***EE 316 – Lab 3 Report*: Programming Logic on the Basys3 FPGA Board**

Ayan Basu [EID: ab73287] (Section: 17760)

[redacted]

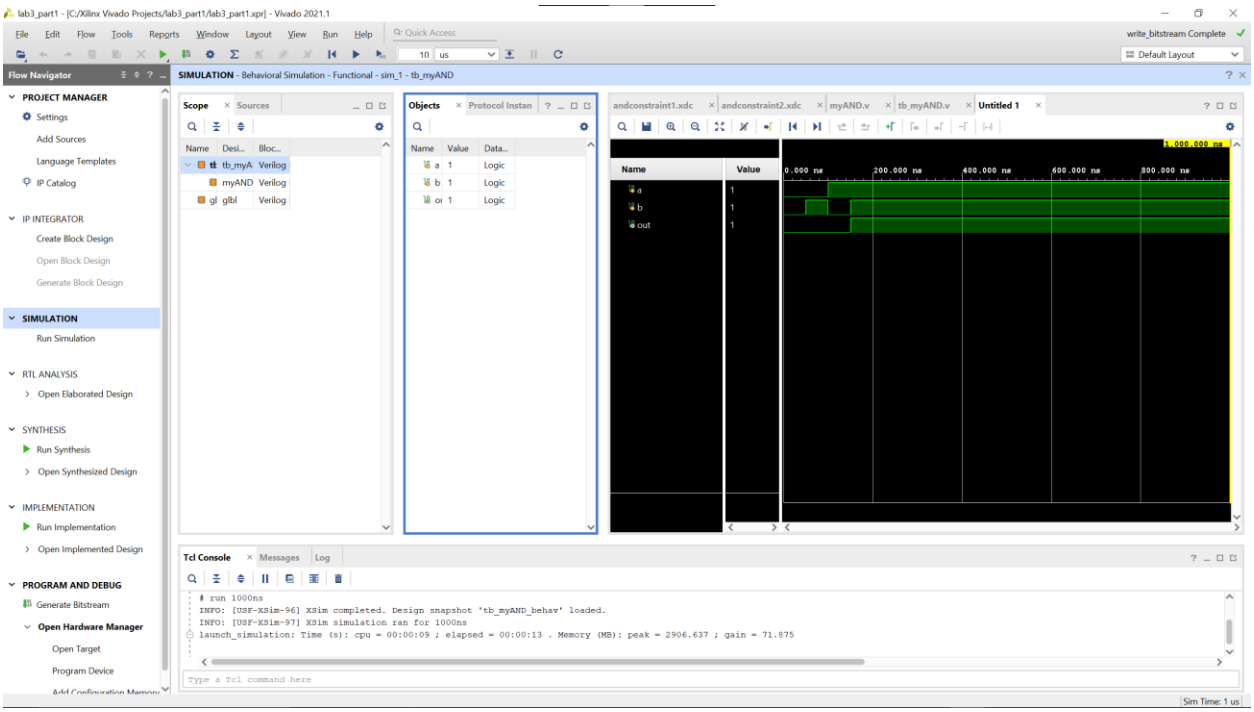
***PART 1*: AND Gate**

Constraint File

```
set_property PACKAGE_PIN V17 [get_ports {b}]
        set_property IOSTANDARD LVCMOS33 [get_ports {b}]
set_property PACKAGE_PIN V16 [get_ports {a}]
        set_property IOSTANDARD LVCMOS33 [get_ports {a}]


set_property PACKAGE_PIN U16 [get_ports {out}]
        set_property IOSTANDARD LVCMOS33 [get_ports {out}]
```

Waveform



***PART 2*: Sprinkler Valve Controller**

Constraint File

```
## Switches
set_property PACKAGE_PIN V17 [get_ports {c}]
        set_property IOSTANDARD LVCMOS33 [get_ports {c}]
set_property PACKAGE_PIN V16 [get_ports {b}]
        set_property IOSTANDARD LVCMOS33 [get_ports {b}]
set_property PACKAGE_PIN W16 [get_ports {a}]
        set_property IOSTANDARD LVCMOS33 [get_ports {a}]
set_property PACKAGE_PIN W13 [get_ports {e}]
        set_property IOSTANDARD LVCMOS33 [get_ports {e}]

## LEDs
set_property PACKAGE_PIN U16 [get_ports {d0}]
        set_property IOSTANDARD LVCMOS33 [get_ports {d0}]
set_property PACKAGE_PIN E19 [get_ports {d1}]
        set_property IOSTANDARD LVCMOS33 [get_ports {d1}]
set_property PACKAGE_PIN U19 [get_ports {d2}]
        set_property IOSTANDARD LVCMOS33 [get_ports {d2}]
set_property PACKAGE_PIN V19 [get_ports {d3}]
        set_property IOSTANDARD LVCMOS33 [get_ports {d3}]
set_property PACKAGE_PIN W18 [get_ports {d4}]
        set_property IOSTANDARD LVCMOS33 [get_ports {d4}]
set_property PACKAGE_PIN U15 [get_ports {d5}]
        set_property IOSTANDARD LVCMOS33 [get_ports {d5}]
set_property PACKAGE_PIN U14 [get_ports {d6}]
        set_property IOSTANDARD LVCMOS33 [get_ports {d6}]
set_property PACKAGE_PIN V14 [get_ports {d7}]
        set_property IOSTANDARD LVCMOS33 [get_ports {d7}]
```

***PART 3*: BCD to 7-Segment Display**

Truth Table & Karnaugh Maps

(8^{th} bit corresponds to dp, decimal point, which is always 1, or off)

in	a	b	c	d	e	f	g	out
0000	0	0	0	0	0	0	1	“0”
0001	1	0	0	1	1	1	1	“1”
0010	0	0	1	0	0	1	0	“2”
0011	0	0	0	0	1	1	0	“3”
0100	1	0	0	1	1	0	0	“4”
0101	0	1	0	0	1	0	0	“5”
0110	0	1	0	0	0	0	0	“6”
0111	0	0	0	1	1	1	1	“7”
1000	0	0	0	0	0	0	0	“8”
1001	0	0	0	0	1	0	0	“9”
1010	1	1	1	1	1	1	1	“10”
1011	1	1	1	1	1	1	1	“11”
1100	1	1	1	1	1	1	1	“12”
1101	1	1	1	1	1	1	1	“13”
1110	1	1	1	1	1	1	1	“14”
1111	1	1	1	1	1	1	1	“15”

A	00	01	11	10
00	0	1	0	0
01	1	0	0	0
11	1	1	1	1
10	0	0	1	1

a = AB + AC + BC’D’ + A’B’C’D

B	00	01	11	10
00	0	0	0	0
01	0	1	0	1
11	1	1	1	1
10	0	0	1	1

b = AB + AC + BC’D + BCD’

C	00	01	11	10
00	0	0	0	1
01	0	0	0	0
11	1	1	1	1
10	0	0	1	1

c = AB + AC + B'CD'

D	00	01	11	10
00	0	1	0	0
01	1	0	1	0
11	1	1	1	1
10	0	0	1	1

d = AB + AC + BC'D' + BCD + A'B'C'D

E	00	01	11	10
00	0	1	1	0
01	1	1	1	0
11	1	1	1	1
10	0	1	1	1

e = D + BC'+ AC

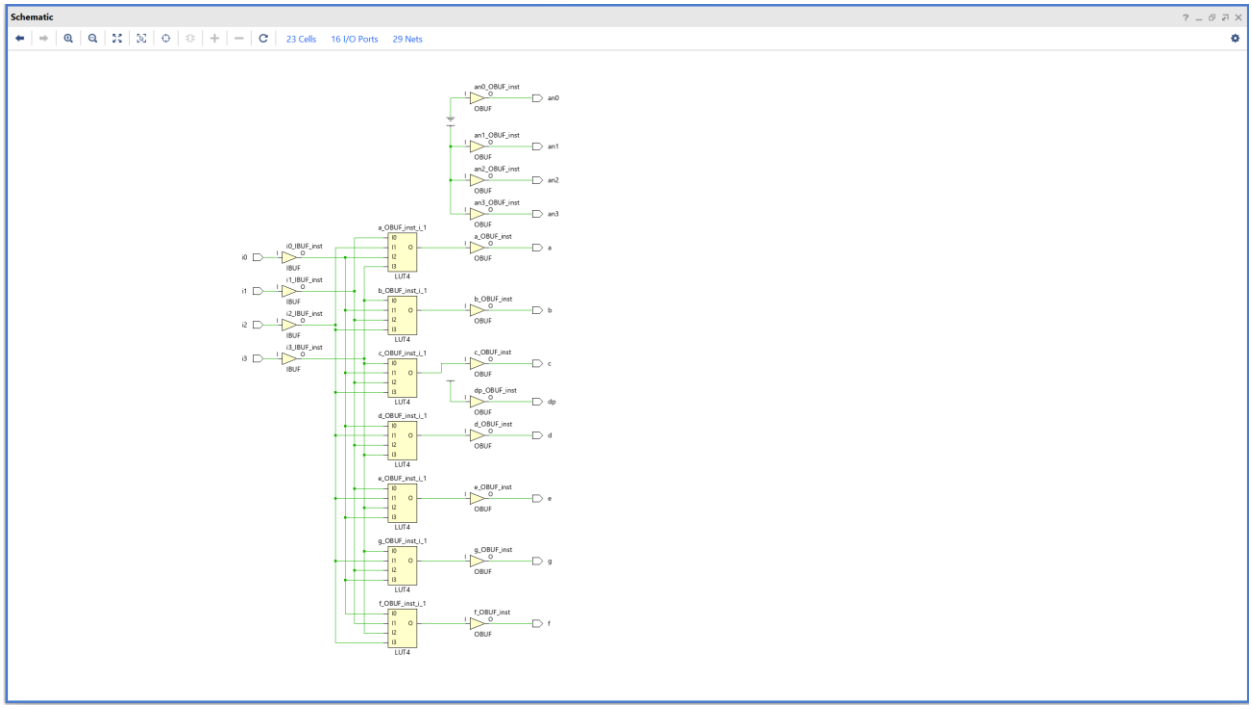
F	00	01	11	10
00	0	1	1	1
01	0	0	1	0
11	1	1	1	1
10	0	0	1	1

f = AB + CD + B'C + A'B'D

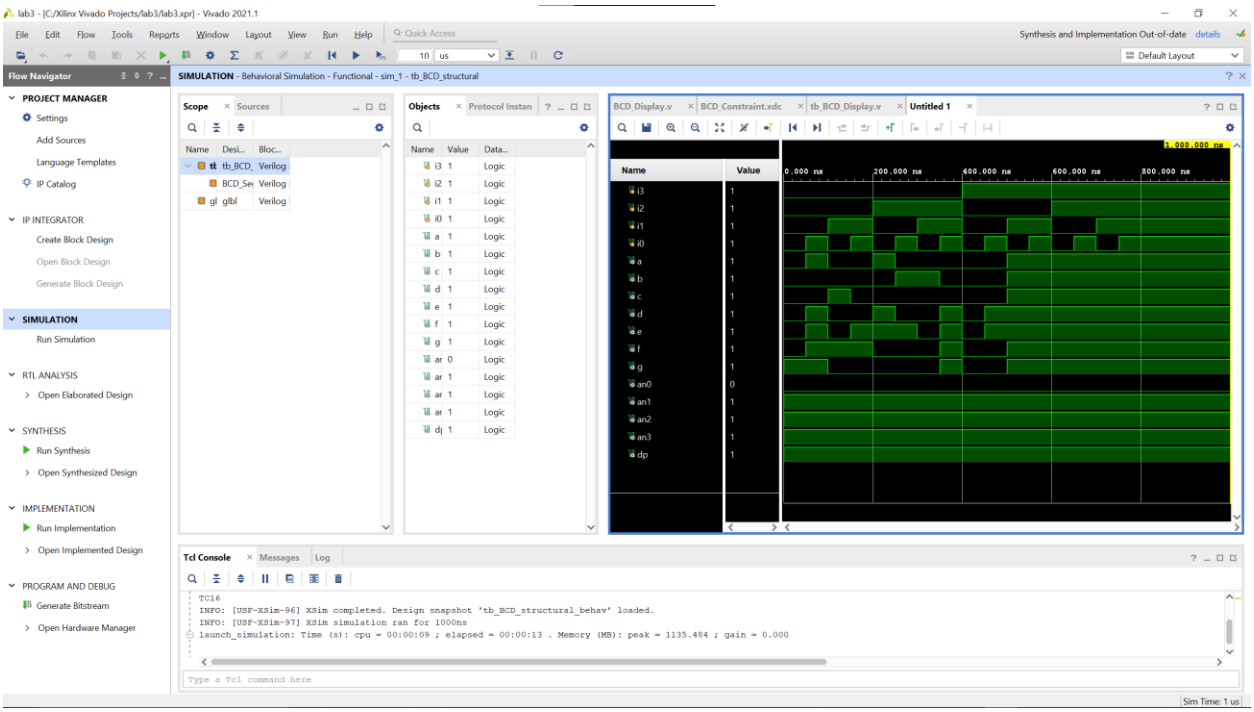
G	00	01	11	10
00	1	1	0	0
01	0	0	1	0
11	1	1	1	1
10	0	0	1	1

g = AB + AC + A'B'C' + BCD

Gate-Level Schematic Design



Simulation



BCD_Display.v (Structural)

```
`timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////////////
// Company:
// Engineer:
//
// Create Date: 09/29/2021 09:13:22 PM
// Design Name:
// Module Name: BCD_Display
// Project Name:
// Target Devices:
// Tool Versions:
// Description:
//
// Dependencies:
//
// Revision:
// Revision 0.01 - File Created
// Additional Comments:
//
//////////////////////////////////////////////////////////////////////////////////


module BCD_Seg_Display(
    input i0,
    input i1,
    input i2,
    input i3,
    output an0,
```

```
    output an1,
    output an2,
    output an3,
    output a,
    output b,
    output c,
    output d,
    output e,
    output f,
    output g,
    output dp
    );



    //wires for outputs
    wire and_m0, and_m1, and_m2, and_m3, and_m4, and_m5, and_m6,
    and_m7, and_m8, and_m9, and_m10, and_m11, and_m12, and_m13, and_m14, and_m15, not_i3,
not_i2, not_i1, not_i0;


      assign an0=1'b0;
      assign an1=1'b1;
      assign an2=1'b1;
      assign an3=1'b1;
      assign dp = 1'b1;


    //Instantiating Not gates
    not n0 (not_i3, i3);
    not n1 (not_i2, i2);
    not n2 (not_i1, i1);
```

```
not n3 (not_i0, i0);

//make AND gates
and m0 (and_m0, i3, i2);
and m1 (and_m1, i3, i1);
and m2 (and_m2, i2, not_i1 ,not_i0);
and m3 (and_m3, not_i3, not_i2, not_i1, i0);
and m4 (and_m4, i2, not_i1, i0);
and m5 (and_m5, i2, i1, not_i0);
and m6 (and_m6, not_i2, i1, not_i0);
and m7 (and_m7, not_i3, not_i2, not_i1, i0);
and m8 (and_m8, i2, i1, i0);
and m9 (and_m9, not_i3, i2, not_i1);
and m10 (and_m10, not_i3, i0);
and m11 (and_m11, i1, i0);
and m12 (and_m12, not_i3, not_i2, i0);
and m13 (and_m13,not_i3, not_i2, not_i1);
and m14 (and_m14, not_i2, i1);
and m15 (and_m15, i3, not_i1, i0);

//set up the OR gates
or agate (a, and_m0, and_m1, and_m2, and_m3);
or bgate (b, and_m0, and_m1, and_m4, and_m5);
or cgate (c, and_m0, and_m1, and_m6);
or dgate (d, and_m0, and_m1, and_m2, and_m7, and_m8, and_m15);
or egate (e, and_m0, and_m1, and_m9, and_m10, i0);
or fgate (f, and_m0, and_m1, and_m11, and_m12, and_m14);
or ggate (g, and_m0, and_m1, and_m13, and_m8);
```

```
endmodule
```

Tb_BCD_Display.v (Structural)

```
module tb_BCD_structural;


    reg i3;
    reg i2;
    reg i1;
    reg i0;


    wire a,b,c,d,e,f,g, an0, an1, an2, an3, dp;


    BCD_Seg_Display uut(
        .i3(i3),
        .i2(i2),
        .i1(i1),
        .i0(i0),
        .a(a),
        .b(b),
        .c(c),
        .d(d),
        .e(e),
        .f(f),
        .g(g),
        .an0 (an0),
```

```
    .an1 (an1),
    .an2 (an2),
    .an3 (an3),
    .dp (dp)
);

initial begin


    i3 = 0;
    i2 = 0;
    i1 = 0;
    i0 = 0;

    #50

    i3 = 0;
    i2 = 0;
    i1 = 0;
    i0 = 0;
    $display ("TC01");
    if ({a,b,c,d,e,f,g}!= 7'b0000001)$display ("Result is Wrong");

    i3 = 0;
    i2 = 0;
    i1 = 0;
    i0 = 1;
    #50
    $display ("TC02");
```

```
if({a,b,c,d,e,f,g}!= 7'b1001111) $display ("Result is Wrong");

i3 = 0;
i2 = 0;
i1 = 1;
i0 = 0;
#50
$display ("TC03");
if( {a,b,c,d,e,f,g}!= 7'b0010010) $display ("Result is Wrong");

i3 = 0;
i2 = 0;
i1 = 1;
i0 = 1;
#50
$display ("TC04");
if( {a,b,c,d,e,f,g}!= 7'b0000110) $display ("Result is Wrong");

i3 = 0;
i2 = 1;
i1 = 0;
i0 = 0;
#50
$display ("TC05");
if({a,b,c,d,e,f,g}!= 7'b1001100) $display ("Result is Wrong");

i3 = 0;
i2 = 1;
i1 = 0;
```

```
i0 = 1;
#50
$display ("TC06");
if({a,b,c,d,e,f,g}!= 7'b0100100) $display ("Result is Wrong");

i3 = 0;
i2 = 1;
i1 = 1;
i0 = 0;
#50
$display ("TC07");
if({a,b,c,d,e,f,g}!= 7'b0100000) $display ("Result is Wrong");

i3 = 0;
i2 = 1;
i1 = 1;
i0 = 1;
#50
$display ("TC08");
if({a,b,c,d,e,f,g}!= 7'b0001111) $display ("Result is Wrong");

i3 = 1;
i2 = 0;
i1 = 0;
i0 = 0;
#50
$display ("TC09");
if({a,b,c,d,e,f,g}!= 7'b0000000) $display ("Result is Wrong");
```

```
i3 = 1;
i2 = 0;
i1 = 0;
i0 = 1;
#50
$display ("TC10");
if({a,b,c,d,e,f,g}!= 7'b0001100) $display ("Result is Wrong");

i3 = 1;
i2 = 0;
i1 = 1;
i0 = 0;
#50
$display ("TC11");
if({a,b,c,d,e,f,g}!= 7'b1111111) $display ("Result is Wrong");

i3 = 1;
i2 = 0;
i1 = 1;
i0 = 1;
#50
$display ("TC12");
if({a,b,c,d,e,f,g}!= 7'b1111111) $display ("Result is Wrong");

i3 = 1;
i2 = 1;
i1 = 0;
i0 = 0;
#50
```

```
$display ("TC13");
if({a,b,c,d,e,f,g}!= 7'b1111111) $display ("Result is Wrong");

i3 = 1;
i2 = 1;
i1 = 0;
i0 = 1;
#50
$display ("TC14");
if({a,b,c,d,e,f,g}!= 7'b1111111) $display ("Result is Wrong");

i3 = 1;
i2 = 1;
i1 = 1;
i0 = 0;
#50
$display ("TC15");
if({a,b,c,d,e,f,g}!= 7'b1111111) $display ("Result is Wrong");

i3 = 1;
i2= 1;
i1 = 1;
i0 = 1;
#50
$display ("TC16");
if({a,b,c,d,e,f,g}!= 7'b1111111) $display ("Result is Wrong");

end
```

endmodule

BCD_Constraint.xdc (Constraint File)

```
## Switches
set_property PACKAGE_PIN V17 [get_ports {i0}]
   set_property IOSTANDARD LVCMOS33 [get_ports {i0}]
set_property PACKAGE_PIN V16 [get_ports {i1}]
   set_property IOSTANDARD LVCMOS33 [get_ports {i1}]
set_property PACKAGE_PIN W16 [get_ports {i2}]
   set_property IOSTANDARD LVCMOS33 [get_ports {i2}]
set_property PACKAGE_PIN W17 [get_ports {i3}]
   set_property IOSTANDARD LVCMOS33 [get_ports {i3}]

##7 segment display
set_property PACKAGE_PIN W7 [get_ports {a}]
   set_property IOSTANDARD LVCMOS33 [get_ports {a}]
set_property PACKAGE_PIN W6 [get_ports {b}]
   set_property IOSTANDARD LVCMOS33 [get_ports {b}]
set_property PACKAGE_PIN U8 [get_ports {c}]
   set_property IOSTANDARD LVCMOS33 [get_ports {c}]
set_property PACKAGE_PIN V8 [get_ports {d}]
   set_property IOSTANDARD LVCMOS33 [get_ports {d}]
set_property PACKAGE_PIN U5 [get_ports {e}]
   set_property IOSTANDARD LVCMOS33 [get_ports {e}]
set_property PACKAGE_PIN V5 [get_ports {f}]
   set_property IOSTANDARD LVCMOS33 [get_ports {f}]
set_property PACKAGE_PIN U7 [get_ports {g}]
   set_property IOSTANDARD LVCMOS33 [get_ports {g}]
```

```
set_property PACKAGE_PIN V7 [get_ports dp]
   set_property IOSTANDARD LVCMOS33 [get_ports dp]


set_property PACKAGE_PIN U2 [get_ports {an0}]
   set_property IOSTANDARD LVCMOS33 [get_ports {an0}]
set_property PACKAGE_PIN U4 [get_ports {an1}]
   set_property IOSTANDARD LVCMOS33 [get_ports {an1}]
set_property PACKAGE_PIN V4 [get_ports {an2}]
   set_property IOSTANDARD LVCMOS33 [get_ports {an2}]
set_property PACKAGE_PIN W4 [get_ports {an3}]
   set_property IOSTANDARD LVCMOS33 [get_ports {an3}]
```