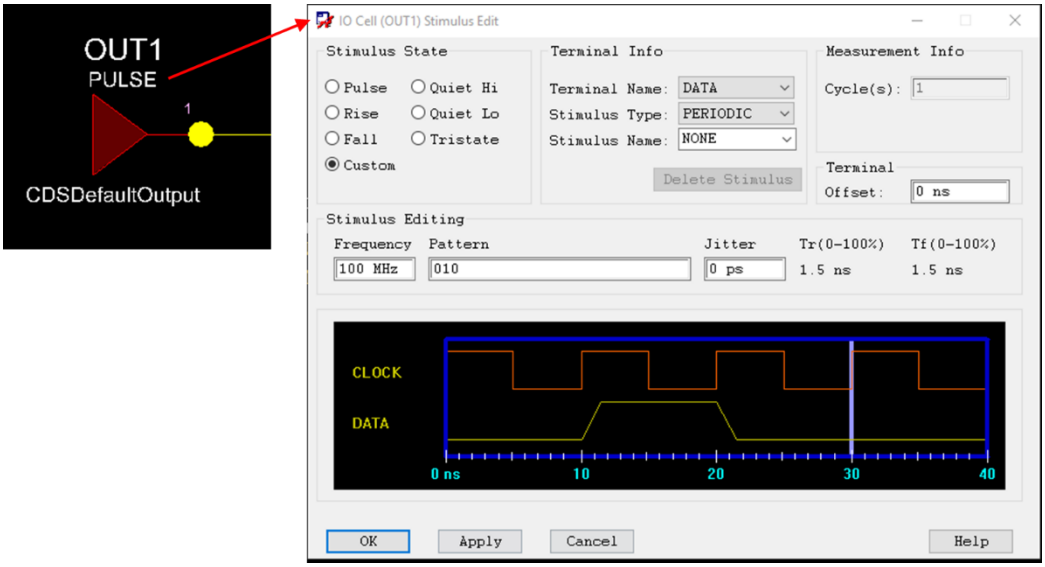


Now the basic driver and receiver are in place, the configuration can be changed. For example, the driver and receiver could be changed to CDSDefault_1p8v and the Transmission Line units changed to length, rather than delay. The Driver and Receiver types get changed from the Parameters panel on the right, left-click on the “+” entries to open the settings for each section. Select the Time delay mode in Circuit and select “length” from the drop-down. Select the driver / receiver, bufferModel entries and the “more” icon appears at the end of the entry, left-click on this to open a list of available parts and change to the 1p8v models.

See the following page for the screenshots.

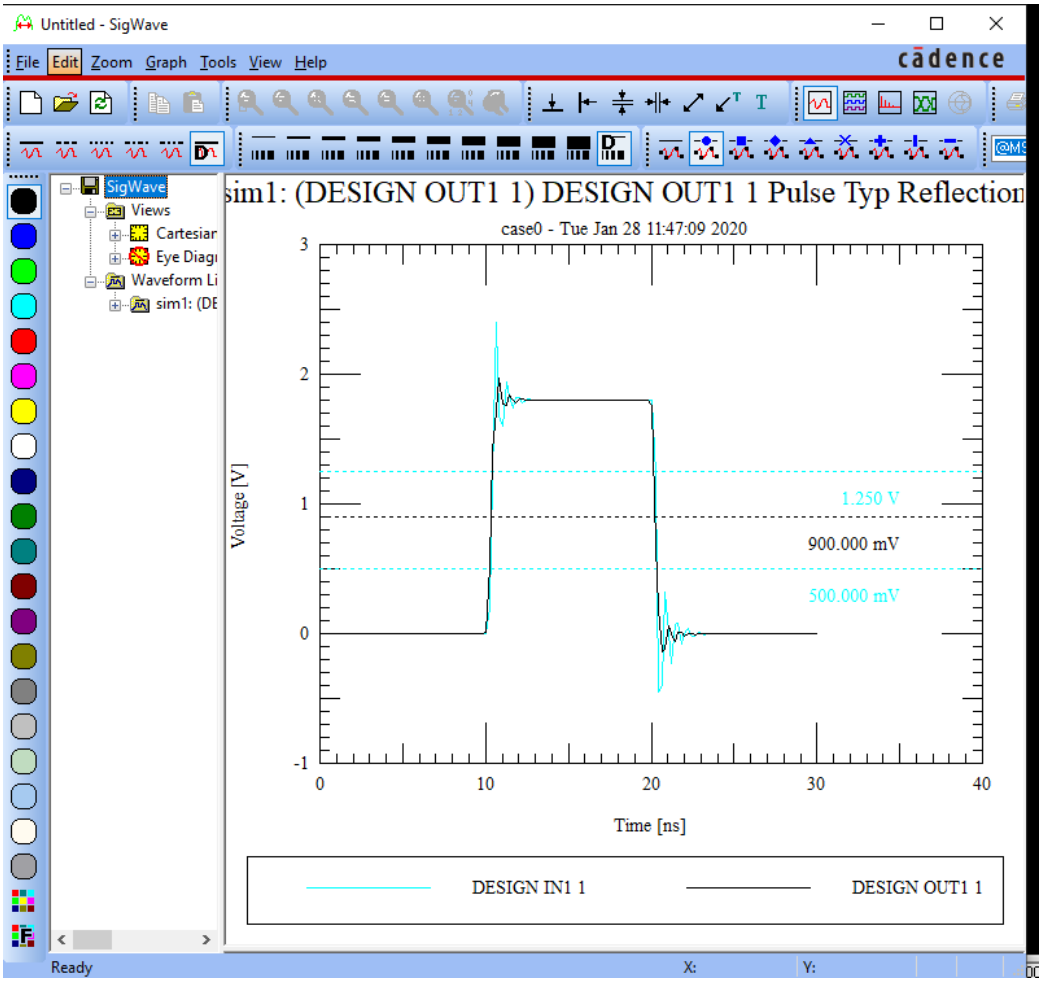


Change the Stimulus to Custom, left-click on the “Pulse” value for the driver and the Stimulus Edit window will open for the driver, change the Stimulus State to Custom, the frequency to 100MHz and the pattern to “010”, close the window after making the changes.



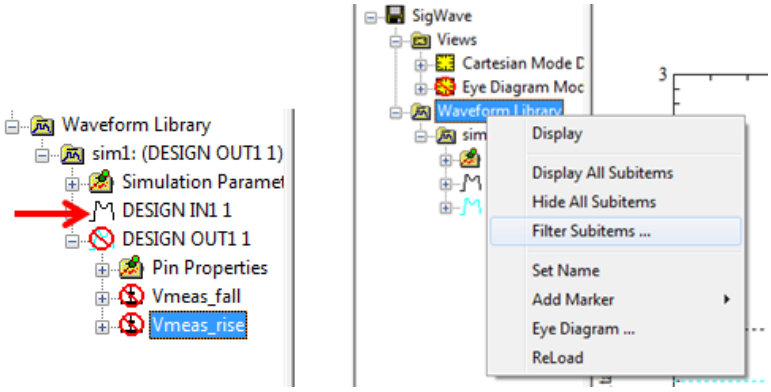
For the first simulation, locate the Impedance and Length entries for TL1, open the Circuit "+" entries, if required to see them, change the Impedance to 50 Ohms and the Length to 500 Mil (thou). (Double-click in the value entry to enable editing and highlight the entire entry with the mouse to replace the value)

Once the values have been set, use Analyze>Simulate from the menu to launch and run the simulation. The SigWave window will open with the simulation waveforms displayed – input and output waveforms in this case – there is some ringing at the receiver and, some reflected to the output. Although the ringing does go below the device thresholds, it does consume about 60% of the available margin, so this could be worth investigating.

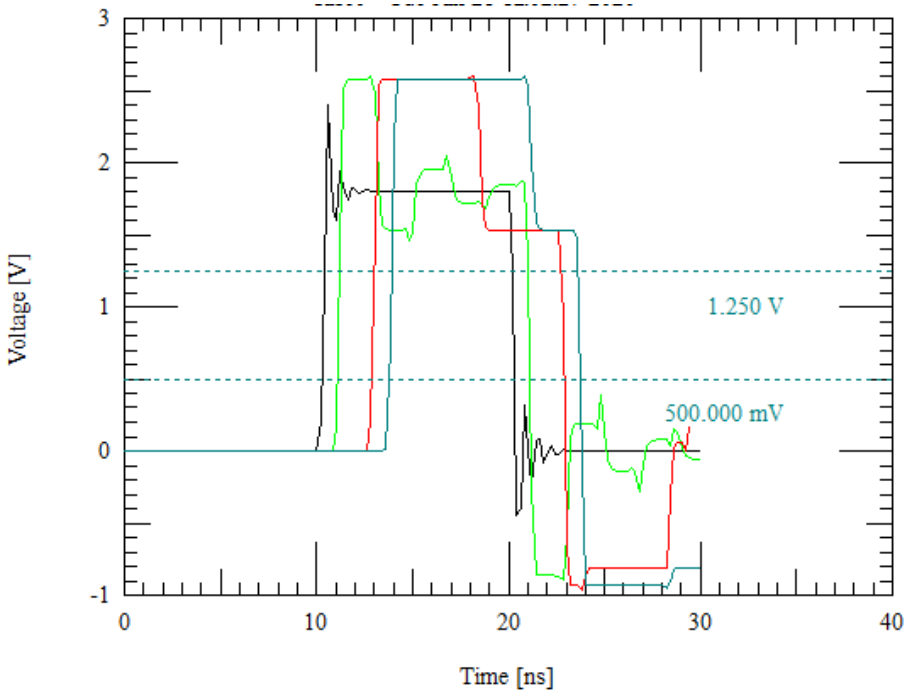


Open the Waveform Library "+" and double-click on a waveform entry, see arrow in the screenshot, to hide the waveform in the display, in this case, hide the Output waveform to leave just the input waveform displayed.

(The right-click>Filter Subitems can also be used to add or remove items from the waveform display)

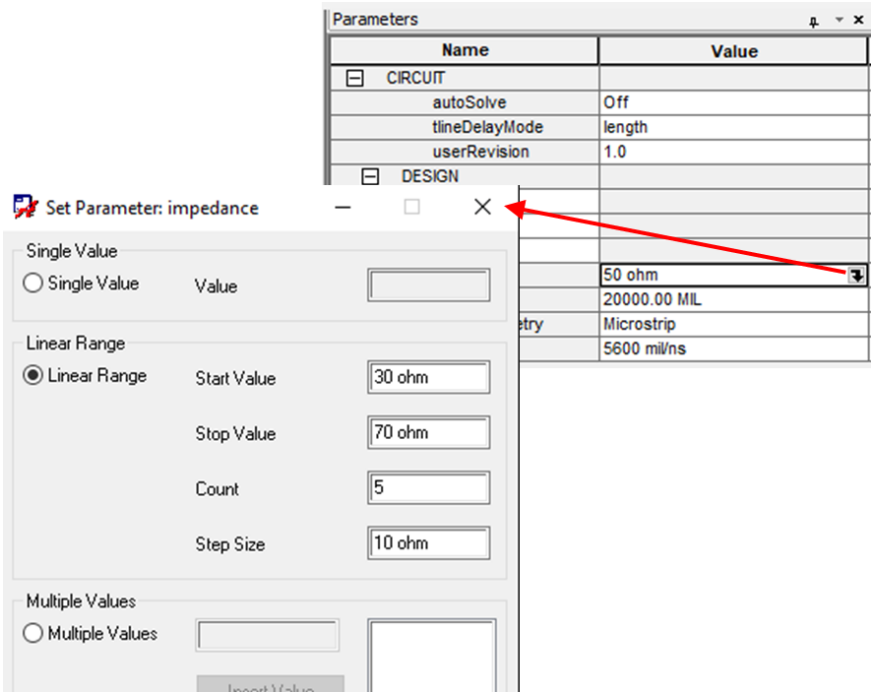


The SigWave window keeps a "Waveform Library", in this case, the SigWave Window will be minimised, the circuit parameters changed and the resulting waveforms compared. Leave the SigWave window and go back to the SigXplorer window and change the Length parameter to 5000 Mils and use Analyze>Simulate to run the simulation again. Repeat these steps for lengths of 15000 and 20000 Mils. (Note that the waveforms displayed in SigWave will automatically change to the last simulation run) Setup the displayed waveforms to get just the receiver waveform displayed for each run.



The green trace above, for the 5000 Mils case, shows the signal being repeatedly "bounced" from end to end of the trace. This will be the setting used for the next section to investigate the effects of changing the impedance of the microstrip on the reflections.

IF you are running SigXplorer from a PCB SI license, OrCAD PCB SI and above, return to the SigXplorer window and set the microstrip length to 5000 Mils. Then click in the Impedance Value for the microstrip and use the "arrow" to open the Set Parameter: Impedance window, set the Linear Range, start at 30, end at 70, a count of 5 for steps of 10 Ohms.



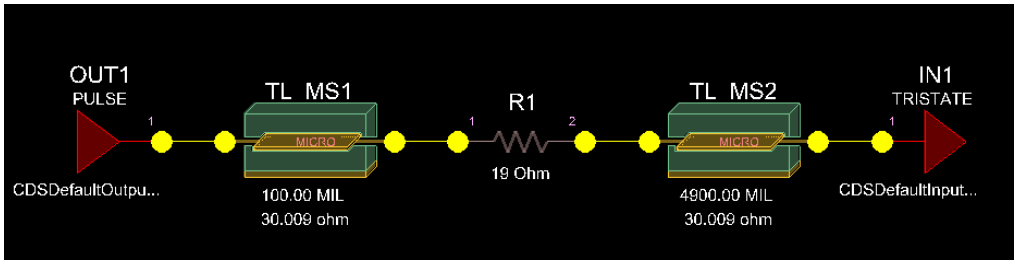
(IF SigXplorer is being run from an OrCAD PCB Designer Professional license, Sweep Simulations will not be available)

Check that the SigWave window is closed, then use the Analyze>Simulate to run the simulations, accept the "Continue" to run the 5 simulations; this is a guard against accidental settings that may cause numerous simulations to run potentially "taking over" the machine. After the simulations have completed a Results window will open in the SigXplorer window, this can be used to compare the results. In this case see that the lower impedance results in greater noise margins and less overshoot.

lode	TL1.impedance	Glitch	Monotonic	NoiseMargin	OvershootHigh	OvershootLow	Pr
	30	PASS	PASS	265.339	2428.82	-843.37	0.8
	40	PASS	PASS	144.942	2556.13	-867.397	0.8
	50	PASS	PASS	106.045	2602.21	-896.514	0.8
	60	PASS	PASS	133.36	2616.92	-910.89	0.8
	70	PASS	PASS	86.4	2623.02	-915.214	0.8

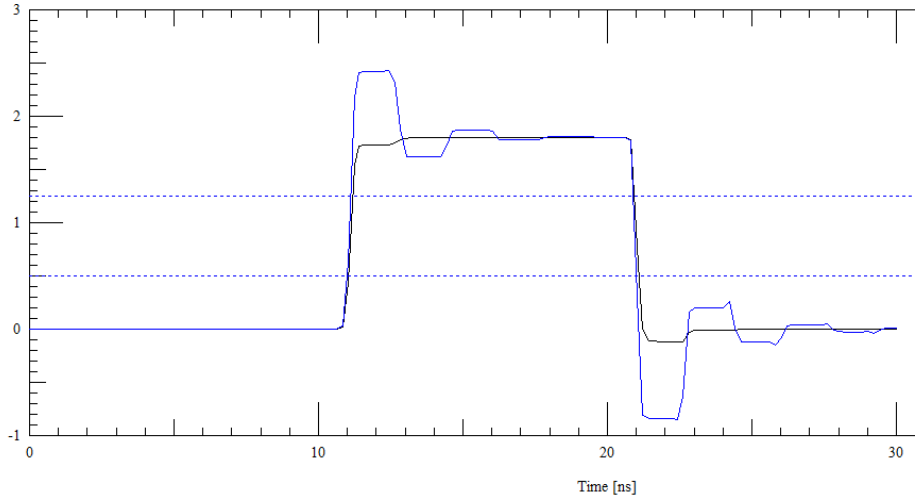
Termination and Stack-up

In this section, a new topology will be created with the driver and receiver as before but with two sections of microstrip and a terminating resistor between them.



Change the resistor value to 19 Ohms and set the parameters of the microstrip sections to be d1Thickness=3.05 Mils, traceWidth=12 Mils, length for TL_MS1 to 100 Mils and TL_MS2 to 4900 Mils. This will clear the Impedance display in the canvas, to restore this, select the TL_MS parts in turn and right-click>View Trace Parameters to update the Impedance values, they should both be 30 Ohms. The output impedance of the driver is about 11

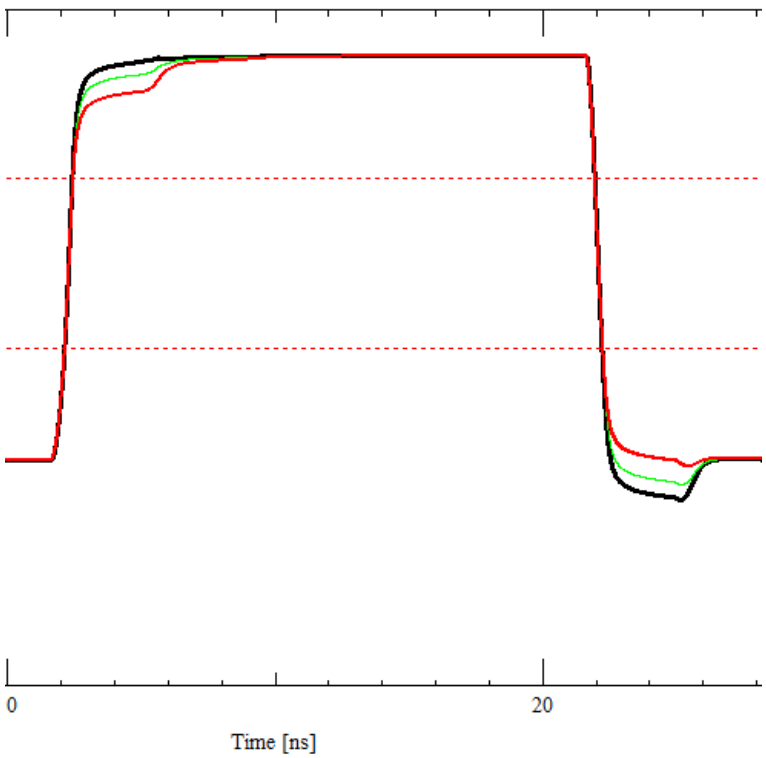
Ohms, with the 19 Ohm series termination, the 30 Ohm transmission line will be matched and there will be no overshoot on the resulting waveform. Setup the Custom stimulus as before for 100 MHz, 010 pattern and run the simulation. Return to the SigXplorer window, don't close the SigWave window, and change the resistor value to 0, effectively removing it from the circuit, then run the simulation again. Enable the receiver traces for both simulations and observe the effect of the terminating resistor.



(The blue trace has the 0 Ohm terminator)

If you are using a PCB SI license, OrCAD PCB SI and above, you can try sweeping the value of the terminating resistor from 5 to 25 Ohms in 5 Ohm steps and examine the Measurements.

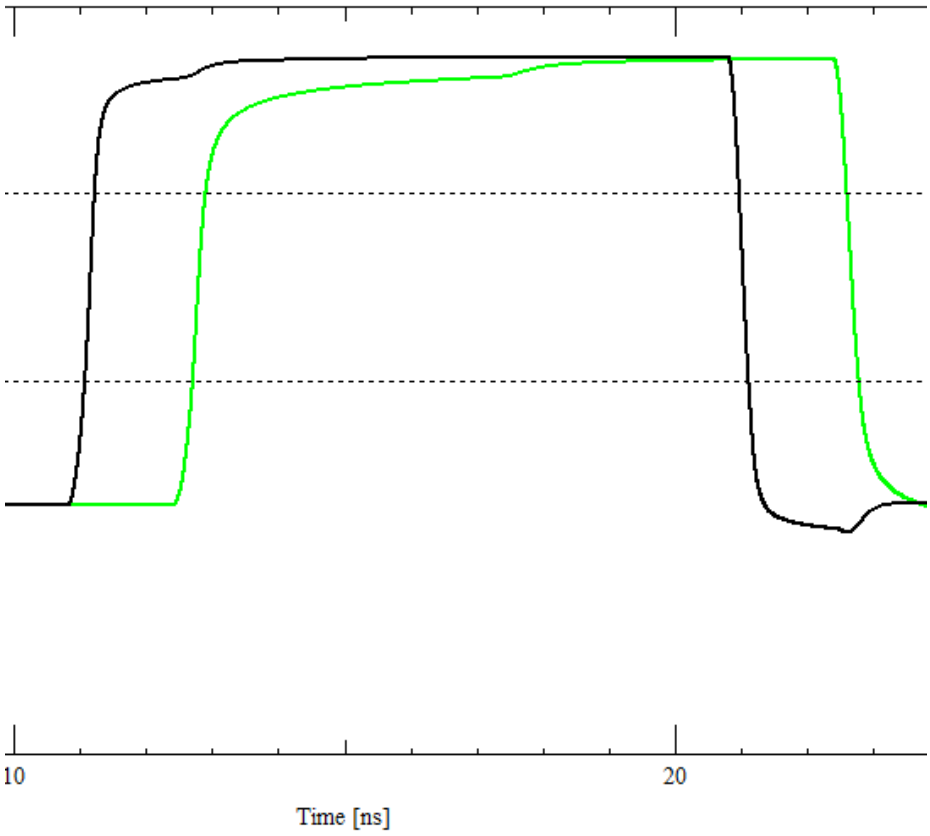
To further investigate this, close SigWave, if it is running, and run some more simulations. With an OrCAD PCB SI license, use the Multiple Values option for the resistor value, add values 17, 19 and 22 Ohms and then run the simulations. With an OrCAD PCB Designer Professional license, run three simulations for resistor values of 17, 19, and 22 Ohms. Set the SigWave display to view the three waveforms at the receiver.



The Black trace is the 17 Ohms case and the Red trace is the 22 Ohms case. See that the smaller terminator value has a “clean” rising edge but has some amount of negative overshoot on the falling edge, whereas the larger value has removed the negative overshoot, the rising edge is now starting to show some undershoot. Like many things in engineering, the final solution is likely to be a compromise based upon the availability of “real” components values for the terminating resistor.

Let’s verify that our termination solution is improving our design. Set the Termination value to 19 Ohms and change the length parameter value for TL_MS2 to 14900 Mils (15000 Mils in total) and re-run the simulation.

See that the Green trace in the next screenshot, the longer trace case, still exhibits some decent control over the signal quality so the solution is “largely immune” to the trace length.

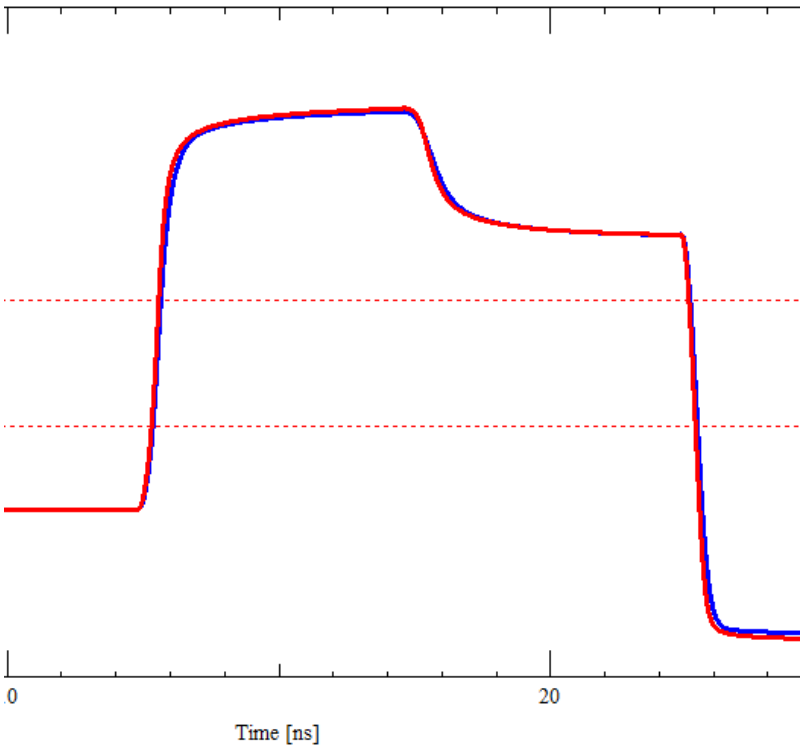


So far, the driver and termination resistor are close together, is this the “better” way? What happens if the termination is moved to the “receiver end”?

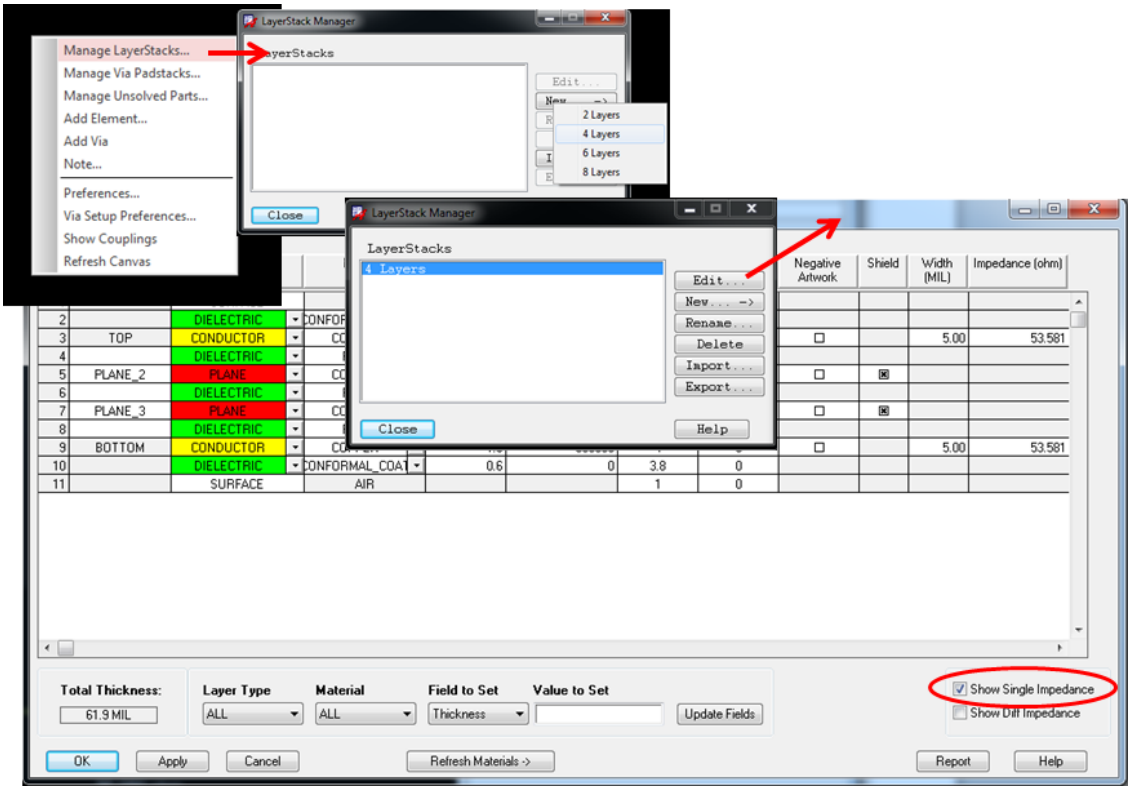
Close SigWave and edit the traceLength parameters for the TL_MS parts so that TL_MS1 is 14900 Mils and TL_MS2 is 100 Mils, then re-run the simulation.

See that the waveforms for 0 Ohm termination (red) and the “receiver end” termination (blue) in the next screenshot are virtually inseparable.

So, the driver and terminator close together act “as one”, the “receiver end”, series termination, looks more like a waste of board space!



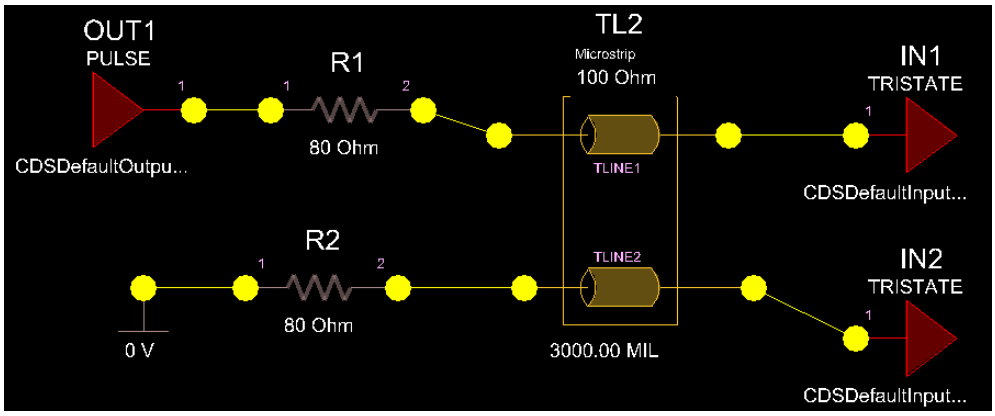
To get the 30 Ohm impedance required some quite extreme parameters, a trace width of 12 Mils and a dielectric thickness of 3 Mils. The next stage is to look at how the layer stackup might affect things. The important thing about the impedances is that they are matched, the actual values are less important. Close SigWave and, back in SigXplorer, right-click in the background canvas and select Manage LayerStacks, select New and 4 layers, select the 4 Layers entry created and use Edit to review the stackup. Check the Show Single Impedance to see what the impedance of the 5 mil trace will be, 53 Ohms in this case.



Close the LayerStack Manager dialog. Go to the TL_MS parts in turn, select them and right-click>Change Layer and select TOP as the layer. Set the traceWidth values to 5, also note that many of the “generic” parameters have now been removed when a specific layer is used, and right-click>View Trace Parameters on each of the TL_MS parts to get the trace impedance updated. Since the trace impedance is now about 53 Ohms, the termination should be 42 Ohms. Also verify that the Driver to termination trace length is 100 Mils, and that the termination to receiver trace length is set to 4900 Mils, then re-run the simulation. The resulting waveform should demonstrate that matched impedance gives good results for signal quality, indicating the importance of matching the impedances, rather than the actual value of the impedances in the connection.

Crosstalk

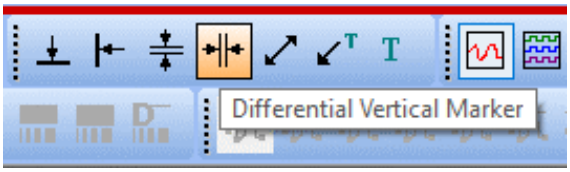
So far connections have been considered in isolation, not a very likely scenario in a real design, so the influence of the activity of one connection on the signal quality of another needs to be considered. Start SigXplorer and draw a new topology, use the 1p8v driver as before, add two receivers, one for the active connection and another for the passive, “victim”, connection. Add a Coupled Transmission Line, resistors and a source to get the topology below. When adding the source, the value will be 5V, set this to 0 and the canvas will be updated accordingly. Edit the Coupled Transmission Line parameters and set the Impedance to 80 Ohms, note that the differential Impedance is the displayed value.



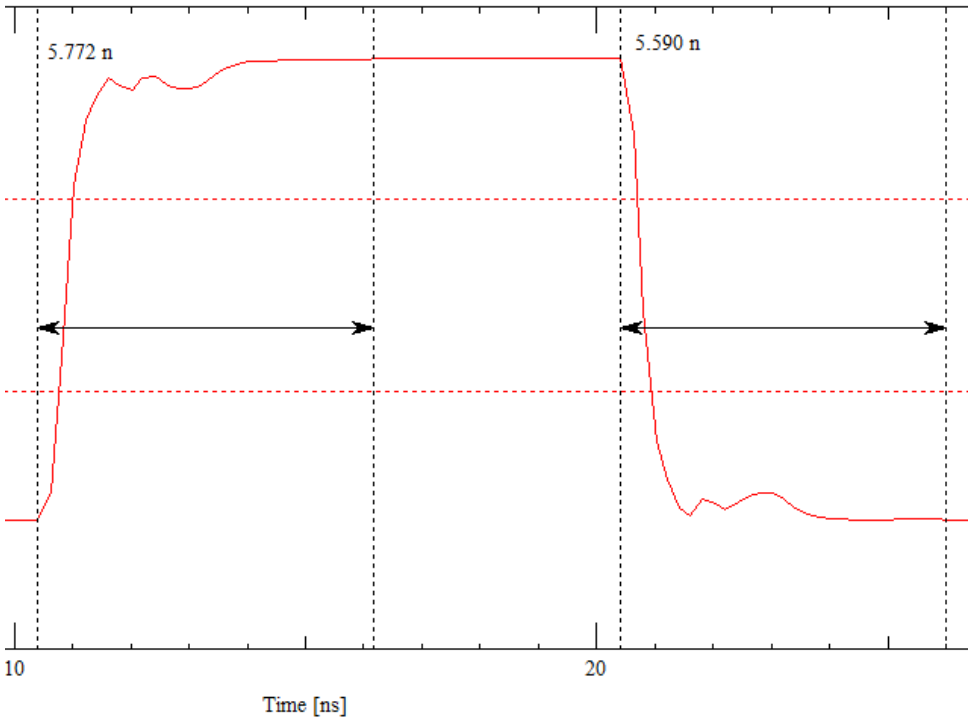
Parameters

Name	Value
CIRCUIT	
autoSolve	Off
tlineDelayMode	length
userRevision	1.0
DESIGN	
IN1	
bufferModel	CDSDefaultInput_1p8v
mappingTag	
IN2	
bufferModel	CDSDefaultInput_1p8v
mappingTag	
OUT1	
bufferModel	CDSDefaultOutput_1p8v
mappingTag	
R1	
resistance	80 Ohm
R2	
resistance	80 Ohm
TL1	
diffImpedance	100 Ohm
diffVelocity	5600 mil/ns
impedance	60 ohm
length	3000.00 MIL
traceGeometry	Microstrip
velocity	5600 mil/ns
V1	
voltage	0 V

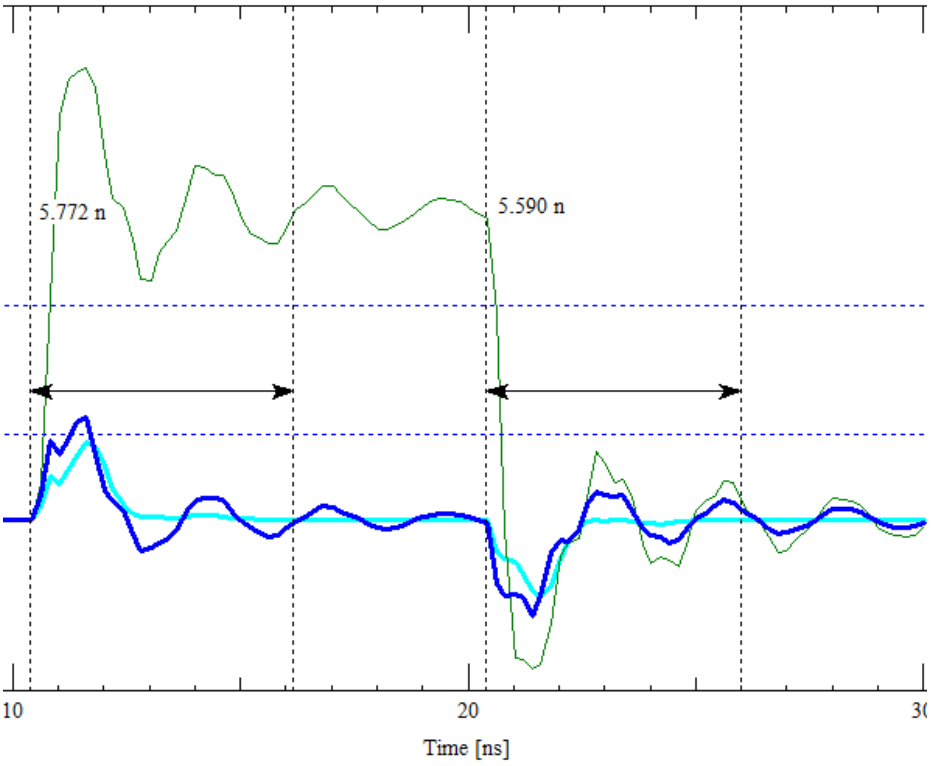
Select the IN1 and IN2 names in sequence and rename IN1 to Receiver and IN2 to Victim; this will make identifying the waveforms a little easier. (The names of IN1 and IN2 will work correctly, if preferred) Change the PULSE source to CUSTOM and 100 MHz and a pattern of 010, as in the previous examples and save the topology, then re-run the simulation. In SigWave, set the display to show the active receiver waveform only, then click in the waveform display to activate the Zoom buttons and select Zoom>Specific Size (there is also a toolbar icon for this), set the Minimum X to 0, the Minimum Y to -0.5V, the Maximum X to 30ns and the Maximum Y to 2.0V, then OK to accept the changes. Now add a Differential Vertical Marker from the toolbar.



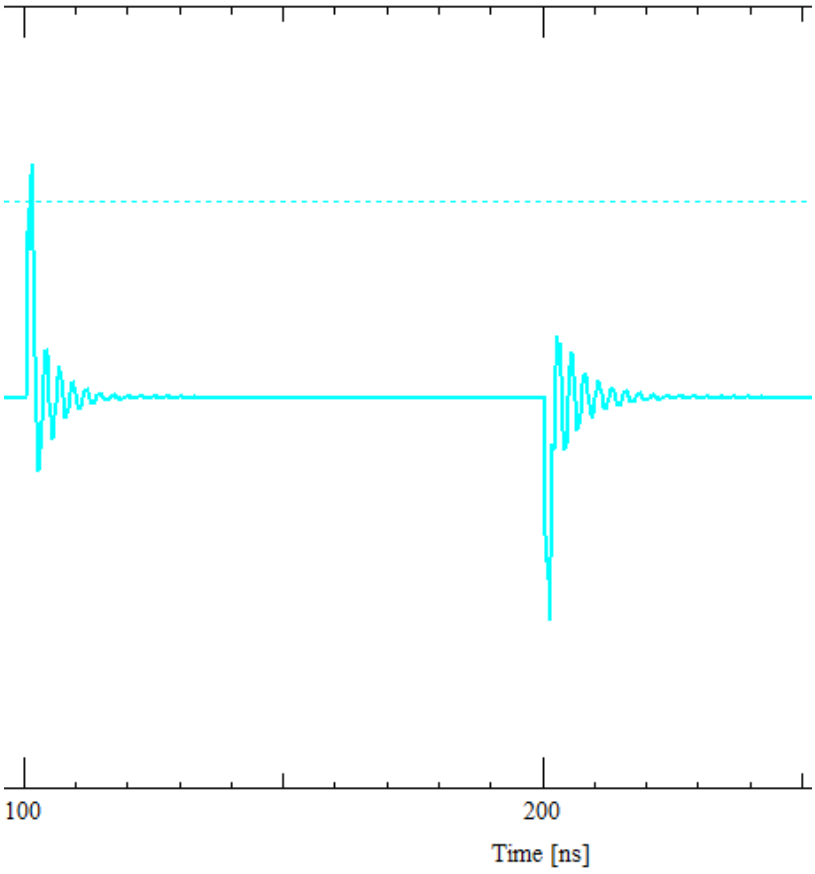
Place the first Differential Vertical marker between the start and end of the rising transition and a second between the start and end of the falling edge. The marker will be added to the waveform, click on one of the vertical dotted lines and hover to get the “double arrow” cursor, then drag the vertical dotted lines to the required locations.



Now hide the Receiver waveform and display only the Victim waveform. Leave SigWave open, go back to SigXplorer and remove the termination for the driver connection (set the termination value to 0) and re-run the simulation. Set the waveform in SigWave to display the Victim in each case. Use the Line Color and Line Width to tweak the displayed waveforms to improve visibility.



Note that the “un-terminated” driver gets enough energy at the receiver to cross the Vil threshold! Just as a reminder of what could possibly go wrong, change the custom stimulus to use a 10MHz frequency and re-run the simulation.



Few systems would be analysed for such a low frequency signal but just look at the excessive ringing in the victim, almost 30% of the signal voltage. Also note that additional transitions from other adjacent connections would only compound the misery!

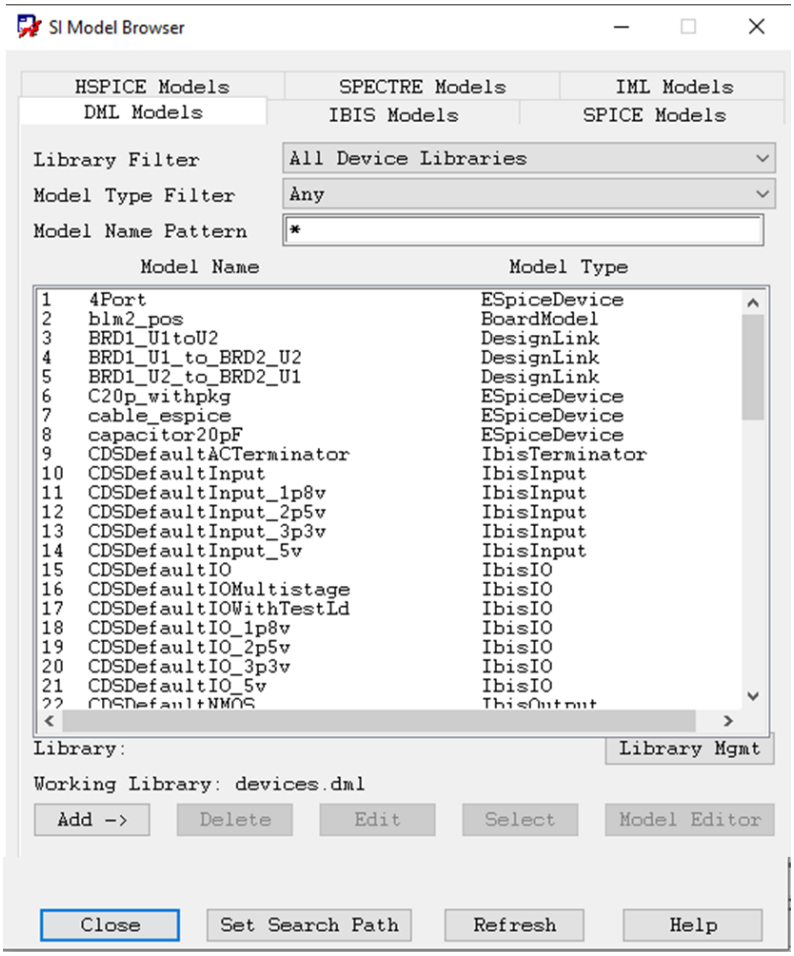
A slight liberty has been taken with this scenario since the Victim terminator is set to a 0V source, a more realistic case would be where the Victim Driver were in a stable Low, or High state. Close SigWave and return to SigXplorer. Change the Custom stimulus back to 100MHz and restore the driver terminator to 80 Ohms. Delete the 0V source and add a driver for the Victim connection, change the driver type to the 1p8v device used for the other connection. The initial stimulus for the victim driver will be Tristate, change this to Quiet_Lo. Hover over the Measurements tab at the bottom of the SigXplorer window and change the selection from Reflection to Crosstalk, and then run the simulation.

"Making it Real"

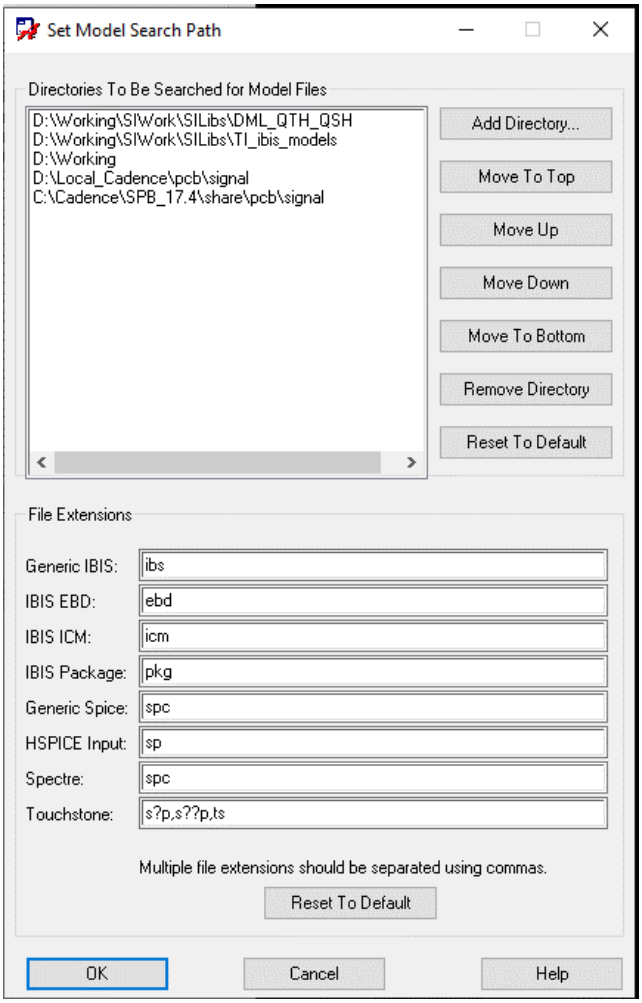
So far, the simulations have used "generic" items for the driver and receiver models in the simulation, that may be OK for preliminary investigation but it's unlikely that any components you use in a real design will have the Cadence default models on the pins, how to get some "real" device details into the simulation? The good news is that device vendors offer lots of support for modelling their components for Signal Integrity analysis. Probably the most common are the IBIS, Input / output Buffer Interface Specification; format models provided by IC vendors, there are also SPICE and the Cadence specific DML models available. For the purposes of demonstration, some Texas Instruments IBIS models and some Samtec DML connector models will be used.

Adding the vendor SI models to SigXPlorer

Start SigXPlorer from any license, then use Analyze>Libraries Browser to get to the library configuration.

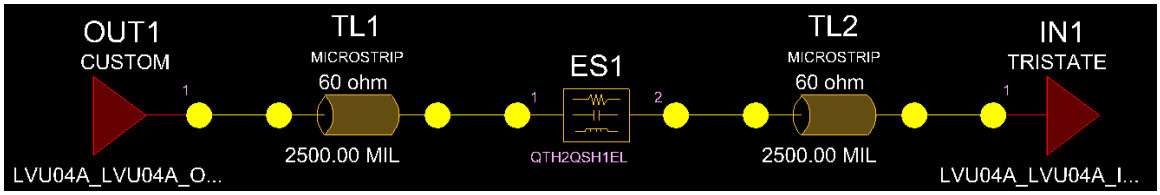


Note that the models are distributed by category and visible through the tabs. Use the Set Search Path to open the settings for the library search path(s), this example has the libraries stored in an SILibs directory but “any location” could be used. Left-click>OK and use Refresh to update the list of models.



In this specific case, there are some QTH parts added from the Samtec connector DML models and two IBIS models added from the Texas Instruments models, these are just some extracted samples; there are very many models available from these, and other, vendors! For the IBIS models, go to the IBIS models tab, highlight the ibis model entry and use the Select button to activate the models, create a unique name for each model when prompted. The selected IBIS models will then be available for simulations in SigXplorer but the “Translate” button will need to be used to create the DML models to make the IO models available for selection in other environments, like PCB Editor. Add a driver, receiver and some interconnect, set the interconnect length to 5000 Mils, as in the first reflection case of this note. Now the Driver and Receiver models can be changed to some “real” ones, select the driver and receiver in turn and change the bufferModel to use some imported models, then run the simulation and examine the resulting waveform. The selected IBIS models may have different voltage and drive level models provided, the bufferModel can be changed to see the effects that changing these characteristics have on the simulation results.

For the case of the provided connector DML models, these need to be added to the canvas using right-click>Add Element, the element type and specific element can then be filtered from the available items and added to the canvas. In this specific example, a connector has been added between two sections of interconnect.



The results look like some series termination is required but the simulation runs correctly with the new models. The previous simulation examples could be repeated with the "real" bufferModels in place, if required.

Extracting data from the PCB for simulation

For this section a very simple schematic of 4 blocks of inverter, resistor, and inverter will be created with the drivers in one package and the receivers in another, the sample soic14 and smdres footprints will be used for physical packaging. When extracted to SigXplorer, the resulting topology will resemble the driver, interconnect, terminator, interconnect, receiver network of previous examples.

Once the netlist is loaded, the parts are placed in the PCB Editor canvas, the exact placement is not critical but the unrouted lengths will be extracted to the topology file. Currently, there are no SI related properties assigned so these need to be assigned in PCB Editor. (A later section will discuss SI and Schematic Capture) Use Setup>More>SI Design Setup to prepare the design for SI analysis, leave all the setup categories selected and start the wizard with Next. Take all the signal nets since there are not many and exclude any Power nets. (If a "real" design is used, restrict the nets to the ones of interest). Check the library configuration, change the model directories, if required to include any specific models required. Assign the voltages to the power nets in the design, select the Net Name and "Assign Voltage to Selected Net". There may be some non-power pins assigned to the Power Nets, use the "All" button for Resolve to get these errors resolved. "OK" the resolution or ignore any further errors for now. Manually edit the Cross-section, if required, a cross-section could be imported from another BRD, or Technology, file. Set the component classes, if required, especially check that any discrete devices are not incorrectly set to IC, this step can be used to force discrete components to the correct class. If there is no model for a device, select the Device and use Create New Model for the IC parts, as required, the Discrete parts can use the Create Default Model option to create the model(s). The resistor to inverter nets have no driver directly connected, accept this as OK since the nets will be driven through the terminator, also ignore that the Power net may have nothing driving it, this will obviously matter in practice but the simulation will work regardless. Take Reflection as the selected simulation type and "tlsim" as the simulator. The SI Audit will show the "un-driven" nets, if any. Finish the wizard and save the BRD file, this can be given a different name to reflect the SI setup if required.

Pre-route

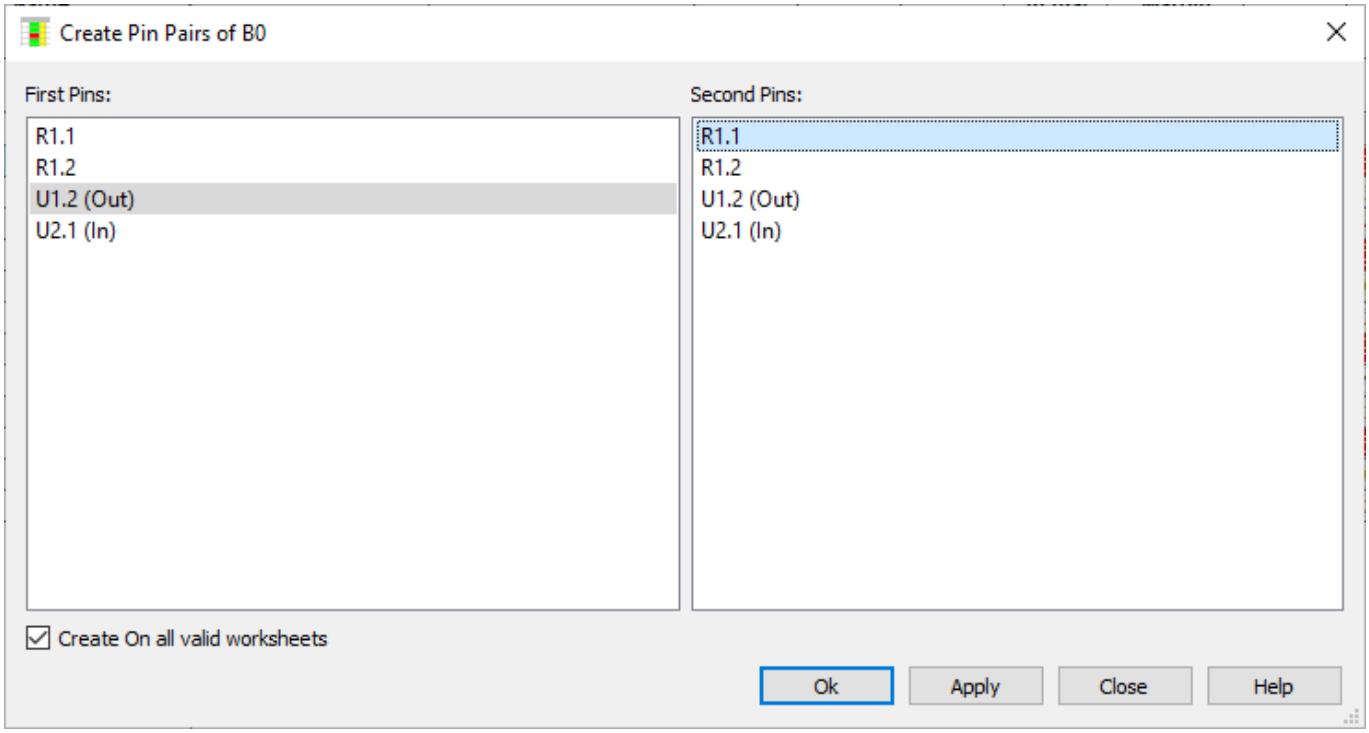
Use one of these methods to get the Topology extracted to SigXplorer:

Either, Tools>Topology Extract (at least an OrCAD PCB Designer Professional license) to get the Nets listed for extraction, select the net (Xnet) in the list and use Save As to save the Topology file, this can then be opened in SigXplorer. This method can be used to extract the un-routed, pre-route, topology, or the routed, post route, topology if the net has been routed.

Or, go to Constraint Manager, Electrical Domain, select Total Etch Length and a Net and right-click>SigXplorer, and SigXplorer will start with the extracted topology. This method will always extract the un-routed, pre-route topology unless the net is routed and, at least, an OrCAD PCB Designer Professional license is being used to run PCB Editor and in Constraint Manager Tools>Options and "Extract for simulation" and "Include routed interconnect" are enabled.

Once the topology has been opened in SigXplorer, by using either method, the network can be simulated as before. Change the Driver and Receiver bufferModel properties to use the “real” models rather than the CDS Defaults. (See Appendix A at the end of this document for a discussion of using the SI Model Library tools to assign “real” IO buffer models to created models) Simulate the network as in the previous examples to determine the driver / terminator length, the terminator receiver length and the terminator value. Remember from the previous simulations that the Driver / Terminator length needs to be “quite short” for the terminator to be effective. Once the length requirements have been determined from simulation, they need to be set in the Constraint Manager so that the PCB Editor DRC system can check the required rules against the actual lengths in the design.

As an example, to “fill in the numbers”, the sample design has a total Manhattan Length of about 2700 Mils for the driver to receiver, the driver / terminator length is set to 100 Mils, the terminator / receiver is set to 2700 Mils and the terminator value set to 33 Ohms. The terminator value will need to be changed in the schematic, but the rules can be applied in Constraint Manager before back annotating the changes to the schematic. In PCB Editor, open Constraint Manager, go to the Electrical Domain, Net, Routing, Min / Max Propagation Delay. Select each of the “B” XNets in turn and right-click>Create Pin Pair. Select the “Out” and pin 1 of the terminator.



Left-click>OK to create the Pin Pair. OK the message about unused Pin Pairs. Repeat the Create Pin Pair for the “In” Pin and Pin 2 of the terminator. There isn’t any “templating” of the topology or rules without a PCB SI license, or Allegro High-Speed Option so all the Pin Pairs need to be created individually for the 4 nets.

Once the Pin Pairs are created, the constraints can be applied, set Driver / Terminator to 110 / 120 Mils and Terminator / Receiver to 2650 / 2750 Mils, a total of 8 Pin Pairs. The Placement can be evaluated by enabling the DRC Unrouted in Analyze>Analysis Mode, Electrical. This will then use the Manhattan, orthogonal, route between components to evaluate if the constraint is met, or not. (When routed the routing may be able to “beat” the Manhattan route but it is a useful starting point) Once the rules are analysed, the placement can be revised, if

necessary. Note that the 10 Mils allowance for placement may make the placement somewhat more difficult than it might be.

Objects			Pin Delay		Prop Delay			Prop Delay		
Name	Referenced Electrical CSet	Pin Pairs	Pin 1	Pin 2	Min	Actual	Margin	Max	Actual	Margin
			mil	mil	mil			mil		
*	*	*	*	*	*	*	*	*	*	*
TEST							-542.5 mil			2.5 mil
B0							-152.5 mil			2.5 mil
R1.2:U2.1					2650 mil	2497.5 mil	-152.5 mil	2750 mil	2497.5 mil	252.5 mil
U1.2:R1.1					110 mil	117.5 mil	7.5 mil	120 mil	117.5 mil	2.5 mil
B1							-57.5 mil			7.5 mil
R2.2:U2.3					2650 mil	2592.5 mil	-57.5 mil	2750 mil	2592.5 mil	157.5 mil
U1.4:R2.1					110 mil	112.5 mil	2.5 mil	120 mil	112.5 mil	7.5 mil
B2							-57.5 mil			7.5 mil
R3.2:U2.5					2650 mil	2592.5 mil	-57.5 mil	2750 mil	2592.5 mil	157.5 mil
U1.6:R3.1					110 mil	112.5 mil	2.5 mil	120 mil	112.5 mil	7.5 mil
B3							-542.5 mil			7.5 mil
R4.2:U2.9					2650 mil	2107.5 mil	-542.5 mil	2750 mil	2107.5 mil	642.5 mil
U1.8:R4.1					110 mil	112.5 mil	2.5 mil	120 mil	112.5 mil	7.5 mil

Note that, in this case, the Terminator / Receiver lengths all have a negative margin indicating that some delay might be required in addition to the Manhattan routing and it would be likely that the connections could be routed successfully.

Here is an example screen shot with the routing completed.

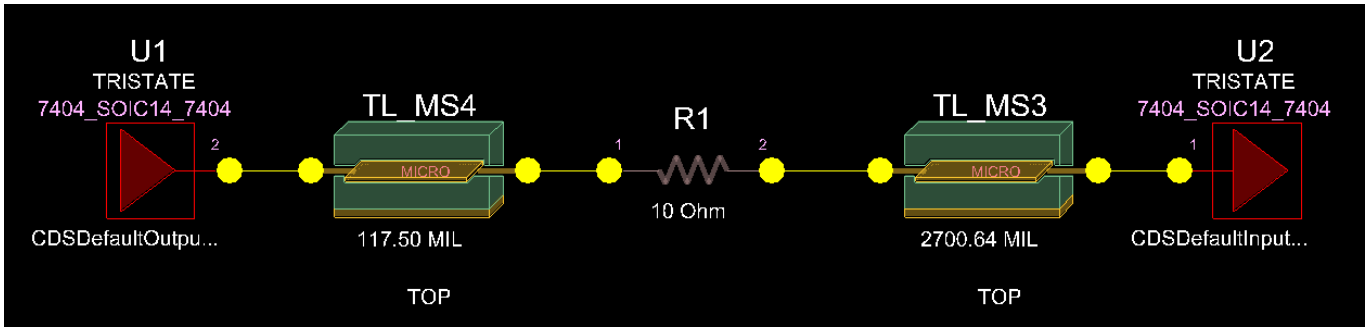
Objects			Pin Delay		Prop Delay			Prop Delay		
Name	Referenced Electrical CSet	Pin Pairs	Pin 1	Pin 2	Min	Actual	Margin	Max	Actual	Margin
			mil	mil	mil			mil		
*	*	*	*	*	*	*	*	*	*	*
TESTSI2							2.5 mil			2.5 mil
B0							7.5 mil			2.5 mil
R1.2:U2.1					2650 mil	2700.64 mil	50.64 mil	2750 mil	2700.64 mil	49.36 mil
U1.2:R1.1					110 mil	117.5 mil	7.5 mil	120 mil	117.5 mil	2.5 mil
B1							2.5 mil			7.5 mil
R2.2:U2.3					2650 mil	2713.21 mil	63.21 mil	2750 mil	2713.21 mil	36.79 mil
U1.4:R2.1					110 mil	112.5 mil	2.5 mil	120 mil	112.5 mil	7.5 mil
B2							2.5 mil			7.5 mil
R3.2:U2.5					2650 mil	2673.21 mil	23.21 mil	2750 mil	2673.21 mil	76.79 mil
U1.6:R3.1					110 mil	112.5 mil	2.5 mil	120 mil	112.5 mil	7.5 mil
B3							2.5 mil			7.5 mil
R4.2:U2.9					2650 mil	2727.86 mil	77.86 mil	2750 mil	2727.86 mil	22.14 mil
U1.8:R4.1					110 mil	112.5 mil	2.5 mil	120 mil	112.5 mil	7.5 mil

Note that for Min / Max Propagation Delays and for Relative Propagation Delay, Pin Delay is considered if the Package Pins have Pin Delay values assigned. This may mean that the actual routing distance within the board may be somewhat less when the delay within the package is considered.

The PCB Editor properties can be back annotated to the schematic, the terminator values changed to reflect the values required from simulation, the netlist re-loaded and the board routed in preparation for Post Route analysis.

Post-Route

In the Constraint Manager, check Tools>Options and ensure that the options for “Extract for Simulation” and “Include Routed Interconnect” are both selected, then select the XNet to extract and the right-click>SigXplorer. SigXplorer will start with the extracted topology displayed. Note the different models that have been applied for the interconnect.

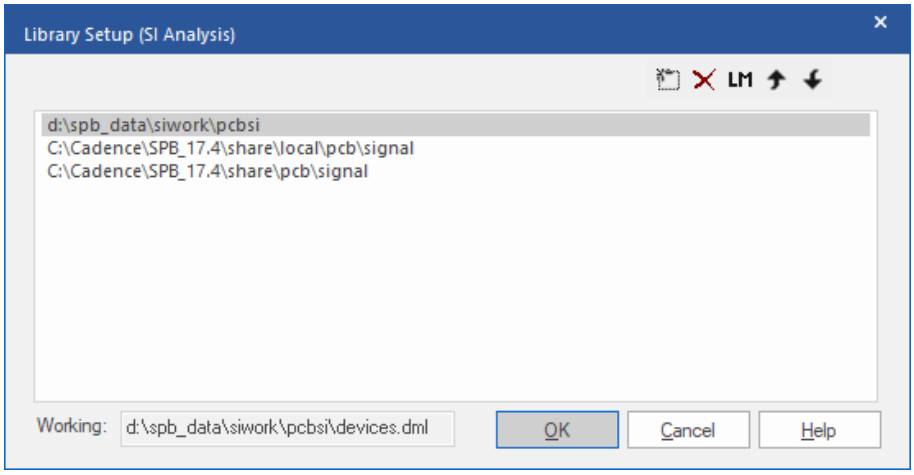


The Stimulus can be applied to the driver and the Simulation can be run to verify that the routed design still meets the Signal integrity requirements for the connection.

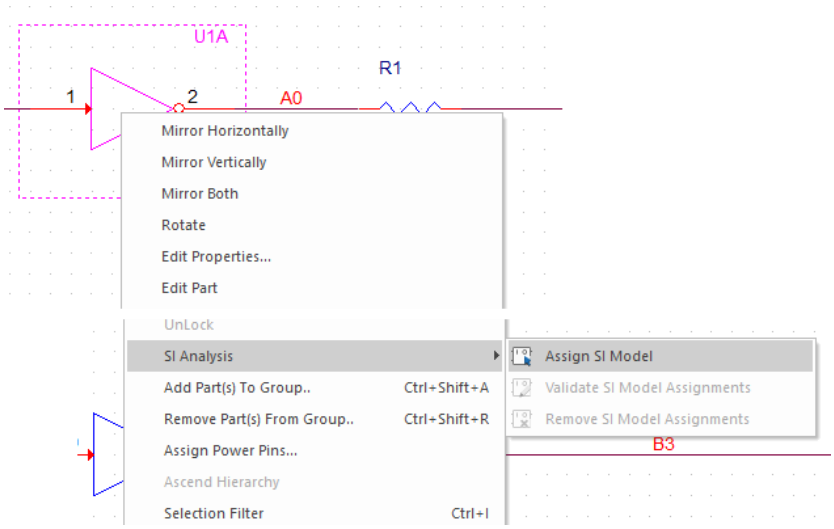
The SI Topology support in OrCAD Capture

Configuring the SI Libraries

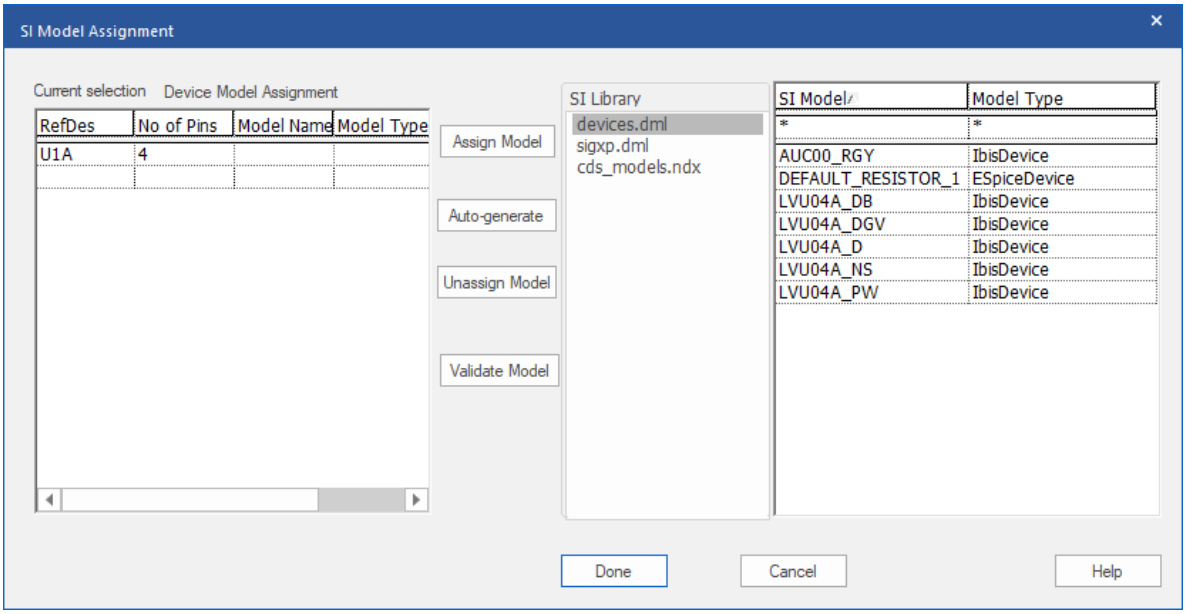
With the design opened, select the DSN file entry in the Project Manager window and then go to SI Analysis>SI Library Setup, use the Add a New Library icon to add the new library location and move this to the Top of the list to find these models in preference to others.



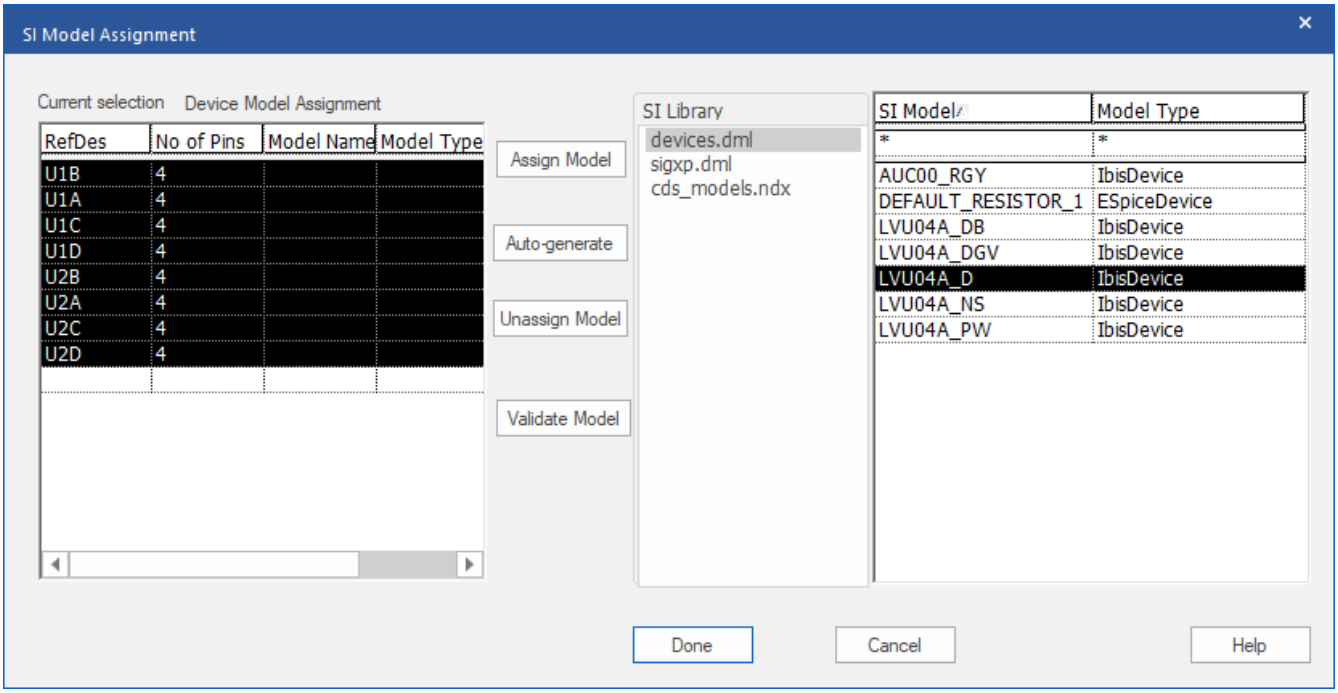
Once the SI Libraries have been configured, the schematic page can be opened, and the part can be selected. Then use right-click>SI Analysis>Assign SI Model.



When the SI Model Assignment dialog opens, select the devices.dml entry from list to get the "04" models listed.



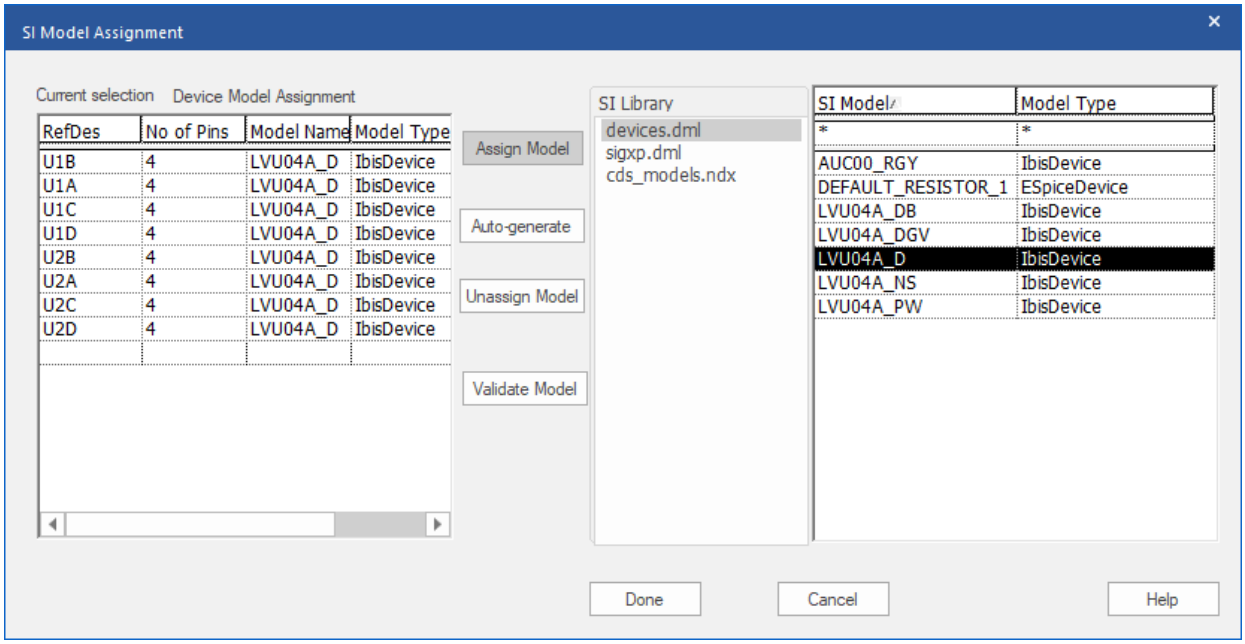
Now the required model can be selected from the list. In this case, the one of the LVU04A models needs to be assigned to the drivers and receivers. In this trivial case the drivers and receivers all use the same model so these can be assigned in a single step if multiple parts are selected.



Use the Assign Mode button once the Signal Model and part(s) are selected.

(Equally, all the parts could be selected in the schematic and models assigned by selecting the models and parts required)

After assignment:

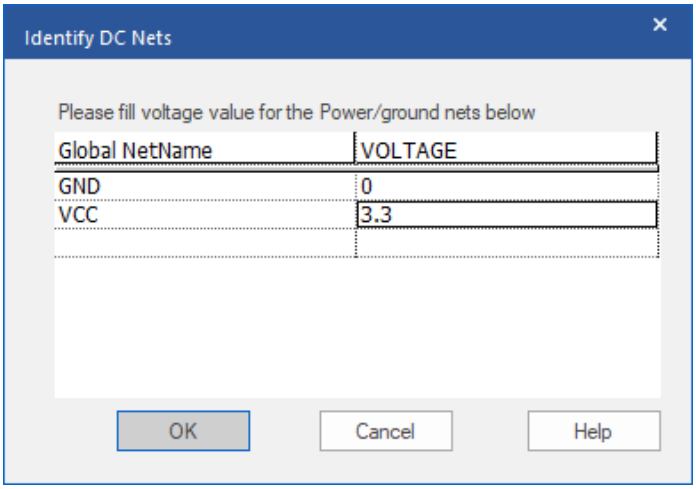


Left-click>Done to exit Model Assignment.

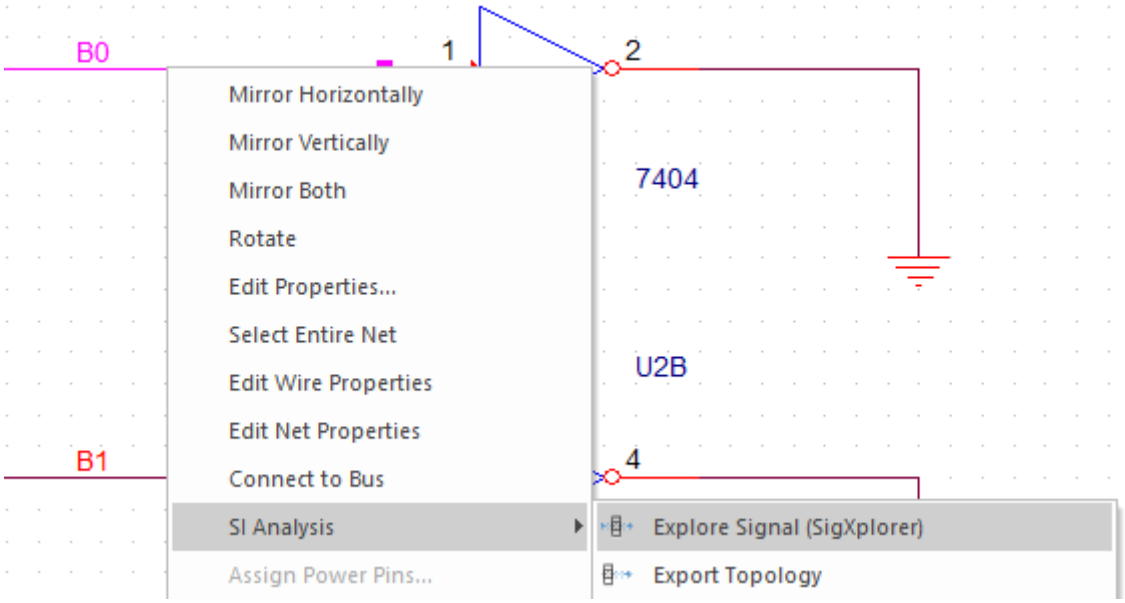
For the discrete devices, the required model could be selected from the configured libraries, if the model already exists, or Auto Assign used to assign default models. Note that discrete devices will need to have values recognised by SPICE to be automatically assigned. If the component values use Part Numbers, rather than circuit values, the models will need to be assigned from the existing Signal Models. (The Discrete Models could also be

assigned from the Project Manager Window, select the DSN file and use SI Analysis>Auto Assign Discrete SI Models, again the parts will need to have values recognised by SPICE to be automatically assigned Signal Models)

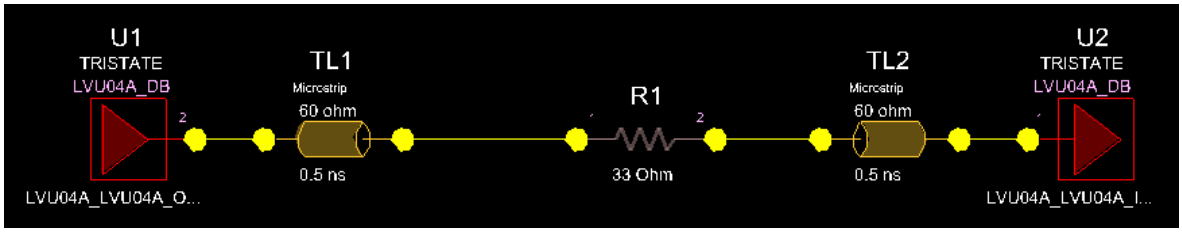
As in the SI Design Setup in PCB Editor, the DC Nets will need to be assigned, select the DSN file in the Project Manager window and use SI Analysis>Identify DC Nets, set the voltages for the DC nets.



Once the SI Models have been assigned and the DC voltages defined, the topology can be exported to SigXplorer for analysis. With an OrCAD PCB SI, or higher, license you can select the net in the schematic page and right-click>SI Analysis>Explore Signal (SigXplorer). For an OrCAD PCB Designer Professional license, you would need to select the net in the schematic page and right-click>SI Analysis>Export Technology which writes a netname.top file that can be opened from SigXplorer.



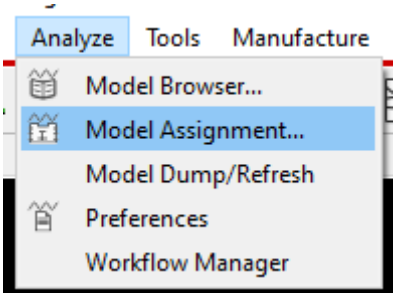
Signal Explorer opens with the extracted Extended Net for pre-route Analysis as before.



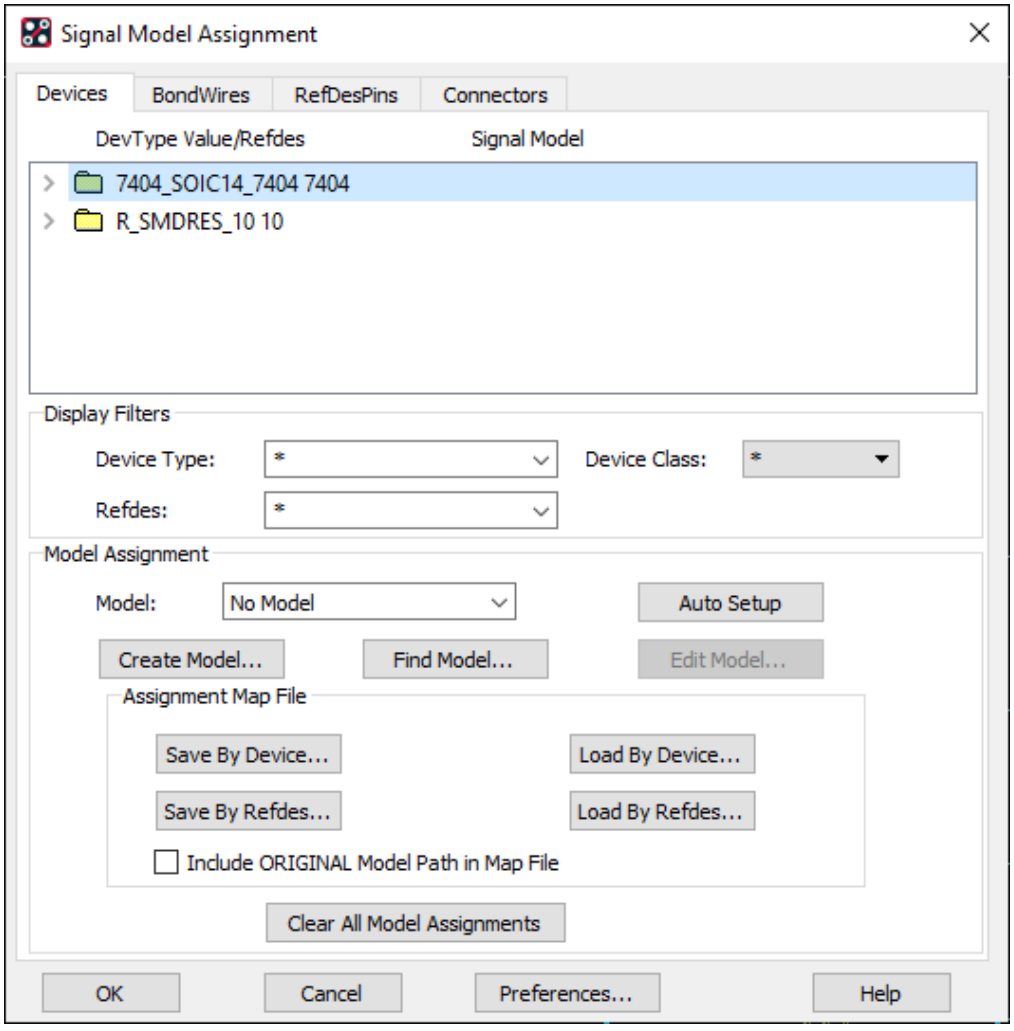
The simulation can be configured as described previously to determine the required topology for the connection.

Appendix A: Creating SI Models in PCB Editor

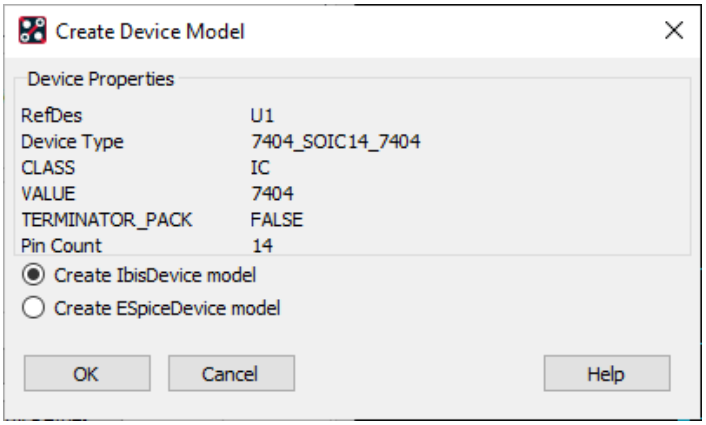
In PCB Editor, use Analyze>Model Assignment,



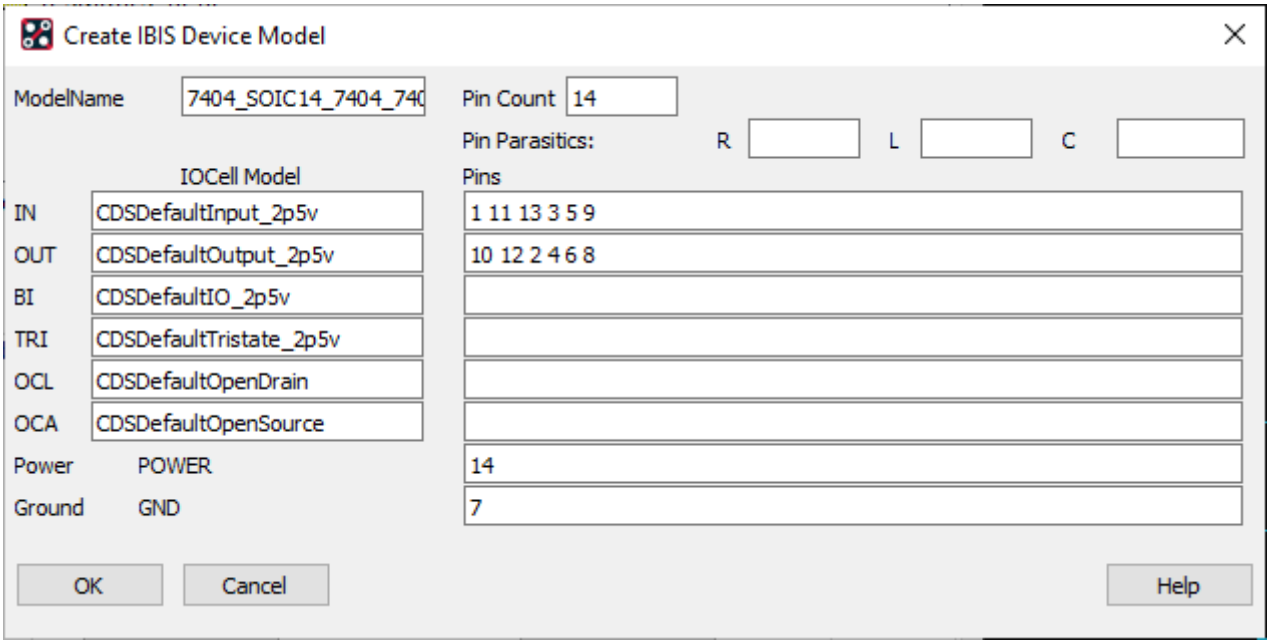
You may need to accept / correct any reported issues from the SI Design Setup to get to the Model Assignment dialog.



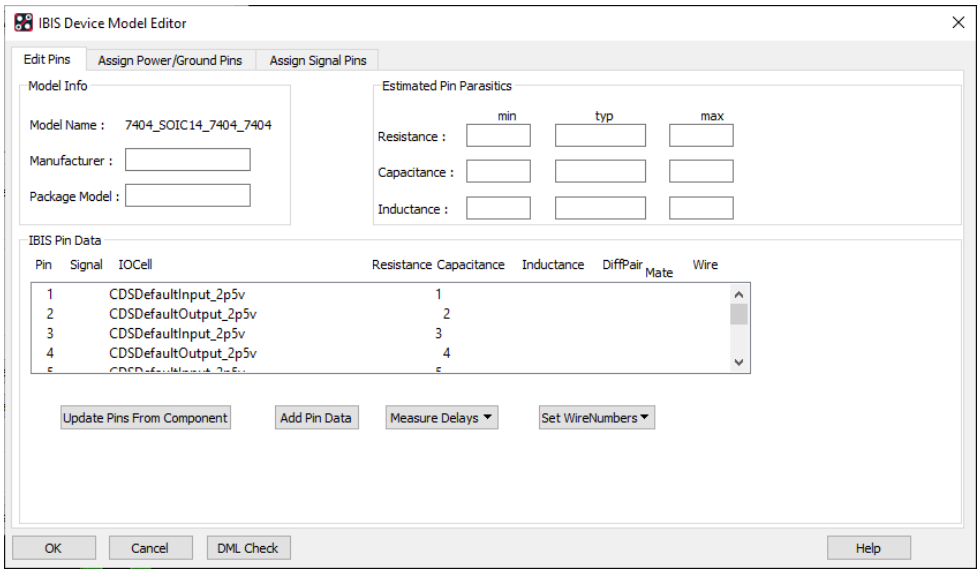
In here you can select, create and edit the model(s). In this case the 7404… item is selected for creation / editing. Create an Ibis Device, if a device doesn't exist.



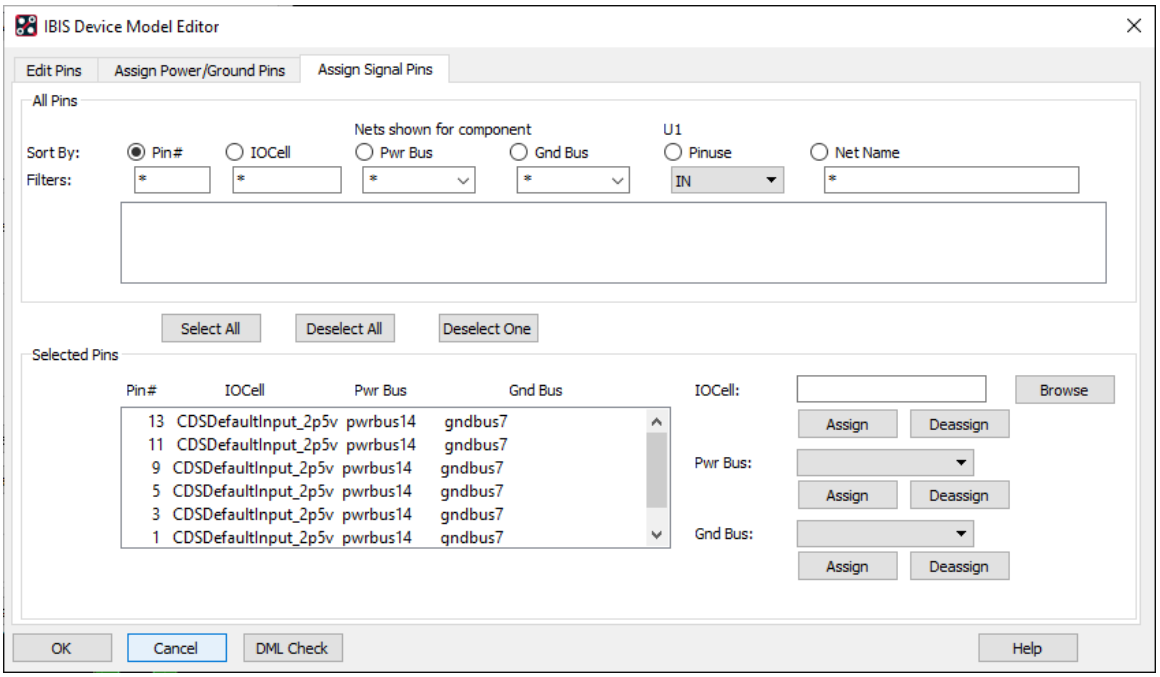
Left-click>OK to get to the default model created.



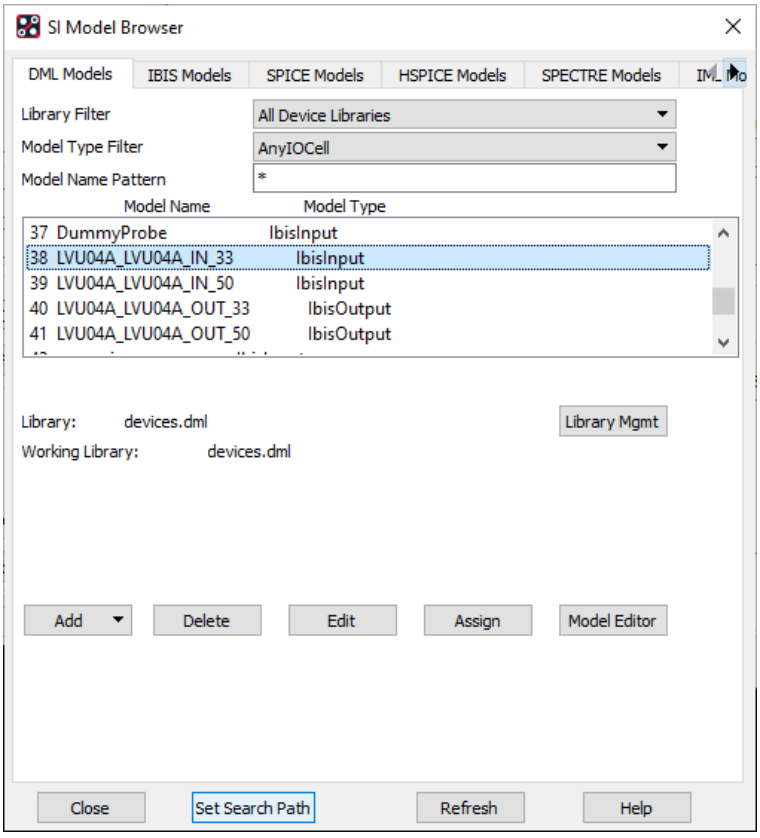
Here the default IO models have been assign and the Pin Numbers are correct. To get the “real” models assigned, left-click>OK to take the defaults for now. Then, back in Signal Model Assignment, use Edit Model to change the default assignments.



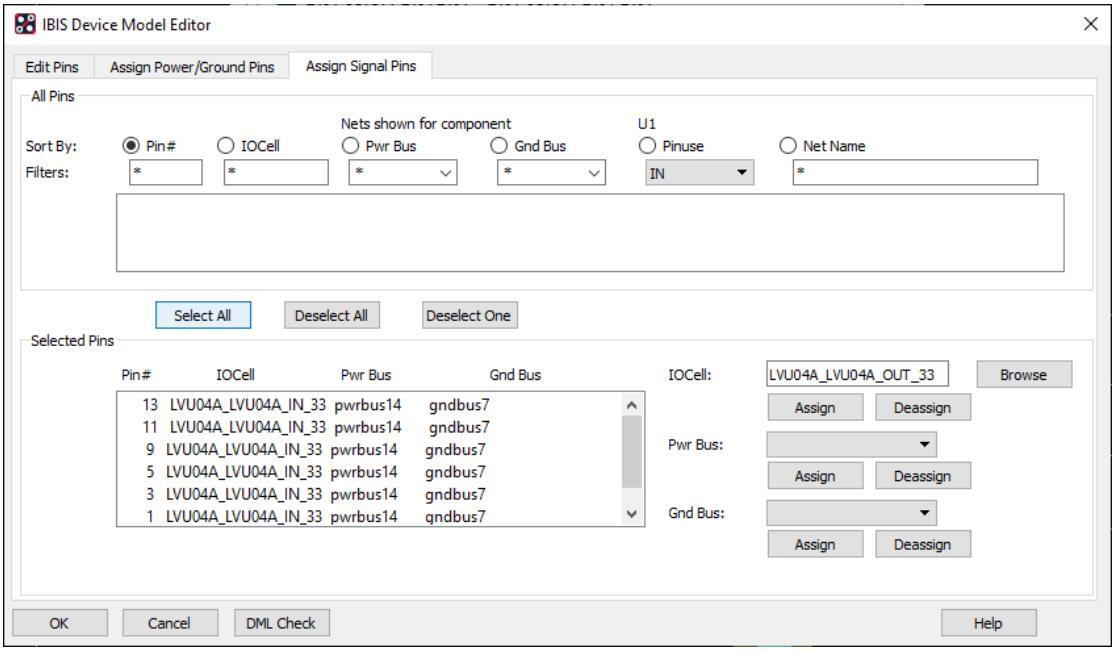
Go to the Assign Signal Pins tab, pick IN as the Pinuse and use Select All to move all the IN pins to the editing window.



Then use the Browse button to search the list of configured IO models.

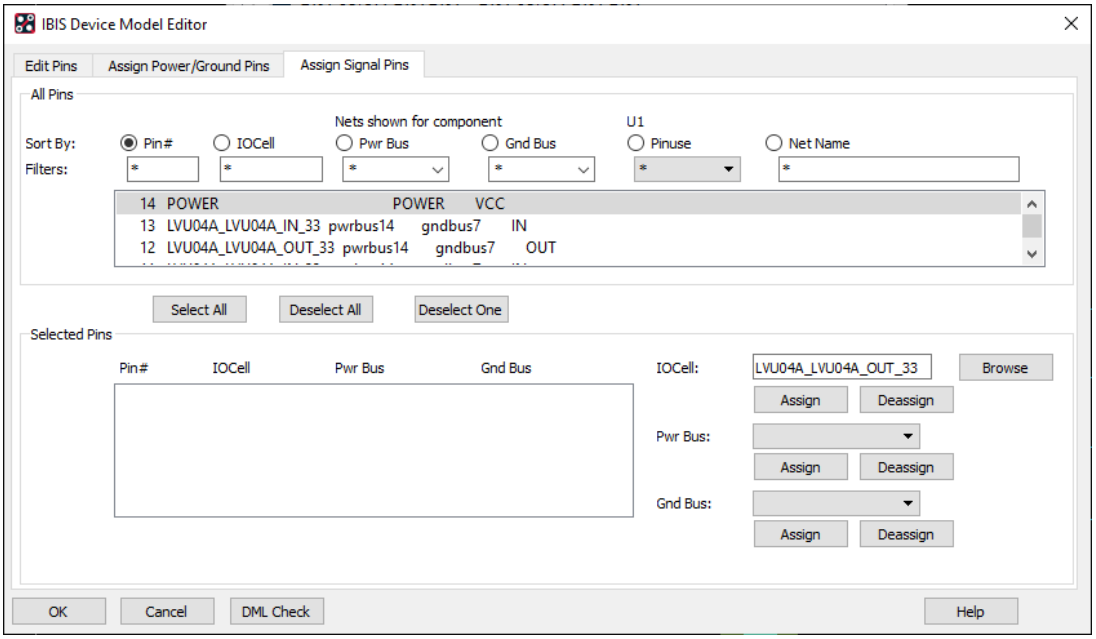


In this case the LVU04A Input for 3.3V needs to be selected, then click on Assign to select the model, switch back to the IBIS Device Model Editor window, there is no need to Close the SI Model Browser window, then use the Assign in the IBIS Device Model Editor window to assign the model selected in the SI Model Browser to the IO pins selected in the model to the IBIS Device Model Editor. As shown in the next screenshot.

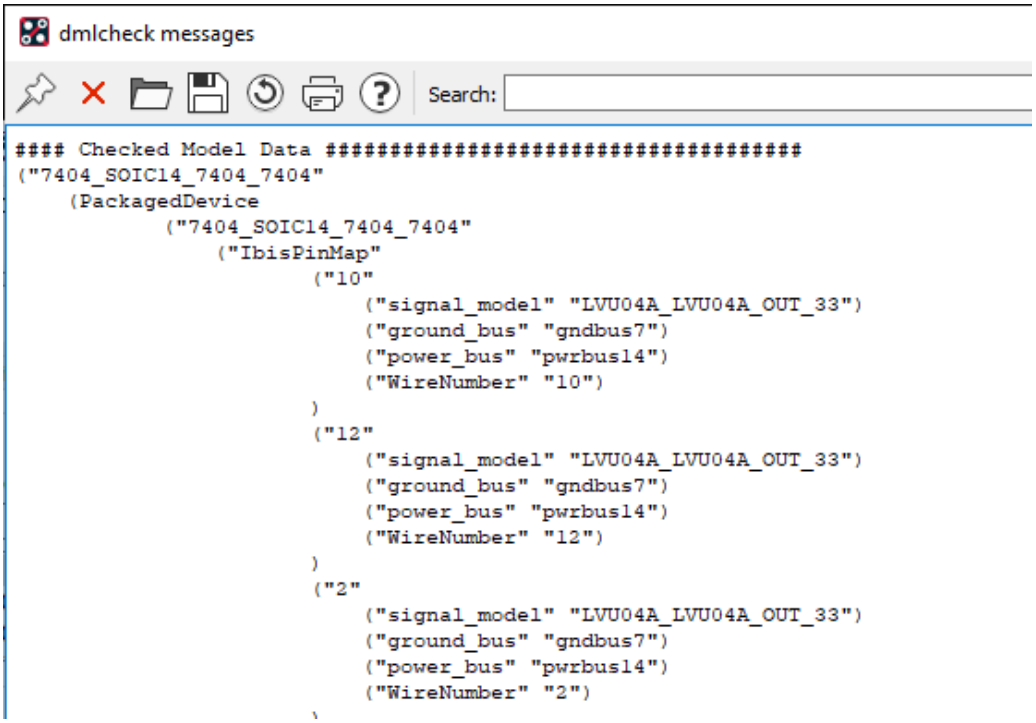


Then use Deselect All to move the IN pins back to the “All Pins” list and repeat the process for the OUT pins to assign the LVU04A output 3.3V to the output pins. The result is as shown in the next screenshot.

(Close the SI Model Browser window)



OK on the IBIS Device Model Editor window will close the window, run the DML check and display the resulting DML file.



Close this window and close the Signal Model Assignment window, you will get a message that the model assignment has been changed.