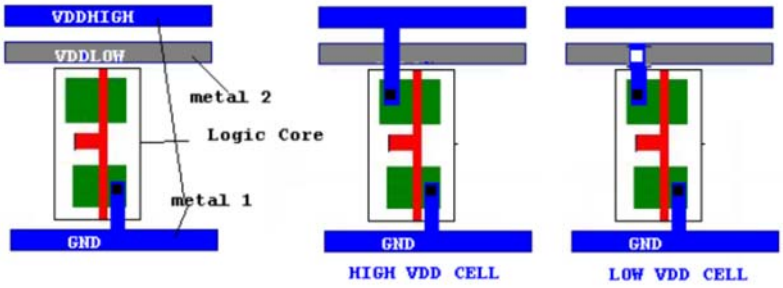
NAME_____

EN1600: Design and Implementation of VLSI Systems

Homework Assignment #4
Due **Friday, November 11, 2016, by 5pm**

1) Consider the C^2MOS master-slave positive edge-triggered flip-flop discussed in class.
   (a) Sketch the transistor-level implementation. Size the transistors for approximately equal rise/fall delay. As with other homework, assume a minimum sized transistor has W_{min}=150nm, and L_{min}=50nm.
   (b) As we discussed in class C^2MOS flip-flops can embed logic between master and slave stages as long as it is non-inverting. For this assignment, you need to embedded a 3-input AND function between the stages. Connect the output of the master latch to one of the inputs of the AND function. The output of the AND should be connected to the data input of the slave latch. Size the AND logic appropriately.
   (c) In Virtuoso, create a schematic view for the circuit you designed.
   (d) Create a layout cell for your schematic. Run DRC and LVS.
   (e) Extract the netlist from the layout of your circuit using Calibre PEX.
   (f) Use a clock period of 5ns and assume that the overlap of the $\overline{clk}$ and *clk* signals is 200ps. Setting the 2 side-inputs of your AND function to be at V_{dd}, simulate your design using the calibre cell view for the following cases:
      i. The 0-0 overlap case
      ii. The 1-1 overlap case
   (g) Derive the setup and hold time constraints for your flip-flop using simulation results.
   (h) [BONUS QUESTION]: As discussed in class, one of the nice features of the C^2MOS flip-flop is that it is insensitive to clock skew, as long as the rise and fall delays of the clocks are not too slow. Simulate a case that shows the flip-flop failing due to slow rise/fall of the clock signals. You may have to change the clock period for these experiments.

2) In this problem you will use standard cells to do a layout of a complex circuit network using a **dual-V_{dd}** approach. This involves first creating a small library of the standard cells needed and then doing a hierarchical design of the complex structure. **Before you start, please read the guidelines given in the assignment and make sure you understand all the details.**
   (a) You have already created a library of standard cells from previous homework assignments. You can reuse those cells for this homework, but in addition, you will need to add 2 different rails for V_{dd} (one for high V_{dd}, and the other for low V_{dd}). You can copy the cells from the previous homeworks to a new cell that will be used in the **dual-V_{dd}** library.



The following are the guidelines/requirements for the standard cells in the library:

STANDARD CELLS

- **U**se 1.1V as high-V_{dd} and 0.7V as low- V_{dd} when you simulate your cells.
- VDDHIGH, VDDLOW and GND rails should run in straight lines from left to right. These rails should frame the core of the standard cells (i.e., the VDDHIGH and VDDLOW rails should run across the top of the core and the GND rail should run across the bottom.)
- On the high-V_{dd} cells you will make a connection to the VDDHIGH rail and for the low-V_{dd} cells you will leave the VDDHIGH rail without any connection and connect to the VDDLOW rail. The rails should not have any bends or jogs in them and should connect from one cell to the next by abutment.
- All cells should connect by abutment without any DRC errors.

INPUTS/OUTPUTS

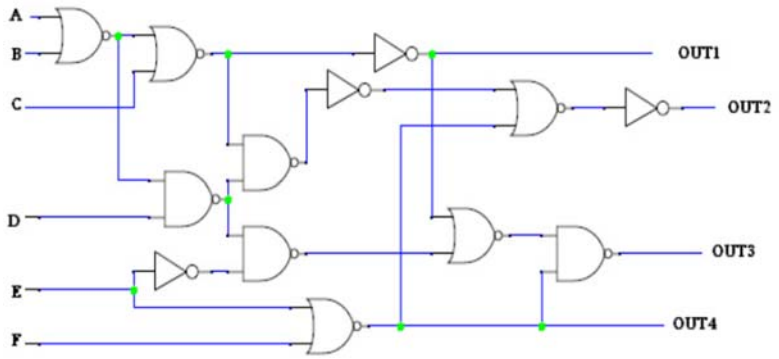
- Each input should be brought to each cell only once, and then split inside the cell.
- Do not route anything besides inputs and outputs outside your cells. The whole idea behind standard cells is that you can array them in all directions, and if you route outside the defined cell space then you prevent this flexibility in layout.
- Pay attention to how inputs and outputs are run to and from the cells. Don't assume that the gate will communicate with an adjacent gate. In other words, your cells are supposed to be generic. *Do not optimize them for any specific application*.

METAL ROUTING

- For ease of connection, use a Metal 1 track for GND and VDDHIGH rails. Use a Metal 2 track for VDDLOW rail.
- Inputs and outputs must come in and out of cells using *metal 2 only*. Vias must already be in place for the inputs and outputs within the cell.
- In order to connect inputs and outputs of a cell, a metal 1track must run *horizontally* below the row of cells. Then, **metal 2** wires will be connected *vertically* from the **metal 1** tracks to the input and output "pins" of the cell.

POLYSILICON ROUTING

- Polysilicon may be used to route signals a short distance within your cells. If you are routing a signal in polysilicon, make it wider than minimum. If you are making a transistor, use minimum width polysilicon (i.e., the length of the transistor is).
- If there is a choice between lengthening your polysilicon or your diffusion to fix a DRC, *always make the polysilicon longer*.
- Diffusion is approximately a factor of 10 greater in resistance.



(b) For the circuit shown above, the path from input A to output OUT2 is the critical path.

i. Layout the complex function using all high-V_{dd} standard cells.

ii. Extract the calibre view using PEX and simulate your design. Here you have 6 different inputs so simulating all 64 possible input combinations is not feasible. Simulate the following five input vectors <ABCDEF> = <000000>, <010100>, <101001>, <010110> and <111111>. Run only one simulation that combines the five input vectors. For this simulation, you can also estimate the power dissipation. You will have to extract the two power supply currents separately (one for VDDHIGH and one for VDDLOW) and create an expression for the instantaneous power in the ADE L calculator. You can then get the average power by integrating over one cycle of your simulation and then dividing by the duration of the cycle.
The expression in your calculator should look similar to the following:

```
integ((v("/vdd!" ?result "tran")*i("/V0/PLUS" ?result "tran")), 1u, 2u)/1u
```

iii. For the input vector transition from <010100> to <101001> measure the propagation delay for input A rising to OUT2 falling.

iv. Estimate the power dissipation for the high-V_{dd} design.

v. For the low-V_{dd} (V_{dd} = 0.7V), repeat part iii) and measure the propagation delay. You can do this part after either doing the layout again using low- V_{dd} standard cells or you can just change the supply voltage on your schematic to 0.7V and simulate to measure the delay for low-V_{dd} case.

vi. The whole point of asking you to design standard cells with two different V_{dd} values was to use a dual-V_{dd} based design to layout the complex function. Layout the complex function using a mixture of low-V_{dd} and high-V_{dd} cells. Try to use as many low-V_{dd} cells as possible but remember that a low-V_{dd} cell cannot drive a high-V_{dd} cell. Assume that there are flip-flops at the end of the output (you don't have to design them) that will restore the logic to the appropriate logic level.

vii. On a copy of the circuit, clearly label which gates were from the low-V_{dd} and high-V_{dd} cells.

viii. Repeat parts (ii), (iii) and (iv) for this dual-V_{dd} design.

ix. Comment on the differences in the values of the propagation delay that you see between the three different cases.

x. What was the total area of your dual-V_{dd} design?

NOTE: you don't have to run LVS again for each dual-V_{dd} cell since the schematic would be the same as the one for the standard cell from the previous homeworks. However, you will have to run LVS on the top view layout/schematic.