EE241B Advanced Digital Integrated Circuits **Spring 2018.**

HOMEWORK 3. **Due: Thursday, March 22, 2018. in class.**

This is an individual assignment!

1. Extracting and simulating the synthesized design.

In the first lab, we have created a design by using a strictly digital ('VLSI') design flow. Then in the second lab, we extracted and simulated the synthesized design using custom design tools. In this third lab we will close the loop and build the remaining part of the flow where we will characterize a custom cell for use in the synthesis flow.
To do this, please work through the tutorial entitled: *Introduction to the Custom Design Flow: Building a standard cell*, then turn in:

a) Report the critical path of the 3-to-8 decoder reported by IC Compiler before and after using your new NAND2 cell. Attach a layout of your cell placed inside IC Compiler (set level=99 to see inside the instance).
b) Using a Monte Carlo simulation of 300 points, plot the mean and sigma for the input-rising/output-fall delay of INVX0_RVT (10ps input slope, 1fF load) for VDD between 0.2 and 1 on the same plot. Then plot σ/μ vs. VDD on another plot. Lastly, plot the delay of a 6σ cell relative to the mean vs. VDD on a third plot. Explain conceptually why variation has more of an effect at low voltages.
c) Generate a histogram of the delay at 0.6V and at 1.05V and try to fit a normal curve (use histfit in Matlab). Are either of them Gaussian? If a distribution is asymmetric, explain why one side has the long tail and not the other. (Optional: use the qqplot command to compare two vectors: one vector from your simulations, and the other sampled from a normal distribution with the μ and σ found from the fit using normrnd)
d) Compute σ_{Vth} for the NMOS in INVX0_RVT. Then, using Ion from the alpha-power-law model (K=1.7e-4, Vth=0.366, alpha=1.36), write out the equation for the delay of an inverter (simply assuming maximum current for a transition from VDD to VDD/2 is close enough: $\Delta T = C\Delta V/I$, where ΔV =VDD/2 and C is approximately 1.5fF). Then use Matlab to run a Monte Carlo simulation on this model for 0.6V and 1.05V by using normrnd with a μ=0 and $\sigma=\sigma_{Vth}$ and adding this value to the threshold (which is exactly what HSPICE is doing for Monte Carlo simulation, but HSPICE is solving more complex equations). How do the distributions compare to part c)? Based on the equation you wrote, should the delay be Gaussian?

2. SRAM

Consider a conventional 6-T SRAM cell, sized to be stable at nominal operating conditions. The cell is supposed to be operated within a conventional, precharged bitline array. We would like to operate it at a reduced supply voltage so we are considering assist techniques. Let's analyze the effect of peripheral signals on the operation of the cell.

a) How does increased wordline voltage affect the read stability of the cell?
b) How does increased wordline voltage affect the read access time of the cell?

c) How does increased wordline voltage affect the writeability of the cell?
d) How does shortened wordline pulse affect the read stability of the cell?
e) How does lengthened wordline pulse affect the writeability of the cell?
f) How does increased cell supply voltage (without changing other signal levels) affect the writeability of the cell?

3. SRAM redundancy and ECC

We would like to investigate impact of variations on SRAM yield. Assume that read and write margins for the cells vary as Gaussian random variables with an average value of μ and a standard deviation of σ. The large array that needs to be put together consists of subarrays with 128 columns and 64 rows.

a) We would like to design a 36MByte array with no redundancy and a 90% yield. What should be the value of μ/σ for each of the cells to achieve this yield?
b) What is the μ/σ for the sense amplifiers needed to attain this yield? You can assume that cell and sense amp failures are independent.
c) What amount of redundancy (as a percentage of columns added per array) is needed to improve the parametric yield to 95% when designing an array from a)? Redundant columns are enabled through fuses during chip testing.
d) For comparison purposes, design an error correction code (may want to search some literature) that corrects for one error per wordline and detects two errors. What is the overhead of such a scheme? If error correction is being used instead of redundancy to correct for bit faults, and applied on the original design with a 90% yield what would be the final yield?

4. Timing

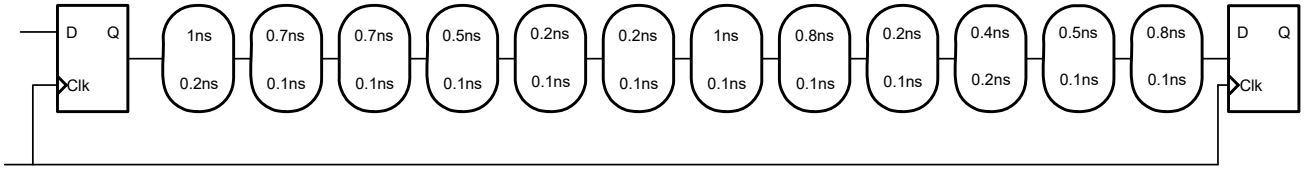

Figure 2.

A pipeline stage is shown in Figure 2. Static combinational logic is placed between two edge-triggered flip-flops. Each cloud of combinational logic has been annotated with its maximum (top) and minimum (bottom) propagation delays.
Your task is to minimize the cycle time by using deeper pipelining. You are allowed to add two more cycles of latency, by adding either latches or flip-flops between combination logic clouds in Figure 2. Available flip-flops have Clk-Q delay of 200ps, setup time of 100ps and zero hold time. Available latches have both Clk-Q and D-Q delays of 150ps, setup time of 100ps, and zero hold time. You can ignore the clock skew. You can add latches or flip-flops in between any two logic clouds, but you cannot break up a logic cloud.
Insert an appropriate number of pipelining elements (latches or flip-flops that minimizes the cycle time with two extra cycles of latency. Clock duty cycle is 50%