

CSCE 612
Lab 1
Estimation of Logic Speed in the AMI C5N Process
Due Date: 2 / 9

Introduction

MOSFETs are extremely complex devices. As such, simulation of well-developed models is usually preferable to pencil-and-paper analysis. Our MOSFET models consist of BSIM Level-3 models and include information that describes the behavior of the MOSFET's gate capacitance, parasitic capacitance, effective resistance, and I-V behavior over a broad range of operating environments. This information is encoded as a set of process characteristics that are interpreted by the simulator.

In this lab, you will use two simulation techniques for measuring logic speed in the AMI C5N process. Both are common techniques for estimation of the maximum speed at which you can expect to operate logic implemented in a particular fabrication process (given a predetermined maximum logic depth, which is the maximum number of logic levels within a pipeline stage).

Schematic simulations are sufficient for this assignment. In other words, you will not be required to simulate extracted layout.

Unit Transistor Width

Before we begin, we need to make a final decision regarding the minimum width for the NMOS (and thus PMOS, assuming PMOSs have 2X width) in the AMI C5N process.

Nominally, we'd want our minimum-sized NMOS transistor to have $W=4\lambda$, $L=2\lambda$ and our minimum-sized PMOS transistor to have $W=8\lambda$, $L=2\lambda$. This translates into $W_n=1.2\mu\text{m}$, $W_p=2.4\mu\text{m}$ with both transistors having the minimum channel length of $0.6\mu\text{m}$.

However, the minimum width setting allowed by the NCSU design kit is $1.5\mu\text{m}$ (5λ or 10 grid units). In addition, I've been told by experienced AMI C5N designers that there should be a minimum of two contacts per diffusion area for any C5N MOSFET. In order to fulfill this requirement, the minimum width must be bumped up to $3\mu\text{m}$, which is 10λ or 20 grid units!

Therefore, we will assume that a unit transistor width is $3\mu\text{m}$, meaning that a minimum-sized logic gate would have $W_n=3\mu\text{m}$ and $W_p=6\mu\text{m}$. Therefore, $3\mu\text{m}$ represents a unit width, or size multiple of 1.

Task 1: Estimating FO4 Delay

Our first technique for determining logic speed is to determine the delay for a fan-out-of-4 inverter (FO4). In the tutorials you were guided through a simplistic FO4 delay estimation.

However, the tutorials' estimation used an ideal (pulse) input for the inverter under test. This hinders accuracy because the delay of the inverter under test is dependent on its input slope (rise and fall time). One way to obtain a realistic input slope is to drive this inverter with a pair of FO4 inverters in series.

In addition, the “load inverters” in the tutorial did not drive a realistic load themselves. This is unrealistic, because the input capacitance of the load inverters also depends on *their* load. This is because the Miller Effect causes the total input capacitance (specifically C_{gd}) of the load inverter to increase with the speed at which we allow it to switch). Therefore, we must “load the load inverter” to make the FO4 estimation more accurate.

Figure 1 shows a test circuit. In this circuit, the NMOS and PMOS transistor widths (in multiples of $W=10\lambda$) are shown below and above each gate, respectively (the Xn notation refer to the names of the inverters only – they don't represent the sizes of each inverter). The input to the X1 inverter is the same pulse source as used in the tutorial. Use the nominal models for both the NMOS and PMOS.

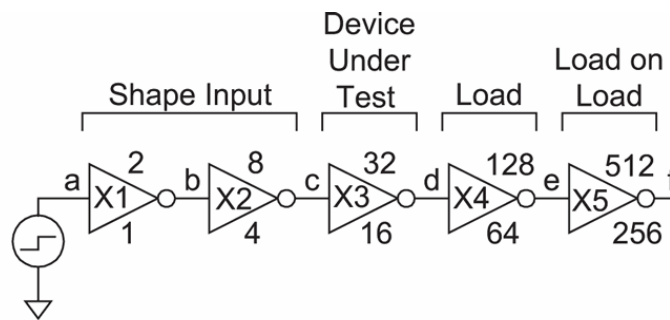


Figure 1

Notice in Figure 1 that instead of physically driving 4 independent inverters, the inverter under test is driving a single inverter whose width is 4 times its own. The capacitance of this gate is equivalent to driving 4 separate, fanned-out inverters of the same size.

Design this circuit hierarchically and simulate it. Plot the input and output of the device under test (X3) and determine its delay. Perform this test using a supply voltage of 2.5 V and 5.0 V.

Task 2: Ring Oscillator

Another way logic speed is measured for a fabrication process is by constructing a ring oscillator. A ring oscillator is constructed by connecting an odd number of inverters in series, as shown in Figure 2.

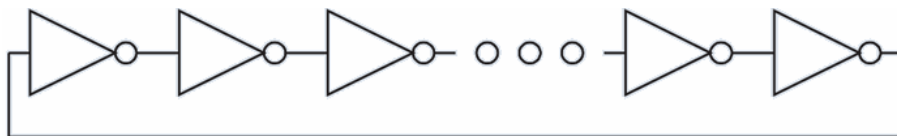


Figure 2

This produces a 50%-duty cycle square wave on each connection between two adjacent inverters (each having varying phase). This is caused by the constant imbalance between the number of 0's and 1's in the ring. This is why the ring oscillator will only work if there are an odd number of inverters composing the ring.

Ring oscillators are often used to compare the speeds of two different processes or to judge if a particular chip is faster or slower than nominally expected.

Construct a 31-stage ring oscillator using $W=10\lambda$ inverters. Use the nominal models for the NMOS and PMOS. When you simulate the inverter, you'll have to set an initial condition on one of the connections between a pair of inverters. To do this in Spectre, select Simulation | Convergence Aids | Initial Condition. Type in 2.5 for the node voltage, select one of the nodes on the schematic, and click OK.

Note that your delay numbers for this task will not match the delays you measured in task 1, since in this case we're using FO1 inverter delays instead of FO4 inverter delays. The goal of this task is to measure the period of the oscillator. Once you've measured the period, you may divide this value by 31 to approximate the delay of a single inverter.

Perform this test using a supply voltage of 2.5 V and 5.0 V.

What to Submit

For **task 1**, submit all appropriate schematic plots and simulation waveform plots.

For the designs, submit a schematic of the top level design as well as the design of all 5 inverters. If you have a **hierarchical** design (with instanced inverters), you'll need to submit a separate schematic plot for each inverter. If you have a **flat** design, you'll just need to submit a single schematic plot.

For the waveform plots, your waveform plot should clearly show the rising and falling delay for the inverter under test. In these plots, be sure you're zoomed in such that the delay between the input and output edges are clearly shown with the actual delay values determined using delta cursors. Use the 10%/90% swing technique (explained in the tutorials) to determine the locations of the edges (90% for rising edge, 10% for falling edge).

For example, to measure the rising delay for the inverter, measure the time difference between the time at which the input falls (10% of supply voltage) to the time at which the output rises (90% of supply voltage). Do this for the 2.5 V supply and the 5.0 V supply simulations.

Note: the purpose of this exercise is to measure the logic delay of the inverter under test. This is different from measuring the rise and fall times of the inverter output, which only measures the inverter's switching speed and not its logic delay. The rising delay and falling delay **may not** be equal. However, the rising and falling time of the inverter's output ***should indeed*** be (approximately) equal, otherwise the pull-up and pull-down network are not balanced.

For **task 2**, submit appropriate schematic plots (top-level and base inverter) and a plot of the simulation waveforms. For the waveform plot, submit a waveform zoomed such that the plot clearly shows one period of the oscillator with delta cursors showing the period time. Do this for the 2.5 V supply and the 5.0 V supply simulations. For both plots, divide the period by 31 in order to estimate an FO1 delay.