Laboratory 4: Gate Delay and Power

ECE 429: Introduction to VLSI Design

By: Alexander Lukens (A20417036)
Instructor: Prof. Ken Choi
Lab Date: 10/01/2020
Due Date: 10/12/2020

Acknowledgement: I acknowledge all of the work (including figures and codes) belongs to me and/or persons who are referenced.

Signature: Alexander Lukens

1. Introduction

In this laboratory session, students will be introduced to the design of a 2-input NAND gate using CMOS logic. This gate will be implemented entirely in the Cadence Virtuoso design suite, and will be simulated using SPICE. Specifically, a schematic will be created for the transistor-level circuit at various different transistor sizes (90nm, 180nm, and 270nm). A testing circuit will then be constructed that will test the NAND gate for all possible transitions (00 to 11, 01 to 11, or 10 to 11). A SPICE netlist for this testing circuit will be exported from Virtuoso, and instructions added to determine the rising/falling delays and power consumption of the circuit.

The results of these SPICE simulations will then be used to determine the logical effort and parasitic delay of the NAND gate.

2. Theory/Pre-Lab

Cadence Virtuoso is a popular electronic design automation (EDA) software suite. This suite is used to design custom circuit designs. Virtuoso enables designers to create a design as a template and then implement it numerous times in a more complex circuit. The characteristics of a specific hardware design can be approximated by defining the parameters of each transistor. For example, the FreePDK45 library is an open source component library that defines various templates with characteristics of a generic 45nm design process. When using this library, designers can approximate how an CMOS design would perform (power, speed, etc) when fabricated on a 45nm process.

Once the CMOS circuit is designed in Cadence Virtuoso, the circuit can be exported to a "netlist". This netlist contains all the information necessary to reproduce the circuit (circuit element and node information, simulation characteristics, etc. Also included in the netlist is the directions for analyses to perform on the circuit. One such analysis is transient analysis, where the response of the circuit is modeled with respect to time. The total run time, start time, and time-stepping can be adjusted to provide granular results.

The delay of a CMOS circuit can be modeled in a variety of different methods. One such method is the Linear delay model, which uses logical effort, electrical effort, and the parasitic delay of a logic gate to determine the total delay of a gate or logic path.

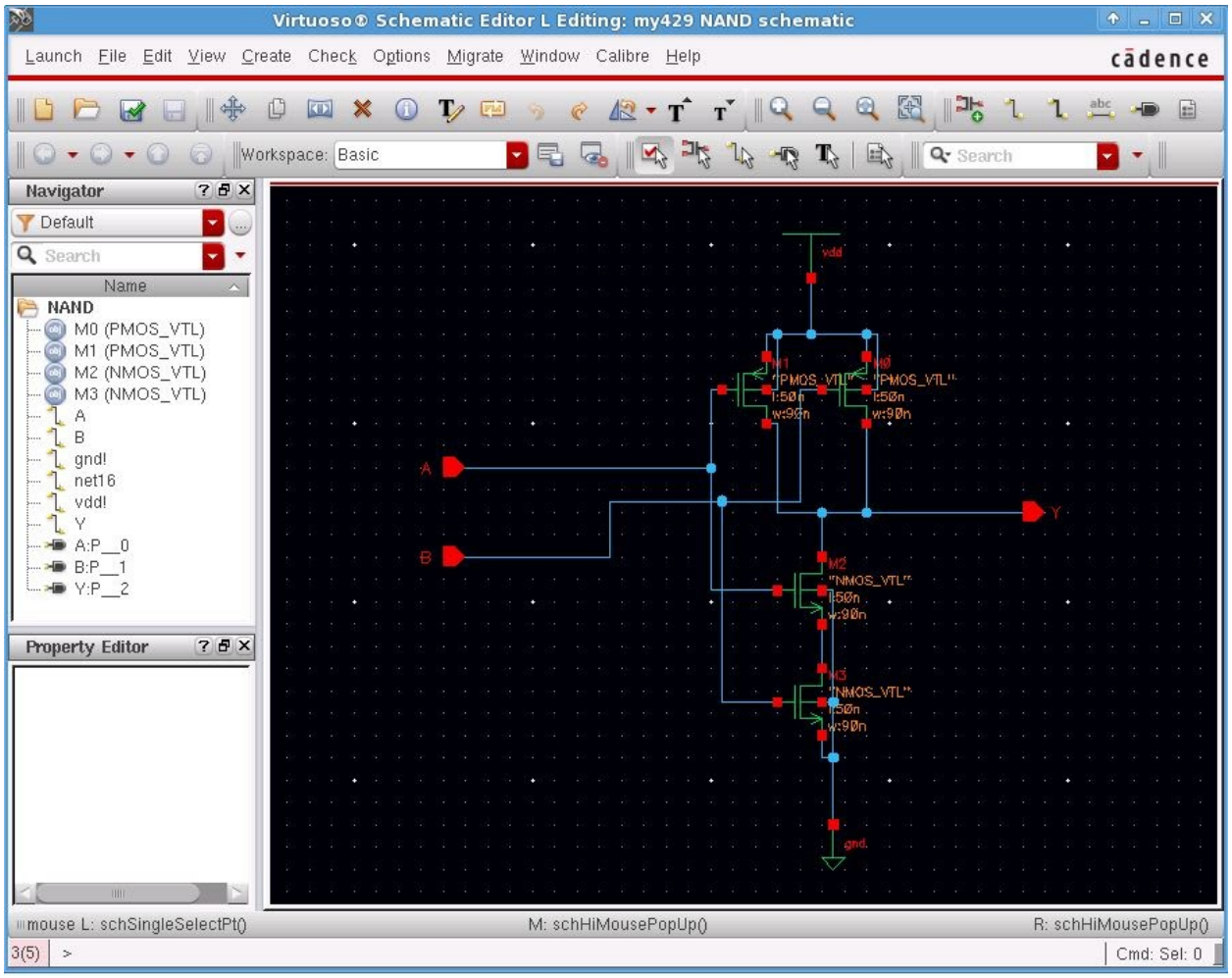
$$D = g * h + p$$

Where g is the logical effort of the gate, h is the electrical effort of the gate, and p is the parasitic effort of the gate. The logical effort of a gate is intrinsic to the operation it performs, and for a 2-input NAND gate is 2. The electrical effort of the gate is proportional to the ratio of capacitance it drives (C_{out}/C_{in}). The parasitic delay is intrinsic to the transistor sizes used.

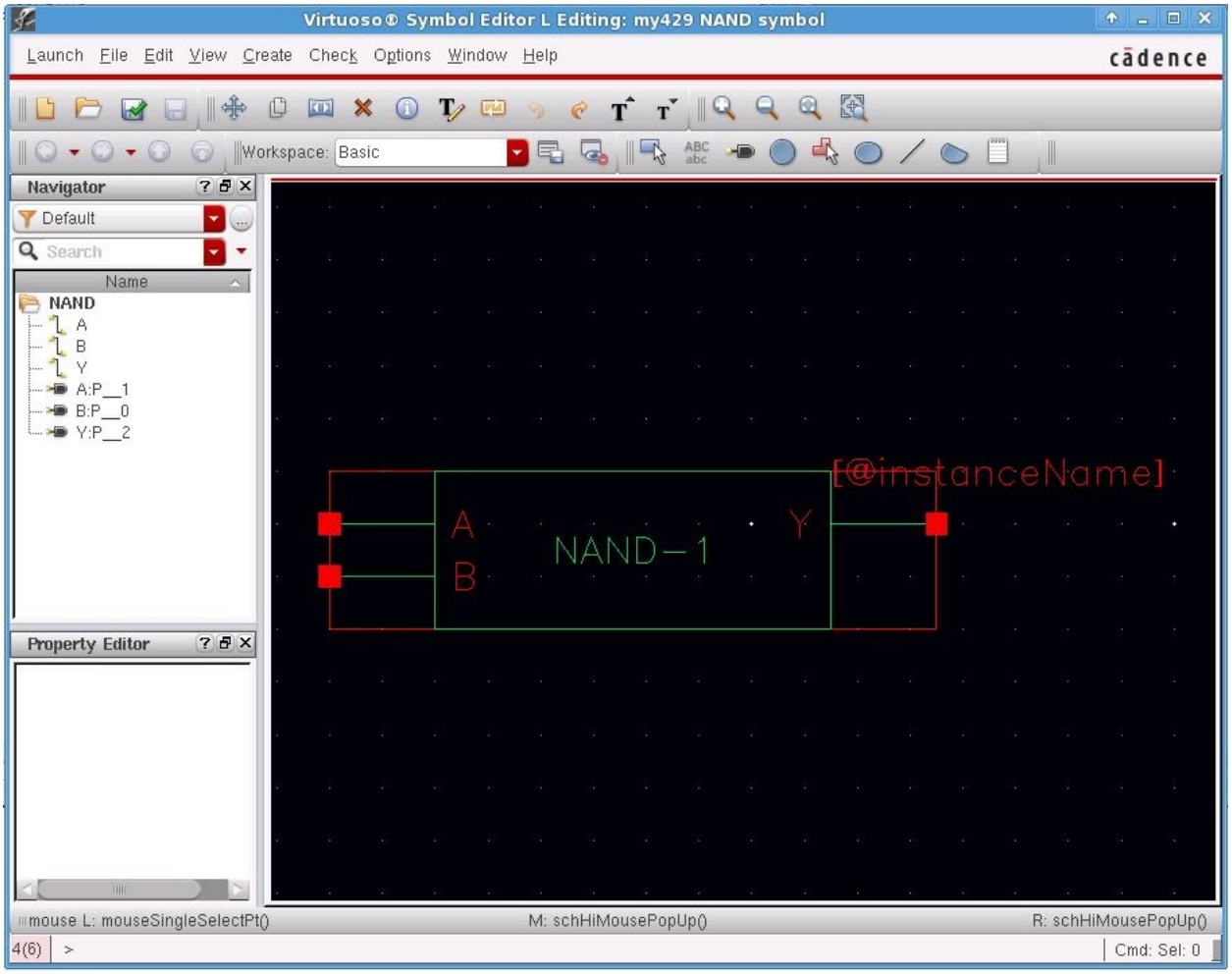
3. Implementation

In this laboratory, students will design and verify a CMOS 2 input NAND schematic. To start, students created a new cell to hold the various implementations of the NAND gate. Then, students created a schematic cellview under the NAND cell. This cellview will hold the high level schematic about how the circuit will be designed using the FreePDK45 library components.

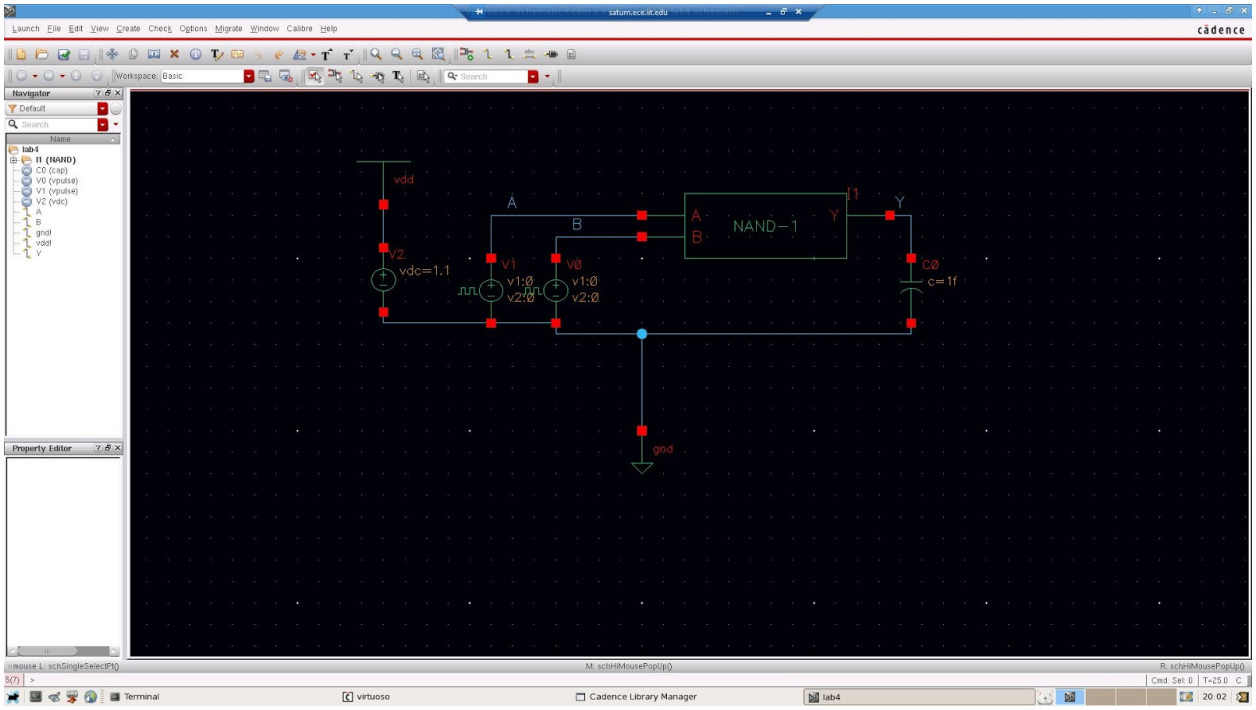
The schematic will include information such as the input and output pins, the PMOS and NMOS pull-up and pull-down networks, and information about the sizing of the transistors. This will allow students to test the functionality of the transistor level circuit to ensure the NAND functions as expected.



Schematic Cellview for 2-input NAND Gate



Symbol Cellview for 2-input NAND



Testing circuit for 2-input NAND

```
** Design view name: schematic
.GLOBAL vdd!


.TRAN 1e-12 5e-9 START=0.0 SWEEP CAP POI 4 1f 2f 4f 8f

.OP

.TEMP 25.0
.OPTION
+    ARTIST=2
+    INGOLD=2
+    PARHIER=LOCAL
+    PSF=2
.INCLUDE "/apps/FreePDK45/ncsu_basekit/models/hspice/hspice_nom.include"

** Library name: my429
** Cell name: NAND
** View name: schematic
.subckt NAND a b y
m1 y a vdd! vdd! PMOS_VTL L=50e-9 W=90e-9 AD=9.45e-15 AS=9.45e-15 PD=300e-9 PS=300e-9 M=1
m0 y b vdd! vdd! PMOS_VTL L=50e-9 W=90e-9 AD=9.45e-15 AS=9.45e-15 PD=300e-9 PS=300e-9 M=1
m3 net16 b 0 0 NMOS_VTL L=50e-9 W=90e-9 AD=9.45e-15 AS=9.45e-15 PD=300e-9 PS=300e-9 M=1
m2 y a net16 0 NMOS_VTL L=50e-9 W=90e-9 AD=9.45e-15 AS=9.45e-15 PD=300e-9 PS=300e-9 M=1
.ends NAND
** End of subcircuit definition.

** Library name: my429
** Cell name: lab4
** View name: schematic
xi1 a b y NAND
c0 y 0 CAP
v1 a 0 PULSE 0 1.1 0 10p 10p 490p 1n
v0 b 0 PULSE 0 1.1 0 10p 10p 490p 1n
v2 vdd! 0 DC=1.1

.measure tpdr
+ TRIG v(b) VAL='0.55' FALL=1
+ TARG v(y) VAL='0.55' RISE=1
.measure tpdf
+ TRIG v(b) VAL='0.55' RISE=1
+ TARG v(y) VAL='0.55' FALL=1
.measure pwr AVG P(v2) FROM=0 TO=5n

.END
```

Netlist for 00->11 transition, 90nm

lab4_180nm_01_11.sp (~/Desktop/ECE429/lab4/180nm)

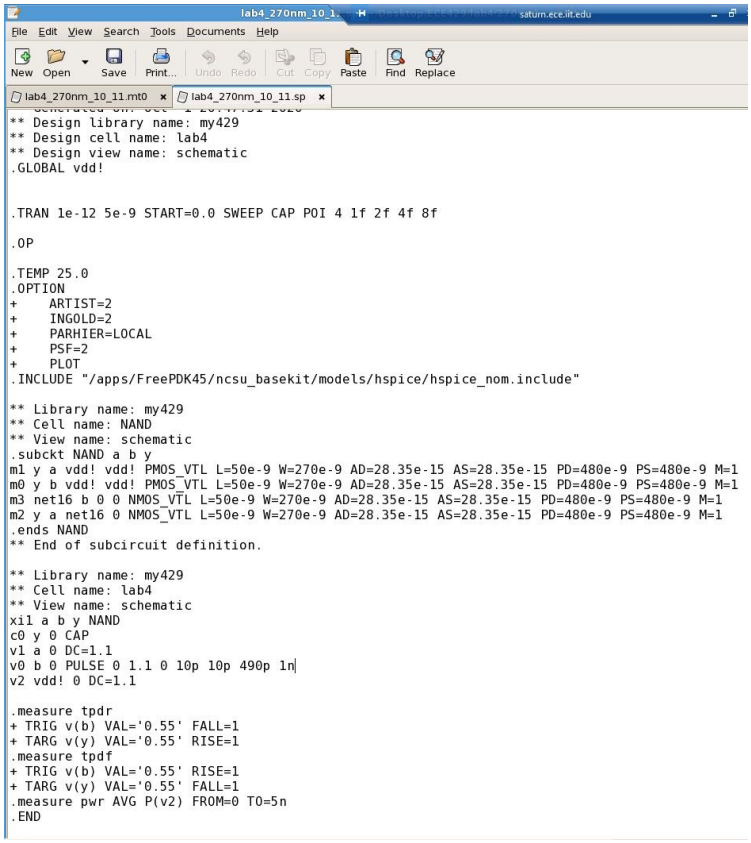
File Edit View Search Tools Documents Help

New Open Save Print... Undo Redo Cut Copy Paste Find Replace

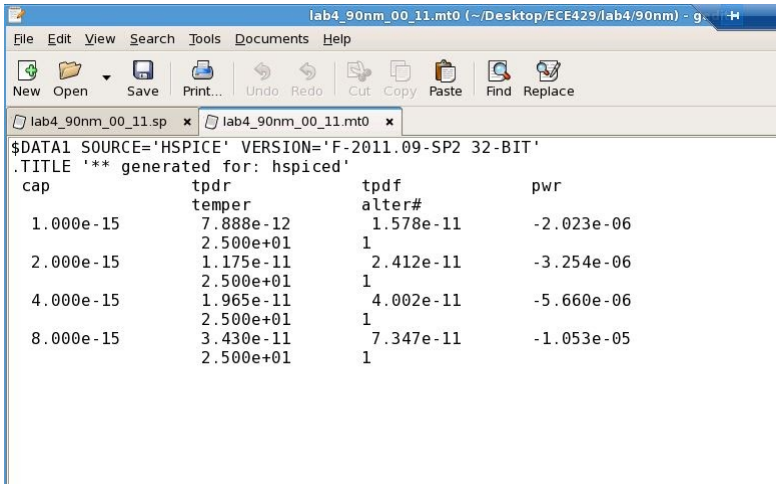
lab4_180nm_01_11.mt0 | lab4_180nm_01_11.sp

```
** Design cell name: lab4
** Design view name: schematic
.GLOBAL vdd!


.TRAN 1e-12 5e-9 START=0.0 SWEEP CAP POI 4 1f 2f 4f 8f

.OP

.TEMP 25.0
.OPTION
+    ARTIST=2
+    INGOLD=2
+    PARHIER=LOCAL
+    PSF=2
.INCLUDE "/apps/FreePDK45/ncsu_basekit/models/hspice/hspice_nom.include"

** Library name: my429
** Cell name: NAND
** View name: schematic
.subckt NAND a b y
m1 y a vdd! vdd! PMOS_VTL L=50e-9 W=180e-9 AD=18.9e-15 AS=18.9e-15 PD=390e-9 PS=390e-9 M=1
m0 y b vdd! vdd! PMOS_VTL L=50e-9 W=180e-9 AD=18.9e-15 AS=18.9e-15 PD=390e-9 PS=390e-9 M=1
m3 net16 b 0 0 NMOS_VTL L=50e-9 W=180e-9 AD=18.9e-15 AS=18.9e-15 PD=390e-9 PS=390e-9 M=1
m2 y a net16 0 NMOS_VTL L=50e-9 W=180e-9 AD=18.9e-15 AS=18.9e-15 PD=390e-9 PS=390e-9 M=1
.ends NAND
** End of subcircuit definition.

** Library name: my429
** Cell name: lab4
** View name: schematic
xi1 a b y NAND
c0 y 0 CAP
v1 a 0 PULSE 0 1.1 0 10p 10p 490p 1n
v0 b 0 DC=1.1
v2 vdd! 0 DC=1.1

.measure tpdr
+ TRIG v(a) VAL='0.55' FALL=1
+ TARG v(y) VAL='0.55' RISE=1
.measure tpdf
+ TRIG v(a) VAL='0.55' RISE=1
+ TARG v(y) VAL='0.55' FALL=1
.measure pwr AVG P(v2) FROM=0 TO=5n

.END
```

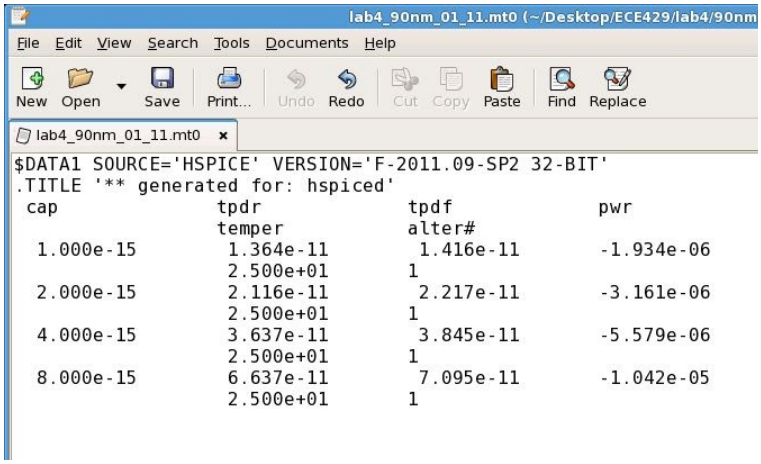
Netlist for 01 -> 11 transition, 180nm



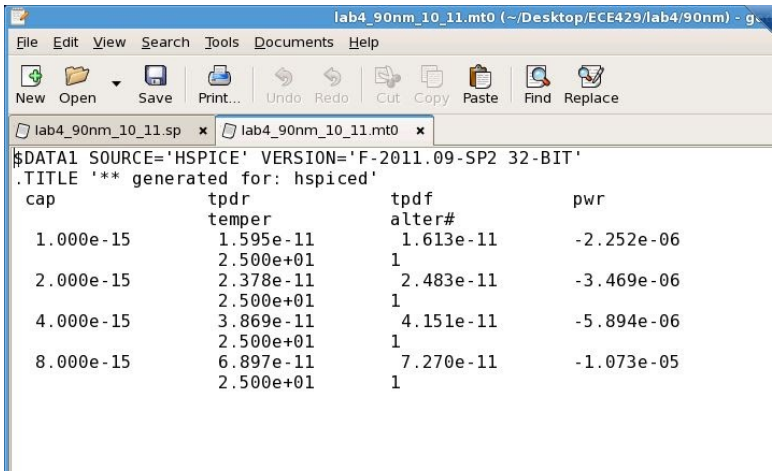
Netlist for 10 -> 11 transition, 270nm



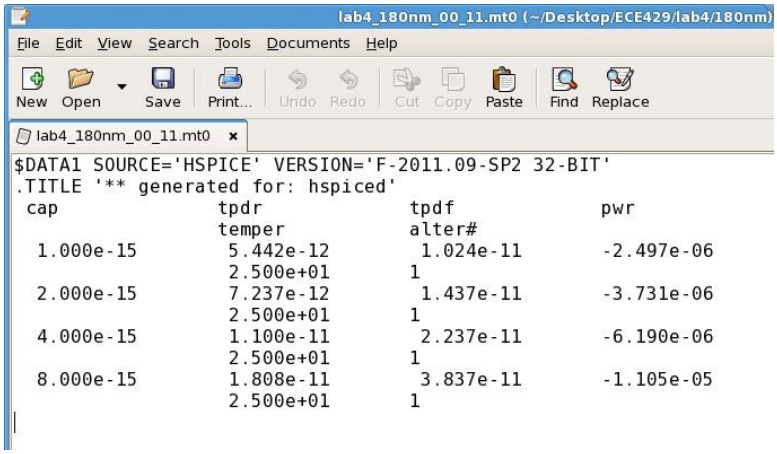
90nm 00-> 01 transition mt0 file



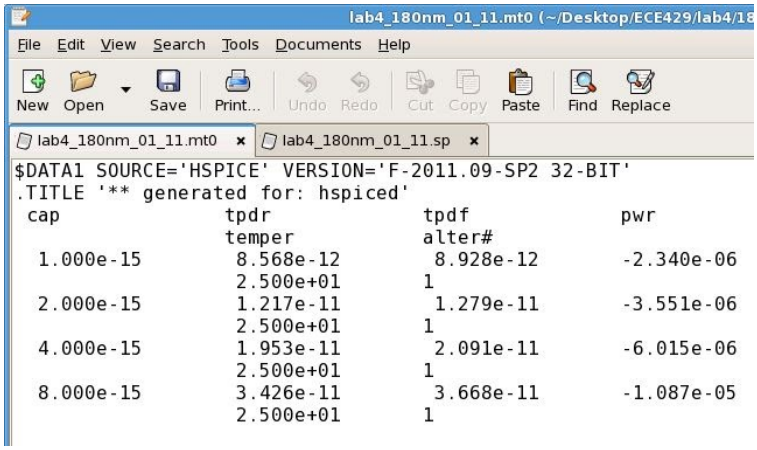
90nm 01 -> 11 transition mt0 file



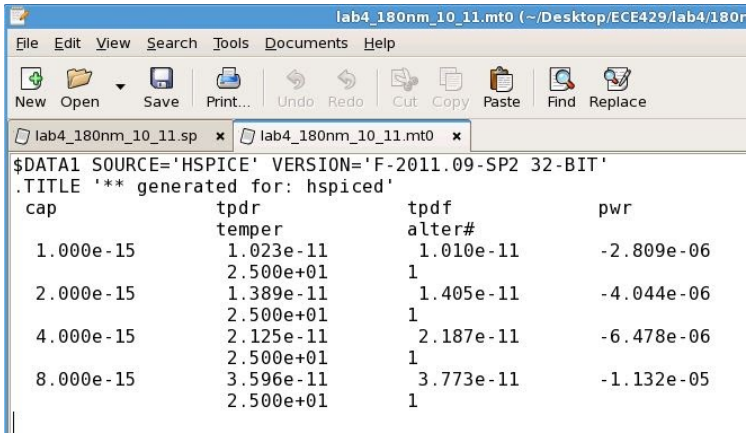
90nm 10 -> 11 transition mt0 file



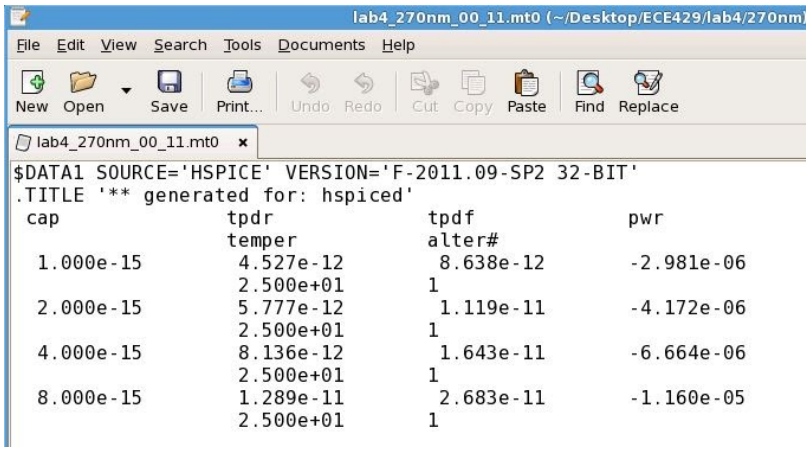
180nm 00 -> 11 transition mt0 file



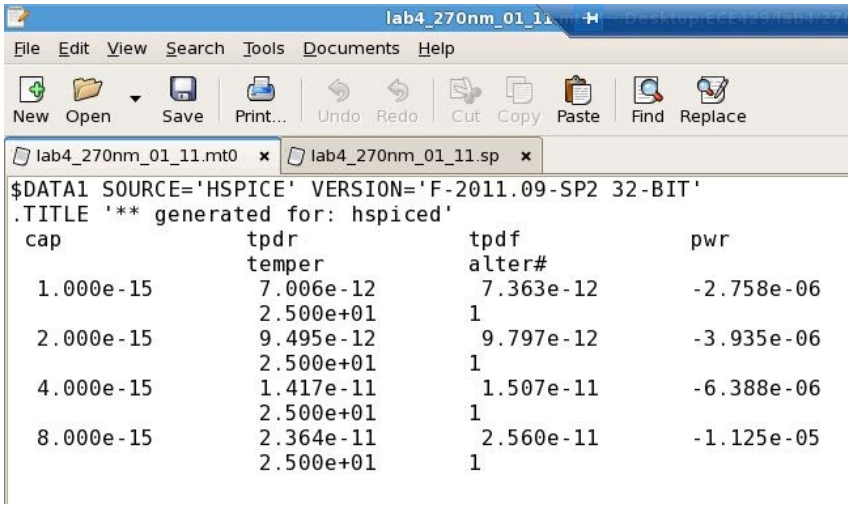
180nm 01 -> 11 transition mt0 file



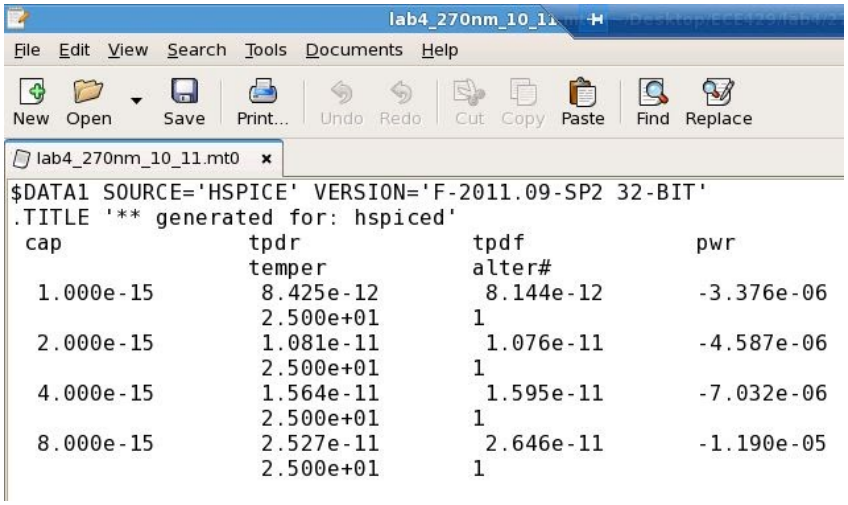
180nm 10 -> 11 transition mt0 file



270nm 00 -> 11 transition mt0 file



270nm 01 -> 11 transition mt0 file



270nm 10 -> 11 transition mt0 file

Condensed Maximum-Value Result Tables

w=90nm l=50nm	Load Capacitances			
	1f	2f	4f	8f
Rising Propagation Delay (ps)	15.95 (10 to 11)	23.78 (10 to 11)	38.69 (10 to 11)	68.97 (10 to 11)
Falling Propagation Delay (ps)	16.13 (10 to 11)	24.83 (10 to 11)	41.51 (10 to 11)	73.47 (00 to 11)
Average Power Consumption (uW)	2.070	3.295	5.711	10.560

w=180nm l=50nm	Load Capacitances			
	1f	2f	4f	8f
Rising Propagation Delay (ps)	10.23 (10 to 11)	13.89 (10 to 11)	21.25 (10 to 11)	35.96 (10 to 11)
Falling Propagation Delay (ps)	10.24 (00 to 11)	14.37 (00 to 11)	22.37 (00 to 11)	38.37 (00 to 11)
Average Power Consumption (uW)	2.549	3.775	6.228	11.080

w=270nm l=50nm	Load Capacitances			
	1f	2f	4f	8f
Rising Propagation Delay (ps)	8.425 (10 to 11)	10.81 (10 to 11)	15.64 (10 to 11)	25.11 (10 to 11)
Falling Propagation Delay (ps)	8.638 (00 to 11)	11.19 (00 to 11)	16.43 (00 to 11)	26.83 (00 to 11)
Average Power Consumption (uW)	3.038	4.231	6.695	11.583

Deliverables

- **What input transitions will you expect to have the maximum rising or falling delay? Why? Do the experimental results match your expectations?**

I expect a transition where only one input is changing (01 to 11) or (10 to 11) to have a larger rising delay (t_{pdr}) than the case where both inputs are changing (00 to 11). This is because when one input is rising, it is the "worst case", and the PMOS pull-up network will have a larger resistance, resulting in a higher rising delay time.

I expect the falling propagation delay to be largest on the (00 to 11) transition, as all transistors are transitioning in this case. When compared to a single input transition, where other transistors are completely stable and only one is changing, a double input transition will take longer.

The experimental data generally supports these expectations. At the 90nm size, the (10 to 11) transition had the largest delay values for all situations except the 8f falling delay. The 8f falling delay was largest for the (00 to 11) transition, although all propagation times of the (00 to 11) transition were very similar to the (10 to 11) delay.

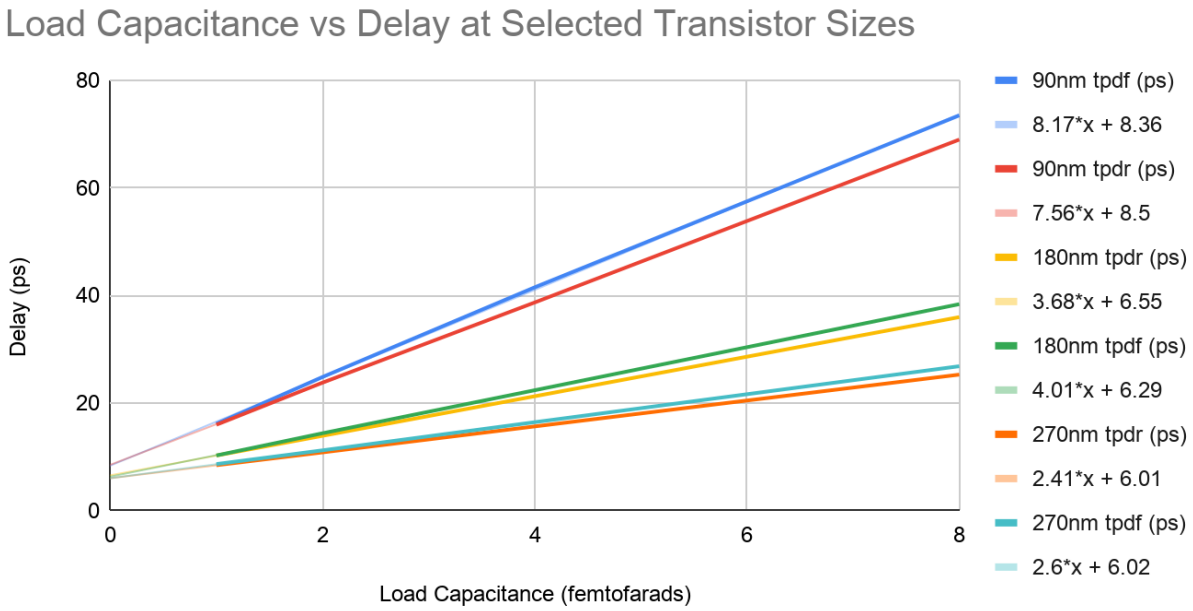
At the 180nm and 270nm sizes, the experimental data completely meets expectations. The rising propagation delay was consistently largest on the (10 to 11) transitions, and the falling propagation delay was largest on the (00 to 11) transitions.

- **Does the relationship between transistor sizes, load capacitances, and propagation delays follow the linear delay model? If yes, use figures to visualize the relationships and then derive the parasitic delay (in ps) and the logical effort (in ps·fF/nm) for your NAND gates.**

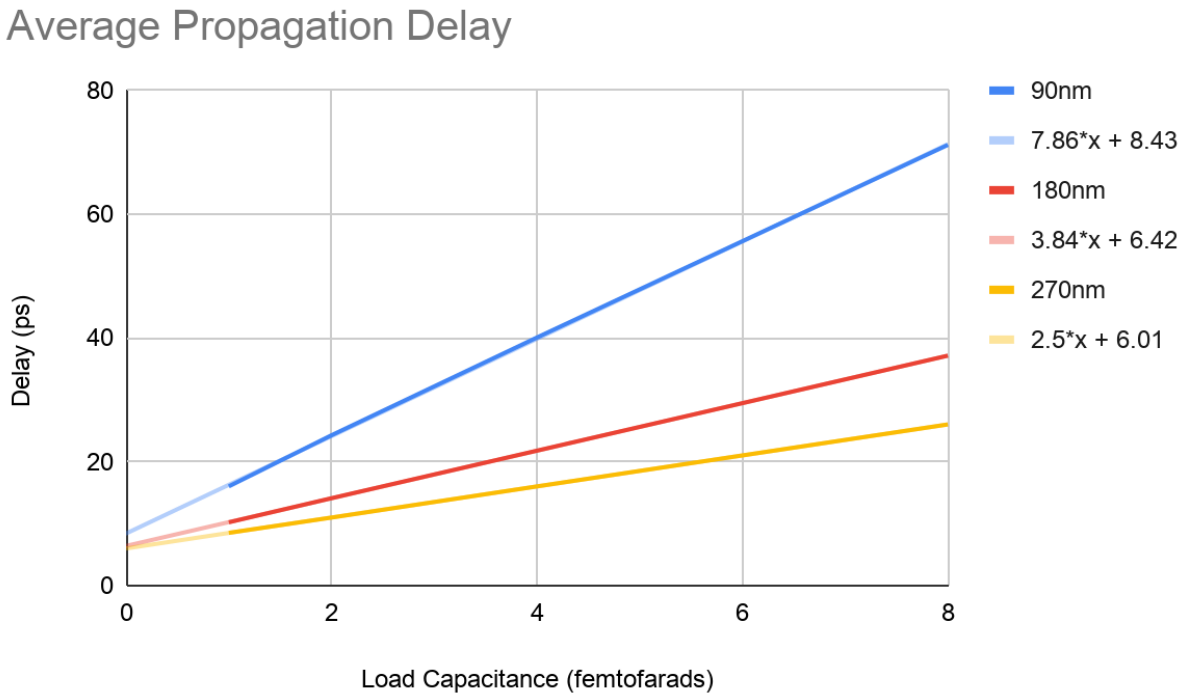
The relationship between transistor sizes, load capacitances, and propagation delays does largely follow the linear delay model. This can be seen by plotting the data and analyzing the results. In the linear model, the total delay of a gate is determined to be

$$D = g * h + p$$

Where g is the logical effort of the gate, h is the electrical effort of the gate, and p is the parasitic effort of the gate. The logical effort of a gate is intrinsic to the operation it performs, and for a 2-input NAND gate is 2. The electrical effort of the gate is proportional to the ratio of capacitance it drives (C_{out}/C_{in}). The parasitic delay is intrinsic to the transistor sizes used. This means that at zero output capacitance, the delay is completely due to the parasitic delay.



This graph shows the rising and falling delays of each transistor size as separate plots. To simplify this graph, the average propagation delay was computed by taking the average of the rising and falling delays at each transistor size and load capacitance. This graph is shown below.



Extrapolating the data to find the capacitance at zero (using linear trendlines), we find that the parasitic delay of the gates at each transistor size are as follows:

Transistor Size (nm)	Parasitic delay (ps)
90	8.43
180	6.42
270	6.015

To find the logical effort of each gate, we must then use the calculated parasitic delay for each transistor size, the total propagation delay, and the computed electrical effort to solve for the logical effort. Using the linear delay model, we can find the equation for logical effort of a gate to be $g = (D - p)/h$.

Using the 90nm gate as having size=1, 180nm as having size=2, and 270 as having size=3, we find that the input capacitance for each NAND gate is:

Transistor Size (nm)	Input Capacitance (unit-capacitance)
90	2
180	4
270	6

The electrical effort is then computed using the load capacitance and the input capacitance

Electrical Effort Computation	Load Capacitances			
	1	2	4	8
90nm	0.5	1	2	4
180nm	0.25	0.5	1	2
270nm	0.1666666667	0.3333333333	0.6666666667	1.333333333

This results with the logical effort being computed as follows:

Logical Effort Computation	Load Capacitances (femtofarads)			
	1	2	4	8
90nm	15.22	15.875	15.835	15.6975
180nm	15.26	15.42	15.39	15.3725
270nm	15.099	14.955	15.03	15.02625

Average Logical Effort	15.35

Because the input capacitance is given in terms of the unit transistor capacitance, the current logical effort calculations do not accurately reflect the logical effort of the gate. If the unit-capacitance value was known, we could then determine the actual logical effort of the gate

- **How power consumptions change as transistor sizes and load capacitances change?**

At all transistor sizes, the power consumption increased as the load capacitance size increased. This is to be expected, as it is harder to drive a larger load capacitance. Additionally, as transistor size increased, the power consumed by the circuit also increased.

- **Among dynamic power, static power, and short circuit power, what are measured in this lab?**

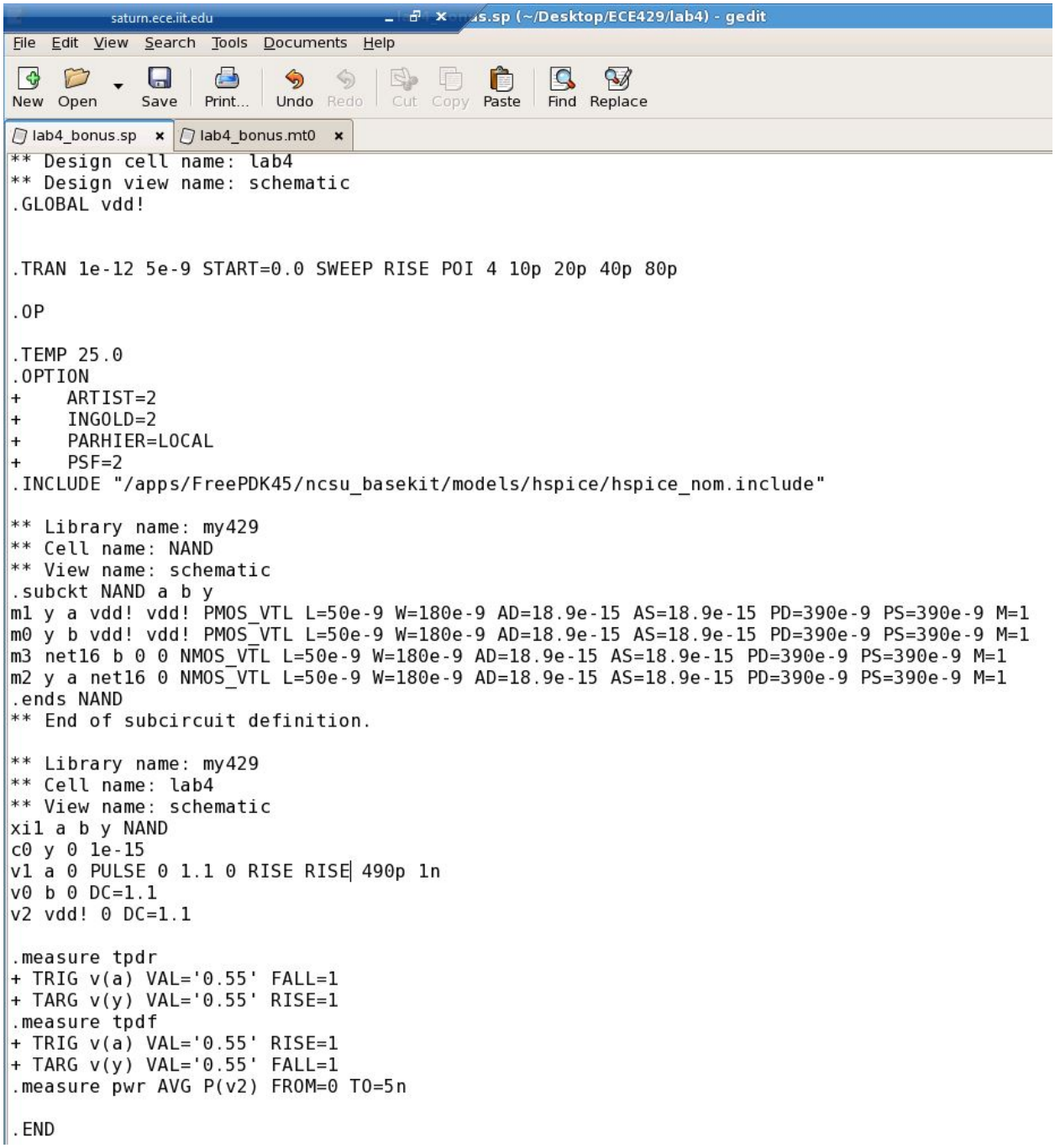
This laboratory measures all power consumed by the circuit (including dynamic, static, and short circuit power) because the power is measured in the SPICE simulation by analyzing the power provided by the Vdd source that supplies the circuit. In other words, the complete power used by the circuit is measured.

Bonus Work

Input slew rates, or the slope of the input signal to a CMOS circuit, has a considerable effect on the delay of the CMOS circuit. As described in textbook section 4.4.6.1, When the rise-time of the input is increased, the CMOS circuit takes longer to become fully ON (increasing the output delay). This effect is proportional to the rise time, meaning that increasingly longer rise-times will result in higher delays. Therefore, to minimize delay, the rise time of the input

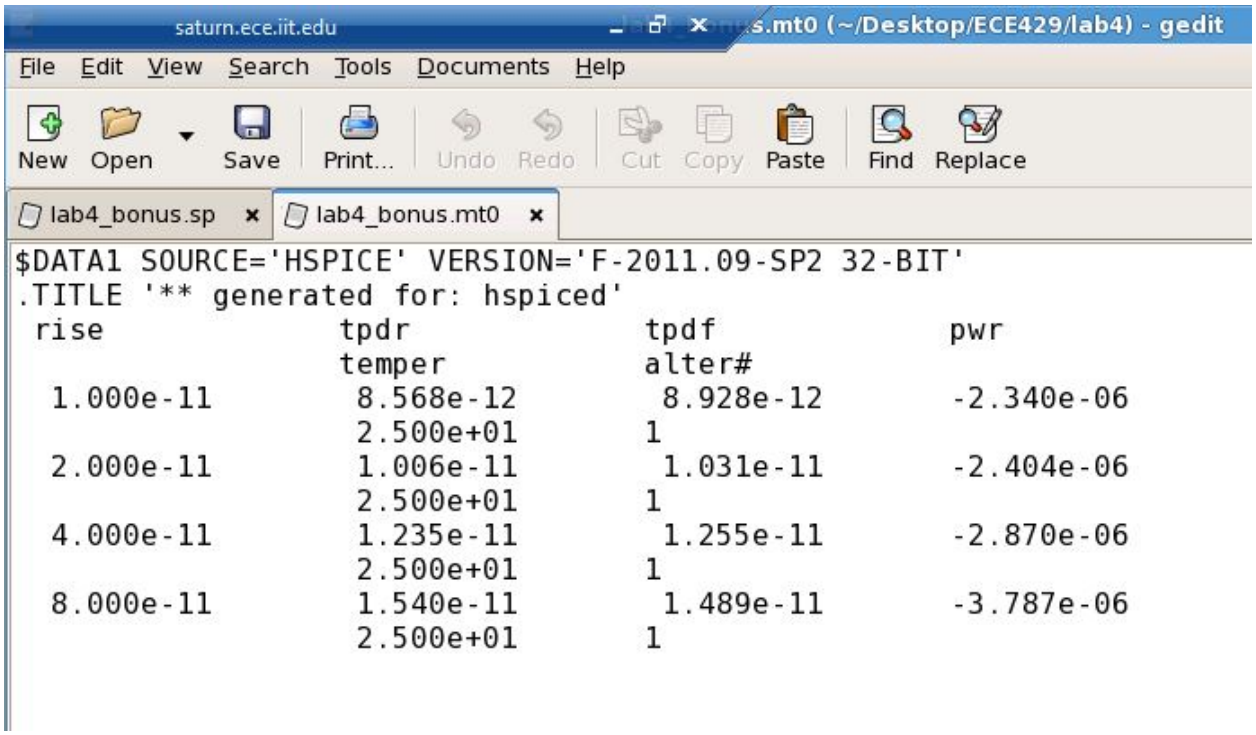
should be minimized (ideally would be a step function). In order to produce the fastest circuits, rise.

To test this relation, the rise and fall time could be varied much like the Capacitance was varied in this laboratory session. All other values, such as transistor size and load capacitance would remain constant. A netlist for this test is shown below.



Spice netlist for varying rise time

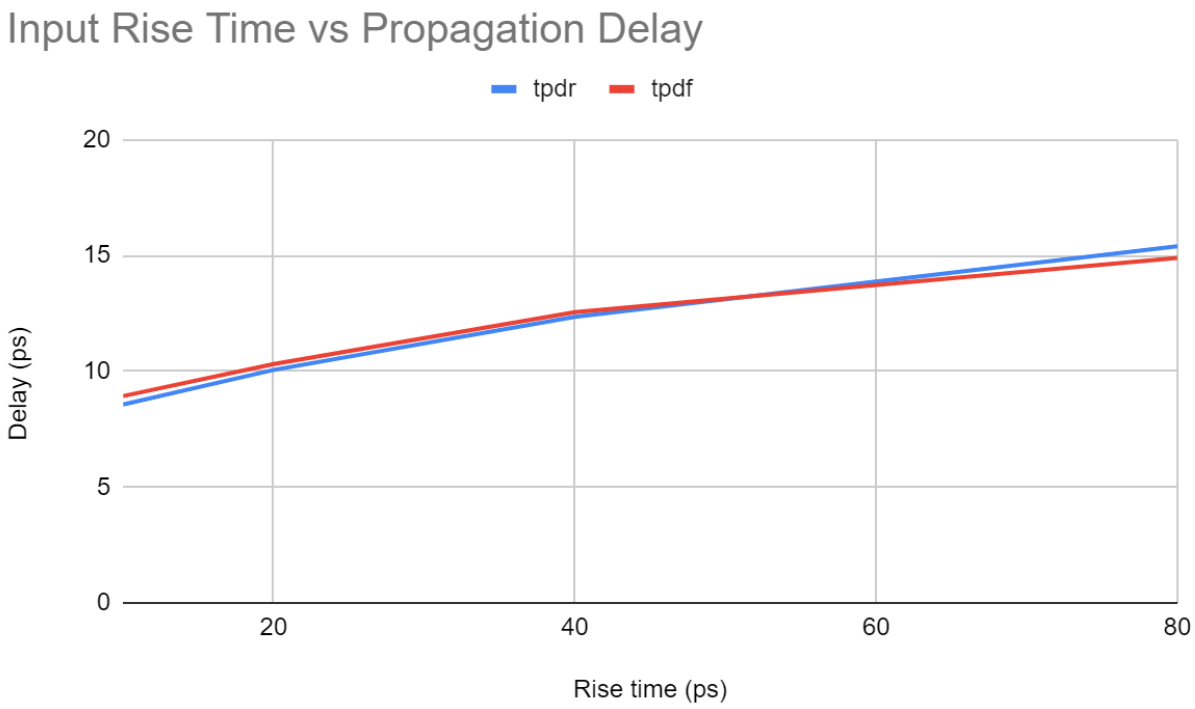
The above netlist varies the rise/fall time of the voltage source attached to input A. This results with the delay of the gate being increased relative to the input rise time. The load capacitance remains constant at 1fF, and only the 01 to 11 transition is tested using this netlist. The results of the spice simulation are shown below.



Spice simulation results for varied input rise time

Input rise time (ps)	Tpdr (ps)	Tpdf (ps)
10	8.568	8.928
20	10.06	10.31
40	12.35	12.55
80	15.4	14.89

Tabulated result data



Graphing the data for propagation delay of the gate (rising and falling propagation delays) shows a strong correlation between the slope of the input signal and the delay of the gate.

4. Conclusion

This lab should be considered a success. Students were successfully able to implement and simulate a 2 input NAND gate using CMOS logic. Students were successfully in using a testing circuit to generate SPICE netlists for the NAND gate at various transistor widths (90nm,180nm, and 270nm). Students then used these netlists to analyze the rising and falling propagation delay of the NAND gate at various different load capacitance values. This data was then tabulated and utilized to determine the parasitic delay and logical effort of the NAND gate.

Students were also successful in relating the slope of the input (input rise/fall time) to the propagation delay of the gate (tpdr and tpdf) in the bonus work section.

In future laboratory sessions, the NAND gate designed in this circuit will be utilized to build and simulate more complex CMOS circuits, such as an AND gate using a NAND and NOT gate. Students will later design and implement a physical layout for the NAND gate and compare the functionality of the layout with the schematic cellview.

5. References

- Choi, Ken. “ECE 429 Laboratory 4 Manual.” Illinois Institute of Technology, 01 Oct. 2020.
- Kim, Victoria. “ECE 429 Guideline for Writing Report & Grading Criteria.” Illinois Institute of Technology, 01 Oct. 2020.