Introduction to CMOS VLSI Design (E158)

Harris

Lab 2: Full Adder Design

In this lab, you will design a full adder at the schematic and layout levels.

1. Full Adder

The objective of this lab is to complete a design of a nontrivial circuit, a full adder. There are many ways to approach this problem. You may assemble your adder from various simpler logic gates or may design it as a single complex cell. Refer to your textbooks for further references on full adders. Warning: at least one of the figures in Principles of CMOS VLSI Design regarding adders is incorrect. Be certain you understand a design before blindly copying it!

Copy your lab1_xx library to lab2_xx for this lab.

2. Schematics

Draw schematics of the adder in fulladder{sch}. Name the inputs *a*, *b*, and *c* and the outputs *sum* and *cout* to match the fulladder{ic} provided. It is wise to simulate your schematic to verify it works before proceeding to the time-consuming layout process.

3. IRSIM Simulation

Take advantage of IRSIM command files in your simulation. A sample `fulladder.cmd` file is in the lab directory. It includes tests of four of the input combinations. Open the command file with a text editor and look it over. The syntax of the command file is:

h sig:	set sig high (logic 1)
l sig:	set sig low (logic 0)
s [time]:	simulate (for [time] nanoseconds, otherwise use default step)
assert sig val:	check that sig has the value val. Emit warning if it does not
x signal:	set signal to invalid level

Assertions are very useful for larger designs because they allow automatic testing of your design without inspecting the waveforms for correctness.

Modify the command file to include the other input combinations with appropriate assertions. Be sure the file is in the same working directory with your lab 2 library. To

invoke the command file, start IRSIM on your fulladder{sch}. Use the Tools • Simulation (Built-in) • Read Vectors from Disk command to read your fulladder.cmd. Check the waveforms to verify that you obtain the correct sum and carry outputs.

4. Layout

Draw a layout of your adder in fulladder{lay}. The layout should obey the same constraints as your gates from Lab 1:

- power and ground run horizontally in Metal 2 on a 80λ center-to-center spacing
- all transistors, wires, and well contacts fit between the power and ground lines
- all transistors should be within 100λ of a well contact
- avoid long routes in diffusion
- *a* and *b* appear on left side of layout in metal1
- *sum* appears on right side of layout in metal1
- *c* appears near the bottom of a cell in metal1
- *cout* appears near the top of the cell directly above *c* in metal1
- metal 2 use is acceptable, but leave space for at least five horizontal metal2 lines to run over the top of the cell

Remember that you will have to connect *cout* of one adder to *c* of the next when you assemble a ripple carry adder next lab. Therefore, don't place any obstructions that would prevent the connections. *cout* output will be vertically connected to the *c* of next full adder directly above it. There should be no metal1 arcs below *c* or above *cout*.

You may find it necessary to add a large rectangle of N-well or P-well to surround your transistors and eliminate spacing problems. Use the Edit • New Pure Layer Node dialog to create a rectangle of either type of well and double-click on the well to set the size. Pure layer nodes don't offer connectivity information to Electric, so use them only for wells.

Use IRSIM to simulate your adder layout. Use the same command file that you created for the schematic simulation.

Check your layout with DRC, ERC, and NCC.

5. What to Turn In

Please provide a hard copy of each of the following items:

1. Please indicate how many hours you spent on this lab. This will not affect your grade, but will be helpful for calibrating the workload for the future.
2. What was unclear in this lab writeup? How would you change it to run more smoothly next time?
3. A printout of your fulladder schematic (and any subcells, if applicable).

4. A printout of your `fulladder.cmd` file.
5. A printout of your fulladder layout.
6. Simulation waveforms demonstrating correct operation of the fulladder layout.
7. What is the verification status of your layout? Does it pass DRC? ERC? NCC?

Optimization Contest

A prize will be given for the best full adder design. This primarily means smallest area. However, easy access to inputs and outputs and clean layout are also important.

Extra Credit

As you are probably aware by now, Electric has plenty of bugs and idiosyncrasies. A major goal of this class is to improve the stability and ease-of-use of Electric. Please email your bug reports directly to Prof. Harris in the format described in Lab Manual 1.