

DIGITAL DESIGN ASSIGNMENT REPORT



Assignment ID: 2

Student Name: 李子南

Student ID: 12011517

PART 1: DIGITAL DESIGN THEORY

李子南 12011517

$$\textcircled{1} \quad a. F_1 = \sum (m_i), F_2 = \sum (\bar{m}_j)$$

$E = F_1 + F_2 = \sum (m_i) + \sum (\bar{m}_j)$ contains the sum of minterms of F_1 and F_2

$$b. G = F_1 \cdot F_2 = (\sum (m_i)) \cdot (\sum (\bar{m}_j)) = \sum (m_i \cdot \bar{m}_j)$$

$$m_i \cdot \bar{m}_j = m_i \quad \text{if } i=j$$

$\therefore G$ contains only the minterms that are common to F_1 and F_2

$$2. a. F(x, y, z) = \sum (3, 5, 7) = \prod (0, 1, 2, 4, 6)$$

$$b. F(A, B, C, D) = \prod (3, 5, 8, 11, 12, 15) = \sum (0, 1, 2, 4, 6, 7, 9, 10, 13, 14)$$

$$\begin{aligned} 3. a. F &= (b+d)(a'+b'+c)(a'+c') \\ &= (b'a' + b'b' + b'c + a'd + b'd + cd)(a'+c') \\ &= (b'+d)(a'+c') \\ &= a'b' + a'd + c'b' + cd \rightarrow SOP \\ &= a'b'c(c+c')(d+d') + a'd(b'+b)(c'+c') + cb'(a+a')(d+d') + cd(b+b')(a+a') \\ &= m_0 + m_1 + m_2 + m_3 + m_4 + m_6 + m_{10} + m_{11} + m_{15} \\ &= \sum (0, 1, 2, 3, 5, 7, 10, 11, 15) \rightarrow \text{canonical form} \end{aligned}$$

$$\begin{aligned} b. F &= xy + x'z' + x'yz \\ &= (xy + x'z' + x'y)(xy + x'z' + z) \\ &= (y + x'z')(xy + z + x') \cdot xy \\ &= xy(x'y + z + x') \\ &= (xy + xyz) \\ &= xy + z z' \\ &= (xy + z)(xy + z') \\ &= (z+y)(x+z)(x+z')(z'+y) \rightarrow POS \\ &= (z+y + xx')(x+z+yy')(x+z'+yy')(z'+y+xx') \\ &= M_0 M_1 M_2 M_3 M_4 M_5 \\ &= \prod (0, 1, 2, 3, 4, 5) \rightarrow \text{canonical form} \end{aligned}$$

$$4. a. \textcircled{1} y'z' + yz' + x'z$$

$$\begin{aligned} &= z' + x'z \\ &= (z' + z)(z' + x) \\ &= z' + x' \end{aligned}$$

$$\begin{aligned} \textcircled{2} \quad &x' + xz' \\ &= (x' + x)(x' + z') \\ &= x' + z' \end{aligned}$$

a is true

$$b. \quad 0x'y + xz + y'z'$$

$$= x'y(z+z') + xz(y+y') + y'z'(x+x')$$

$$= \sum(0, 2, 3, 4, 5, 7)$$

$$0 \quad xy' + xz' + yz$$

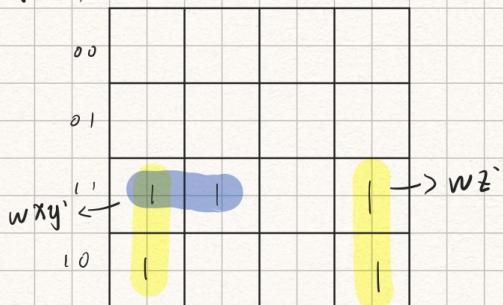
$$= xy'(z+z') + xz'(y+y') + yz(x+x')$$

$$= \sum(0, 2, 3, 4, 5, 7)$$

b is true

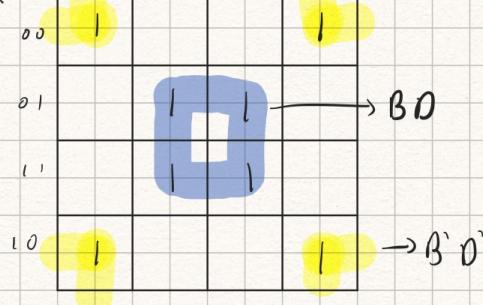
00	01	11	10
1		1	1
1	1	1	1

5. a. $00 \quad 01 \quad 11 \quad 10$



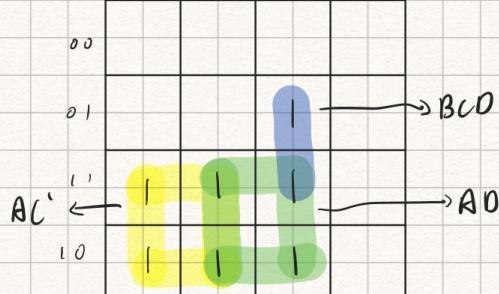
$$F(w, x, y, z) = wz' + wxy'$$

b. $00 \quad 01 \quad 11 \quad 10$



$$F(A, B, C, D) = BD + B'D'$$

c. $00 \quad 01 \quad 11 \quad 10$



$$F(A, B, C, D) = AC' + AD + BCD$$

A Karnaugh map for four variables (A, B, C, D) showing the minterms for the given function. The rows are labeled A' (top), A (middle), and A'' (bottom). The columns are labeled D' (left), D (middle), and D'' (right). The minterms are represented by shaded cells:

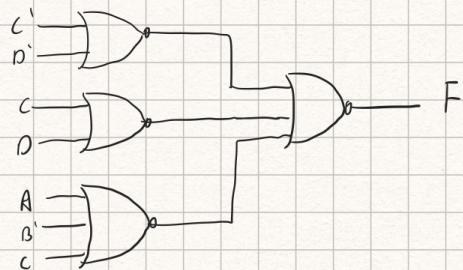
- $A'B'C'D$: Cell (A', B', C', D')
- $A'BC'D$: Cells (A', B', C, D') and (A', B', C', D)
- $A'BC'D'$: Cell (A', B', C, D')
- $ABC'D$: Cells (A, B, C', D) and (A, B, C, D')
- $ABC'D'$: Cells (A, B, C', D') and (A, B, C, D')

Arrows point from the labels to their respective groups of cells.

$$F(A, B, C, D) = AB'C + ACD + B'C'D + ABC'$$

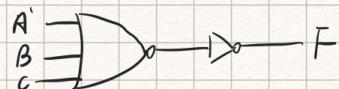
		00	01	11	10
		00	1	0	1
A B C	D	00	0	0	0
		01	0	0	0
11	0	1	0	1	
10	0	1	0	1	

$$F(A, B, C, D) = (C + D)(C' + D')(A + B' + C)$$



b.	00	01	11	10
00)
01				
11				
10	0	0		

$$F(A, B, C, D) = (A' + B + C)$$



7.	a.	00	01	11	10
		X			1
$C'D'$	←	1		1	X
		1			
		X			1

 $\rightarrow A'BC$ $\rightarrow B'C'D'$

$$F(A, B, C, D) = C'D' + B'C'D' + A'BC$$

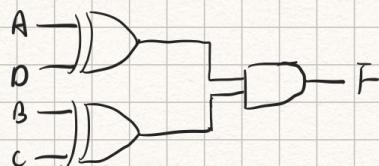
$$= \sum(0, 2, 4, 6, 7, 8, 10, 12)$$

b.	00	01	11	10
$A'C'$	←	X	X	1
	01	1	X	
	11	1		
	10	X		1

 $\rightarrow B'C'D'$ $\rightarrow AD'$

$$F(A, B, C, D) = AD' + A'C' + B'C'D' = \sum(0, 1, 2, 4, 5, 8, 10, 12, 14)$$

$$\begin{aligned}
 8. \quad F &= A'B'C'D + AB'C'D' + ABC'D' + A'B'CD \\
 &= A'D(B'C + B'C) + AD'(B'C + BC') \\
 &= (B'C + B'C)(A'D + AD') \\
 &= (B \oplus C)(A \oplus D)
 \end{aligned}$$



PART 2: DIGITAL DESIGN LAB (TASK 1)

DESIGN

```
module seg(
    input  a, b, c, d, e,
    output reg [7:0] seg_out,
    output reg[7:0]seg_en
);
    always @*
    begin
        case (e)
            1'b0:
                casex({a,b,c,d})
                    4'b1xxx:begin
                        seg_en=8'b1111_1110;
                        seg_out=8'b0100_0000;
                        end // 0
                    4'b01xx:begin
                        seg_en=8'b1111_1110;
                        seg_out=8'b0111_1001;
                        end // 1
                    4'b001x:begin
                        seg_en=8'b1111_1110;
                        seg_out=8'b0010_0100;
                        end // 2
                    4'b0001: begin
                        seg_en=8'b1111_1110;
                        seg_out=8'b0011_0000;
                        end // 3
                    default: seg_en=8'b1111_1111;
                endcase
            1'b1:
                casex({a,b,c,d})
                    4'bxxxx1: begin
                        seg_en=8'b1111_1110;
                        seg_out=8'b0011_0000;
                        end // 3
                    4'bxx10:begin
                        seg_en=8'b1111_1110;
                        seg_out=8'b0010_0100;
                        end // 2
                    4'bx100:begin
                        seg_en=8'b1111_1110;
                        seg_out=8'b0111_1001;
                        end // 1
                    4'b1000:begin
                        seg_en=8'b1111_1110;
                        seg_out=8'b0100_0000;
                        end // 0
                    default: seg_en=8'b1111_1111;
                endcase
            endcase
        end
    endmodule
```

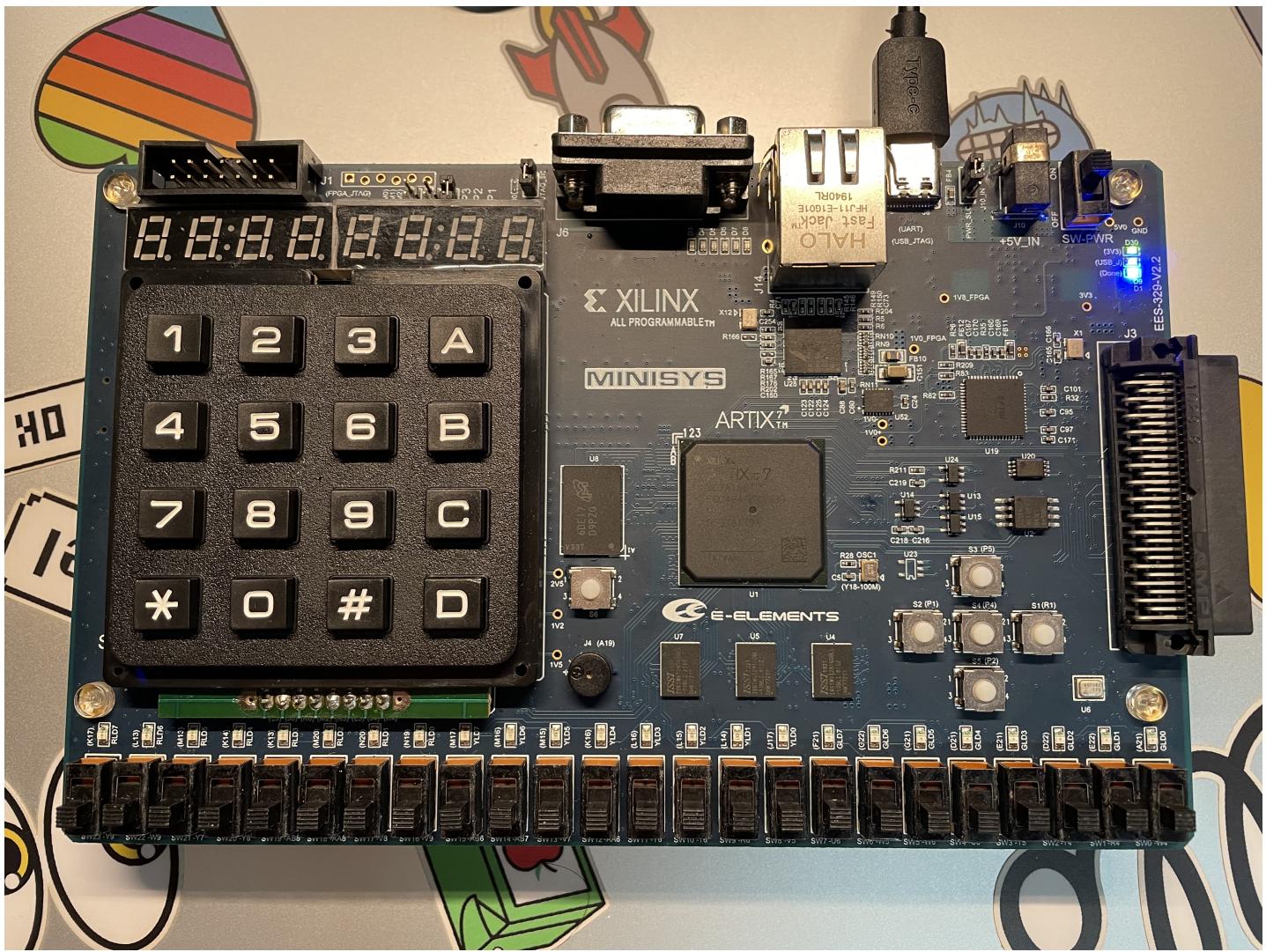
CONSTRAINT FILE AND TESTING

```
set_property IOSTANDARD LVCMOS33 [get_ports {seg_en[0]}]
set_property IOSTANDARD LVCMOS33 [get_ports {seg_en[1]}]
set_property IOSTANDARD LVCMOS33 [get_ports {seg_en[2]}]
set_property IOSTANDARD LVCMOS33 [get_ports {seg_en[3]}]
set_property IOSTANDARD LVCMOS33 [get_ports {seg_en[4]}]
set_property IOSTANDARD LVCMOS33 [get_ports {seg_en[5]}]
set_property IOSTANDARD LVCMOS33 [get_ports {seg_en[6]}]
set_property IOSTANDARD LVCMOS33 [get_ports {seg_en[7]}]
set_property IOSTANDARD LVCMOS33 [get_ports {seg_out[0]}]
set_property IOSTANDARD LVCMOS33 [get_ports {seg_out[1]}]
set_property IOSTANDARD LVCMOS33 [get_ports {seg_out[2]}]
set_property IOSTANDARD LVCMOS33 [get_ports {seg_out[3]}]
set_property IOSTANDARD LVCMOS33 [get_ports {seg_out[4]}]
set_property IOSTANDARD LVCMOS33 [get_ports {seg_out[5]}]
set_property IOSTANDARD LVCMOS33 [get_ports {seg_out[6]}]
set_property IOSTANDARD LVCMOS33 [get_ports {seg_out[7]}]
set_property IOSTANDARD LVCMOS33 [get_ports {a}]
set_property IOSTANDARD LVCMOS33 [get_ports {b}]
set_property IOSTANDARD LVCMOS33 [get_ports {c}]
set_property IOSTANDARD LVCMOS33 [get_ports {d}]
set_property IOSTANDARD LVCMOS33 [get_ports {e}]
set_property PACKAGE_PIN AB8 [get_ports {e}]
set_property PACKAGE_PIN Y9 [get_ports {a}]
set_property PACKAGE_PIN W9 [get_ports {b}]
set_property PACKAGE_PIN Y7 [get_ports {c}]
set_property PACKAGE_PIN Y8 [get_ports {d}]
set_property PACKAGE_PIN A18 [get_ports {seg_en[7]}]
set_property PACKAGE_PIN A20 [get_ports {seg_en[6]}]
set_property PACKAGE_PIN B20 [get_ports {seg_en[5]}]
set_property PACKAGE_PIN E18 [get_ports {seg_en[4]}]
set_property PACKAGE_PIN F18 [get_ports {seg_en[3]}]
set_property PACKAGE_PIN D19 [get_ports {seg_en[2]}]
set_property PACKAGE_PIN E19 [get_ports {seg_en[1]}]
set_property PACKAGE_PIN C19 [get_ports {seg_en[0]}]
set_property PACKAGE_PIN E13 [get_ports {seg_out[7]}]
set_property PACKAGE_PIN C15 [get_ports {seg_out[6]}]
set_property PACKAGE_PIN C14 [get_ports {seg_out[5]}]
set_property PACKAGE_PIN E17 [get_ports {seg_out[4]}]
set_property PACKAGE_PIN F16 [get_ports {seg_out[3]}]
set_property PACKAGE_PIN F14 [get_ports {seg_out[2]}]
set_property PACKAGE_PIN F13 [get_ports {seg_out[1]}]
set_property PACKAGE_PIN F15 [get_ports {seg_out[0]}]
```

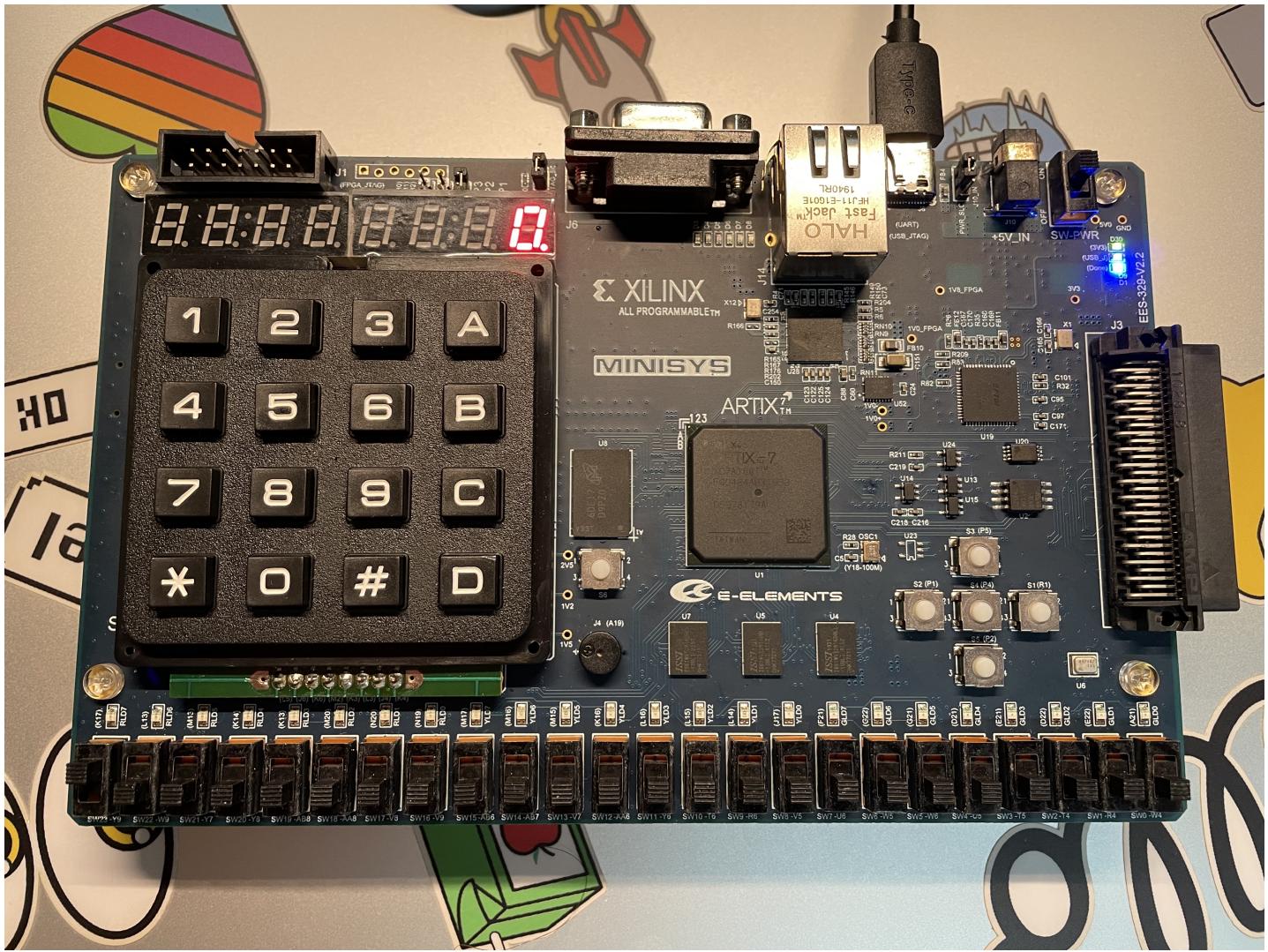
Test result

下方按钮从左到右依次代表0号, 1号, 2号, 3号, 优先级切换开关

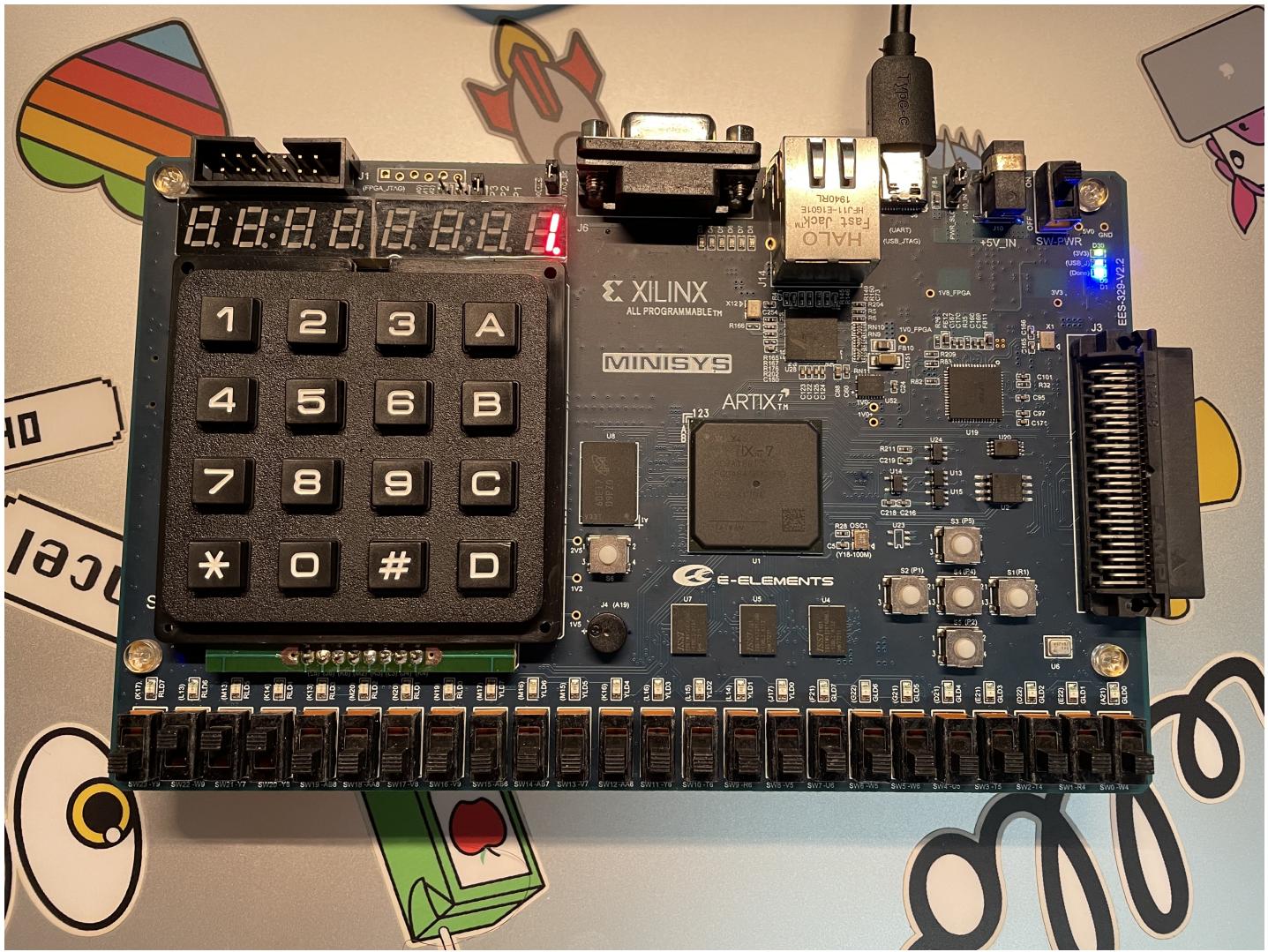
- 所有开关都为关闭状态, 灯不亮。



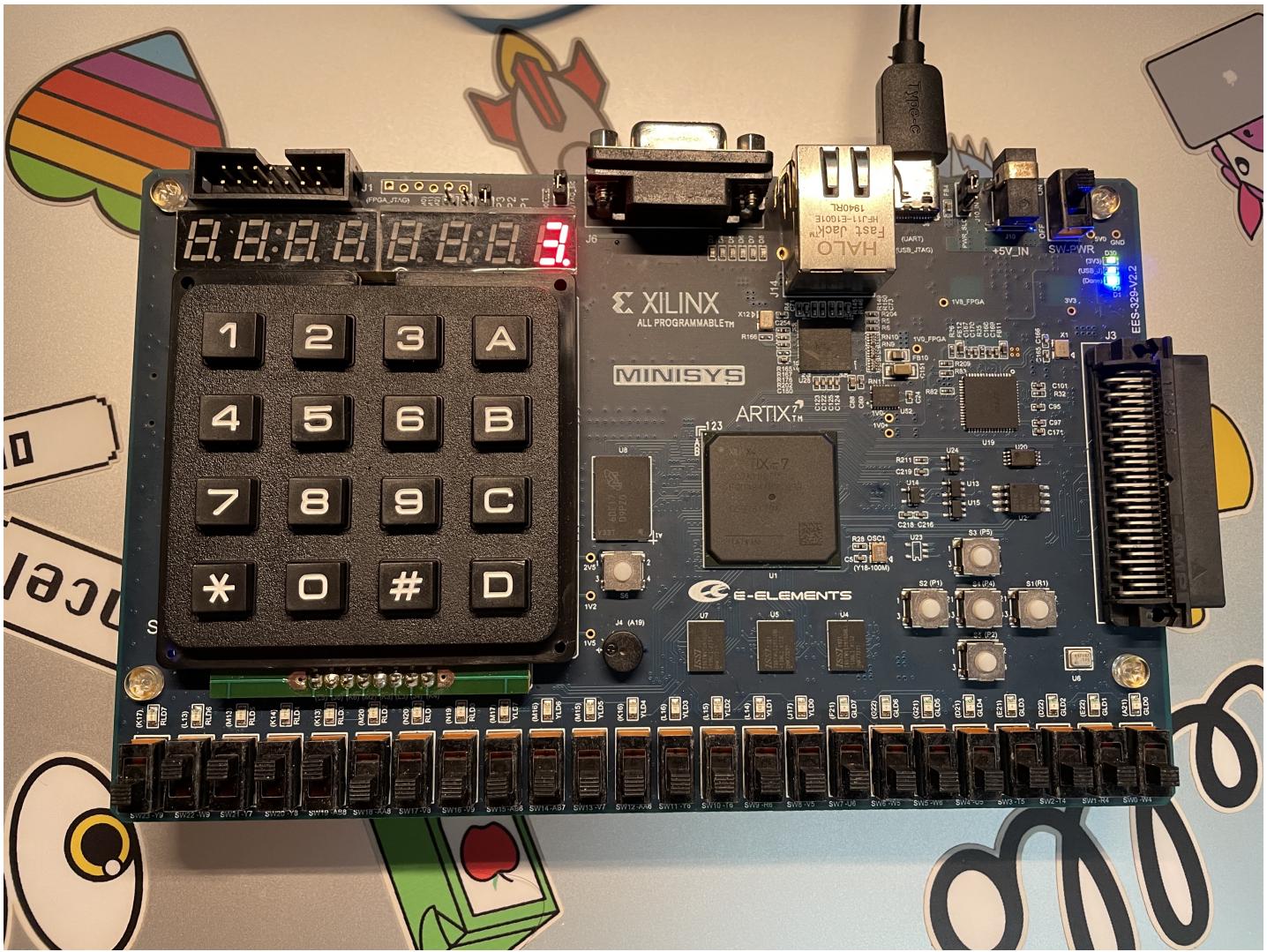
2. 0号开关打开，显示为0



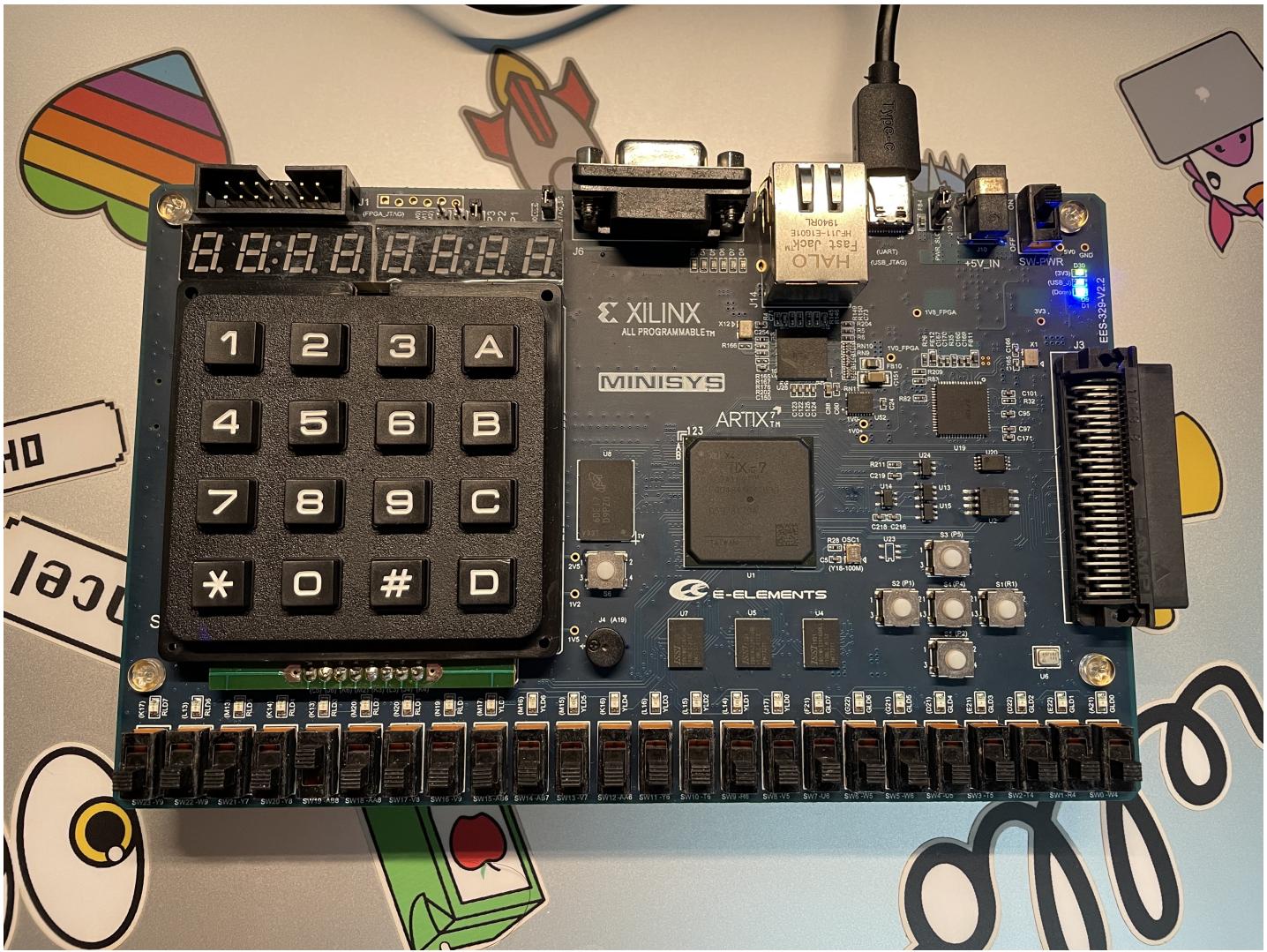
3. 1, 2, 3号开关同时打开，优先级开关关闭。按照优先级显示1号病房。



4. 1, 2, 3号开关同时打开，优先级开关开启。按照优先级显示3号病房。



5. 所有病房开关关闭，优先级开关开启，依然不显示。



THE DESCRIPTION OF OPERATION

一开始我将设计文件中的seg_en改为了1位宽，但无法生成比特流文件，后来再回去翻ppt才搞懂，seg_en应该是8位，亮的灯比特征为1。之后成功生成了比特流文件并烧写进了板子里，但还是不能正常运行，2号和3号病房的开关变成了0和1，后面发现是因为我把程序写在了两个模块中，并且在参数传递的时候出现了问题。之后我将所有代码整合到一个模块里，就可以正常运行了。

PART 2: DIGITAL DESIGN LAB (TASK 2)

$$\begin{aligned}
 F(a,b,c) &= a \oplus b \oplus c \\
 &= (a'b + ab') \oplus c \\
 &= (a'b + ab')c' + (ab')c(a' + b)c \\
 &\geq a'b c' + ab' c' + a'b' c + abc \\
 &= \sum(1, 2, 4, 7) \\
 &= \prod(0, 3, 5, 6)
 \end{aligned}$$

	00	01	11	10
0		1		1
1	1		1	

It can't be simplified

Truth-table

{a,b,c}	F
000	0
001	1
010	1
011	0
100	1
101	0
110	0
111	1

DESIGN

SOP

```

module Task2_SOP(
  input[0:0] a,b,c,
  output[0:0] out
);
  assign out = (~a & b & ~c) | (a & ~b & ~c) | (a & b & c) | (~a & ~b & c);
endmodule

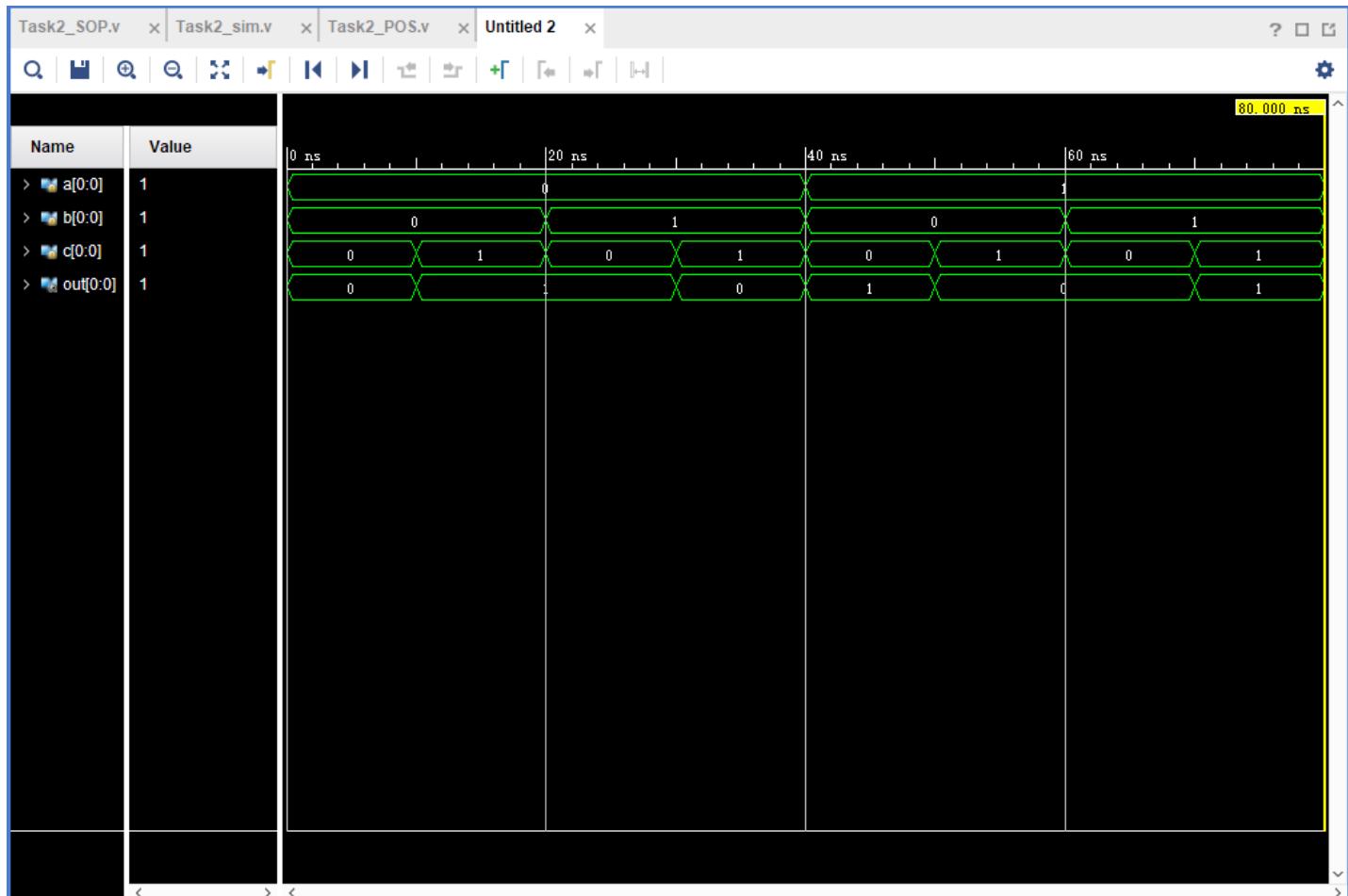
```

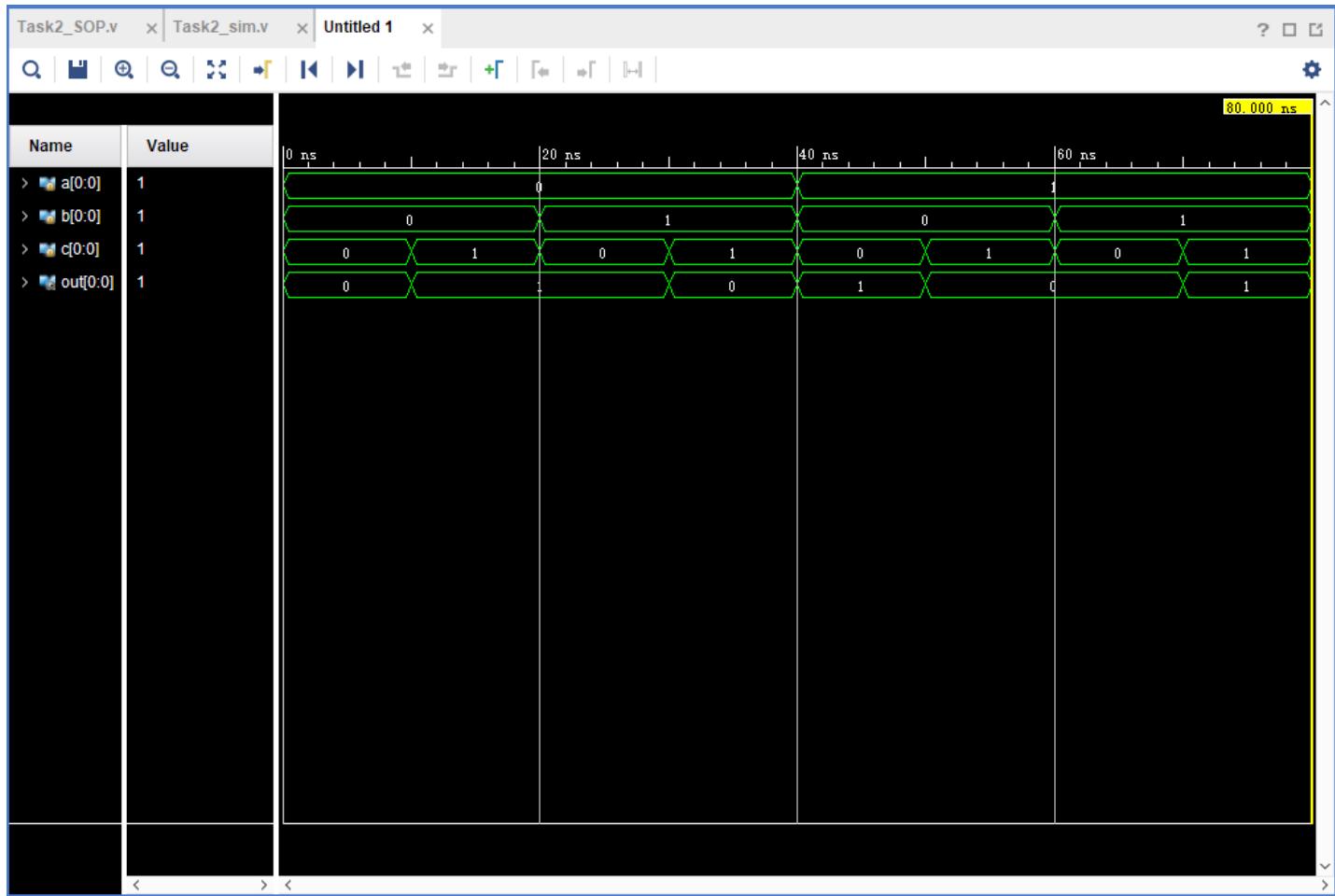
POS

```
module Task2_POS(
    input[0:0] a,b,c,
    output[0:0] out
);
    assign out = (a | b | c) & (a | ~b | ~c) & (~a | b | ~c) & (~a | ~b | c);
endmodule
```

SIMULATION

```
module Task2_sim();
reg[0:0] a,b,c;
wire[0:0] out;
    Task2_POS pos(a,b,c,out); //or Task2_SOP sop(a,b,c,out)
initial
begin
{a,b,c} = 3'b000;
while({a,b,c} < 3'b111)begin
    #10 {a,b,c} = {a,b,c} + 1;
end
#10 $finish;
end
endmodule
```





根据真值表，模拟结果正确。

THE DESCRIPTION OF OPERATION

本题我先是在纸上化简了原式，得到了SOP形式和POS形式，之后写出了设计文件和仿真文件，但我SOP的设计文件第一次写漏了一个最小项，让我疑惑了半个小时，后面发现了就没事了。