Lab 10 – Worksheet

Name: Breeha and Kulsoom	ID:08051, 08283	Section:

Task 1.

Code: Design module & testbench

Provide appropriately commented code for designed module, the code should contain meaningful variable naming.
**Add snippet or Copy paste the code written in the Editor window. Make sure the irrelevant area of the snip is cropped.*

Design Module

```
module Control_Unit(
    input [6:0] opcode,
    output reg [1:0] ALUop,
    output reg branch,
    output reg memread,
    output reg memtoreg,
    output reg memwrite,
    output reg ALUsrc,
    output reg regwrite
    );
 always@(*) begin
    case(opcode)
        7'b0110011: //R-Type
        begin
            ALUop=2'b10;
            branch=1'b0;
            memwrite=1'b0;
            memread=1'b0;
            regwrite=1'b1;
            memtoreg=1'b0;
            ALUsrc=1'b0;
        end
        7'b0000011: //I-Type (ld)
        begin
            ALUop=2'b00;
            branch=1'b0;
            memwrite=1'b0;
```

```
        memread=1'b1;
        regwrite=1'b1;
        memtoreg=1'b1;
        ALUsrc=1'b1;
      end
      7'b0100011: //S-Type (sd)
      begin
        ALUop=2'b00;
        branch=1'b0;
        memwrite=1'b1;
        memread=1'b0;
        regwrite=1'b0;
        memtoreg=1'bx;
        ALUsrc=1'b1;
      end
      7'b1100011: //SB-Type (Beq)
      begin
        ALUop=2'b01;
        branch=1'b1;
        memwrite=1'b0;
        memread=1'b0;
        regwrite=1'b0;
        memtoreg=1'bx;
        ALUsrc=1'b0;
      end
  endcase
  end
endmodule
```

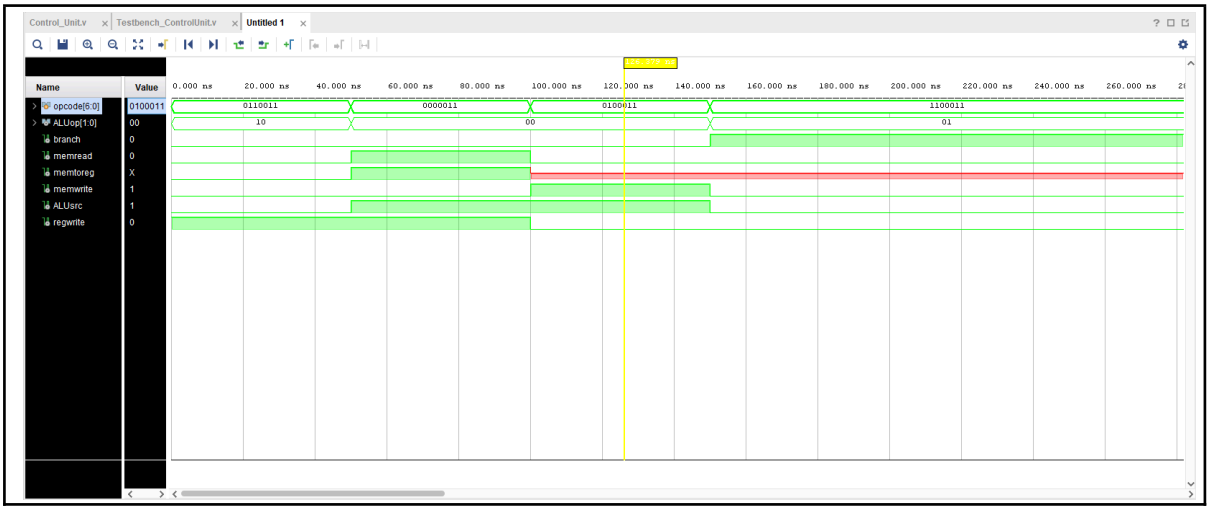
TestBench

```
module Testbench_ControlUnit();
reg [6:0] opcode;
wire [1:0] ALUop;
wire branch;
wire memread;
wire memtoreg;
wire memwrite;
wire ALUsrc;
wire regwrite;
Control_Unit cu(opcode,ALUop,branch, memread, memtoreg, memwrite,ALUsrc, regwrite);
initial
begin
opcode=7'b0110011;

#50
opcode=7'b0000011;
```

```
#50
opcode=7'b0100011;

#50
opcode=7'b1100011;
end
endmodule
```

Results (Waveforms)

**Add snip of relevant signals' waveforms. Make sure the irrelevant area of the snip is cropped.*



Comments

**Observation/Comments on the obtained results/working of code.*

Here we passed 6 bit opcode to design module then on basis of the opcode the values are assigned to the selector bits.

Task 2.

Code: Design module & testbench

Provide appropriately commented code for designed module, the, code should contain meaningful variable naming.

**Add snippet or Copy paste the code written in the Editor window. Make sure the irrelevant area of the snip is cropped.*

Design Module

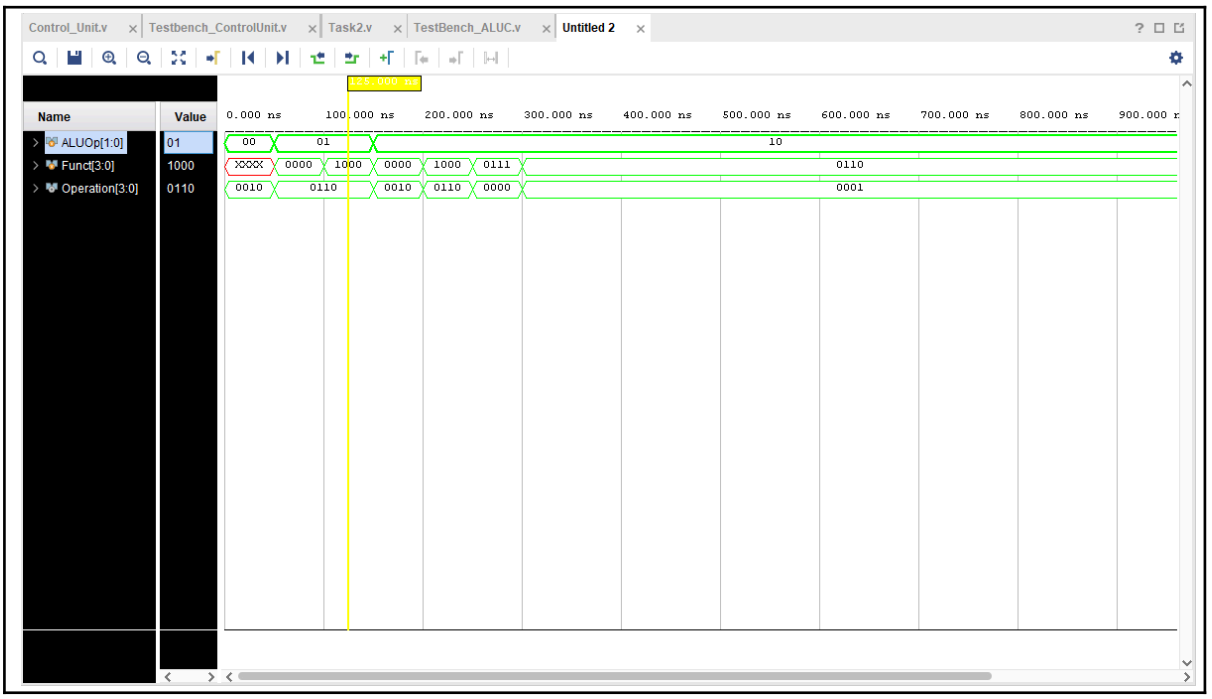
```
module ALU_Control(
   input  [1:0] ALUOp,
   input  [3:0] Funct,
   output reg [3:0] Operation

   );
   always@(*) begin
   case(ALUOp)
      2'b00:
      begin
      Operation=4'b0010;
      end
      2'b01:
      begin
      Operation=4'b0110;
      end
      2'b10:
      begin
         case(Funct)
            4'b0000:
            Operation=4'b0010;

            4'b1000 :
            Operation=4'b0110;

            4'b0111:
            Operation=4'b0000;

            4'b0110:
            Operation=4'b0001;
           endcase
     end
   endcase
   end
endmodule
```

TestBench

```
module TestBench_ALUC();
    reg  [1:0] ALUOp;
    reg  [3:0] Funct;
    wire [3:0] Operation;

    ALU_Control alc(ALUOp,Funct,Operation);

    initial
    begin
    ALUOp=2'b00;

    #50
    ALUOp=2'b01;
    Funct=4'b0000;

    #50
    ALUOp=2'b01;
    Funct=4'b1000;

    #50
    ALUOp=2'b10;
    Funct=4'b0000;

    #50
    ALUOp=2'b10;
    Funct=4'b1000;

    #50
    ALUOp=2'b10;
    Funct=4'b0111;

    #50
    ALUOp=2'b10;
    Funct=4'b0110;

    end
endmodule
```

Results (Waveforms)

**Add snip of relevant signals' waveforms. Make sure the irrelevant area of the snip is cropped.*



Comments

**Observation/Comments on the obtained results/working of code.*

The ALUop bits are taken as inputs and that determines ALU operations which will be performed and the 4 bit output which represents operation. For the r types instructions, Func field is needed as input. For SB and I type the input for func is dont cares which will not affect the operation.

Task 3

Code: Design module & testbench

Provide appropriately commented code for designed module & its testbench, code should contain meaningful variable naming.

**Add snippet or Copy paste the code written in the Editor window. Make sure the irrelevant area of the snip is cropped.*

Design Module

```
module top_control(
    input [6:0] Opcode,
    input [3:0] Funct,
    output Branch,
    output MemRead,
    output MemToReg,
    output MemWrite,
    output ALUSrc,
    output RegWrite,
    output [3:0] Operation
    );
    wire [1:0] ALUOp;
    Control_Unit cu(Opcode,ALUOp,Branch, MemRead, MemToReg, MemWrite,ALUSrc,
RegWrite);

    ALU_Control alc(ALUOp,Funct,Operation);

endmodule
```

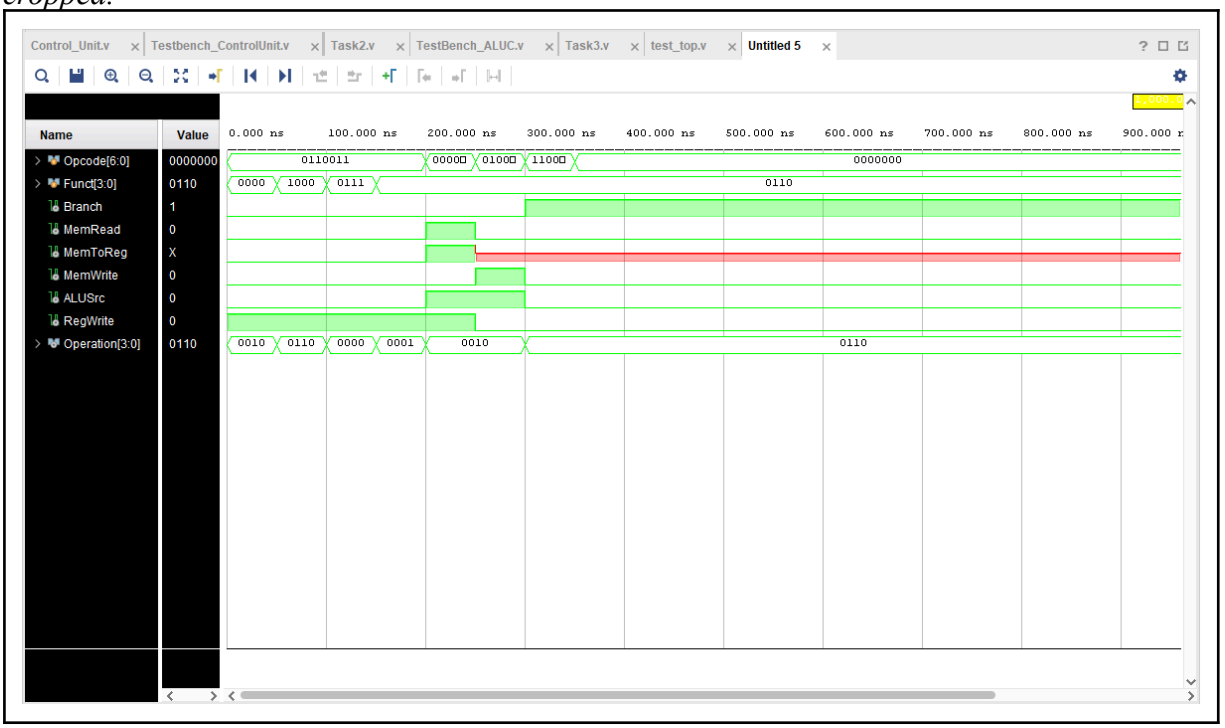
TestBench

```
module test_top();
reg [6:0] Opcode;
reg [3:0] Funct;
wire Branch;
wire MemRead;
wire MemToReg;
wire MemWrite;
wire ALUSrc;
wire RegWrite;
wire [3:0] Operation;

top_control tc(Opcode, Funct, Branch, MemRead, MemToReg, MemWrite, ALUSrc,
RegWrite, Operation);
initial
```

```
begin
Opcode=7'b0110011;
Funct= 4'b0000;

#50
Opcode=7'b0110011;
Funct= 4'b1000;
#50
Opcode=7'b0110011;
Funct= 4'b0111;
#50
Opcode=7'b0110011;
Funct= 4'b0110;

#50
Opcode=7'b0000011;

#50
Opcode=7'b0100011;

#50
Opcode=7'b1100011;

#50
Opcode=7'b0;
end
endmodule
```

Results (Waveforms)

**Add snip of relevant signals' waveforms. Make sure the irrelevant area of the snip is cropped.*



Comments

**Observation/Comments on the obtained results/working of code.*

The tops modules instantiated the previous two modules. They calculate the values of selector bits in control lines and the operation which is to be performed. For R type we need a funct field which is not fixed while others are.

Assessment Rubrics

Marks Distribution:

For description of different levels of the mapped rubrics, please refer the provided Lab Evaluation Assessment Rubrics.

Task No.	LR 2 Code	LR 5 Results	AR 7 Report Submission
Task 1 Control Unit	/20	-	/20
Task 2 ALU Control	/20	-	
Task 3 Top Control	/20	/20	
Total Points	/100 Points		
CLO Mapped	CLO 1		