Introduction to Digital Logic
EECS/CSE 31L

Assignment 2

EECS Department
Henry Samueli School of Engineering
University of California, Irvine

October, 18, 2014

Due on Saturday 10/25/2014 11:00pm. Note: this is a one-week assignment

1 Implementing a 1-bit ALU [100 points + 5 bonus points]

The goal of this assignment is to practice more VHDL coding and learn how to build a 1-bit Arithmetic and Logic Unit (ALU) at data flow (component instantiation in allowed).

1.1 Assignment Description

The objective of this project is to assess your understanding of how to implement a complicated combinational hardware block using the codes you have developed in the last assignment. In this assignment you are supposed to implement a 1-bit ALU in VHDL and verify to see if it is working as expected.

1.1.1 ALU Entity

Code 1: Sample entity of 1-bit ALU in VHDL

```
ENTITY alu_1bit IS
 PORT(
  A : IN STD_LOGIC;
  B : IN STD_LOGIC;
  cin : IN STD_LOGIC;
  opsel    : IN STD_LOGIC_VECTOR(2 DOWNTO 0);
  mode    : IN STD_LOGIC;
  output    : OUT STD_LOGIC;
  cout    : OUT STD_LOGIC
 );
END alu_1bit;
```

1.1.2 ALU Description

The ALU is supposed to have the following functionalities both in Arithmetic and Logic computing.

Table 1: ALU operation description

mode	**opsel**	**Micro-operation**	**Description**
0	000	$a + b$	Add
0	001	$a + \bar{b}$	Sub with borrowed carry
0	010	a	Move
0	011	$a + \bar{b} + 1$	Sub
0	100	$a + 1$	Increment
0	101	$a - 1$	Decrement
0	110	$a + b + 1$	Add & Increment
1	000	$a\ AND\ b$	bit-wise AND
1	001	$a\ OR\ b$	bit-wise OR
1	010	$a\ XOR\ b$	bit-wise Exclusive OR
1	011	$\bar{a}$	Compliment
1	101	$shl\ a$	1 bit shift left

1.2 Assignment Deliverables

Your submission should include the following:

- Design report of 1-bit ALU
- Waveform snapshot for each function
- VHDL code file

Note: Remember to name your files as **assignment2_STUDENT-ID_alu.vhd** and **assignment2_STUDENT-ID_alu.pdf**.