ECE 3450: Digital Electronics
Fall 2009, Lab #2

Dr. Pallav Gupta
pallav.gupta@villanova.edu

I Objectives

The goals of this lab are the following:

1. Design a nontrivial circuit - a fulladder.
2. Perform schematic and layout entry, simulation, DRC, ERC, and NCC for the fulladder.

II Courses Forum

The forum is located at http://pandim.ece.villanova.edu/phpbbforum. Please use it to ask questions.

III Submission Instructions

The lab must be submitted via the web at http://pandim.ece.villanova.edu. Instructions on how to submit are on the courses forum. Apply the following information on the Remository form when submitting your lab.

Filename:	`lab2-[first name]-[last name].zip`, *e.g.*, `lab2-john-doe.zip`
Location:	`ECE 3450/LAB2`
Title:	`Lab 2`
Author:	Your name, *e.g.*, `John Doe`

Note: Follow the above convention strictly. Failure to do so will result in a **zero**. We request your lab in this specific format so that the automated scripts at the back-end can run smoothly without breaking. Please adhere to it.

See Section X on how to name and organize the files that you will need to submit for this lab.

IV Collaboration

You must complete this lab independently; you are not allowed to work in pairs or a group. However, you are welcome to discuss the material with your colleagues.

V Prerequisites

You must have completed Lab #1 before proceeding further. If not, then it your responsibility to do that first. Otherwise, you will be totally lost in this lab.

VI Provided Files

All the files for this lab are provided in `lab2.zip`. Unpack this archive and you will see the following directory structure:

```
lab2/
├─README                                  readme
├─lab2-xx.jelib                          Electric library
└─sim/                                   IRSIM test vectors
  └─fulladder.cmd
```

The Electric file contains `fulladder{sch}`, `fulladder{ic}`, and `fulladder{lay}`. The `sim/fulladder.cmd` file contains a few of the test vectors to simulate the circuit using IRSIM. You will need to provide the remaining test vectors.

VII Designing the Fulladder

Recall that a fulladder has three inputs a, b, c and two outputs s, *cout*. There are many ways in which a fulladder can be designed. You can choose to assemble it from various simpler logic gates or design it as a single complex cell. If you opt to use logic gates, try to think in terms of NANDs/NORs rather than ANDs/ORs; you will save transistors.

You can refer to the textbook and/or other references to look up possible designs. Better yet, if you spend some time drawing and analyzing the Karnaugh maps of the fulladder (*Hint*: try re-expressing s as a function of *cout* (or $\overline{cout}$), a, b, c), you might be able to deduce a simple, yet elegant simplified static CMOS structure. This is because s, *cout* are symmetric functions. This will result in a single complex cell that computes both s and *cout*. Whatever you do, make sure you understand your design. Do not blindly copy from a book! Your schematic and layout must match!

Important: If your design requires complementary inputs (*i.e.*, $\bar{a}$, $\bar{b}$, $\bar{c}$), you will need inverters. Do not assume that the complements are available as extra inputs!

If you choose to build the fulladder from other logic cells (*i.e.*, hierarchical design), you can use the cells that you designed in Lab #1. Simply copy the cells from that lab into your library.

My personal recommendation is to design the fulladder using a single complex cell.

VII.A Schematic

Draw the schematic of the fulladder in `fulladder{sch}`. Name the inputs `a`, `b`, `c` and the outputs `s`, `cout` to match the icon provided in `fulladder{ic}`. Set the widths of all the nMOS and pMOS transistors to 8 and 16, respectively. Note that this is only applicable if you are building the fulladder as a complex cell. If you are building it from other cells, do not change the widths of the transistors in those cells. Just use the cells and interconnect them appropriately.

Make sure you export the inputs, outputs, and power and ground lines. It is helpful to label the internal wires in the schematic so they have meaningful names in case you need to debug later. Perform DRC and IRSIM simulation on the schematic. You will need to add the missing test vectors in `fulladder.cmd`.

VII.B Layout

The layout must obey the same constraints as those in Lab #1. In particular, keep the following in mind:

1. Power and ground lines run horizontally in 8λ-wide `metal1` on a 90λ center-to-center spacing; appropriate well contacts should be placed on the rails every 4λ.

2. All transistors, wires, and well contacts fit between the power and ground lines.
3. All transistors should be within 100λ of a well contact.
4. Avoid long routes in diffusion.
5. All inputs and outputs must be placed in `metal2` contacts and aligned with well/substrate contacts.
6. `a`, `b`, `c` appear on left side of cell in `metal1` (runs horizontally); `s` appears on right side of cell in `metal2` (runs vertically); `cout` appears near the the top of cell directly above `c` in `metal1`.
7. `metal2` use is acceptable, but leave space for at least five vertical `metal2` wires to run over the top of the cell.
8. You may find it necessary to add a large rectangle of `N-well` or `P-well` to surround the transistors and eliminate spacing problems.
9. Export all inputs, outputs, power, and ground; inputs and outputs should be aligned with the well contacts.
10. Create a neat and clean layout to avoid problems in the following labs.

Important: Before doing any layout, it is critical to draw an accurate **stick diagram** of the cell on paper. Once you are confident, it is then easy to transfer it onto an actual cell layout. I strongly recommend that you follow this practice to save yourself a lot of time (and possible misery).

Remember that you will have to connect `cout` of one adder to `c` of the next adder (*e.g.*, ripple carry adder, ALU datapath, *etc*). Therefore, do not place any obstructions that would prevent the connections. `cout` will be vertically connected to the `c` of the next fulladder directly above it. There should be no `metal2` arcs below `c` or above `cout`.

Perform DRC, ERC, NCC, and IRSIM simulation on the layout.

VIII Technical Report

A technical report (not to exceed 10 pages) is to be written that details everything you have done in this lab. You should present the design of the fulladder, and show the schematic and layout. You may also include stick diagrams. Furthermore, you should show the results of the simulations, and whether the layout passed DRC, ERC, and NCC or not. Elaborate on any difficulties faced in this lab and the employed workarounds. Summarize what you have learned from doing this lab.

The technical report must be of the highest standards. Otherwise, it runs a high risk of being rejected which will impact your lab grade. You can consult some technical publications to see how to write a good technical report. It must be written using the LaTeX template that was provided earlier. The template can be downloaded at http://pandim.ece.villanova.edu.

IX Parting Words

Congratulations on finishing this lab! Hopefully, you now have some experience in designing circuits that form the building blocks of larger and complex systems.

X What to Submit

For this lab, you must submit the following files:

1. The Electric library. Name it `lab2-xx.jelib` where `xx` are your initials.

2. The LaTeX PDF file which contains the technical report. Name it `report-xx.pdf` where `xx` are your initials.

Take both the files and archive (zip) them into a folder. Name the folder `lab2-[first name]-[last name].zip`. See Section III on how to submit the archive. Failure to follow these instructions will result in a **zero** for the lab. No ifs, buts, *etc.*

Important: The Electric library must contain the schematic, icon, and layout of the fulladder. If other cells were employed, include them as well. All schematics must pass DRC and IRSIM simulation. The layouts must pass DRC, ERC, NCC, IRSIM simulation, and be drawn as per specification (outlined above). Icons must be of the correct size.

XI Errors

I usually write precise tutorials and bug-free code. However, I am human (do not be surprised) and do make mistakes. In addition, CAD tools get updated frequently and the interface might change, rendering parts of the writeup ineffective. If you find any mistakes or inconsistencies while doing this lab, please bring it to my attention **immediately**. You will earn extra points if what you report is indeed a bug.