



EE307 Digital Electronics and Integrated Circuits

HW4 - Assigned: 1/28/18 Due: 2/2/18

Paper submissions due: 5:30pm

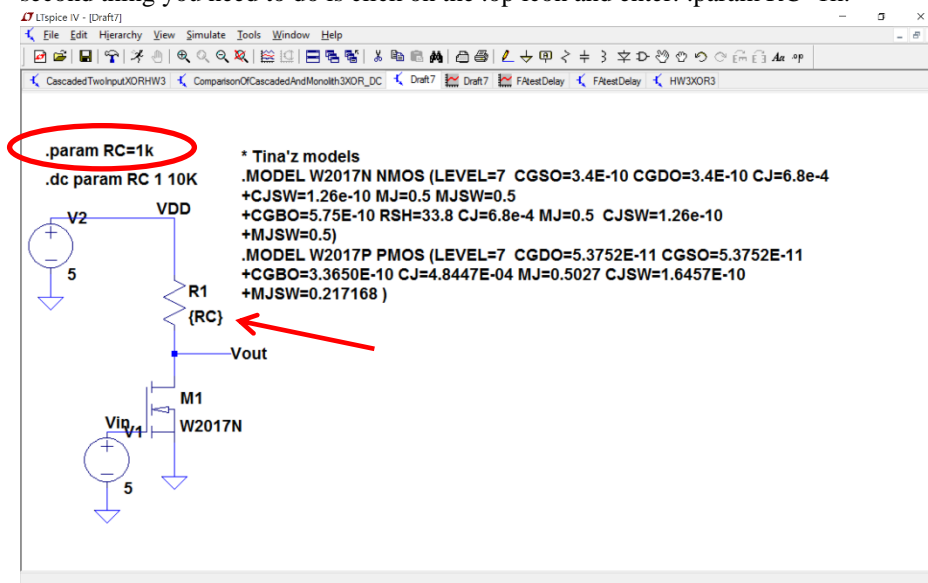
Electronic submissions due: 9:00pm

This week's homework is about transistor resistance, V_m , V_{IL} , V_{IH} , V_{OL} , V_{OH} , a MUX to build and a bit more on using LTSpice.

1. LTSpice: Reading for homework

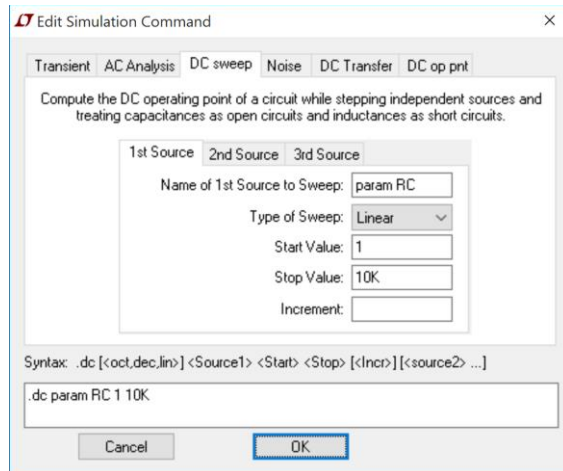
- Comment 1: When you use symbols you only need to include the model file on the toplevel schematic. If you put the model file path in each symbol then for some reason you need to change the path, you'll need to change it in every single symbol. So please remove the path from each individual schematic that has a symbol. Nothing to do for this "question".
- Comment 2: If you have two wires that aren't connected but have the same name, LTSpice will interpret that as them being the same wire and treat them as if they are connected. That means that you don't have to draw long wires between everything that's connected - you just have to name two segments of wires the same. Nothing to do for this "question".
- You may want to change the supply voltage at some time during the quarter so having a VDD pin on your symbols. I think you can put a source with its value set to a variable (See below ".PARAM") but I haven't used it enough to be sure it won't cause problems in the future.
- LTSpice technique 1 for this week: Here's a new technique that will allow you to do two very handy things.
 - It'll let you sweep any value, not just voltages and currents. By "any value" I mean resistances, capacitances, temperature, etc.
 - This technique can also save you a ton of work when you change values used in many places. It allows you to change the value in one place but have the change take effect in every place that it's used.

First let's look at how you can sweep other values than voltage or current by using variables (.PARAM). When doing a graphical (drag and drop) circuit entry, there are two things you need to do to use .PARAM. Instead of giving a resistor value like 1K, put in a name surrounded by curly brackets like {RC}. Names have to follow the same rules you use for other names in your schematics. The second thing you need to do is click on the .op icon and enter: .param RC=1k:



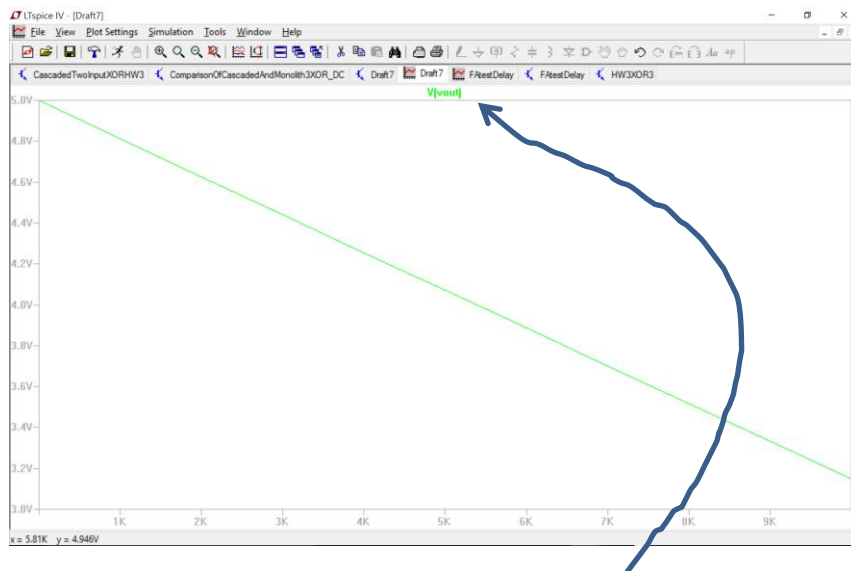
The curly brackets tell LTSpice that you are setting that value to a variable. The .param sets a value to that variable. To sweep the value you need to set the DC sweep as shown at the top of the next page.

The value you set in the .PARAM directive doesn't matter if you are sweeping the variable value in your DC sweep but you have to give it some initial value or the compiler will complain. Let's say you want to find a resistor that will give you an output centered around 3.5V when $V_{GS}=5V$. First set $V_{GS}=5V$ as shown in the above schematic and then set the simulator values on the DC sweep as follows:



Don't forget the 'param' before RC. That tells LTSpice to look for something defined in a .PARAM statement.

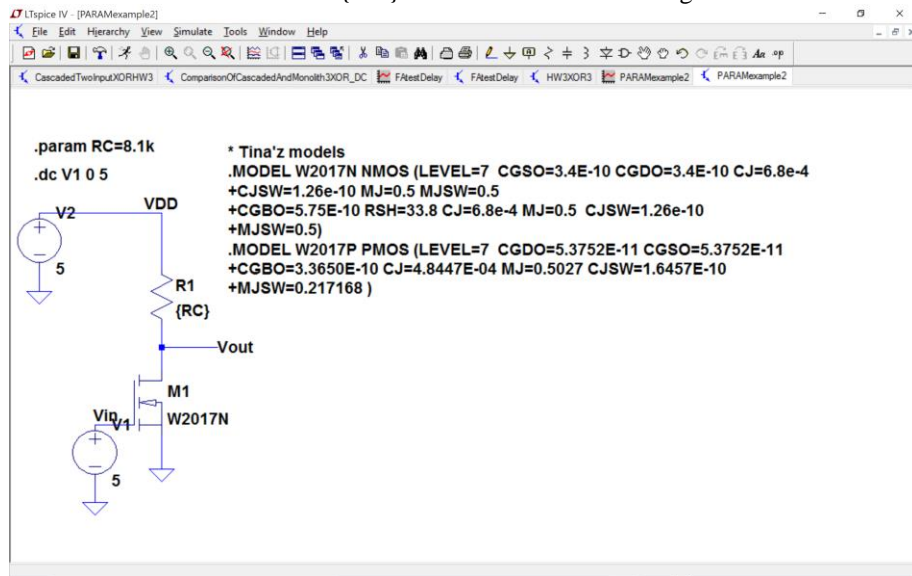
When you run this simulation the output voltage will change as RC changes. Whatever you sweep shows up on the X-axis. Whether you swept a voltage, a resistance, a capacitance, a temperature, a current or whatever:



To find where the output equals 5V use a cursor (right click here and select "Attached cursor – 1st" . That should allow you to drag the cursor along the graph line. (A single left click on the signal name also gives you 1st and a double click on the signal name gives you 1st & 2nd). The position of the cursor is displayed in a separate little window). **NOTE:** that the value that you sweep is shown on the X-axis.

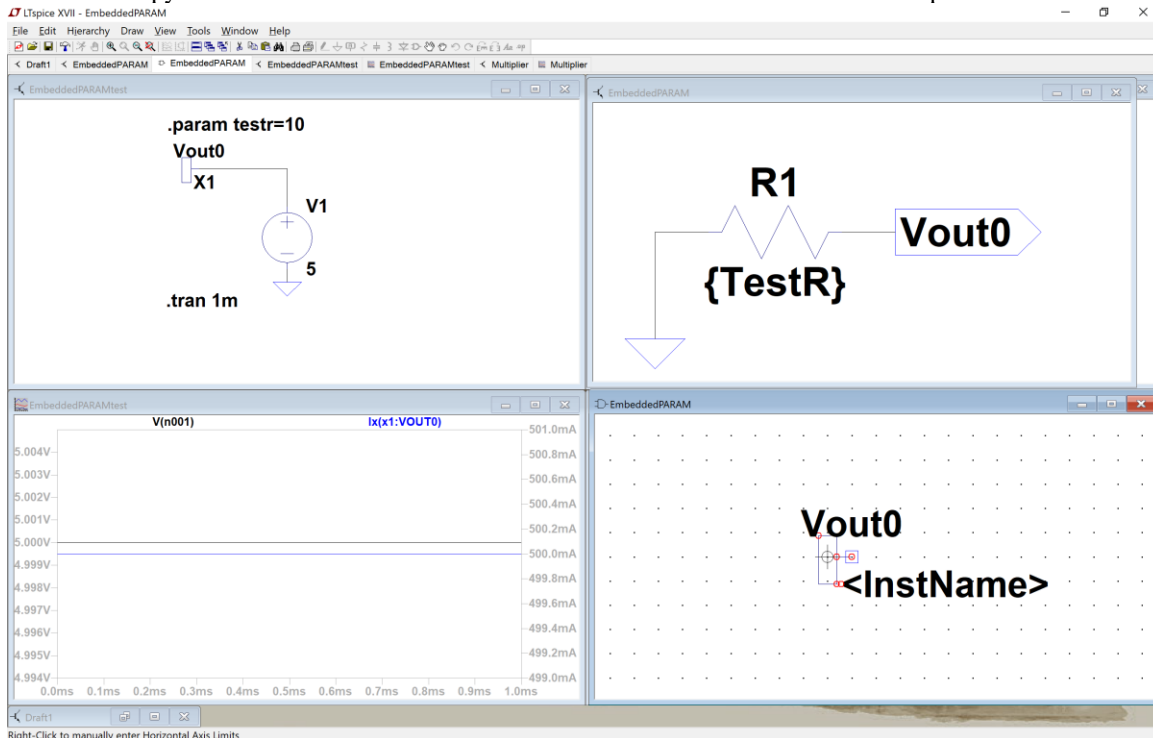
The first use described above is about sweeping a value. The second use, which I'll explain next, just sets a value of a component or a whole bunch of components depending on where you use the variable. When I say "A component value" I mean 1k on a resistor or 1nF on a capacitor or something like that. This use of .PARAM is useful if you want to use the same value in, say, twenty places but you want to

be able to change all of them to a new value easily all at the same time. If you set all those places to the variable (not the value) then you just need to change the value in the .PARAM directive to change all twenty places at once. The picture on the next page doesn't show multiple RTL inverters but if it did, on each one I could set the R value to {RC} and then when I changed the 8.1K in the .PARAM statement all the resistors with {RC} as their value would change to the new value.



You can use “RC” as a value in as many places as you'd like and then when you change its value in the .PARAM directive, all fields that have been designated as having a value of “RC”. You can use as many .PARAM directives as you need. Nothing to do for this “question”.

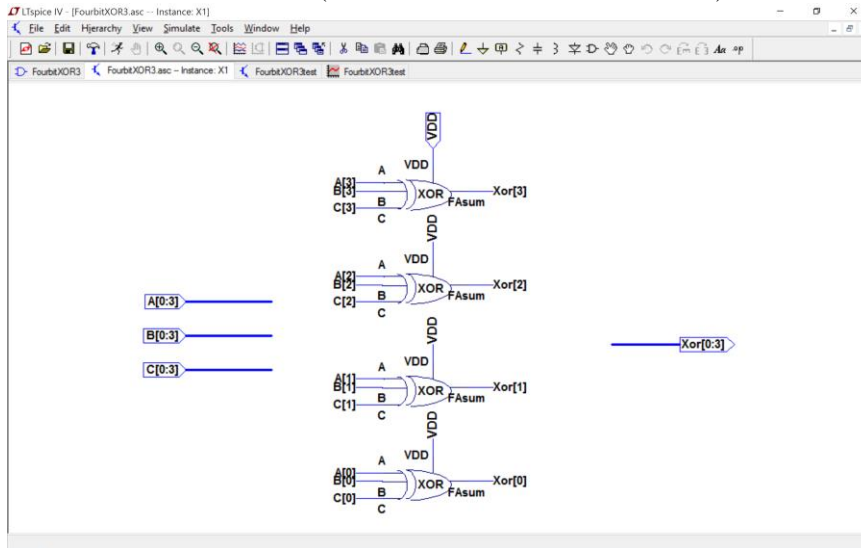
A note on .PARAM: You can use a parameter inside a symbol. If you do that you can specify the value inside the schematic of the symbol or at the toplevel or any level between as long as it is higher in the pyramid than the schematic that uses the PARAM value. Here's an example:



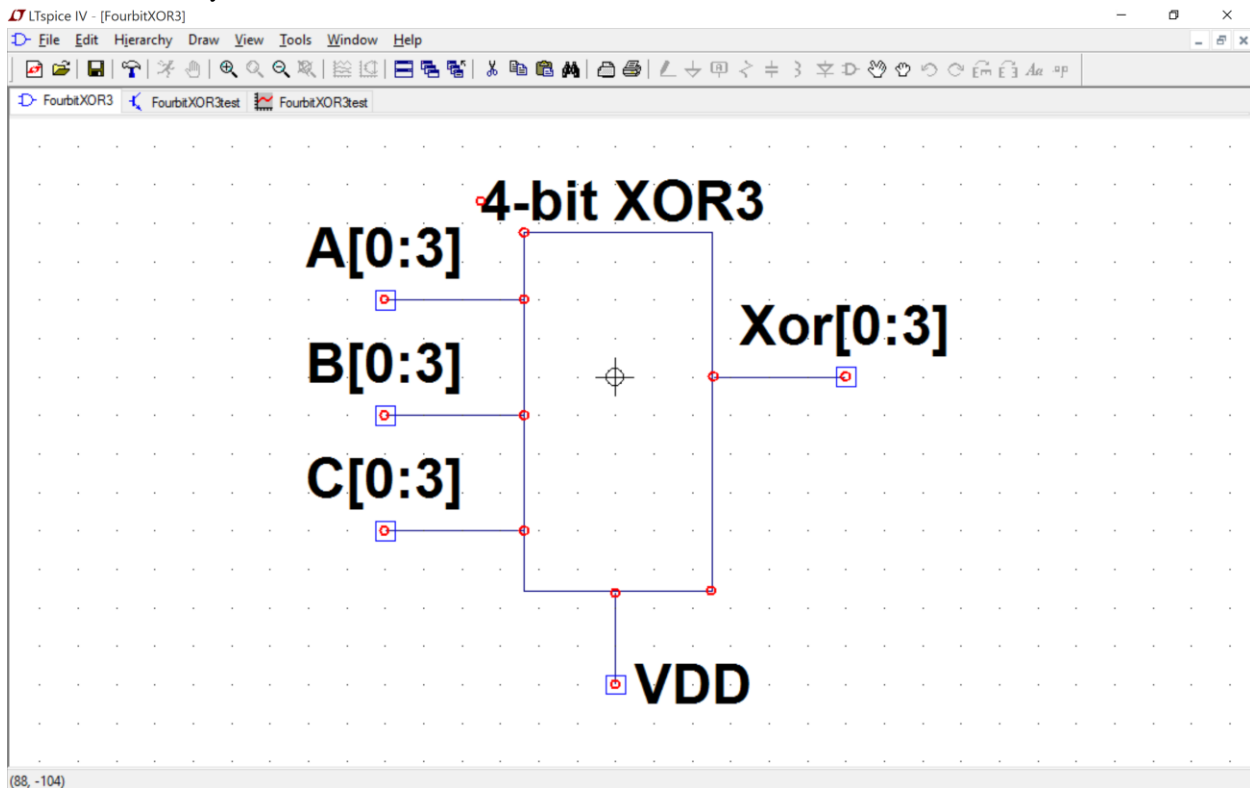
- e. Many of you are already using buses but here's more information. LTSpice technique 2 for this week: This comes from http://ltwiki.org/index.php5?title=Undocumented_LTspice#Bussing_of_Connections_and_Components_.28BUS_shorthand_notation.29

As your circuits start having to deal with more and more inputs, buses will be useful.

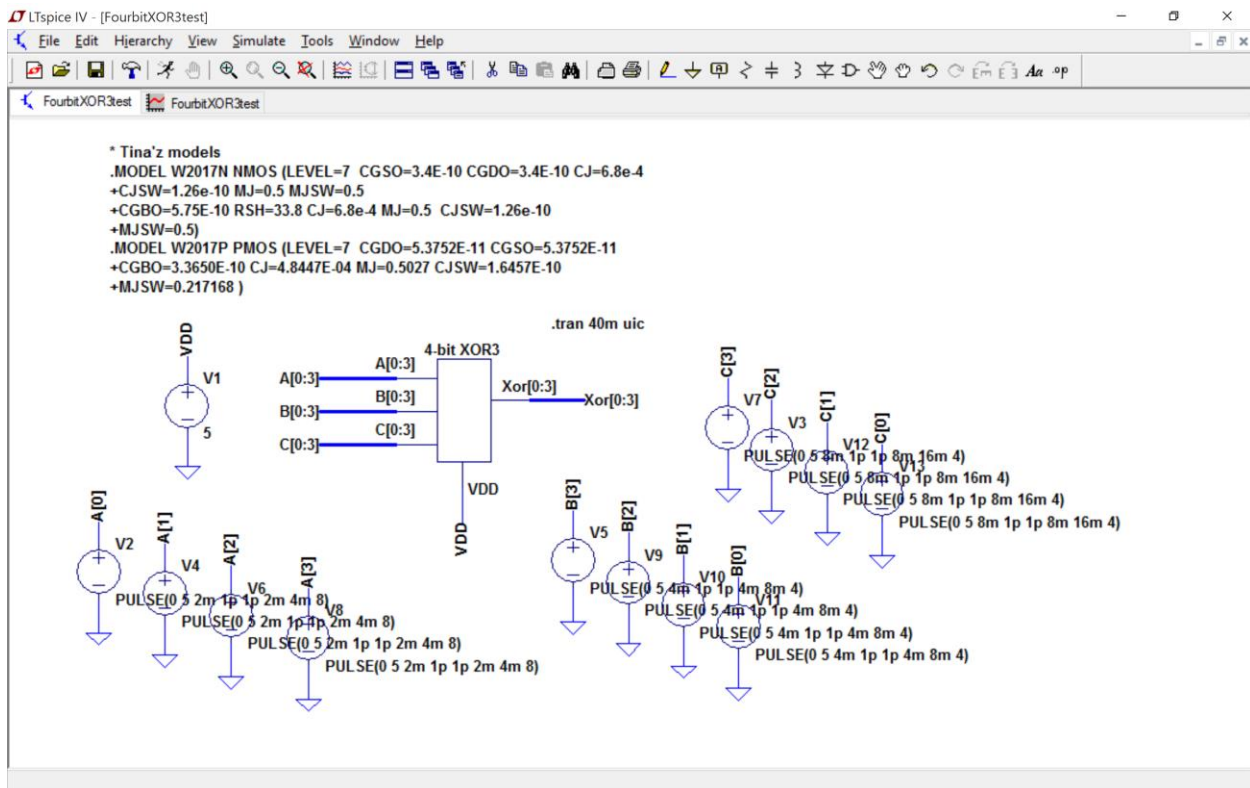
The subcircuit: (I used the 3-bit XOR from last week)



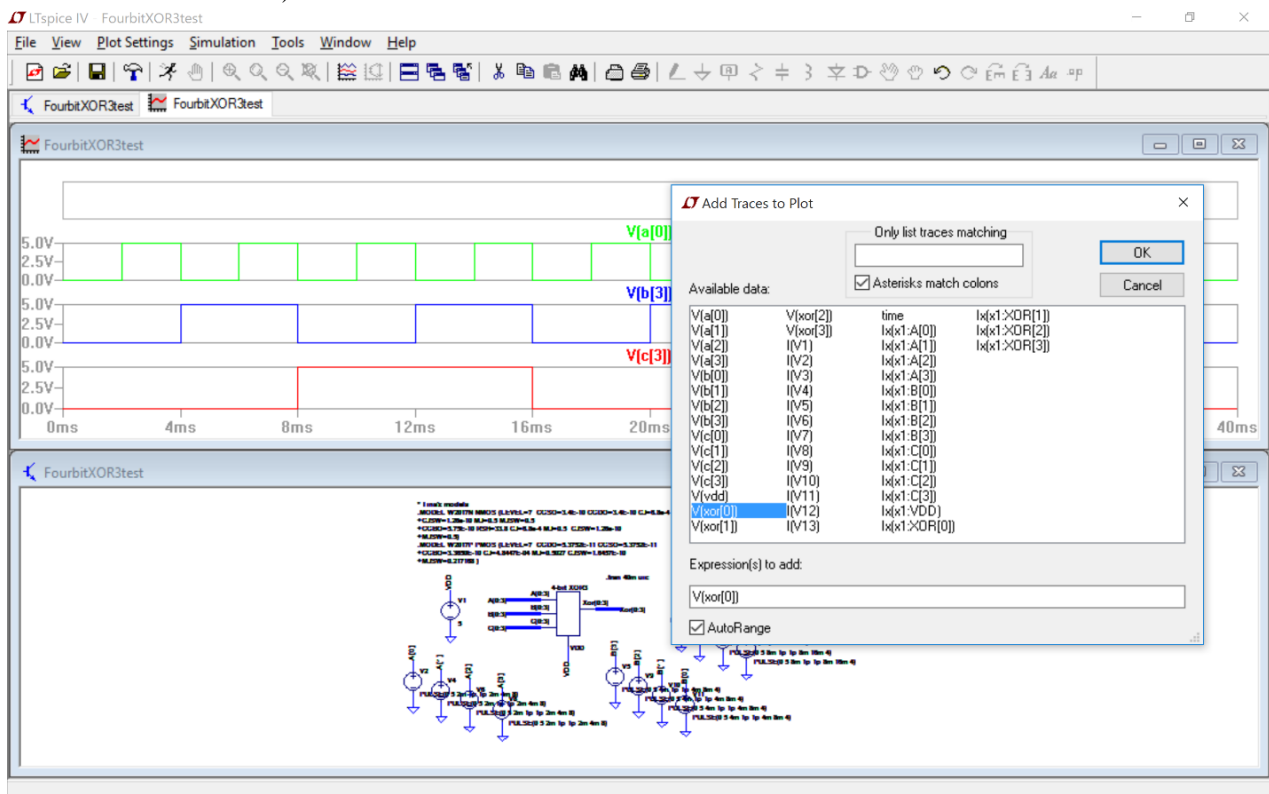
The symbol:



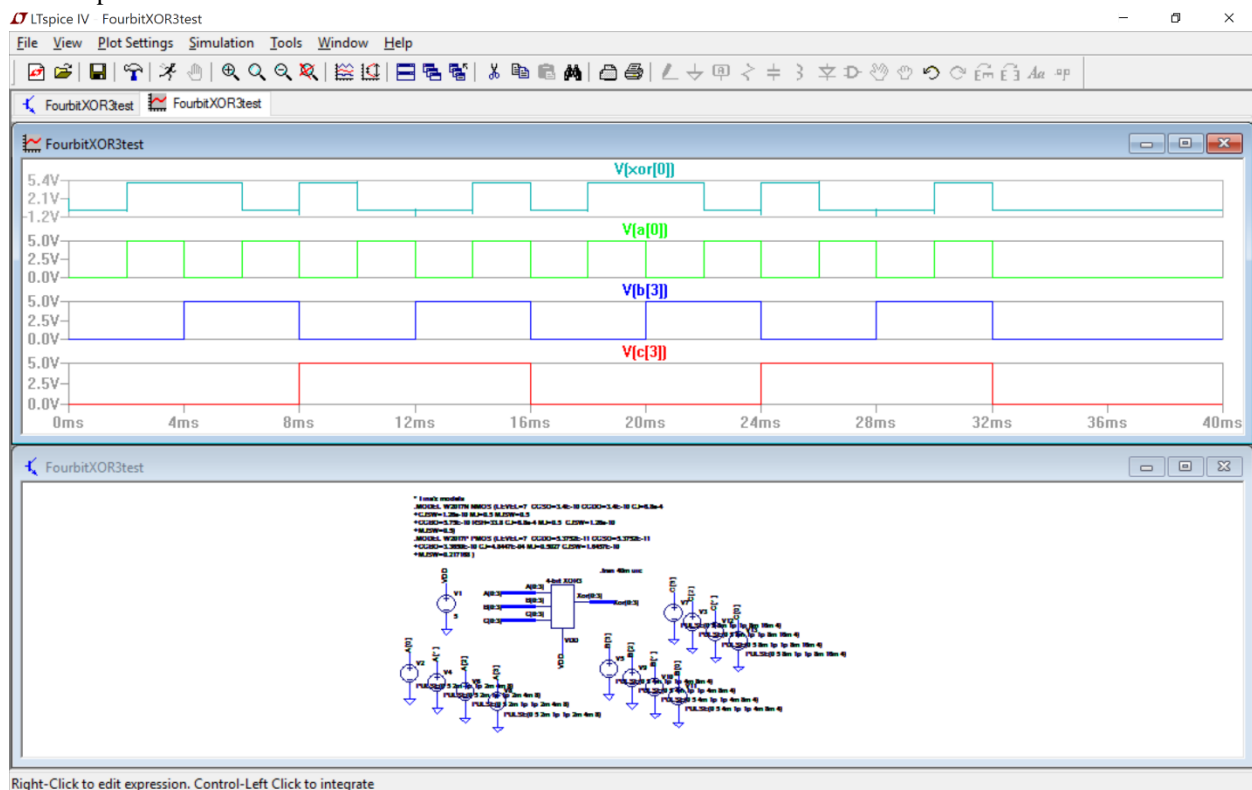
The test circuit:



The results: (Adding signals by right clicking on graph area and adding wire from bus that we are interested in)

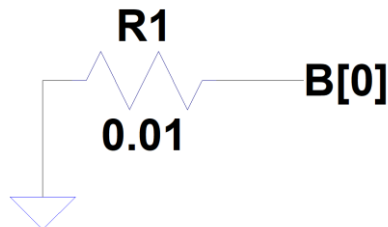


Graph:

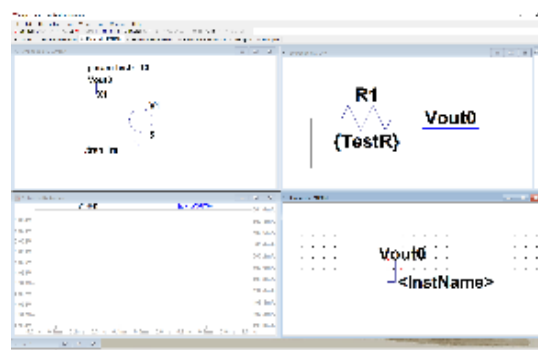


Nothing to do for this “question”.

Sometimes you want to set a single wire of a bus to GND, VDD or connect it to another wire. There’s an “e” component in SPICE that is convenient but it will give infinite current so it won’t give you a realistic behavior. I prefer putting a very small resistor (0.01Ω or so) in series:

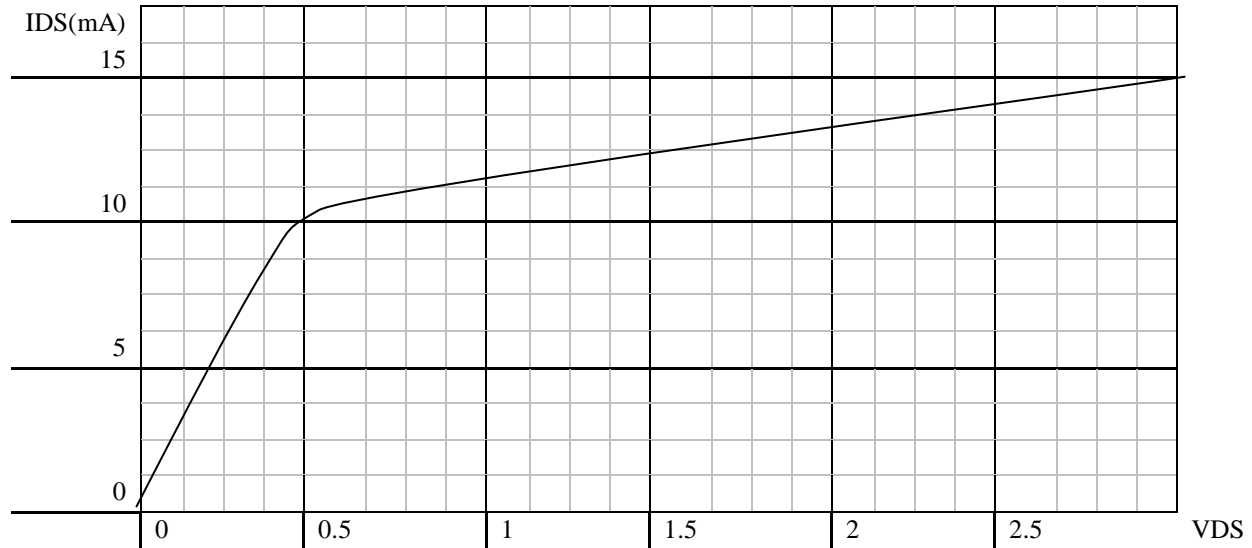


I drew all the parts of a component that can be used to change the name of a wire up where I made a comment on .PARAM.



2. Delay and time constants. Review math to derive rise, fall and propagation delay equations:
 - a. Say you knew the R and C at a node, find an equation for rise time. The slide with the propagation delay may be of use.
 - b. Say you knew the R and C at a node, find an equation for fall time. The slide with the propagation delay may be of use.
 - c. Find an equation for the propagation delay assuming it is completely described by the RC characteristics of the node and the input has an instantaneous input transition.

3. You did the math to find the delays when you know R and C. The next two problems help you find R and C. This problem is about finding R.



- For the graph above, what is R_{on} ? $V_{DD}=3V$. Use a two point average approximation.
- For the graph above, what is R_{on} ? $V_{DD}=3V$. Use the center point approximation.
- Why do you use the active region of the graph when calculating R_{on} ?
- For $\frac{k'}{2} \cdot \frac{W}{L} = \frac{1}{1000} \frac{A}{V^2}$ and $V_{TN}=0.6V$ and $V_{DD}=5V$, what is R_{on} using the center point approximation? (Using the equation instead of the graph).
- What is the resistance of a resistor made of a rectangular N+ doped region of silicon that is $40\mu m$ long and $2\mu m$ wide? (See table at top of next page)
- What is the length of a resistor made in P+PLY that is $2\mu m$ wide and has a resistance of $10K\Omega$? (See table at top of next page)

PROCESS PARAMETERS	RR	N+	P+	P+PLY	POLY	N+BLK	P+BLK	N_W	UNITS
Sheet Resistance	1523.7	6.2	8.1	260.7	6.3	72.3	107.7	321	ohms/sq
Contact Resistance		7.2	6.8	6.5	6.7				ohms
Gate Oxide Thickness		46							angstrom

PROCESS PARAMETERS	M1	M2	M3	M4	M5 (MT)	M6 (ML)	UNITS
Sheet Resistance	57	90	95	92	91	14.29	mohms/sq
Contact Resistance		2.7	2.3	2.25	2.29	0.28	ohms

COMMENTS: BLK is silicide block.

CAPACITANCE PARAMETERS	N+	P+	POLY	D_N_W	N_W	M1	M2	M3	M4	MT	ML	UNITS
Area (substrate)	901	1139	108	220	289	63	41	28	22	19	13	aF/ μm^2
Area (N+active)			7540									aF/ μm^2
Area (P+active)			7542									aF/ μm^2
Area (r well)	873			1088								aF/ μm^2
Area (MOS varactor@1V)			7776									aF/ μm^2
Area (HA varactor)		2744										aF/ μm^2
Area (MiM)			2057									aF/ μm^2
Area (M1)			191									aF/ μm^2
Area (M2)					116							aF/ μm^2
Area (M3)						83						aF/ μm^2
Area (M4)							91					aF/ μm^2
Area (MT)								96				aF/ μm^2
Area (ML)									22			aF/ μm^2

4. V_m , V_{IL} , V_{IH} , V_{OL} and V_{OH} . Assuming that:
- >> the magnitude of V_{TN} is the same as V_{TP} and is 0.5V
 - >> $V_{omax}=6V$ and $V_{omin}=0V$
 - >> $V_{DD}=6V$
 - >> $K'=200\mu A/V^2$ for both. Remember $K'=k'(W/L)$
 - >> λ for both is $0.07 A^{-1}$
- a. Find V_m
 - b. Find gain
 - c. Find V_{IL}
 - d. Find V_{IH}
 - e. Find NML
 - f. Find NMH
5. This week's work on the computer is to build the MUX for the datapath.
- a. Draw the truth table for a MUX that selects between two inputs. The tricky part here is to determine the inputs and the outputs on the MUX.
 - b. Find the logic for the MUX.
 - c. Build the MUX with gates. This component only selects between two single wire inputs. Nothing to show for this question.
 - d. Make a single symbol that takes two 9-bit inputs (just one select signal) and selects between them. Use the MUX you designed in 5c to build this. Make a symbol for it.
 - e. Use your MUX to select between the adder or multiplier outputs. Package the entire adder, multiplier and MUX into a single symbol. You may have to zero some wires since the output of the adder isn't as wide (doesn't have as many bits) as the multiplier. A technique for that was introduced in the reading above (if you are using a bus). Nothing to show for this question.
 - f. Test the datapath. Set $A=1110$ and $B=0011$. Switch the select signal and show the output switch between the two outputs. Please turn in a screen capture showing the output changing with a change in the select value.