ECE451: VLSI Systems and Design
Lab 0: Tutorial
Due: January 12, 2015 (before lecture)

1 Introduction

The purpose of this lab is to get you familiar with the software we use in the labs. We will use the SUE Design Manager and MAX Layout Environment tools from Micromagic Inc. (`www.micromagic.com`). We use **SUE** to do schematic entry and simulations, and **MAX** for doing full-custom layout. At the end of this lab you should become familiar with the tools as well as the design methodology for the labs. Although you may be able to find other ways of doing things, please refrain from diverging from the tutorial. If you discover any problems with the tutorial or the Micromagic tools, please post them on the discussion board or send an email to `sz.lulec@mail.utoronto.ca`. DO NOT contact Micromagic Inc. directly.

Please read FAQ.pdf file that is posted under the Course Material in portal. This file contains issues that are frequently confronted by students during lab sessions. Refering to this file may save you a lot of time. Also, please check this file regularly since there will be updates throughout the semester.

You work **individually** in this lab. Starting from the next lab, you will work in pairs. Please find a partner as soon as possible.

2 Software Setup

The Micromagic software is accessible from all UG machines. To setup, simply append these lines to **the end** of the `.cshrc` file in your home directory, then logout and re-login:

```
# ECE451
source /cad2/ece451/SOURCEME
```

Now you should be able to start SUE and MAX by typing command `sue` and `max` respectively.

There are a limited number of licenses available so you should start doing the labs as early as possible.

3 SUE Tutorial

1. Type `mmi_tutorial` at a command prompt.
2. Select *SUE* in the *Which Tutorial* field and click *View PDF Tutorial in Browser*. This will start up Adobe reader in a browser.
3. Start the tutorial on page 1 (which is page 5 of the PDF).
4. Skip everything from Section *Using Pulsegen* (page 37) to the end of this part (page 54).

5. Try *Cross Probing with MAX* (page 55) after doing the MAX tutorial (Section 5). Ignore the warning about the outdated file format of `FA.max` when it is opened in MAX.

   If MAX does not launch after clicking *Done* in the *MAX Cross Probe Init* window, do the following: Start MAX using the command `max -tech mmi18 FA.max` in the same directory (../tutorial/sue/). Discard the warnings. Now you should have both FA.sue and FA.max opened in SUE and in MAX, respectively. Select *MAX Cross Probe Init* in *Sim* tab in SUE. Select *MAX* for *Which MAX* option and write *mmi18* for the *MAX Technology*. Click *Done*. The rest is same as the tutorial.

4 IRSIM Tutorial

IRSIM is an open-source switch-level simulator. Unlike SPICE, it models transistors as *ideal switches*, ignoring most of the higher-order and analog properties. A tutorial of IRSIM is available online at:

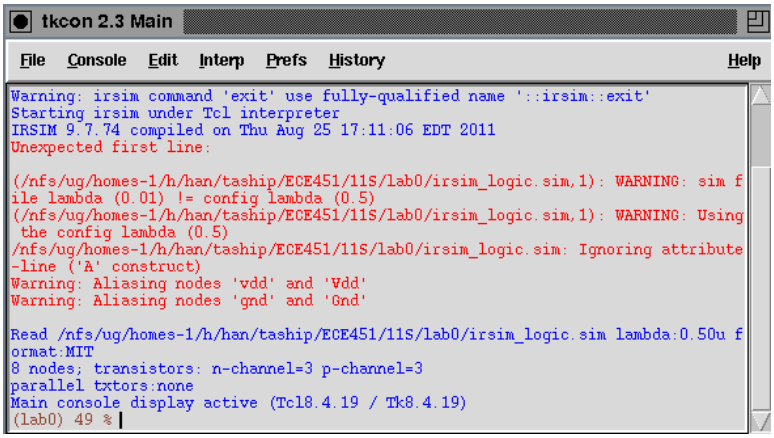
`http://opencircuitdesign.com/irsim/tutorial/tutorial.html`

While the tutorial presents IRSIM as a standalone tool that takes a circuit netlist (`.sim` file) from any schematic capture or layout editor, we use it as an integrated component of SUE and MAX. Therefore, ignore the instructions in Section *Creating a .sim file* and *Starting IRSIM* and do the following instead. However, do read these sections for a general understanding of the `.sim` file format and the command-line arguments of IRSIM.

1. Start SUE by typing `sue` at a command prompt.

2. Construct the CMOS circuit in the figure in Section *Creating a .sim file*. Save it in `irsim_logic.sue`.

3. To generate `irsim_logic.sim` and launch IRSIM, go to *Sim* →*Change Simulation Mode* ..., select *Sim* and click *Done*. Now go to the *Sim* menu again. Notice that the menu items are different from those in the default SPICE simulation mode. Simply click on *Sim It* (hotkey: *h*). You should see the following window. You can safely ignore all the warnings shown in red.



4. Compare the generated `irsim_logic.sim` with the one in the original tutorial. What extra information does `irsim_logic.sim` contain?

5. Take a look at the SUE command prompt. You should see the following messages:
   `Wrote sim netlist to <...>/irsim_logic.sim`
   `Starting irsim/analyzer with irsim <a prm file> <...>/irsim_logic.sim`
   The IRSIM command in the second message is similar to the one described in the original tutorial.

6. Go through the following sections in the original tutorial:

   Running IRSIM

   Using the analyzer window

   Vectors

   Quitting IRSIM

   Automation through command files

If you are interested, read Appendix A for additional information about IRSIM.

5 MAX Tutorial

1. Type `mmi_tutorial` at a command prompt.
2. Select *MAX* in the *Which Tutorial* field and click *View PDF Tutorial in Browser*. This will start up Adobe reader in a browser.
3. Start from Section *Starting up MAX* in Part 1 on page 5 (which is page 7 of the PDF).
4. Starting from page 11 in Part 1, use technology mmi18 instead of mmi25. Whenever asked to use command `max -tech mmi25`, use `max -tech mmi18` instead.
5. Skip Section *Viewing DRC Errors in MAX and with the Calibre Interface* in Part 2 (page 53-61).
6. Skip Part 3 (page 67-80).
7. Do Part 4 (page 81-118) and skip the rest of the tutorial.

6 Submission

1. `pulsegen.sue`: schematic of pulsegen as on page 37 of the SUE tutorial.
2. `pulsegen.ps`: (colored) SPICE plot of pulsegen as on page 26. Go to *Print* →*Print to File* in the NST window.
3. `irsim_logic.sue`: schematic of the example in IRSIM tutorial.
4. `irsim_logic.ps`: IRSIM Analyzer plot of `irsim_logic.sue`. Go to the *Print* menu in the Analyzer window.
5. `my_nand.max`: layout of the NAND cell as on page 116 of the MAX tutorial.
6. `README`: a 3-line text file that identifies yourself. Write your first and last name in the first and second line. Write your student number in the third line.

You will submit the above files, **using the provided names**, electronically with the following command:

```
submitece451s 0 README ...
```

You can check your submission with the command `submitece451s -l 0`

Do not forget to submit the README file!!

A More on IRSIM

The following is extracted from "CMOS Circuit Verification With Symbolic Switch-Level Timing Simulation" by Clayton B. McDonald and Randal E. Bryant in IEEE Trans. On CAD of Integrated Circuit and Systems, vol 20, no.3, March 2001.

> As a testbed for STS, we have implemented a symbolic version of the timing simulator IRSIM[15]. IRSIM is itself derived from two earlier simulators, RSIM and nRSIM. RSIM [18] introduced the concept of event-driven switch-level timing simulation based on Elmore delays [7], [14], which are delay estimates computed as resistance-capacitance products. It models transistors as switched linear resistors and all capacitors are connected to ground. Fig. 2. shows a simple circuit and its representation under the RSIM model. RSIM contained a simple and somewhat pessimistic model of nodes with unknown (X) values. NRSIM [5] improved on this model of X values and introduced several other enhancements, such as an improved model of charge-sharing effects and the simulation of transient voltage spikes. Lastly, IRSIM implemented an incremental simulation model, where circuit updates could be analyzed with only partial resimulation.

[5] C. Y. Chu, "Improved Models for Switch-Level Simulation", Ph.D. dissertation, Stanford Univ., Stanford, CA, 1988.
[7] W. Elmore, "The transient response of damped linear networks with particular regard to wideband amplifiers," J. Appl. Phys., vol. 19, no. 1, pp.55-63, Jan. 1948.
[14] J. Rubinstein, P. Penfield, and M. A. Horowitz, "Signal delay in RC tree networks," IEEE Trans. Computer-Aided Design, vol. CAD-2, pp.202-211, July 1983.
[15] A. Salz and M. A. Horowitz, "IRSIM: An incremental MOS switch-level simulator," in Proc. Design Automation Conf., June 1989, pp.173-178.
[18] C. J. Terman, "RSIM: A logic-level timing simulator," in Proc. Int. Conf. Computer Design, Oct. 1983, pp. 437-440.