Introduction to Digital Logic
EECS/CSE 31L

Midterm Project

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

October, 17, 2016

Due on Sunday 10/31/2016 11:00pm. Note: **NO exception for late submission.**

1 Implementing a 128-bit ALU

The goal of this assignment is to practice multi-component logic design in SystemVerilog and learn how to build a 1-bit Arithmetic and Logic Unit (ALU) and expand it to 128 bits at structural levels (You are NOT allowed to use *always* blocks.)

1.1 Assignment Description

The objective of this project is to assess your understanding of how to implement a more complicated combinational hardware block using the codes you have developed in the first part of this assignment last week. This assignment asks you to design and implement a 1-bit ALU first. Then, you are supposed to implement a 128-bit ALU using 1-bit ALU component you have already developed. Also consider using ***generate*** construct for instantiating the 1-bit ALU components.

1.1.1 ALU Entity

Code 1: Port definition of 128-bit ALU in SystemVerilog

```
module alu_128bit ( op1, op2, opsel, mode, result, c_flag, z_flag, o_flag, s_flag );

  parameter DWIDTH = 128;

  input  logic [DWIDTH-1:0] op1    ;
  input  logic [DWIDTH-1:0] op2    ;
  input  logic [       2:0] opsel  ;
  input  logic              mode   ;
  output logic [DWIDTH-1:0] result ;
  output logic              c_flag ;
  output logic              z_flag ;
  output logic              o_flag ;
  output logic              s_flag ;


endmodule;
```

1.1.2 ALU Description

The ALU is supposed to have the following functionalities both in Arithmetic and Logic computing. a and b symbols, in the Table 1, are by default 128-bit data inputs, shown as *op1* and *op2* in the port description, respectively. The *result* port will have the operation's result. The *c_flag, z_flag, o_flag, s_flag* ports represent the carry out, zero, overflow, and sign flags, respectively.

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	1-bit logical shift left

1.2 Project Deliverables

Your submission should include the following:

- Design report of 128-bit ALU including any assumptions you have made and the list of errors you have got and how you have solved them.
- Your report should include the detail architecture of your design
- Include a preliminary evaluation of your block timing in terms of number of gates
- Waveform snapshot for each function with a brief explanation
- SystemVerilog code file

Note1: Remember to name your files as **mtp_GROUP-ID_alu.sv** and **mtp_GROUP-ID_alu.pdf**.

Note2: Compress all your files in "zip" or "tar" format and then submit the compressed file.

Note3: Remember to upload the ".tex" file if you have prepared your report using Latex.

Note4: Remember to include the group ID and the name and student ID of each group member and the the description of the task each member has performed in the report.