



Lab 1: Intro to Logic with Multimedia Logic
Worth 35 points (30 lab + 5 report)

Lab Objectives:

In this first lab, you will learn how to use the Multimedia Logic application in Windows to do schematic entry and simulation. MML is a free schematic entry and simulation tool. If you would like to get the tool for home (you still have to come to lab though for checkoff) here is a link to it <http://www.softronix.com/logic.html>. You will use this program to build some simple circuits.

You will also gain experience designing combinatorial logic circuits, and using combinatorial logic as part of more complex circuits.

Lab1 Preparation:

- Read through this Lab1 assignment.
- Review basic Logic Gates (Beginning of Chapter 3).

Schematic Comments Requirement

For *every lab assignment* in this course (write-ups, MML files, flowcharts, and code), you are required to include the following information in a header on the top of the first page:

- Your name and email address
- Lab number (in this case, Lab 1)
- Lab title
- Due Date
- Your section letter
- The name of your TA

For MML files, each page should contain a header indicating the part of the lab that page contains.

In addition, your MML circuits should contain comments to help your grader understand and you're your circuit. The following documentation is mandatory:

- Each page should have a header describing which part of the lab the circuit is associated with, and a brief description of that circuit (eg: "Implementation of a truth table")

- Each input should be labelled.
 - If an input is several switches indicating a binary number, indicate which is the most and least significant bit.
- Each output should be labelled.
- If there are particularly complicated or confusing parts of your circuit, brief comments inside should describe what each section does (eg: "Compare bits 1a and 1b")

Comments should also help people who are not familiar with your circuit to understand what it does, and how. In particular, your grader should be able to use your circuit, and even modify it slightly, without you there to explain it to them. Write your comments accordingly.

Part A (5pts): Warm Up

Do the tutorial for Multimedia Logic (Help->Tutorial) or look at the MML_Tutorial.pdf provided. This simple tutorial will walk you through building and simulating simple circuit, save the resulting file as Lab1.lgi.

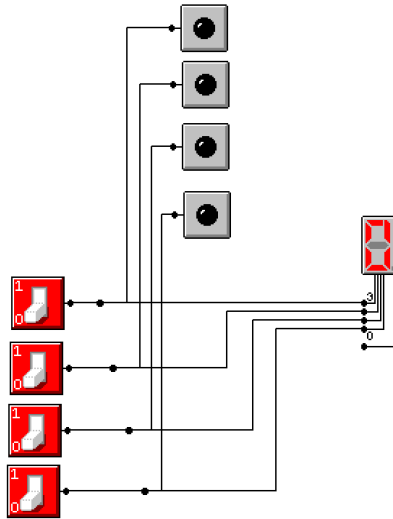
Once you've built and tested the circuit, use the "Text" tool to put the required comments on your schematic.

Part B (5pts): Playing with Numbers

Use the "+" button to add a new page to your MML file, and add the appropriate header comments.

Now, build the schematic below. See if you can make it cleaner than the one given. Simulate the circuit and play around with the switches, getting a feel for inputting a binary number and seeing the result on the display.

The schematic below is not complete until you label your inputs and outputs!



Part C (10pts): Truth Table to Gates

Start a new schematic page and label it PART C. Design logic that implements the following truth table.

IN[2]	IN[1]	IN[0]	Output
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0

Things to note for Part C:

Connect the inputs to “Switch” tools and the output to an LED to verify your circuit works correctly. *There should only be 3 inputs and one final output.* (You may use additional debugging outputs if you like, but label them accordingly).

You may use any circuit topology you like: sum-of-products, product-of-sums, or a gate-reduced topology of your own. Whatever topology you choose, be sure you can reproduce your design process (guessing wildly or copying someone else will result in a poor grade!)

As always, be sure to test your logic afterwards, and put correct comments in your schematic.

Part D (10pts): Guessing Game

Now you will create a fun guessing game in logic! Create a design on a new page labeled PART D that allows the user to play "guess the number." In this game, the user presses a momentary pushbutton which generates a random 2-bit number. Then, the user can switch two toggle switches to attempt to guess that number. When the user is correct, an LED lights up.

Your circuit will need the following components:

- Use the random number generator circuit element. You only need to connect two of the outputs, which are the pins on the right side of the box (question: does it matter which two you use? Why or why not?).
- Use a momentary pushbutton switch to drive the random number "generation". Connect the switch to "C+". (To make a momentary pushbutton, start with a toggle switch and double-click to edit its properties).
- Use two toggle switches for user input, this will be the "guess".
- Use combinational logic to test for equality; basically, is the output of the random number the same as that of the switches? If you're having trouble figuring out how to make this circuit: one strategy is to write a truth table. Another strategy is to test for equality one bit at a time, and combine the results.
- Use an LED to indicate whether the user's guess was correct or not.
- 2 LEDs to indicate the random secret number. This makes it a bad secret, but much easier to grade and debug! (You can cover up these with a sticky note while you play the guessing game, or if you're ambitious, have another toggle switch disable them.)

In total, you should have 3 inputs (one momentary pushbutton and two toggles) and 3 outputs.

Lab Write-up (5 pts)

In *every lab write-up* for this course, we will be looking for the following things:

- Appropriate headers (see above)
- Sensible lab writeup structure (eg: purpose, methods, results, analysis)
- Describe what you learned, what was surprising, what worked well and what did not
- Readability

For this lab in particular, also address:

- How would you make your own 7-segment display from part B if Multimedia Logic didn't provide one for you?

- How do you hypothesize the random number generator works? How can things be really random in a computer when it is made of logic gates, which are supposed to be deterministic?

Lab reports shouldn't need to be long. One page is sufficient.

We do not break down the point values; instead, we will assess the lab report as a whole while looking for the following content in the report.

To alleviate file format issues we want lab reports in plain text. Feel free to use a word processor to type it up but please submit a plain text file *and CHECK IT IN CANVAS to make sure it is readable.*

Deliverables

You will submit two(2) files on canvas:

- Lab1.lgi
- Lab1_report.txt

Check-off

For this lab, as with all labs, you will need to demonstrate your lab when it is finished to the TA or tutor and get it signed off. The staff member will use your circuit, and ask you questions about its functionality and your design process. Checkoffs are due within 1 week of the submission date. *GET CHECKED OFF IN YOUR OWN SECTION.*