CSE 331

Group 7 – Project Report

Section: 04

Faculty: Dr. Dihan Md. Nurudddin Hasan (DMH)

Member Names	ID
Fazla Rabbi Sajid	1513155642
Sourav Sutradhar	1712126042
Aminul Islam Joy	1711867642
Siful Islam Munna	1812682642
Montakim Afsar	1812491042
Md Imam Hossain	1510836042

Contents

Work distribution:

Fazla Rabbi Sajid	Proteus Simulation, Program Flow Chart, Hardware implementation
Sourav Sutradhar	Arduino Code, Literature review, Hardware implementation
Aminul Islam Joy	Logisim Circuit, Circuit Diagram, Hardware implementation
Siful Islam Munna	Arduino Code, Equipment Management, Hardware implementation
Montakim Afsar	Proteus Simulation, KMap Derivation, Hardware implementation
Md Imam Hossain	Logisim Circuit, Background Research, Hardware implementation

Question

Implement the given encryption table using a microcontroller. Use single pole, double throw switch to configure the inputs for high and low conditions. Use LEDs to represent the corresponding output statuses

Input				Output			
I3	I2	I1	I0	O3	O2	O2	O1
0	0	0	0	1	1	0	0
1	0	0	0	1	1	0	1
0	1	0	0	0	1	1	0
1	1	0	0	1	1	0	0
0	0	1	0	1	0	1	1
1	0	1	0	1	0	1	0
0	1	1	0	1	0	1	1
1	1	1	0	0	0	0	0
0	0	0	1	0	0	0	1
1	0	0	1	0	0	1	1
0	1	0	1	1	0	1	0
1	1	0	1	1	1	1	0
0	0	1	1	0	1	0	0
1	0	1	1	1	1	0	0
0	1	1	1	1	1	1	0
1	1	1	1	1	1	0	0

1. General Description

1.1 Arduino UNO

Arduino Uno is a microcontroller board based on the ATmega328. It has 20 digital input/output pins (of which 6 can be used as PWM outputs and 6 can be used as analog inputs), a 16 MHz resonator, a USB connection, a power jack, an in-circuit system programming (ICSP) header, and a reset button. It will be connected with sensors and the GSM and a code will be uploaded to the Arduino uno for the other components to work properly.

The Arduino Uno operates from 6-10V power supply, there are two way of supplying the power to the board. It can be directly powered by the USB connection with PC or we can plug in external AC to DC battery to power the board. The board works efficiently with power supply of 7V to 12V.

1.2 Arduino IDE

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. This software can be used with any Arduino board. We write our code in this compiler and then connect the microcontroller via USB to upload the code in it.

1.3 Proteus 8 Pro

The Proteus Design Suite is a software suite containing schematic, simulation as well as PCB designing. Schematic Capture in the Proteus Design Suite is used for both the simulation of designs and as the design phase of a PCB layout project. It is therefore a core component and is included with all product configurations.

We simulate our circuit in Proteus 8 Pro using Arduino Uno, resistors, single pole double throw switches, LED lights.

1.4 Logisim

Logisim is a logic simulator which permits circuits to be designed and simulated using a graphical user interface. The software is used most often by students in computer science classes to design and experiment with digital circuits in simulation. Circuits are designed in Logisim using a graphical user interface similar to traditional drawing programs, an interface also found in many other simulators.

2. Equipment

We used varieties for software and hardware to finish the projects, those are

1) Logisim
2) Arduino UNO
3) Arduino IDE
4) Proteus 8 Pro
5) 10k Ohm Resistors
6) 330 Ohm Resistors
7) LEDs
8) Single Pole Double Throw switches

3. Method of Derivation

We use the given table for our group 7 to get the sum of products for the outputs O0, O1, O2, O3. We build the table on Logisim and find the expression for each of the output using k-map.

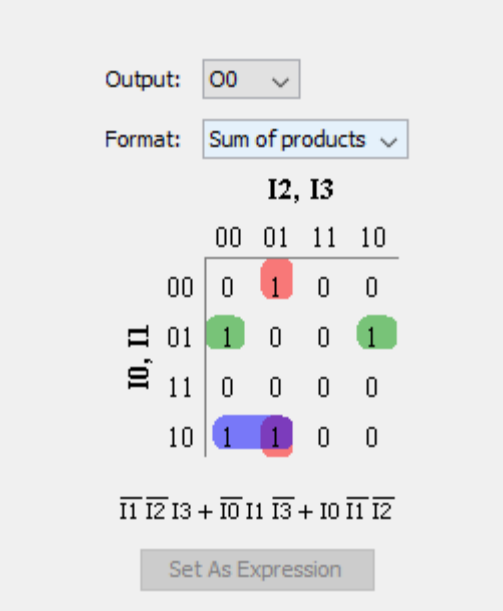
3.1 Truth Table

I0	I1	I2	I3	O0	O1	O2	O3
0	0	0	0	0	0	1	1
0	0	0	1	1	0	1	1
0	0	1	0	0	1	1	0
0	0	1	1	0	0	1	1
0	1	0	0	1	1	0	1
0	1	0	1	0	1	0	1
0	1	1	0	1	1	0	1
0	1	1	1	0	0	0	0
1	0	0	0	1	0	0	0
1	0	0	1	1	1	0	0
1	0	1	0	0	1	0	1
1	0	1	1	0	1	1	1
1	1	0	0	0	0	1	0
1	1	0	1	0	0	1	1
1	1	1	0	0	1	1	1
1	1	1	1	0	0	1	1

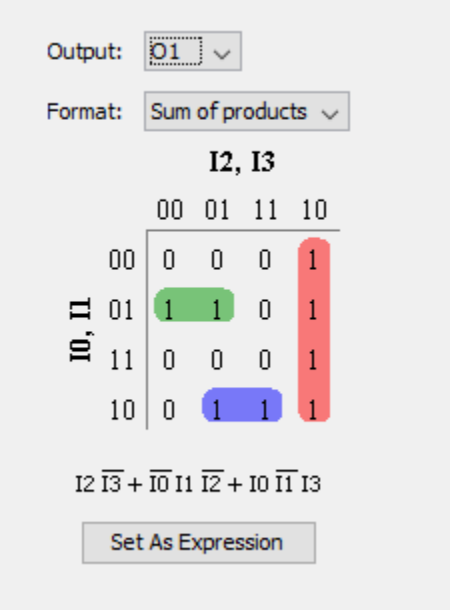
3.2 Derived Result

We use the K-Map to derive the expression for the output.

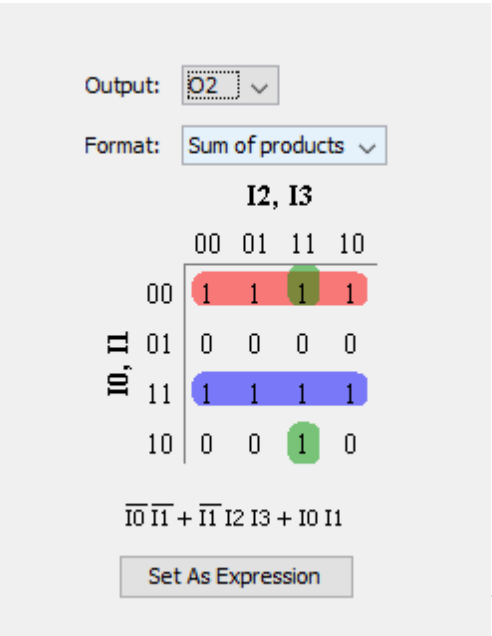
O0:



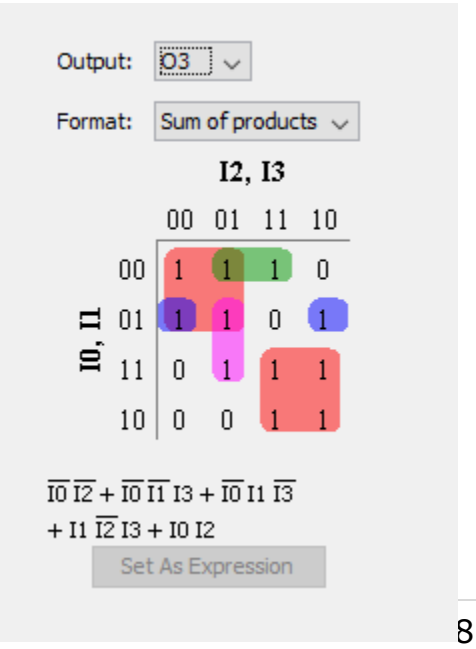
O1:



O2:



O3:



After using the K-Map method we get these expressions:

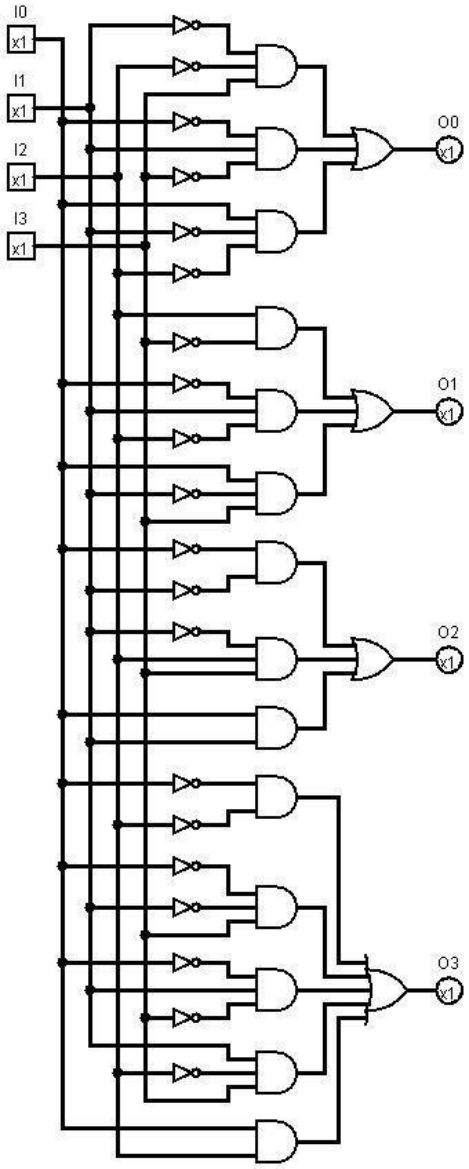
O0 = $\overline{I1}I2\overline{I3} + I0\,\overline{I1}\,\overline{I3} + \overline{I0}\,I1\,I2\,\overline{O1}\,\overline{=}$

$I2\,I3 + I0\overline{I1}\,I2 + \overline{I0}\,I1\overline{I3}$ **O2 =** $\overline{I0}\,I1$

$+ I1\,\overline{I2}\,\overline{I3} + \overline{I0}\,\overline{I1}$

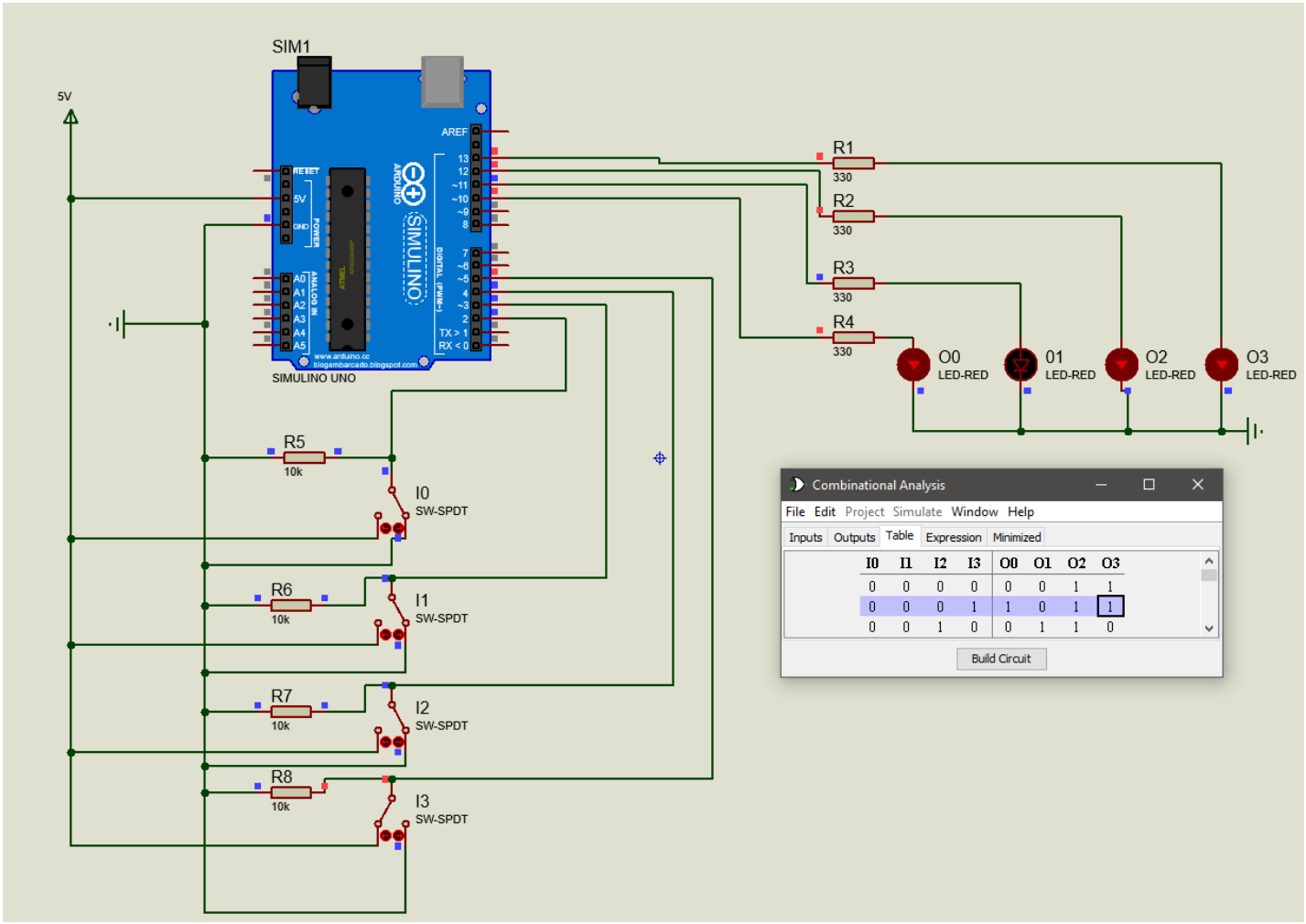
O3 $= \overline{I0\,I2} + \overline{I0}\,\overline{I1}\,I3 + \overline{I0}\,I1\,\overline{I3} + I1\,\overline{I2}\,I3 + I0\,I2$

Logisim Circuit:



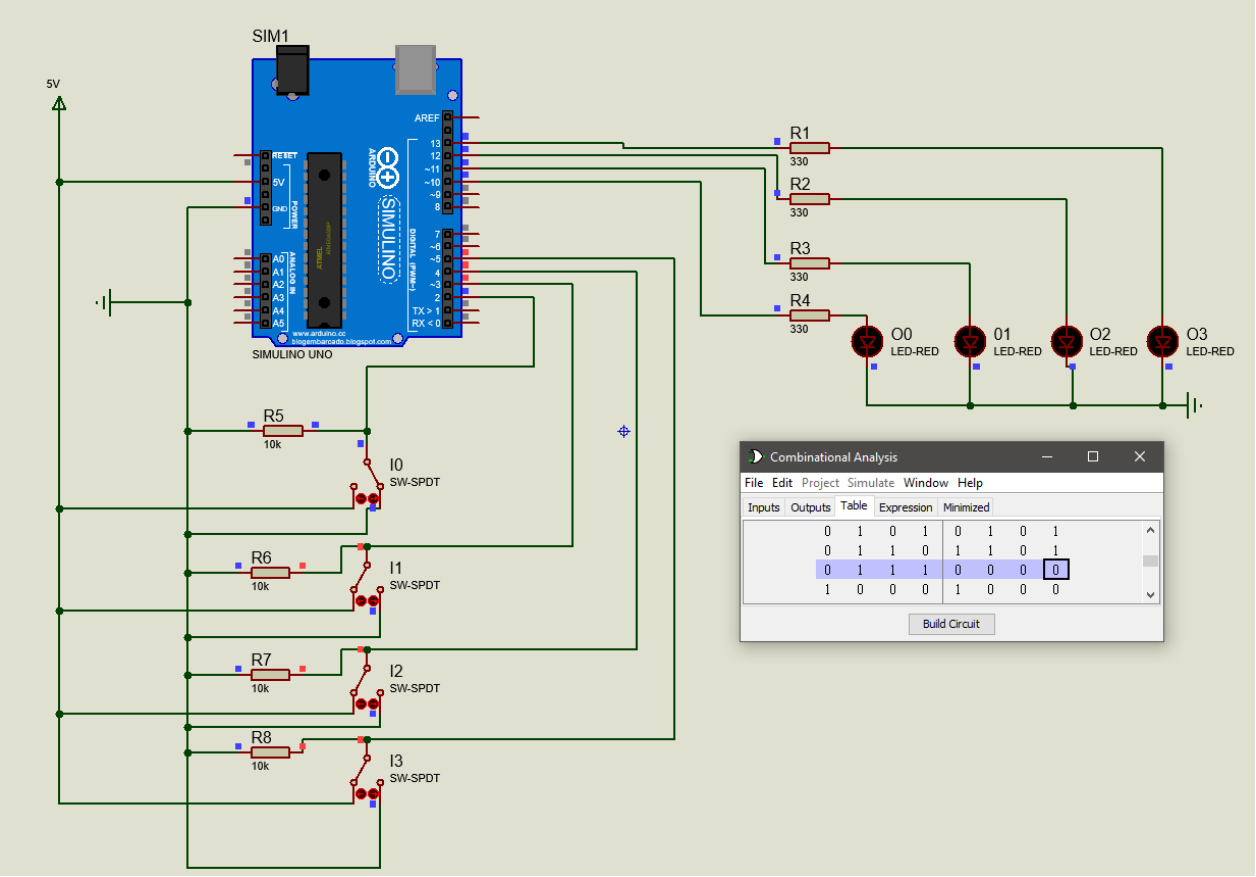
4. Circuit Diagram with Values of Electrical Components

4.1 Figure 1



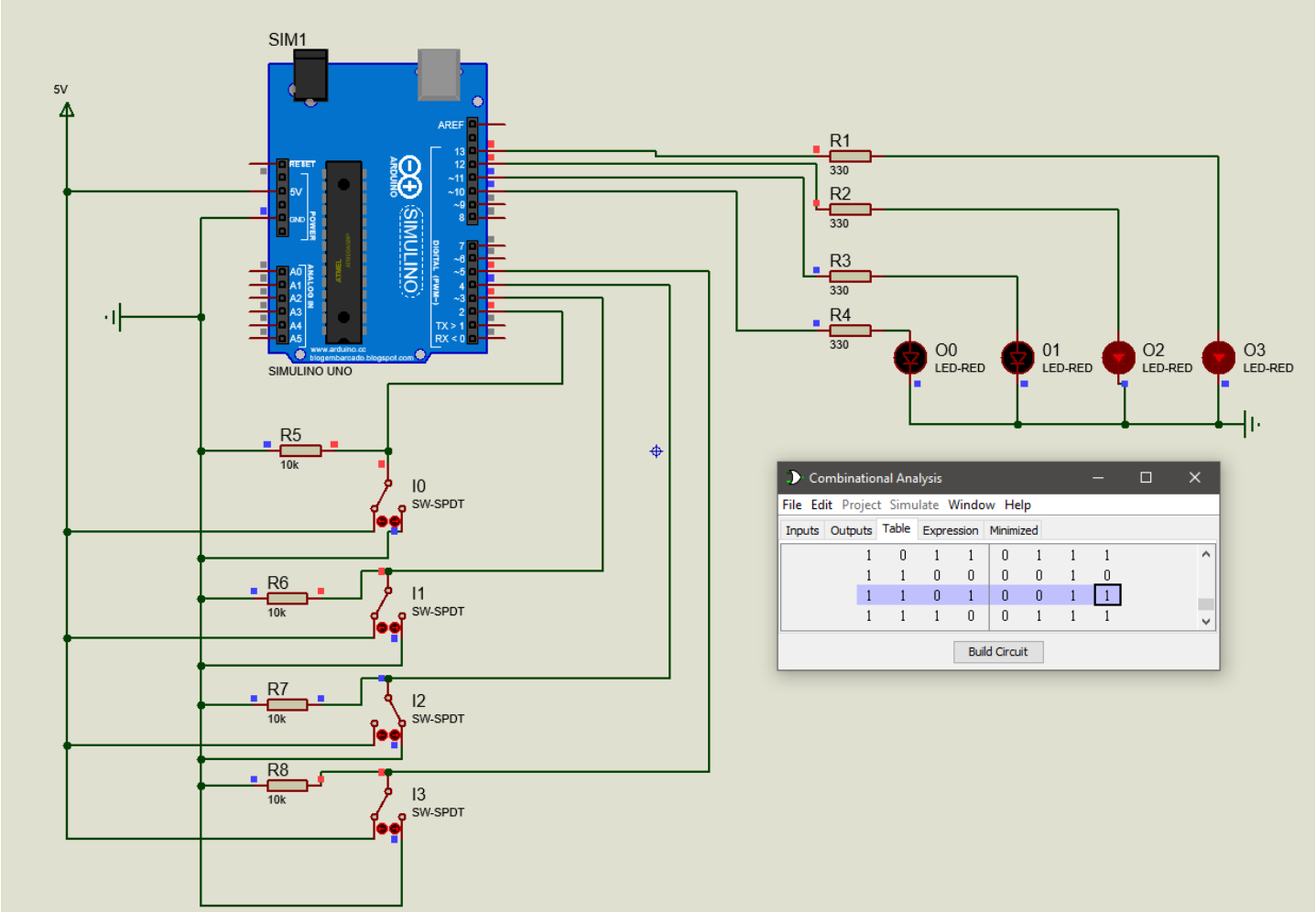
Here we can see the circuit in Proteus 8, as the inputs are I0= 0, I1=0, I2=0, I3=1 and the LED outputs are O0=1, O1=0, O2=1, O3=1.

4.2 Figure 2



In this picture the inputs are I0= 0, I1=1, I2=1, I3=1 and the LED outputs are O0=0, O1=0, O2=0, O3=0.

4.3 Figure 3



In this picture the inputs are I0= 1, I1=1, I2=0, I3=1 and the LED outputs are O0=0, O1=0, O2=1, O3=1.

Proteus Simulation Video:

https://drive.google.com/file/d/1SO25eR3JPteA4oRp9E_01ku8Rm5OkYFs/view?usp=sharing

5. Circuit Operation Principles

We get our output functions using the K Map and the equations are,

$$O0 = \overline{I1}\,I2\,\overline{I3} + I0\,\overline{I1}\,I3 + \overline{I0}\,I1\,I2$$

$$O1 = \overline{I2}\,I3 + I0\,\overline{I1}\,I2 + \overline{I0}\,I1\,\overline{I3}$$

$$O2 = \overline{I0}\,I1 + I1\,\overline{I2}\,\overline{I3} + \overline{I0}\,\overline{I1}$$

$$O3 = \overline{I0\,I2} + \overline{I0}\,\overline{I1}\,I3 + \overline{I0}\,I1\,\overline{I3} + I1\,\overline{I2}\,I3 + I0\,I2$$

The next step is to build the circuit using these logic gates and implementing them on Logisim.

Then we used Proteus 8 pro to build the hardware circuit diagram, the components used in Proteus are

Arduino UNO, this is the microcontroller we used in the simulation.

SPDT Switch, 4 single pole double throw switches to connect with the Arduino pins. The arrangements are, SPDT Switch 1(I0) to Arduino Digital Pin 2, SPDT Switch 2(I1) to Arduino Digital Pin 3, SPDT Switch 3(I2) to Arduino Digital Pin 4, SPDT Switch 4(I3) to Arduino Digital Pin 5.

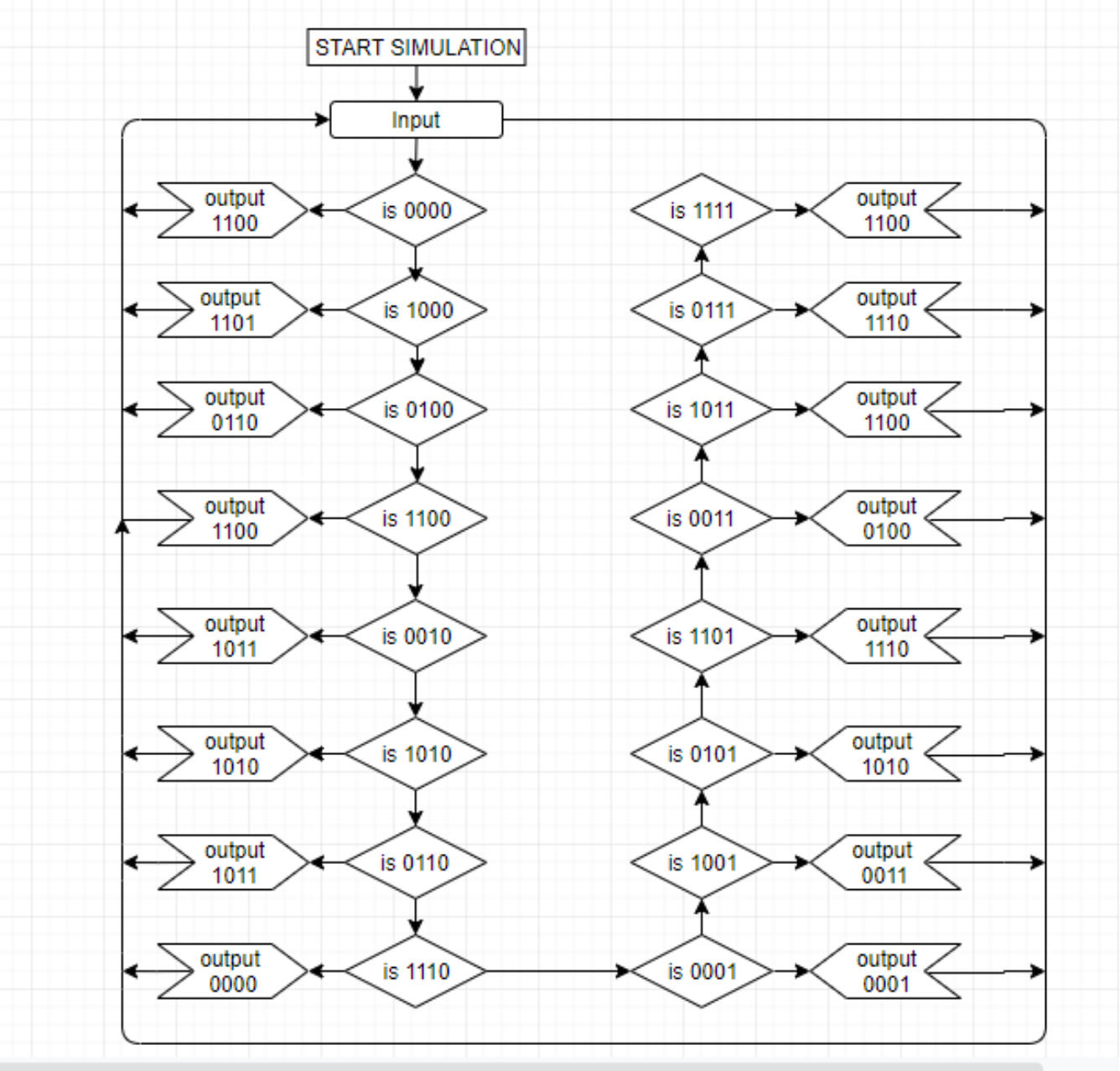
Resistors, we used 4 10K-ohm resistors to connect with the switches and 4 330-ohm resistors to connect with the LED outputs.

LEDs, we connected 4 LED outputs O0, O1, O2, O3 to the Arduino UNO Pin no 10, 11, 12, 13 respectively.

After building the circuit we upload the Arduino code to the UNO and simulated the project.

We uploaded a video on Drive to demonstrate the Proteus simulation of the circuit.

6. Program Flow Chart



7. ARDUINO Program

```
int O0 = 10;
int O1 = 11;
int O2 = 12;
int O3 = 13;

int I0 = 2;
int I1 = 3;
int I2 = 4;
int I3 = 5;


void setup()
{
  pinMode(I0,INPUT);
  pinMode(I1,INPUT);
  pinMode(I2,INPUT);
  pinMode(I3,INPUT);
  pinMode(O0,OUTPUT);
  pinMode(O1,OUTPUT);
  pinMode(O2,OUTPUT);
  pinMode(O3,OUTPUT);

}

void loop()
{

  boolean I0State = digitalRead(I0);
  boolean I1State = digitalRead(I1);
  boolean I2State = digitalRead(I2);
  boolean I3State = digitalRead(I3);
  boolean O0State;
  boolean O1State;
  boolean O2State;
  boolean O3State;

 O0State=
(!I1State&!I2State&I3State)|(!I0State&I1State&!I3State)|(I0State&!I1State&!I2State);
 O1State= (I2State&!I3State)|(!I0State&I1State&!I2State)|(I0State&!I1State&I3State);
 O2State=  (!I0State&!I1State)|(!I1State&I2State&I3State)|(I0State&I1State);
 O3State=
(!I0State&!I2State)|(!I0State&!I1State&I3State)|(!I0State&I1State&!I3State)|(I1State&!
I2State&I3State)|(I0State&I2State);

  digitalWrite(O0,O0State);
  digitalWrite(O1,O1State);
  digitalWrite(O2,O2State);
  digitalWrite(O3,O3State);

  if (digitalRead(I0) == 0 && digitalRead(I1) == 0 && digitalRead(I2) == 0 &&
digitalRead(I3) == 0)
  {digitalWrite(O0,LOW) ;
  digitalWrite(O1,LOW) ;
  digitalWrite(O2,HIGH) ;
  digitalWrite(O3,HIGH);

  }
```

```
  if (digitalRead(I0) == 0 && digitalRead(I1) == 0 && digitalRead(I2) == 0 &&
digitalRead(I3) == 1)
  {digitalWrite(O0,HIGH) ;
  digitalWrite(O1,LOW) ;
  digitalWrite(O2,HIGH) ;
  digitalWrite(O3,HIGH);

  } if (digitalRead(I0) == 0 && digitalRead(I1) == 0 && digitalRead(I2) == 1 &&
digitalRead(I3) == 0)
  {digitalWrite(O0,LOW) ;
  digitalWrite(O1,HIGH) ;
  digitalWrite(O2,HIGH) ;
  digitalWrite(O3,LOW);

  } if (digitalRead(I0) == 0 && digitalRead(I1) == 0 && digitalRead(I2) == 1 &&
digitalRead(I3) == 1)
  {digitalWrite(O0,LOW) ;
  digitalWrite(O1,LOW) ;
  digitalWrite(O2,HIGH) ;
  digitalWrite(O3,HIGH);

  } if (digitalRead(I0) == 0 && digitalRead(I1) == 1 && digitalRead(I2) == 0 &&
digitalRead(I3) == 0)
  {digitalWrite(O0,HIGH) ;
  digitalWrite(O1,HIGH) ;
  digitalWrite(O2,LOW) ;
  digitalWrite(O3,HIGH);

  } if (digitalRead(I0) == 0 && digitalRead(I1) == 1 && digitalRead(I2) == 0 &&
digitalRead(I3) == 1)
  {digitalWrite(O0,LOW) ;
  digitalWrite(O1,HIGH) ;
  digitalWrite(O2,LOW) ;
  digitalWrite(O3,HIGH);

  } if (digitalRead(I0) == 0 && digitalRead(I1) == 1 && digitalRead(I2) == 1 &&
digitalRead(I3) == 0)
  {digitalWrite(O0,HIGH) ;
  digitalWrite(01,HIGH) ;
  digitalWrite(O2,LOW) ;
  digitalWrite(O3,HIGH);

  } if (digitalRead(I0) == 0 && digitalRead(I1) == 1 && digitalRead(I2) == 1 &&
digitalRead(I3) == 1)
  {digitalWrite(O0,LOW) ;
  digitalWrite(O1,LOW) ;
  digitalWrite(O2,LOW) ;
  digitalWrite(O3,LOW);

  } if (digitalRead(I0) == 1 && digitalRead(I1) == 0 && digitalRead(I2) == 0 &&
digitalRead(I3) == 0)
  {digitalWrite(O0,HIGH) ;
  digitalWrite(O1,LOW) ;
  digitalWrite(O2,LOW) ;
  digitalWrite(O3,LOW);

  } if (digitalRead(I0) == 1 && digitalRead(I1) == 0 && digitalRead(I2) == 0 &&
digitalRead(I3) == 1)
  {digitalWrite(O0,HIGH) ;
  digitalWrite(O1,HIGH) ;
  digitalWrite(O2,LOW) ;
  digitalWrite(O3,LOW);
```

```
  } if (digitalRead(I0) == 1 && digitalRead(I1) == 0 && digitalRead(I2) == 1 &&
digitalRead(I3) == 0)
  {digitalWrite(O0,LOW) ;
  digitalWrite(O1,HIGH) ;
  digitalWrite(O2,LOW) ;
  digitalWrite(O3,HIGH);

  } if (digitalRead(I0) == 1 && digitalRead(I1) == 0 && digitalRead(I2) == 1 &&
digitalRead(I3) == 1)
  {digitalWrite(O0,LOW) ;
  digitalWrite(O1,HIGH) ;
  digitalWrite(O2,HIGH) ;
  digitalWrite(O3,HIGH);

  } if (digitalRead(I0) == 1 && digitalRead(I1) == 1 && digitalRead(I2) == 0 &&
digitalRead(I3) == 0)
  {digitalWrite(O0,LOW) ;
  digitalWrite(O1,LOW) ;
  digitalWrite(O2,HIGH) ;
  digitalWrite(O3,LOW);

  } if (digitalRead(I0) == 1 && digitalRead(I1) == 1 && digitalRead(I2) == 0 &&
digitalRead(I3) == 1)
  {digitalWrite(O0,LOW) ;
  digitalWrite(O1,LOW) ;
  digitalWrite(O2,HIGH) ;
  digitalWrite(O3,HIGH);

  } if (digitalRead(I0) == 1 && digitalRead(I1) == 1 && digitalRead(I2) == 1 &&
digitalRead(I3) == 0)
  {digitalWrite(O0,LOW) ;
  digitalWrite(O1,HIGH) ;
  digitalWrite(O2,HIGH) ;
  digitalWrite(O3,HIGH);

  } if (digitalRead(I0) == 1 && digitalRead(I1) == 1 && digitalRead(I2) == 1 &&
digitalRead(I3) == 1)
  {digitalWrite(O0,LOW) ;
  digitalWrite(O1,LOW) ;
  digitalWrite(O2,HIGH) ;
  digitalWrite(O3,HIGH);

  }

  }
```

8. Question and Answers:

1. What is the clock frequency of the microcontroller used?
   The clock frequency is 16 MHz

2. What is the data bus width of the microcontroller used?
   8 bits

3. What is the size of your hex file generated? Attach the hex codes in your report
   The generated HEX file is 7KB.
   Hex file link from ARDUINO IDE :
   C:\\Users\\SHADAL~1\\AppData\\Local\\Temp\\arduino_build_72562/CSE331ProjectGroup7.hex

Drive Link:

https://drive.google.com/drive/folders/1jabA3xVIq3aWd-KlmAaaZHnVQJ9luVA0?usp=sharing

4. Can the project be implemented by using interrupt?
   Yes

   Interrupts in Arduino

   Interrupts are a mechanism by which an I/O or instruction can suspend the normal execution of the processor and gets itself serviced like it has higher priority. It has two types of interrupts.

   External Interrupt:

   These interrupts are interpreted by hardware and are very fast. These interrupts can be set to trigger on the event of RISING or FALLING or LOW levels.

Pin Change Interrupts:

Arduinos can have more interrupt pins enabled by using pin change interrupts. In ATmega168/328 based Arduino boards any pins or all the 20 signal pins can be used as interrupt pins. They can also be triggered using RISING or FALLING edges.

5. Is the main routine required to be an infinite loop? Provide an explanation in favor of your answer.

   Yes, the main routine is required to be an infinite loop because If we want the program to execute once only, we should end it with a while loop.

   Example:

```
main() {
..... while(1) { .....
        }
}
```

   Otherwise it automatically restarts

   To elaborate, main is a function called by the startup code, when main terminates it returns.

   The startup code then executes a reset.

   The startup code is added by the linker.

6. Is there any difference between level triggered and edge triggered operation for the given project?

   Edge Triggering is a type of triggering that allows a circuit to become active at the positive edge or the negative edge of the clock signal.

And Level Triggering is a type of triggering that allows a circuit to become active when the clock pulse is on a particular level.

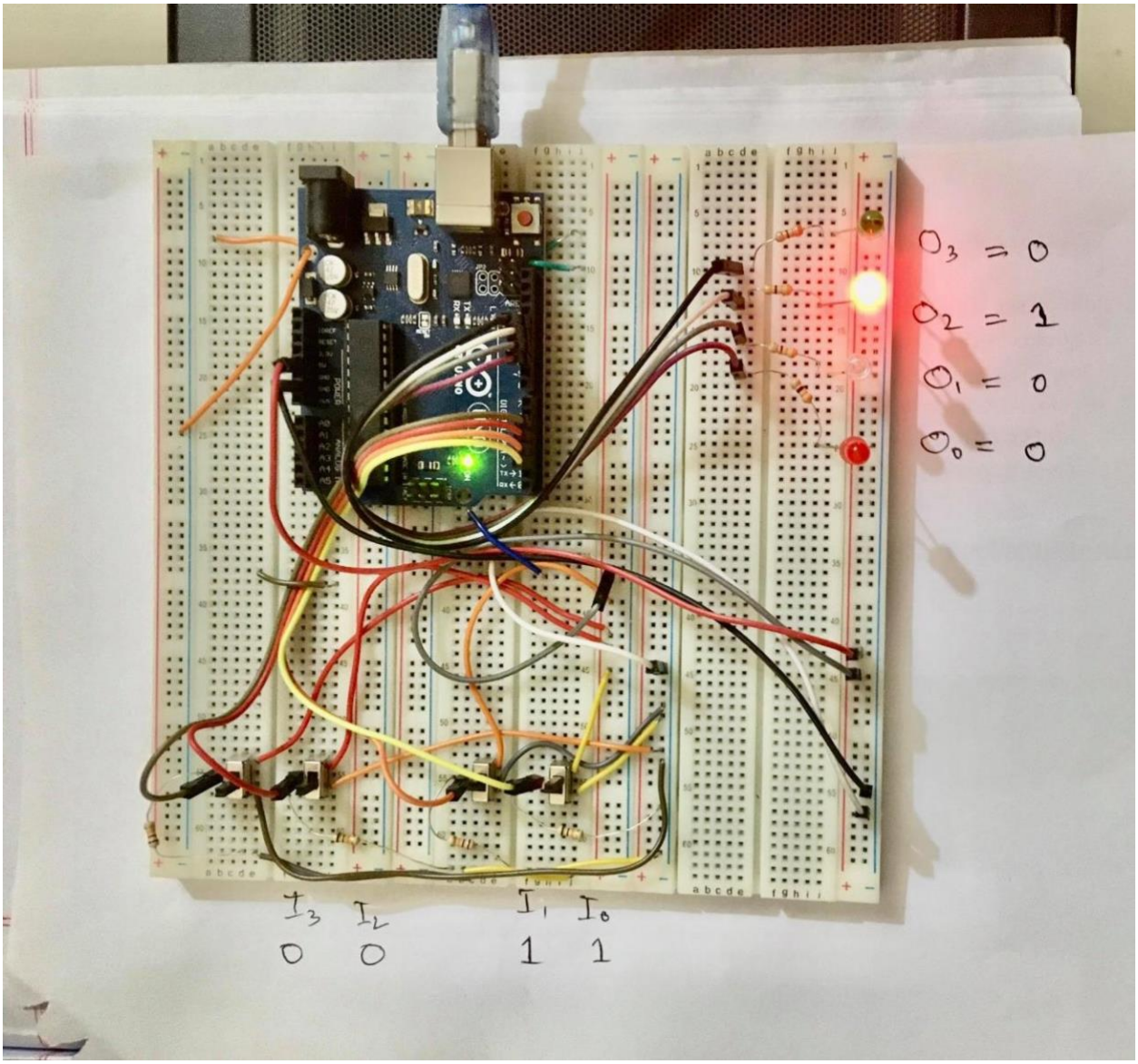
In the given project with the Microcontroller Arduino Uno , the Edge Triggering and the Level Triggering is the same.

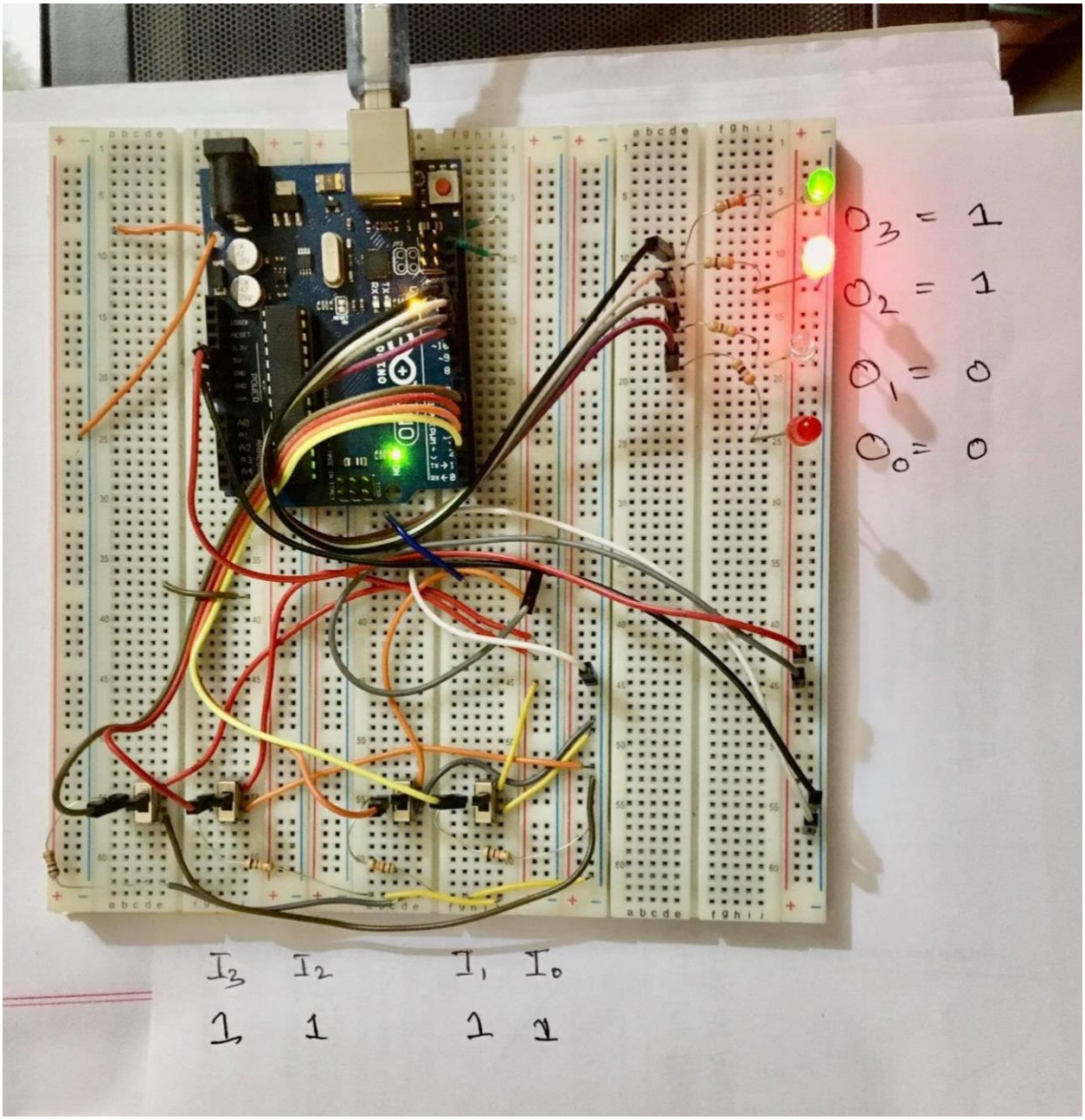
7. Is the project referring to encryption or decryption from input to output?

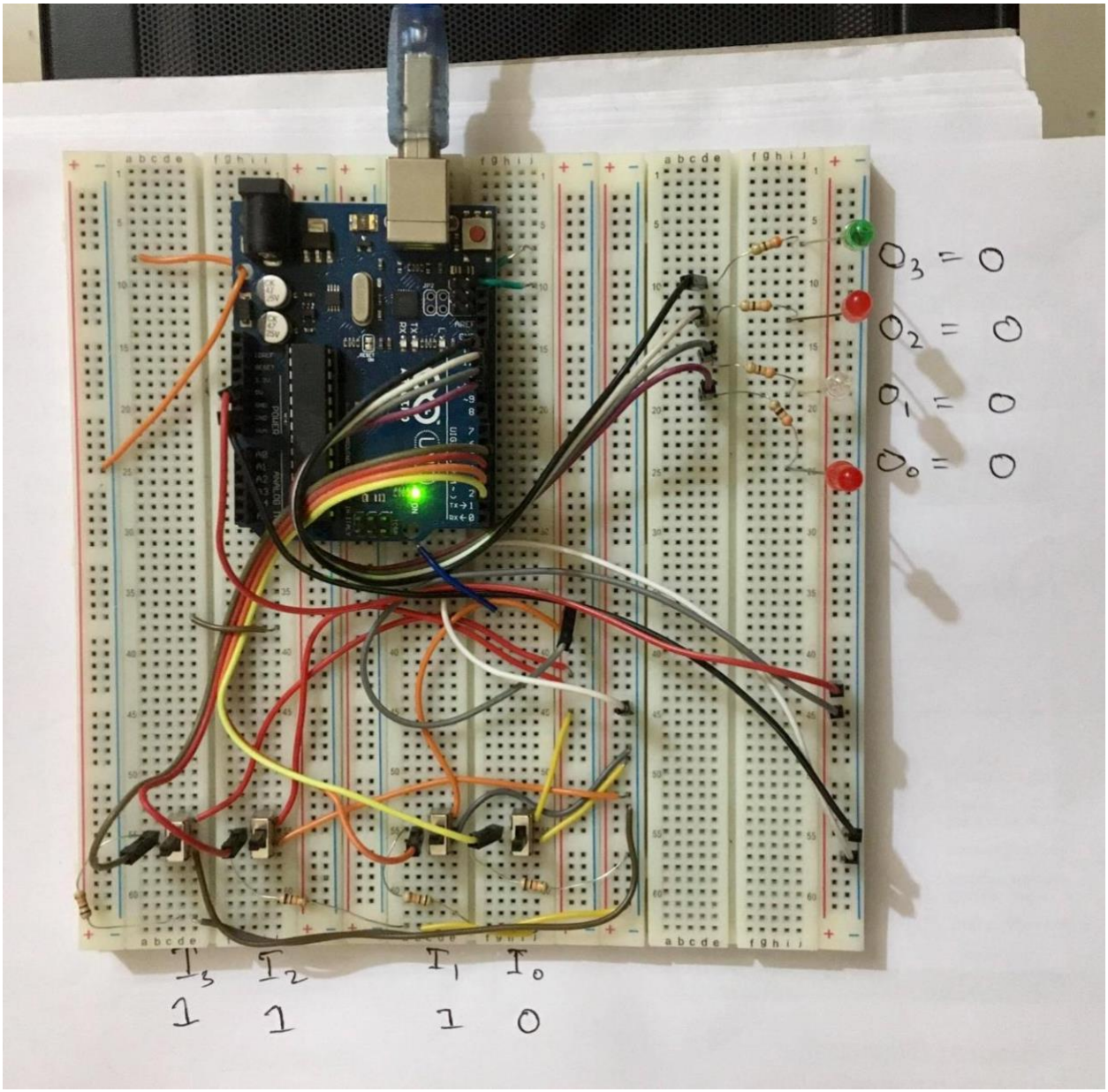
   Decryption from input to output.

9. Hardware Implementation

We implemented the project using Arduino UNO and SPDT switches and the images and video link is given below.




$O_3 = 1$
$O_2 = 1$
$O_1 = 0$
$O_0 = 0$
I_3 I_2 I_1 I_0
1 1 1 1



The link of video demonstration is given below:

Hardware Implementation:

https://drive.google.com/file/d/1U-mY2dl0bFKjExnzwekLVlByH5U4Fqum/view?usp=sharing

References

1. https://forum.arduino.cc/index.php?topic=375806.0
2. https://www.researchgate.net/publication/322159080_Digital_Logic_Gate_Simulation_usin g_Arduino_Microcontroller
3. https://proteus.informer.com/8.0/