Laboratory 2: Basic Boolean Circuits

Contents

Objective:

The objective of this experiment is to investigate methods for evaluating the performance of Boolean circuits. Emphasis will be placed on the basic CMOS inverter but the concepts are applicable to larger circuits. Time domain behavior of the inverter will be considered. Some of the file creation and manipulation commands that will be required in this experiment were covered in the previous laboratory experiment.

Part 0: Set up transistor models

Download the SetupModels script from the class website, like last week use

```
chmod 700 ~/Downloads/SetupModels.sh
```

to make the script executable then run it with,

```
~/Downloads/SetupModels.sh
```

and the models should be set up.

Part 1: Introduction

A digital inverter with Boolean input A and Boolean output Y is shown in Fig. 1a

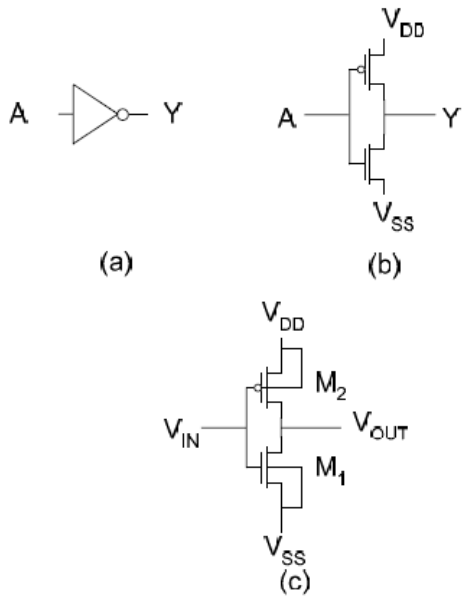


Figure 1 Digital Inverter a) Gate Representation b) Simple Transistor Implementation c) Transistor Implementation Showing Bulk Connections

This gate level representation provides no detail about the underlying circuit used to realize the gate. Fig. 1b shows transistor level detail for the basic CMOS implementation but does not show how the bulk is connected in the circuit. The representation in Fig. 1c shows all connectivity of the CMOS inverter. In the latter view, the input and output variables are labeled as voltage variables rather than as Boolean variables.

Part 2 Simulation of a CMOS Inverter

In this section we will focus on the creation of a schematic of a CMOS inverter and on the simulation of this inverter in Spectre. We will create this schematic in the library created last week after attaching a technology file to it.

Part 2.1 Attaching technology information

In the previous laboratory experiment, a simple circuit consisting only of resistors and capacitors was simulated. In this experiment an inverter implemented with transistors will be simulated. Initially, create a new cell named 'inverter' in schematic view. To use the correct models for the transistors, the "Technology" in which the inverter is designed must be associated with the schematic. By associating a technology (sometimes termed a "technology file") with a design, both the semiconductor manufacturer and the technology that manufacturer will use to fabricate the circuit is specified should we choose to fabricate the design as an IC. For the purposes of this course, we will use the 0.18µm CMOS process technology.

When the process association is complete, the TSMC 0.18µm process information will automatically be used in our designs. Process information in a technology file includes critical parameters such as the minimum device size, the available layer masks, the supply voltage level, and a detailed model of the transistors.

Attach the tsmc 0.2µ C5N technology file to the lablib Library.

Note: *Normally you would attach a technology library, when the "Technology File for New Library" window appears when you created a new library. But since we closed that window last week, we will present the other way here.*

In the CIW select **Tools** in the menu bar and select **Technology File Manager**, a "Technology Tool Box" window will appear. Click on the **Attach** button. An "Attach Technology Library to Design Library" window will appear. Select the library created last week (**lablib**) by scrolling down in the Design Library selector. Then select **NCSU_TechLib_tsmc02** by scrolling down in the Technology Library selector. Click **OK**. Check to be sure it has been attached. Do this by highlighting **Lab2** in the LMW, clicking on **Edit** in the menu bar of the LMW, and clicking on the **Properties** button in the drop down menu. The "….tsmc02" technology should appear in the techLibName field.

Part 2.2 Creation of a Schematic

Consider an implementation of the CMOS inverter of Fig. 1c in the tsmc 0.2µ CMOS process. Use a 3.3V power supply and size the devices M_1 and M_2 with the drawn dimensions given in Table 1.

Table 1: Device Sizes for Inverter

	W	L
M_1	0.3u	0.2u
M_2	0.9u	0.2u

Create a schematic view of the inverter in a new cell in your **lablib** library (for convenience use the name **Inverter** for this cell). After creating the schematic view cell, open the Schematic Editor Window (SEW) and enter the circuit schematic. For the NMOS and PMOS transistors, use the cells nmos4 and pmos4 from the **NCSU_Analog_Parts** library, respectively. NOTE: The transistors nmos4 and pmos4 have 4 terminals, not 3.

When editing the transistor properties, the lengths and widths need to be in the fields labeled "Width" and "Length", not in the "Width (grid units)" or "Length (grid units)" fields. When entering numbers in the Cadence design and simulation environment, you can use engineering notation such as k (10^3), M (10^6), m (10^{-3}), u (10^{-6}), n (10^{-9}), etc. **The suffix should be written right next to the number without spaces or unit**. For example, $10*10^9$meters can be entered as 10n or 10e-9 but not as 10 n or 10nm.

The labeled voltages on the schematic of Fig.1c include the voltage V_{SS} which is often set at ground but for generality we have designated it as Vss. These labeled voltages must be designated with pins on the schematic you enter in Cadence. Pins can be of different types depending upon the intended use of the pin. To add pins, go to **Add --> pin**. The process of adding a pin is the same as adding labels, that is, separate multiple pins by a space and then place them on the schematic. Be sure to wire the pins to the appropriate wires!

The voltages V_{DD} and V_{SS} should be bidirectional (inputOutput) pins. The voltage designated as V_{IN} should be an input pin and the voltage designated as V_{OUT} should be designated as an output pin. Also note that the **specific pin will be automatically connected to the wire labeled with the same name**. **Be careful while naming wires and pins. Later labs will require you to reference these, and all references are case sensitive. Pick a labeling style and stick to it!**

Part 2.3 Symbol Creation

To re-use the inverter we just created efficiently, we need to create a symbol for our schematic design. In the SEW, follow the click sequence **Create→ Cellview→From Cellview**. Verify the location information in the "Library Name" and "Cell Name" fields are correct. Then click on **OK**. A window with the symbol will be opened. Use the buttons in the editing pallet to change the symbol view to make it look like an inverter. A drop down editing menu can also be used by clicking on the right mouse button. Make sure you keep the pins consistent. This process simply creates a graphic - no wiring or electrical connections are used in the lines of the graphic. The critical electrical parts of the symbol are only the red squares where the exterior circuit will attach to the graphic symbol.

Part 2.4 Inverter transient response simulation

Now that we have a symbol for our inverter, let us set up the test bench. Create another schematic using the LMW called **inverter_TB** to test the design. The test bench is shown below.

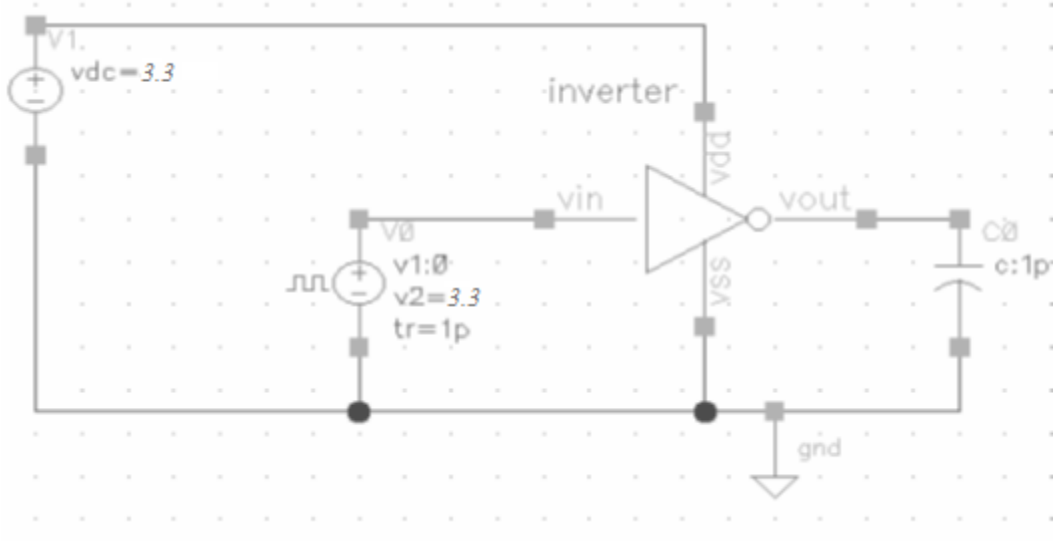


Figure 2: Inverter simulation test bench

Use the voltage sources from **analogLib**. For the pulse input source, you can use either the vpulse or the vsource component. You will need to apply an input square wave of frequency 1MHz with magnitude of 3.3V and rise and fall times of 1 picosecond each. To instantiate your own inverter, use the instantiation procedure as you would for any other component from analogLib but choose the inverter cell instead from your own library. Set the dc power supply voltage to 3.3V and the load capacitor to 1pF. **Check and Save** your work and correct all errors or warnings.

Open the Analog Design Environment and set up the transient analysis to run long enough for 5 complete input cycles. Select the input and output voltages for plotting and start the simulation. Did you create labels for those nodes of interest (recall how we used the labels vIn, vMid, and vOut in lab1)? Debugging and evaluation of results become easier with labels.

Part 2.5: Inverter transfer characteristics simulation

Modify the source and simulation environment to obtain the dc transfer characteristics (V_{OUT} vs V_{IN}) of the inverter for inputs voltages between 0V and 3.3V. *At what value if V_{IN} is the input and output equal? What is the minimum value of VIN that can be applied and still keep the output near Vdd? What is the maximum value of V_{IN} that can be applied and still keep the output near 0V?*

Part 2.6: Inverter driving a load

Return to transient analysis and increase the load to 10 pF. What do you notice at the output? What if you have a 100 pF load? Or a 1uF load? Explain what you think happens.

Revert back to a 10pF load and increase the width of the transistors of the inverter by a factor of ten, then simulate. Revert the widths and increase the lengths of the transistors by a factor of ten, run the new simulation. *What happened, and why did it occur?*

Return your inverter back to its original state when finished.

Useful Equations to consider:

Charging speed (which is also called Slew Rate) = $\frac{I}{C}$

Current of the transistor is directly proportional to W/L ratio $I \propto \frac{W}{L}$

Part 3: Cascaded Inverters

We will now cascade two inverters and analyze how the signal propagates from one stage to the other. Create an inverter_cascade_TB cell so it looks like the following figure:

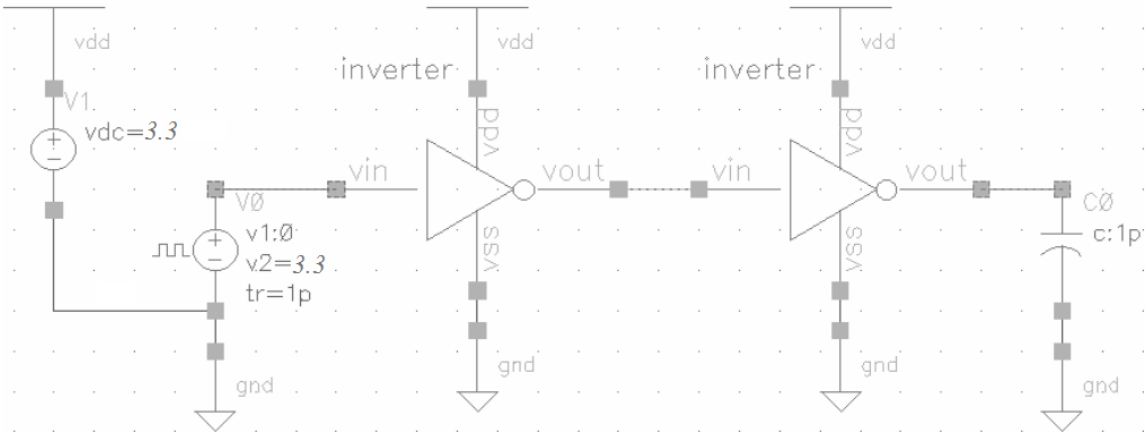


Figure 3: Cascaded inverter simulation

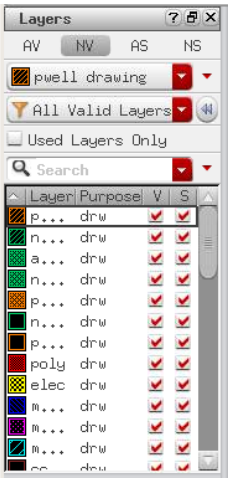
The "antenna" on top of the vdc source and the power supply pin of the inverters is a visual aid and a method of connecting multiple power supply nets together. The antenna shaped cell is called vdd and is available from analogLib.

Run a simulation and plot the three voltages of the circuit: input of inverter 1, output of inverter 1, and output of inverter 2. Label these nodes before running the simulation.

Compare the output of inverter 2 with the input of inverter 1. Do they look the same? Do you see any loss of signal integrity? Is there any noticeable delay?

Part 4: Introduction to Layout

The layout view is a representation of how an integrated circuit will look from the top once it is fabricated. Create a layout view of the inverter cell you created above. This can be done in the schematic view by selecting **Launch** → **Layout XL** or from the Library Manager through **File** → **New** → **Cell View** → **Layout** in the Inverter folder.



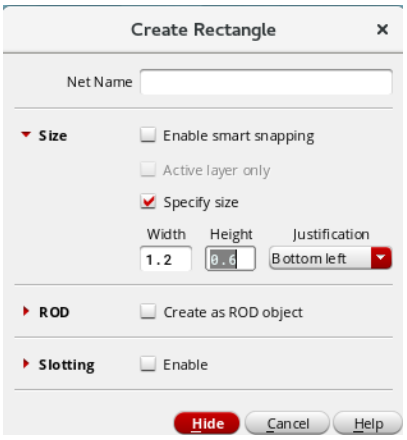
The **first thing to check** when you open a new layout for the first time is to make sure that the technology file is attached correctly. To do this check the layers selections (on the left of the screen) and each layer should be colored differently (see image on the left). If the boxes are not different colors there is a problem and you should bring it up to your TA. You can select different layers by clicking them in this select bar.

The first thing to do in layout view is get used to how it works. Initially the most useful tool is the draw rectangle. To do this, select **Create** → **Shape** → **Rectangle** (**shortcut key 'r'**). This will bring up a small box with options, we will ignore this for now but when something is not working right **check the options box** for that process.

Go to the middle window and **left click once**. Then **move the mouse**. **Click again** and the program will make a rectangle. **Click and Drag will move the screen**, not make a rectangle.

In the layout editor window, notice how the cursor snaps to the closest coordinate unit. To check your grid settings, go to **Options** → **Display** → **Grid Controls**. It is suggested that the *X & Y snap spacing be at 0.15* and *the minor and major spacing be at 1 and 5 respectively*.

Next let's make a rectangle exactly 0.6 microns by 0.2 microns. To do this press 'r' but **do not ignore** the pop up box this time. Instead of using the "draw" option click on the Size drop down and check the "Specify size" box. Put in 0.6 and 0.2 into the two option boxes. Return to the layout and create three boxes of the same size using Metal 1, Metal 2, and Poly by clicking with these options set. (Image on right is old, but close.)



You can use the ruler to check your dimensions (shortcut key "k"). If the screen becomes cluttered with rulers, go to **Tool** → **Clear All Rulers (shortcut key "K")**. Make sure you experiment with the buttons to the left as well as the menu to get comfortable moving, copying, stretching, cropping or rotating a rectangle; the options are endless.

After getting a picture of the part above, delete it. Finally we are going to create a single transistor layout. A transistors physical layout is made of many parts

- Well – a lightly doped region around the transistor that is doped the opposite of the active region. NMOS transistors need a P-well and PMOS transistors need an N-well. Due to the settings in Cadence we assume a P-well crystal so we do not need to place a P-well layer for NMOS devices.
- Select –This is a region surrounding the Active region of a transistor, inside the well. It is required for manufacturing reasons.
- Active – The Active region is the region that is heavily doped and is the portion of the main body of the transistor.
- Gate – This is a poly that passes across the transistor touching both ends of the select. It determines what state the transistor is operating in.
- Source/Drain – Physically identical in our process the source and drain are the two regions the active area is divided into by the gate.
- Vias – Multiple vias are needed to electronically connect parts of the transistor. The vias we will use will be,
    - M1_P – used to connect metal to the source and drain for PMOS devices, as well as the bulk connection for NMOS devices. To attach the bulk of an NMOS you just have to place an M1_P connection from the metal that will connect to Vss to the general bulk.
    - M1_N – used to connect metal to the source and drain for NMOS devices.
    - NTAP – used to connect the bulk of PMOS devices. This is made up of an N-well and a via, attach it to the metal connected to the source side while keeping it outside the select region.

To create a via go to **create** → **via**.

With the above information attempt to create a PMOS transistor, this may be difficult so ask your TA if you get stuck. Once you believe you have made it go to the top tool bar and select **verify** → **extract**. This will create a new cell view called “extracted”. Open it and there should be a box that says pmos4 if you have built the device correctly. Press **Shift+F** to be able to see more information (Width, Length, etc), and **Control+F** to return to the box that says pmos4.

Tip: Familiarize yourself with the user interface before proceeding to the next lab. This will save a lot of time next week. Make sure to check the keyboard shortcuts on the next page.

Useful keyboard shortcuts in schematic view:

Action	Key
Add Instance	i
Add Pin	P
Wire	w
Undo	u
Redo	shift +u
Properties	q
Rotate	r
Copy	c
Check and Save	F8
Zoom to Fit	f
Move	m
Wire Name	L

Useful keyboard shortcuts in layout view:

Action	Key
Create rectangle	R
More detail in layout	shift + f
Less detail in layout	ctrl + f
Stretch rectangle	S
Zoom to Fit	F
create ruler	K
clear all rulers	shift + k
Undo	U
Redo	shift +u
Copy	c
Properties	q