Laboratory #4: Full Adder and System Design Review

Purpose

- Learn about and build a Full Adder.
- Solve a real-world digital logic problem using tools presented in class so far.

Part 1: Full Adder

Background: You are familiar with the Three-Way Light Control expression:

```
L = AB'C' + A'BC' + A'B'C + ABC          (NOT, AND, OR, Lab 2)
  = ((A'B'C)'(A'BC')'(AB'C')'(ABC)')'           (NAND, Lab 3)
  = ((A+B+C)'+(A+B'+C')'+(A'+B+C')'+(A'+B'+C)')'  (NOR, Lab 3)
  = A ^ B ^ C                          (Surprise! a 3-input XOR)
```

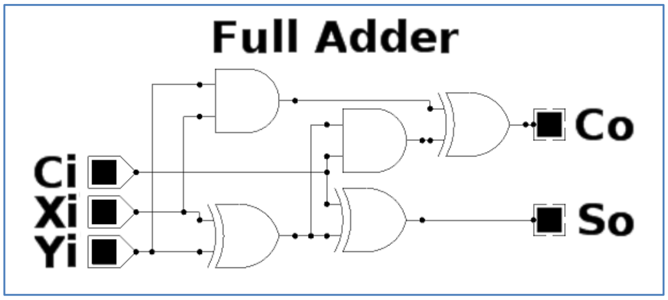
You are also familiar with the Majority Gate expression:

```
M = A'BC + AB'C + ABC' + ABC   (Sum Of Products)
  = AB + BC + AC               (reduced SOP)
  = (C(A ^ B)) ^ AB            (Surprise again!)
```

These two expressions also define a full adder, the foundation of digital arithmetic hardware and Central Processing Units (CPUs). A full adder has three 1-bit inputs: Ci (Carry In), Xi and Yi. It has two 1-bit outputs: Co (Carry Out) and So (Sum Out). Co So (2 bits) equals the Sum Ci + Xi + Yi.

Xi	Yi	Ci	Co	So	Sum = Ci+ Xi + Yi (decimal)
0	0	0	0	0	0
0	0	1	0	1	1
0	1	0	0	1	1
0	1	1	1	0	2
1	0	0	0	1	1
1	0	1	1	0	2
1	1	0	1	0	2
1	1	1	1	1	3

One way to implement a full adder is like this (only five 2-input gates, from two ICs):



Pre-Lab: use Verilog, to model and simulate the operation of the full adder circuit shown above. Verify it satisfies the full adder truth table provided. Call the Verilog files: fulladder.v and fulladder_tb.v. Once you are done with the simulation, replace labels Ci, Xi and Yi with switches SW1, SW2 and SW3 and labels Co and So with “LEDS” E and F, and draw a detailed schematic diagram of the circuit (must include a parts list, reference designators, pin numbers, Vcc and Ground, your name and the date). You have the required data sheets in previous lab handouts.

During-Lab: Build, test and debug the full adder circuit. When you are convinced it is working correctly show it to your instructor. Take a picture of it demonstrating 1+1+1 = 3. Your report for Part 1 should include the Schematic Diagram, Procedure Changes, Results, and Discussion. See the Lab Report Template for details of each section.

Part 2: BCD digit divided by 5

Pre-Lab:

The goal of part 2 of the lab is to design a 1-digit BCD divide-by-5 circuit on your Logic Trainer. Use SW1, SW2, SW3 and SW4 to input BCD digits. Use LEDs E, F, G and H to display a 1-bit quotient and 3-bit remainder. For example, to demonstrate "9 divided by 5 equals 1 remainder 4", SW1=1, SW2=0, SW3=0, SW4=1, LED E is on (quotient = 1), and LEDs F, G, H are on, off, off (remainder 4). For input values greater than 9, **we don't care** what the output is. Use only parts that you've been given data sheets for.

Use Verilog to model and simulate the circuit you plan to build (call the files: BCDdivide5.v and BCDdivide5_tb.v)

SW1	SW2	SW3	SW4	E	F	G	H
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	0	0	0	1	0
0	0	1	1	0	0	1	1
0	1	0	0	0	1	0	0
0	1	0	1	1	0	0	0
0	1	1	0	1	0	0	1
0	1	1	1	1	0	1	0
1	0	0	0	1	0	1	1
1	0	0	1	1	1	0	0
1	0	1	0	–	–	–	–
1	0	1	1	–	–	–	–
1	1	0	0	–	–	–	–
1	1	0	1	–	–	–	–
1	1	1	0	–	–	–	–
1	1	1	1	–	–	–	–

During-Lab: Build, test and debug the circuit. When you are convinced it is working correctly show it to your instructor. Take a picture of it demonstrating 9 divided by 5 to include in your lab report. Your report for Part 2 should include the completed schematic**,** Procedure Changes, Results (photograph), and Discussion. See the Lab Report Template for details of each section.

Summary and Conclusion Questions

1. Imagine talking to a student that will be taking this class in the future. Looking back on your first 4 labs (manual digital logic techniques), what tips would you give them to help make their time and effort more productive?

Pre-lab deliverables (these deliverables are worth 50% of the lab's grade)

Part 1

- fulladder.v, fulladder_tb.v, and a snapshot of the truth table generated
- detailed schematic diagram of the circuit

Part 2

- BCDdivide5.v, BCDdivide5_tb.v, a snapshot of the truth table generated, and the GTKwave simulation waveforms. To improve the readability of the waveforms make sure to group the signals f,g,h and to show their value in decimal format. Do the same for SW1, SW2, SW3 and SW4.
- Detailed schematic diagram of the circuit