Maulik Patel, Ojas Anand, and David Luria

EECE 2060C – 002

November 9, 2017

Post-Lab Report

Laboratory 6: A Simple ALU Implemented Using the Basys FPGA Board

Design Specification

Design a 3-bit ALU with signed 2's complement implementation that will respond correctly to the following instructions:

- When instruction Ins[1:0] = 00 – ALU preforms S[2:0] = A[2:0] + B[2:0]
- When instruction Ins[1:0] = 01 – ALU has S[2:0] = number of logic 1 bits in input A
- When instruction Ins[1:0] = 10 – ALU preforms S[2:0] = A & B
- When instruction Ins[1:0] = 11 – ALU preforms left rotation for input A

Implementation

- Top Module:

```
`timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////////////
// Company:
// Engineer:
//
// Create Date:    09:31:54 10/26/2017
// Design Name:
// Module Name:    mALUStructure
// Project Name:
// Target Devices:
// Tool versions:
// Description:
//
// Dependencies:
//
// Revision:
// Revision 0.01 - File Created
// Additional Comments:
//
//////////////////////////////////////////////////////////////////////////////////
module mALUStructure(ins, A, B, S, OV, Cout);
   input [1:0] ins;
   input [2:0] A, B;
   output reg [2:0] S;
   output reg OV, Cout;
   reg [3:0] temp = 0;
   integer i;

   always @(ins, A, B)
      begin
         OV = 1'b0;
         Cout = 1'b0;

         case (ins)
            2'b00:
               begin            //Instruction A+B
                  temp = A + B;
                  Cout = temp[3];
                  S = temp[2:0];
                  OV = (A[2]==B[2])&(A[2]!=S[2]);
               end
            2'b01:
               begin            //Instruction for counting the number of bits with logic 1 in input A
                  S = A[2] + A[1] + A[0];
               end
            2'b10:
               begin      //Instruction for A AND B
                  S = A & B;
               end
            2'b11:
               begin            //Instruction for left rotation for input A
                  i = A[2];
                  S = A << 1;
                  S[0] = i;
               end
            endcase
      end
endmodule
```

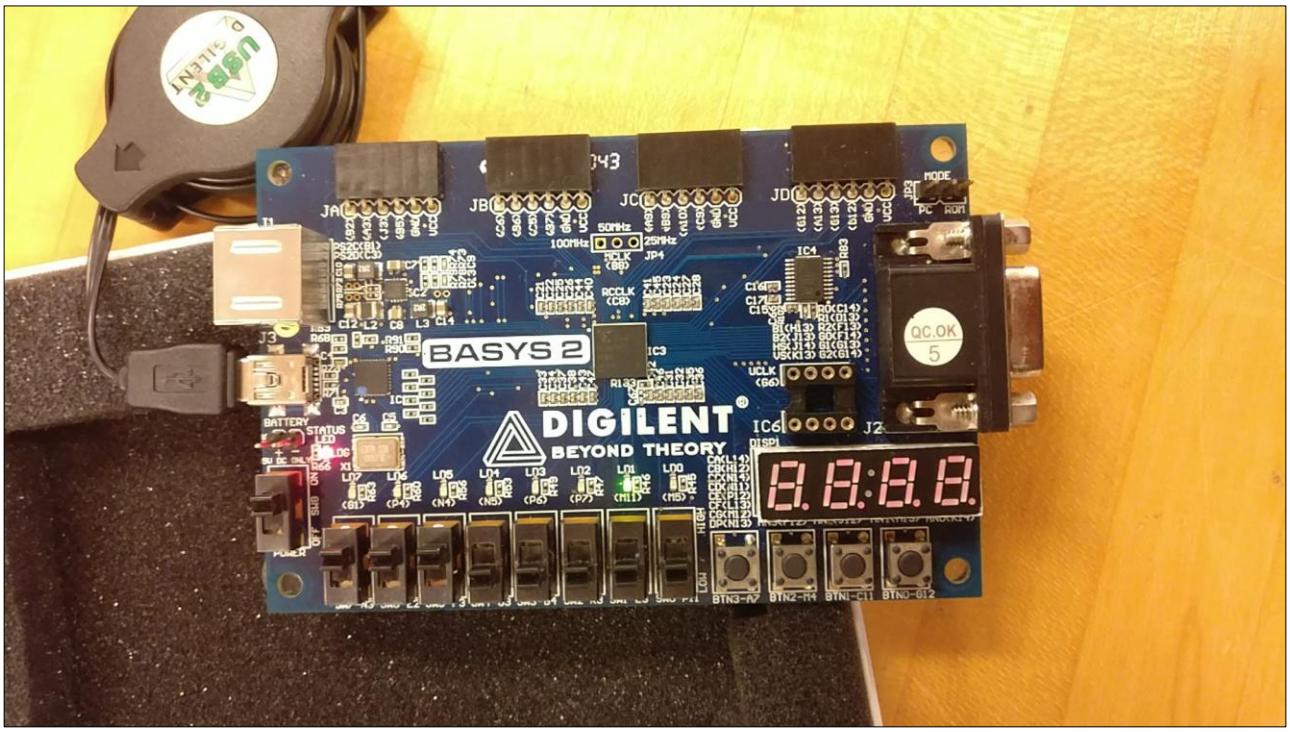
- mALU Module:

- neg2pos Module:

- bcd7seg:
- Testbench:

Results

- Week 1:



Contributions

- Week 1:

	Requirements	Design	Simulation	Test	Implementation	Report
David Luria	35%	7%	30%	40%	35%	20%
Maulik Patel	30%	75%	30%	30%	30%	20%
Ojas Anand	35%	8%	40%	30%	35%	60%