# ELC 2137 Lab 10: 7 Segment TDM

Spencer Stinson

November 10, 2020

## Summary

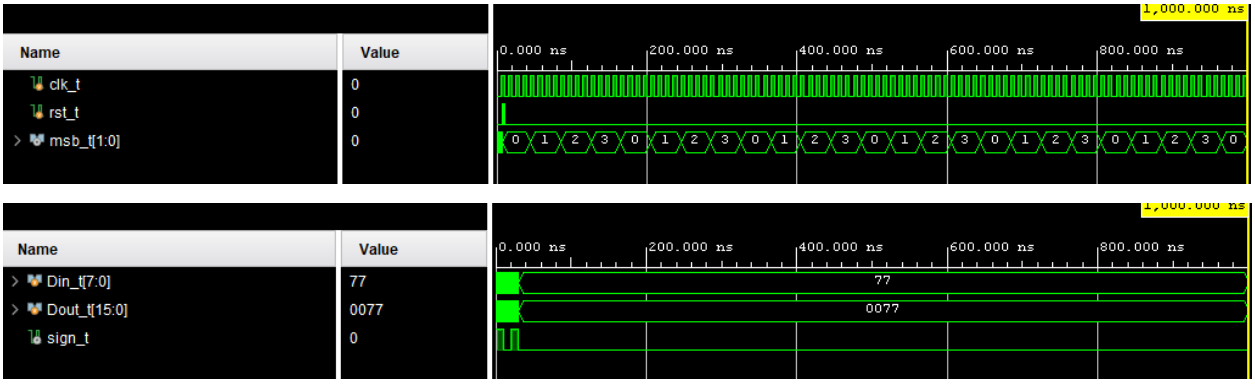
In this lab, we used our knowledge and code from previous labs to create a calculator using our board. This lab helped us to understand how to do conversions in verilog better with the show2c module, how to use the design specs, such as the timing of the light that we used in the Ncount, which we found the value to be 20 to get an accurate reading, and lastly it taught us how to combine these to get an accurate calculator as our final result.

## Q&A

There were no questions for this lab.

## Results

In this section, put your simulation waveforms, results tables, pictures of hardware, and any other required items.



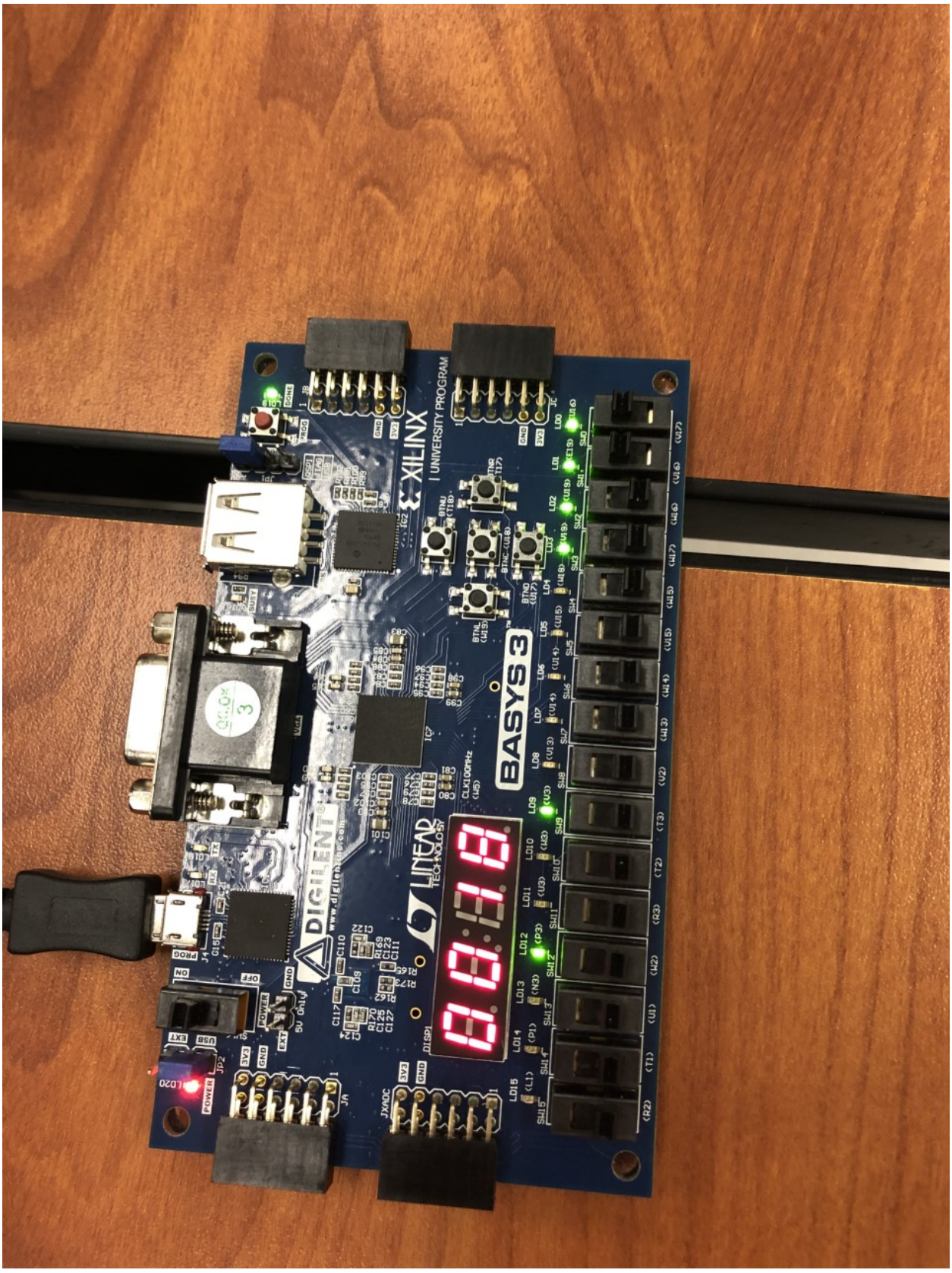



Figure 1: 15 + 3 = 18 displayed on basys board

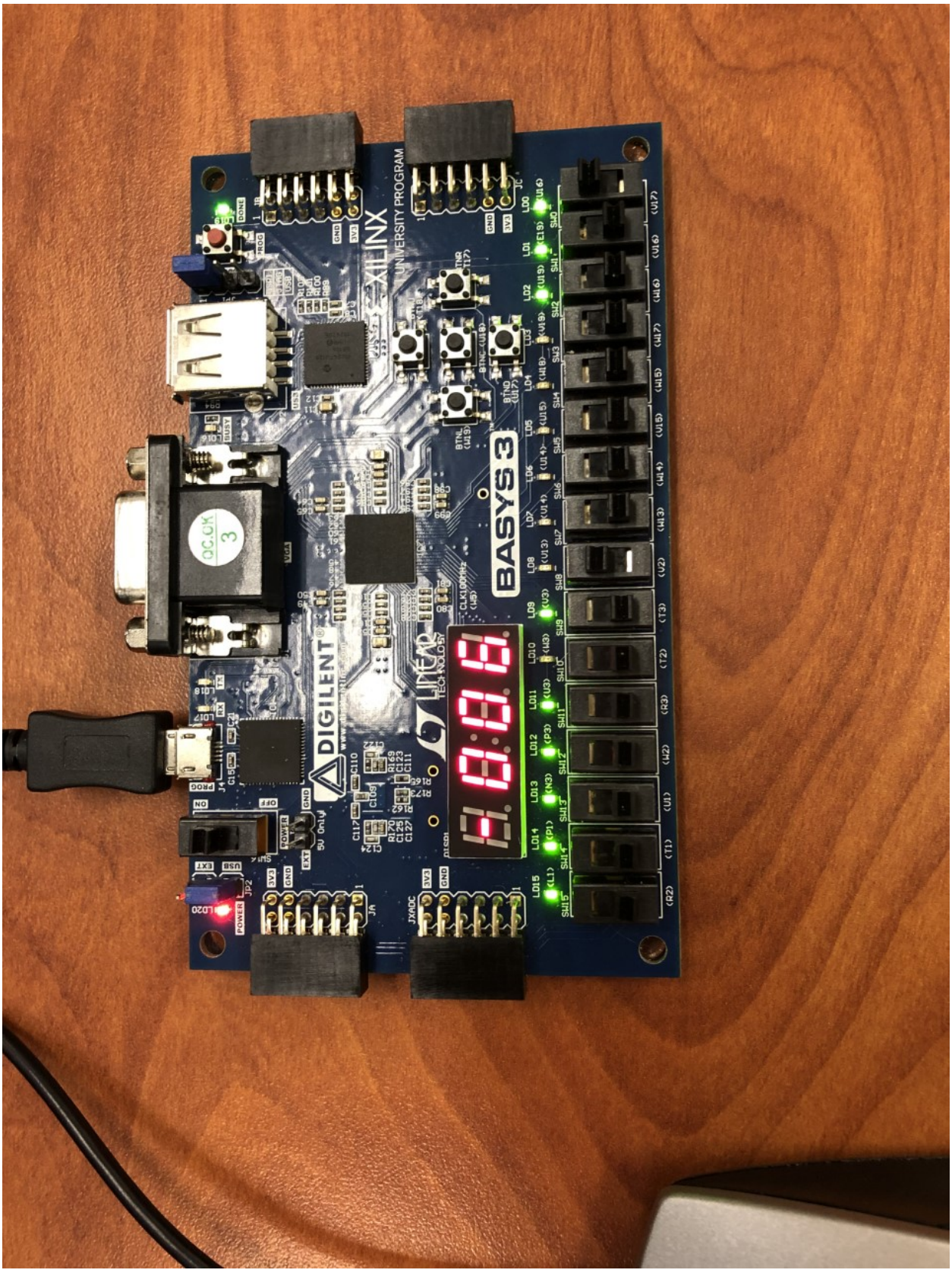



Figure 2: 1 - 7 = -6 displayed on basys board

## Code

Listing 1: Ncount module

```
‘timescale 1ns / 1ps
//
   //////////////////////////////////////////////////////////////////////////////////

// Company:
// Engineer:
//
// Create Date: 10/29/2020 11:49:50 AM
// Design Name:
// Module Name: Ncount
// Project Name:
// Target Devices:
// Tool Versions:
// Description:
//
// Dependencies:
//
// Revision:
// Revision 0.01 - File Created
// Additional Comments:
//
//
   //////////////////////////////////////////////////////////////////////////////////



module Ncount #(parameter N = 4)(
    input clk,
    input rst,
    output [1:0] msb
    );
wire [N-1:0] Qnext ;
wire [N-1:0] Qreg  ;


    register #(.N(N)) Qr(
    .clk(clk),
    .rst(rst),
    .en(1),
    .D(Qnext),
    .Q(Qreg)
    );

  assign Qnext = Qreg +1 ;

  assign msb =  Qreg [N-1:N-2];

endmodule
```

Listing 2: Show 2's compliment module

```
‘timescale 1ns / 1ps
//
   //////////////////////////////////////////////////////////////////////////////////

// Company:
// Engineer:
//
// Create Date: 10/29/2020 01:23:50 PM
// Design Name:
// Module Name: show2c
// Project Name:
// Target Devices:
// Tool Versions:
// Description:
//
// Dependencies:
//
// Revision:
// Revision 0.01 - File Created
// Additional Comments:
//
//
   //////////////////////////////////////////////////////////////////////////////////



module show2c(
    input [7:0] Din,
    output reg [15:0] Dout,
    output sign
    );
    reg [7:0] addout;
     assign addout = ~Din + 8’b00000001;

    always @*

    if (Din[7])
        Dout <= {8’h00, addout};
    else
        Dout <= {8’b00, Din};

    assign sign = Din[7];

endmodule
```

Listing 3: wrapper testing module

```
‘timescale 1ns / 1ps
//
   //////////////////////////////////////////////////////////////////////////////////

// Company:
// Engineer:
//
```

```
// Create Date: 10/29/2020 12:54:21 PM
// Design Name:
// Module Name: wrapper
// Project Name:
// Target Devices:
// Tool Versions:
// Description:
//
// Dependencies:
//
// Revision:
// Revision 0.01 - File Created
// Additional Comments:
//
//
   //////////////////////////////////////////////////////////////////////////////////



module wrapper(
    input clk,
    input btnC,
    output [6:0] seg,
    output dp,
    output [3:0] an
    );
    wire [1:0] msbs;

    Ncount #(.N(20)) nc(
    .clk(clk),
    .rst(btnC),
    .msb(msbs)
    );

    sseg4 s(
    .data(16'b0111),
    .hex_dec(0),
    .sign(1),
    .digit_sel(msbs),
    .seg(seg),
    .dp(dp),
    .an(an)
    );
endmodule
```

Listing 4: calculator final module

```
`timescale 1ns / 1ps
//
   //////////////////////////////////////////////////////////////////////////////////

// Company:
// Engineer:
//
```

```
// Create Date: 11/05/2020 12:29:48 PM
// Design Name:
// Module Name: calc
// Project Name:
// Target Devices:
// Tool Versions:
// Description:
//
// Dependencies:
//
// Revision:
// Revision 0.01 - File Created
// Additional Comments:
//
//
   //////////////////////////////////////////////////////////////////////////////////



module calc(
    input btnU,
    input btnD,
    input [11:0] sw,
    input clk,
    input btnC,
    output [6:0] seg,
    output dp,
    output [3:0] an,
    output [15:0] led
    );

    wire [15:0] ledout;
    wire [15:0] s1dout;
    wire ssign;
    wire [1:0] msbout;

  top_lab9 (
   .btnU(btnU),
   .btnD(btnD),
   .sw(sw),
   .clk(clk),
   .btnC(btnC),
   .led(ledout)
  );

  assign led = ledout;

   show2c s1 (
    .Din(ledout[15:8]),
    .Dout(s1dout),
    .sign(ssign)
   );

   Ncount #(.N(20)) N1 (
```

```
   .clk(clk),
   .rst(btnC),
   .msb(msbout)
   );

   sseg4 s41 (
     .data(s1dout),
     .hex_dec(sw[15]),
     .sign(ssign),
     .digit_sel(msbout),
     .seg(seg),
     .dp(dp),
     .an(an)
    );









endmodule
```

Listing 5: Ncount test bench module

```
`timescale 1ns / 1ps
//
   //////////////////////////////////////////////////////////////////////////////////

// Company:
// Engineer:
//
// Create Date: 10/29/2020 12:21:07 PM
// Design Name:
// Module Name: Ncount_testbench
// Project Name:
// Target Devices:
// Tool Versions:
// Description:
//
// Dependencies:
//
// Revision:
// Revision 0.01 - File Created
// Additional Comments:
//
//
   //////////////////////////////////////////////////////////////////////////////////
```

```
module Ncount_testbench();

reg clk_t, rst_t;
wire [1:0] msb_t;

Ncount #(.N(4)) dut(
    .clk(clk_t),
    .rst(rst_t),
    .msb(msb_t) );

always begin
    clk_t = ~clk_t; #5;
    end

 initial begin
    clk_t = 0; rst_t = 0; #7;
    rst_t = 1; #3;
    rst_t = 0; #3;
    end

endmodule
```

Listing 6: show2c test bench module

```
`timescale 1ns / 1ps
//
   //////////////////////////////////////////////////////////////////////////////////

// Company:
// Engineer:
//
// Create Date: 11/09/2020 08:09:16 PM
// Design Name:
// Module Name: show2c_testbench
// Project Name:
// Target Devices:
// Tool Versions:
// Description:
//
// Dependencies:
//
// Revision:
// Revision 0.01 - File Created
// Additional Comments:
//
//
   //////////////////////////////////////////////////////////////////////////////////



module show2c_testbench();
```

```
  reg [7:0] Din_t;
  reg [15:0] Dout_t;
  wire sign_t;


show2c dut(
.Din(Din_t),
.Dout(Dout_t),
.sign(sign_t) );


initial begin
Din_t = 8'b11111111; #10;
Din_t = 8'b01111111; #10;
Din_t = 8'b11110111; #10;
Din_t = 8'b01110111; #10;
end



endmodule
```