****

Combinational Lock

**Project Report**

**Course: Digital Logic Design**

### **Instructor: Asma Ramay**

**Submitted by:**

**Zain Ali Shah**

**FA19-BCE-092**

**Introduction:**

A combination lock is a type of locking device in which a [sequence](https://en.wikipedia.org/wiki/Sequence) of symbols, usually numbers, is used to open the lock. The sequence may be entered using a single rotating dial which interacts with several discs or *cams*, by using a set of several rotating discs with inscribed symbols which directly interact with the locking mechanism, or through an electronic or mechanical keypad. Types range from inexpensive three-digit luggage locks to high-security safes. Unlike ordinary padlocks, combination locks do not use keys. In which a sequence of numbers is used to open the lock.

**Components:**

Two 7 SEG BCD Display.

Four 4077 XNOR Gates.

One 4-input AND Gate.

One NOT Gate.

Two RED LEDs.

Eight switches for Data Entry and Key Code.

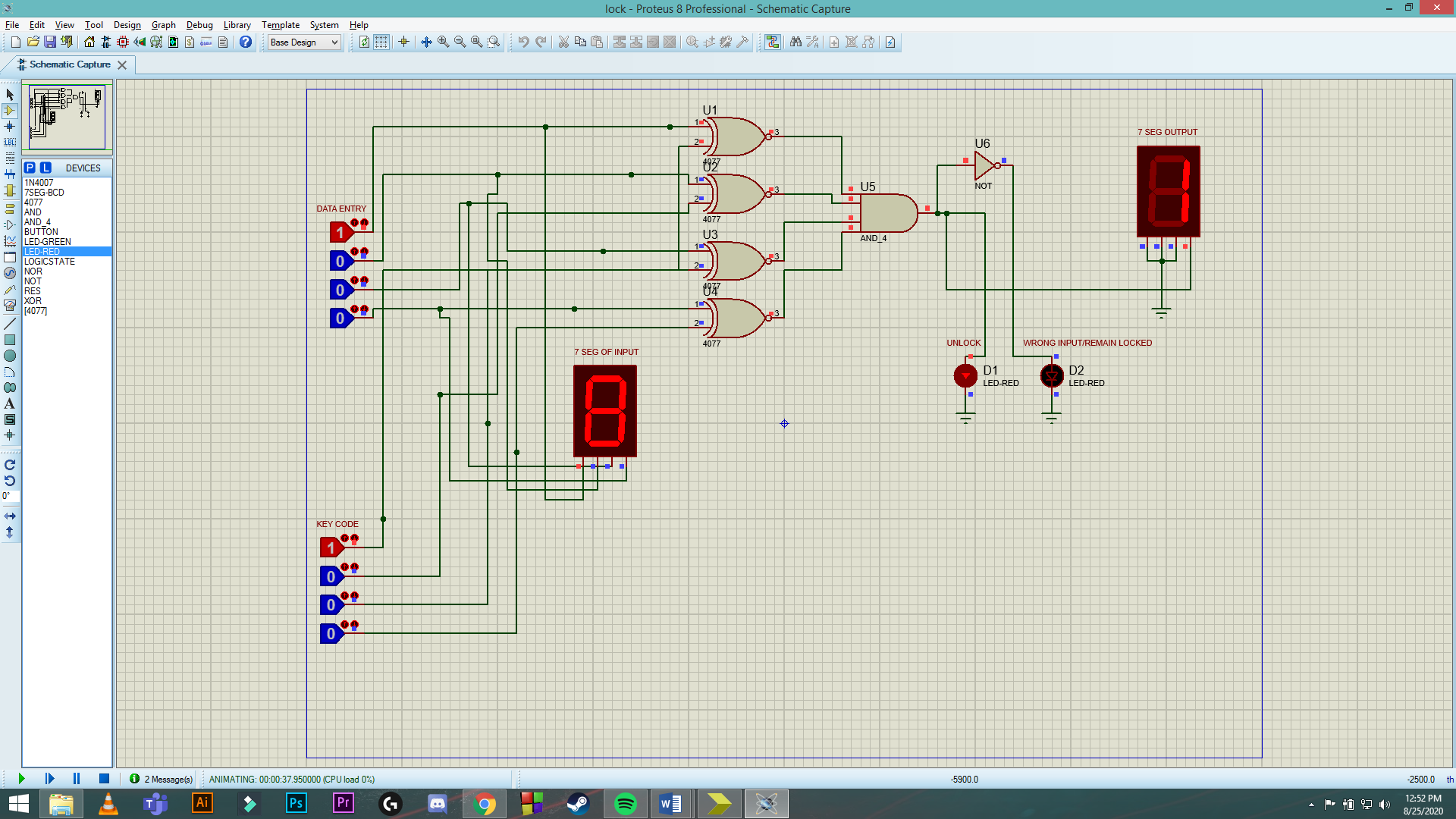
**Working of Combinational Lock:**

Four switches act to hold the correct code for unlocking the lock, while the other switches serve as a data entry point for the person trying to open the lock.

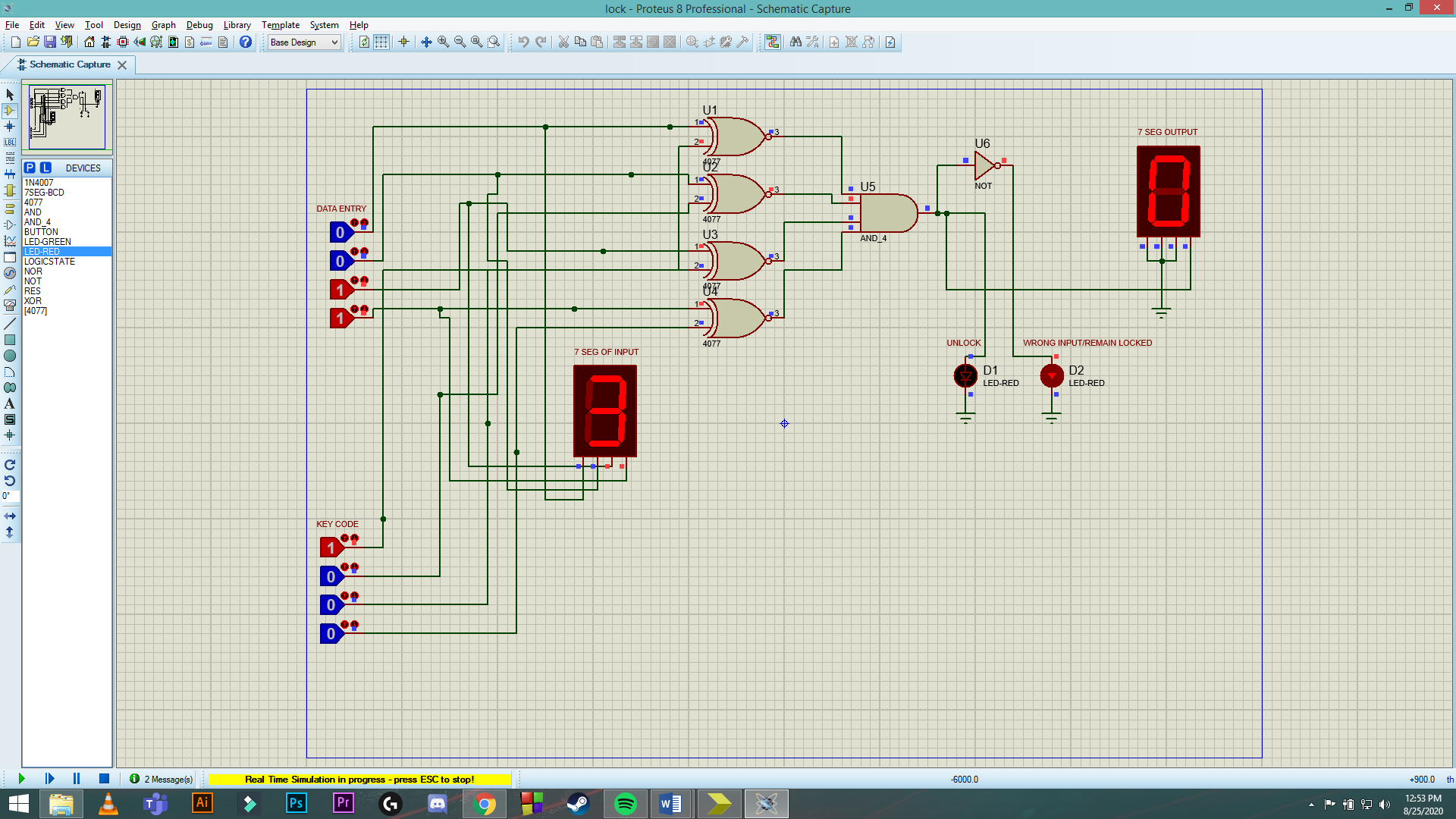
In real life, of course, the switch assembly with the “key” code set on it must be hidden from the sight of the person opening the lock, which means it must be physically located *elsewhere* from where the data entry switch assembly is. This requires two switch assemblies.

**Proteus:**

1. **When the combination is correct:**



1. **When the combination isn’t correct:**



**Circuit Explanation:**

This circuit illustrates the use of XNOR (Exclusive-NOR) gates as bit comparators. Four of these XNOR gates compare the respective bits of two 4-bit binary numbers, each number “entered” into the circuit via a set of switches. If the two numbers match, the output of XNOR gates will be 1, otherwise 0. Now I have connected all XNOR gates output to one 4-input AND gate. If all XNOR gates give 1 output, AND gate will give 1 output. And even if one XNOR gate gives 0, output on AND gate will be zero. Now the output of AND gate is connected to two LEDs. One LED turns on when AND gate output is 1, which means that the key code and Data entry match. The lock is now unlocked. The other LED is connected to AND gate output via a NOT gate. So when the AND gate output is 0, the NOT gate converts it to 1, turning the 2nd LED on and indicating that the Key Code and Data Entry don’t match. For visual representation, I have connected the Data Entry inputs or switches to a 7 Segment BCD display, which will show what digit you’re giving as input. The other 7 Segment BCD display that I have used is to show if the Lock is OPEN or CLOSED. Three inputs of this 2nd display is connected to ground. This means that this display can only show 0 (when the codes don’t match and lock remains CLOSED) and 1 (when the codes match and lock is now OPEN).

The “key” code that must be matched at the data entry switch array should be hidden from view, of course. If this were part of a real security system, the data entry switch assembly would be located ***outside***the door and the key code switch assembly ***behind***the door with the rest of the circuitry.

**Advantages:**

**1) Don’t need any key**

How’s you feel if you don’t need to pick a set of keys. It is a great feeling when you know that now you don’t need to carry your house key with you everywhere. Just dial a pin code and open your main entrance door. Having a digital lock means there are no chance of lost your keys.

**2) Hard to break**

Biometric or digital locks are very hard to break for intruders and burglars. If you install a biometric lock in your house, you are ensured that your personal belongings are safe and secure. The main benefits of having a digital lock that the doors are automatically locked when you closed it.

**3) Maintenance Free**

Most of the homeowners have a misconception that biometric locks need maintenance and repair service after some times. They also think that these locks require a huge amount of electricity for running. But the truth is something else.

**Disadvantages:**

**1) Power Failure**

There is no doubt that digital locks run with a single battery for several years. But some locks run with only electricity. If you have installed an electric lock on your house, you cannot operate it in the absence of electricity.

**2) Forgetful**

With a biometric lock, there is a possibility that you forget your pin that you set to unlock it. Once you forget the code, you are not able to access the door until to a professional not reach you.

Having a keyless door lock system is a great idea, but you need to consider its disadvantages also. If you have lost your keys various times, then digital locks are good for you.

**3) Don’t share the Pin**

Sometimes, in an emergency situation, you need to share your pic code with your neighbors or friends. Before sharing your code with them, make sure you tell the code only that person who you trust. If by mistake, you share it with a wrong person, there are chances of burglary.

**Xilinx Code**:

1. **Main Module:**

module lock(i,j,y,a,bcd1,bcd2

);

input [3:0]i,j;

output y;

output a;

assign z=1'b0;

output [7:0] bcd1,bcd2;

wire x,g,p,q;

xnor xnor1(x,i[0],j[0]);

xnor xnor2(g,i[1],j[1]);

xnor xnor3(p,i[2],j[2]);

xnor xnor4(q,i[3],j[3]);

and and1(y,g,p,q,x);

not g1(a,y);

seg g2(j,bcd1);

seg g3({z,z,z,y},bcd2);

endmodule

1. **7 Segment Module:**

module seg(bcd, segments

);

input [3:0] bcd;

output reg [7:0]segments;

always@(bcd)begin

case(bcd)

0: segments=8'b10000000;

1: segments=8'b11111001;

2: segments=8'b10100100;

3: segments=8'b10110000;

4: segments=8'b10011001;

5: segments=8'b10010010;

6: segments=8'b10000010;

7: segments=8'b11111000;

8: segments=8'b10000000;

9: segments=8'b10010000;

10: segments=8'b10001000;

11: segments=8'b10000011;

12: segments=8'b11000110;

13: segments=8'b10100001;

14: segments=8'b00001110;

15: segments=8'b10001110;

default: segments=8'b11111111;

endcase

end

endmodule

1. **Test Bench:**

module test;

// Inputs

reg [3:0] i;

reg [3:0] j;

// Outputs

wire y;

wire a;

wire [7:0] bcd1;

wire [7:0] bcd2;

// Instantiate the Unit Under Test (UUT)

lock uut (

.i(i),

.j(j),

.y(y),

.a(a),

.bcd1(bcd1),

.bcd2(bcd2)

);

initial begin

// Initialize Inputs

i = 4'b1010;

j = 4'b0101;

#10;

j= 4'b1010;

#10;

i= 4'b1111;

j= 4'b1111;

#10;

i= 4'b0001;

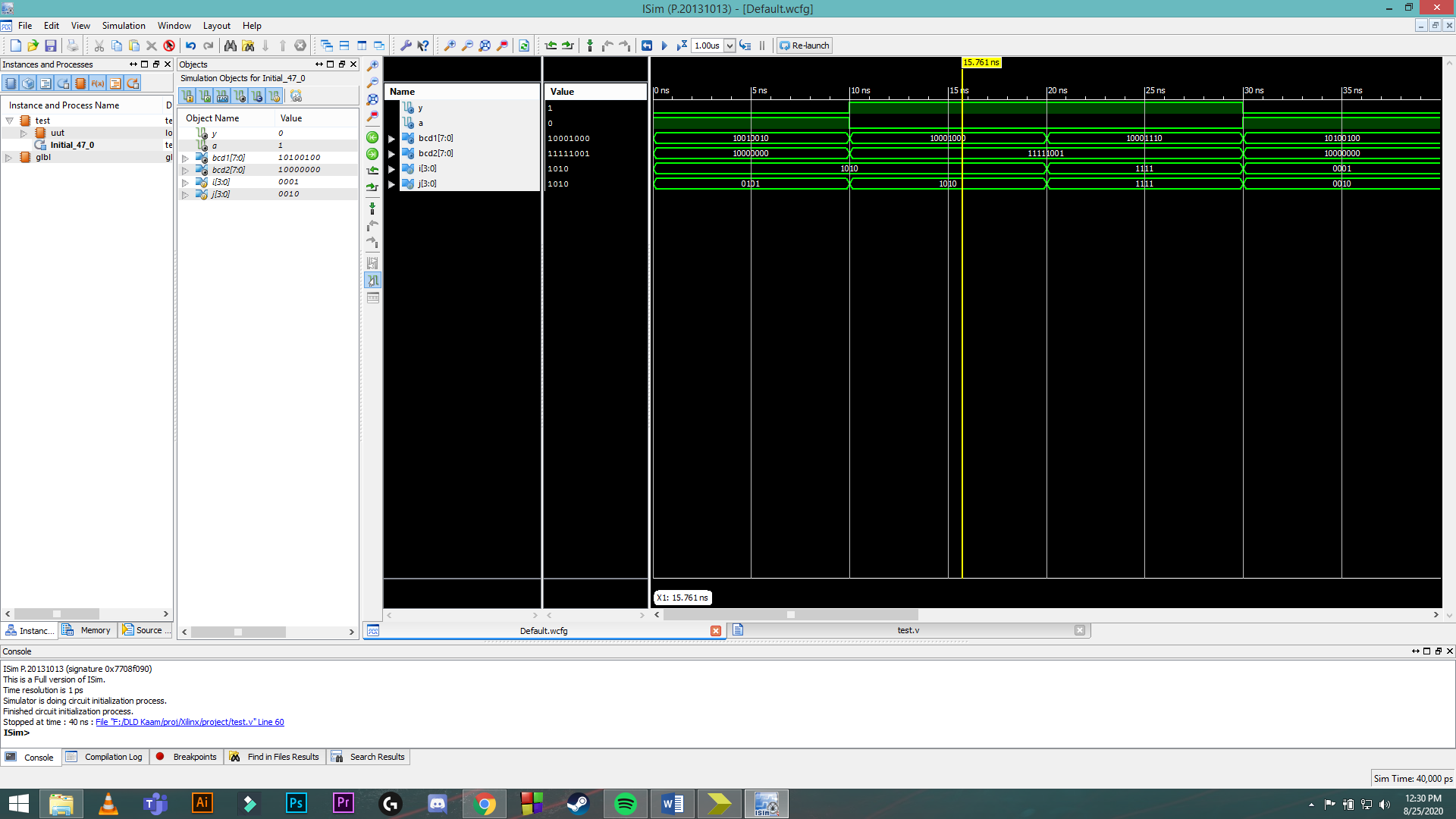
j= 4'b0010;

#10 $finish;

End

Endmodule

1. **Output Graph:**



**Code Explanation**:

1. In the main module, I’ve declared two 4-bit inputs i and j, which represent the key code and data entry respectively.
2. I have declared two outputs, y and a, which represent the UNLOCK and LOCK state of the combinational lock respectively.
3. I have declared two 8 bit outputs, bcd1 and bcd2 representing the BCD display outputs.
4. I have also declared a 1-bit register equal to zero, which goes into the input of bcd2 to make it only show 0 or 1.
5. Then I used four XNOR gates as shown in the circuit and connected all the outputs to AND gate.
6. Then I also used NOT gate as shown in the circuit.
7. Then I called in the 7 segment module to two times, first to show the data entry and the second one to show if the output is 0 (closed) or 1 (open).
8. The 7 segment module consists of the output that it has to show from 0 to 15. The default case for it is 1111111 which means that all LEDs will be off.
9. In the test bench, first I have entered i=1010, which means that if j is the same, y will be 1, otherwise a will be 1. So I entered wrong data entry i.e j=0101. The output in the graph will show that the Lock remains in CLOSED state. Then I entered the correct value of j, so the lock goes to OPEN state. I further tried different data entry and key codes for better clarity.

**Conclusion**:

This project shows how we can use the working of XNOR gates to create a combinational lock and use 7 segment displays for the output of the code, I have used Proteus and Xilinx simulation to show the circuit and output the result with graphs respectively. This is a combinational lock with two input assemblies, one is KEY CODE and the other is DATE ENTRY. The behavior of XNOR gate is very helpful in making these small combinational locks and are very helpful on large scale and industrial/professional use.