Final Report

Group: 1

Semester: Summer 2022

Course Code: CSE331

Course Name: Microprocessor Interfacing and Embedded System

Section: 02

Submitted to: Dr.Dihan Md Nuruddin Hasan (dmh)

Submitted by

Name	ID
Ishrat Jahan	1921909042
Alabhia Talukder	1911294642
Sayema Islam Tithi	1831007642
Sadia Binta Sarowar	1811908642

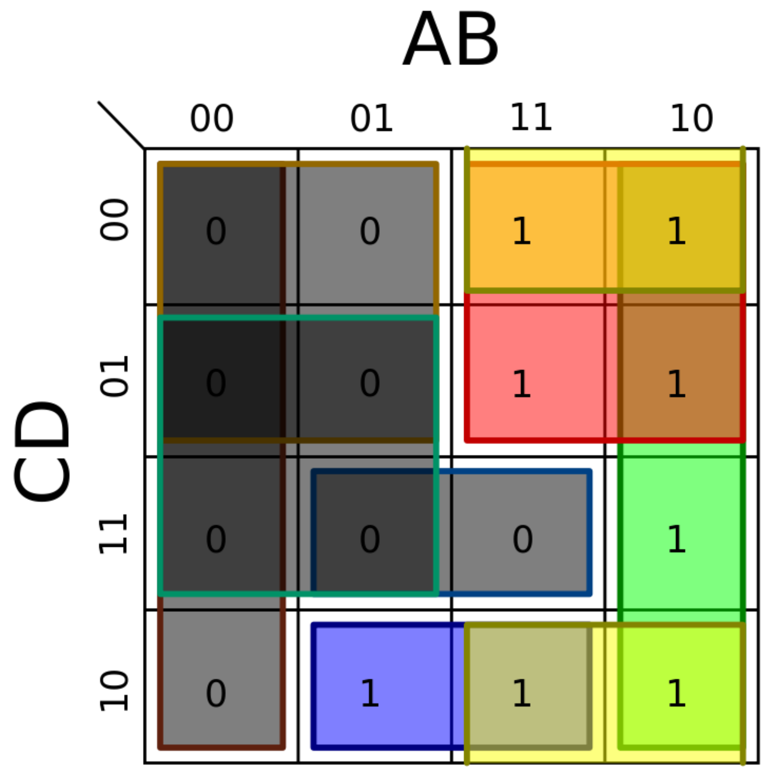
Contents

Question

Implement the given encryption table using 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			
I 3	I 2	I 1	I 0	O 3	O 2	O 2	O 1
0	0	0	0	1	1	0	1
1	0	0	0	1	1	1	0
0	1	0	0	0	1	1	0
1	1	0	0	1	0	1	0
0	0	1	0	1	0	1	0
1	0	1	0	0	0	0	1
0	1	1	0	0	1	1	0
1	1	1	0	0	0	1	1
0	0	0	1	1	1	0	0
1	0	0	1	0	0	1	1
0	1	0	1	1	1	0	0
1	1	0	1	0	0	0	1
0	0	1	1	1	0	1	1
1	0	1	1	0	0	1	1
0	1	1	1	1	1	1	0
1	1	1	1	1	0	0	0

Hints for deriving the logic expression:

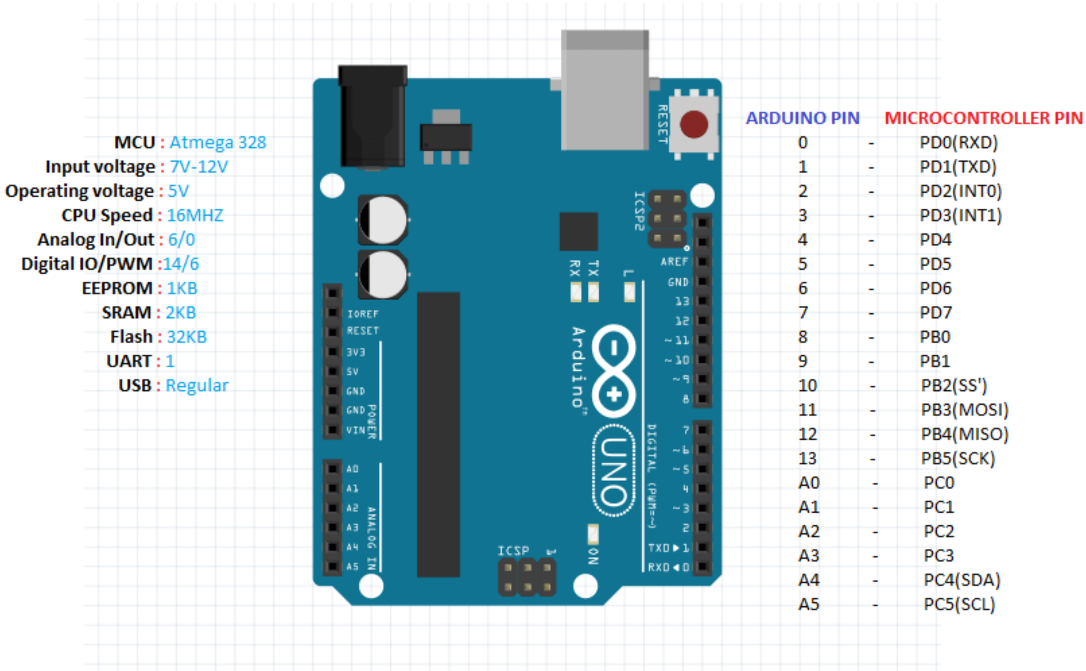


f(A,B,C,D) = E(6,8,9,10,11,12,13,14)

F=AC'+AB'+BCD'+AD'

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

Example microcontroller: (Arduino UNO)



Main Objective

The main objective of the project is to Implement the given encryption table using microcontroller. We will be using single pole, double throw switch to configure the inputs for high and low conditions and Use LEDs to represent the corresponding output statuses.Following this we will code the encryption table in three different arduino boards and compare the power consumption between the three boards.

Methodology

Step 1: Derivation of equations from truth table using k-maps.

Step 2: Draw the logic circuit in Logisim.

Step 3: Code the expression in Arduino IDE.

Step 4: Generate the hex file.

Step 5: Draw the circuit in Proteus.

Step 6: Repeat for three different board.

Step 7: Burn the hex file and see the operation.

Step 7: Use digital multimeter to measure current.

Step 8: Use P=VCC*Current to calculate the power consumption by each board and compare.

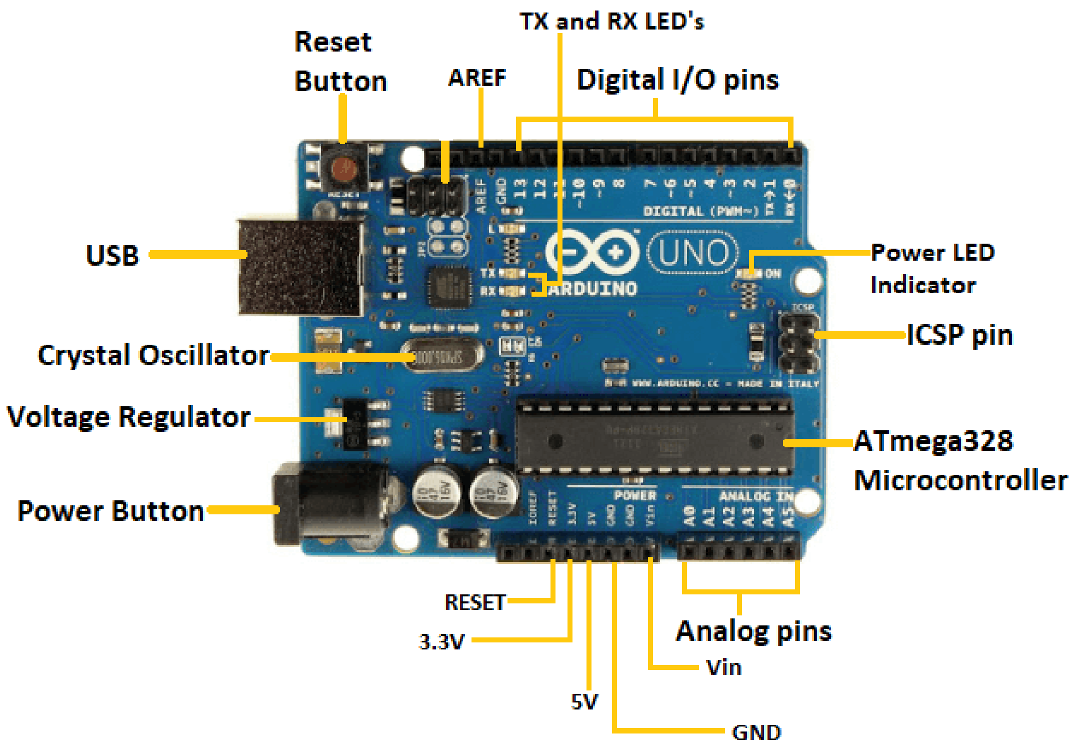
1. General Description

1.1 Arduino UNO

The Arduino UNO is a standard board of Arduino. Here UNO means 'one' in Italian. It was named as UNO to label the first release of Arduino Software. It was also the first USB board released by Arduino. It is considered as the powerful board used in various projects. Arduino UNO is based on an ATmega328P microcontroller. It is easy to use compared to other boards, such as the Arduino Mega board, etc. The board consists of digital and analog Input/Output pins (I/O), shields, and other circuits.

The Arduino UNO includes 6 analog pin inputs, 14 digital pins, a USB connector, a power jack, and an ICSP (In-Circuit Serial Programming) header. It is programmed based on IDE, which stands for Integrated Development Environment. It can run on both online and offline platforms.

The IDE is common to all available boards of Arduino.



1.2 Arduino IDE

It is an open source Arduino Software(IDE) where it's easier to write code and upload it to the board. It can be used with any Arduino board.

1.3 Proteus 8 Pro

The Proteus Design Suite is a proprietary software tool suite used primarily for electronic design automation. The software is used mainly used to create schematics and electronic prints for manufacturing printed circuit boards.

1.4 Logisim

Logisim is a tool for designing and simulating digital logic circuits. With its simple toolbar interface and simulation of circuits as you build them, it is simple enough to facilitate learning the most basic concepts related to logic circuits. It has the capacity to build larger circuits from smaller sub circuits, and to draw bundles of wires with a single mouse drag, Logisim can be used (and is used) to design and simulate entire CPUs for educational purposes.

2. Equipment

Hardware

- Arduino UNO
- Breadboard
- Resistors
- Wires
- Switch
- Red LED

Software

- Proteus 8 Professional Software
- Arduino IDE
- Logisim

3. Method of Derivation

3.1 Truth Table

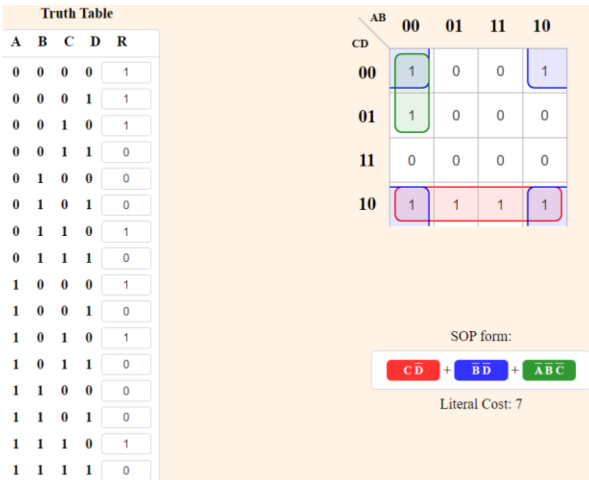
I3	**I2**	**I1**	**I0**	**O3**	**O2**	**O1**	**O0**
0	0	0	0	1	1	0	1
1	0	0	0	1	1	1	0
0	1	0	0	0	1	1	0
1	1	0	0	1	0	1	0
0	0	1	0	1	0	1	0
1	0	1	0	0	0	0	1
0	1	1	0	0	1	1	0
1	1	1	0	0	0	1	1
0	0	0	1	1	1	0	0
1	0	0	1	0	0	1	1
0	1	0	1	1	1	0	0
1	1	0	1	0	0	0	1
0	0	1	1	1	0	1	1
1	0	1	1	0	0	1	1
0	1	1	1	1	1	1	0
1	1	1	1	1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

I3,I2.I1,I0=Input

03,02,01,00=Output

3.2 Derived Result

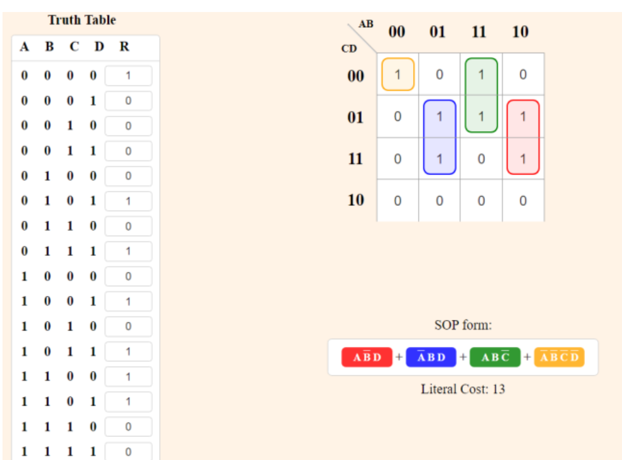
Kmaps



Truth Table

A	B	C	D	R
0	0	0	0	1
0	0	0	1	1
0	0	1	0	1
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	1
0	1	1	1	0
1	0	0	0	1
1	0	0	1	0
1	0	1	0	1
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	1
1	1	1	1	0

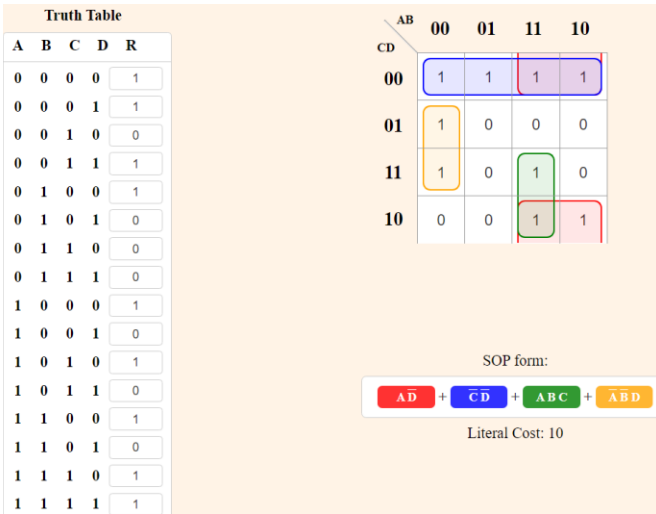
00=I0I1’I3+I0’I1I3+I0I1I2’+I0’I1’I2’I3’



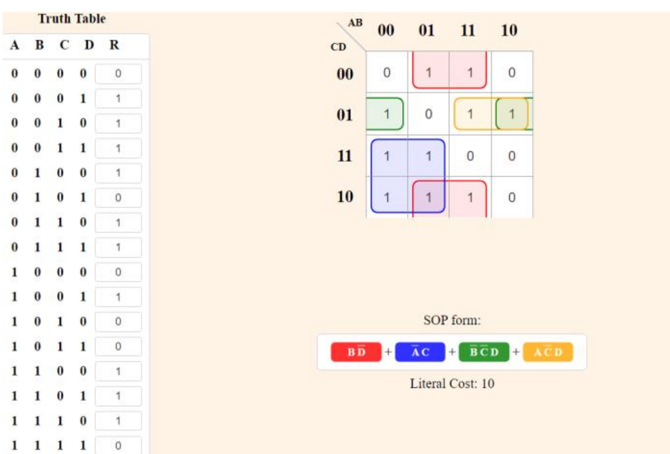
Truth Table

A	B	C	D	R
0	0	0	0	1
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	1
0	1	1	0	0
0	1	1	1	1
1	0	0	0	0
1	0	0	1	1
1	0	1	0	0
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	0
1	1	1	1	0

01=I1I3’+I0’I2+I1’I2’I3+I0I2’I3



02=I2I3’+I1’I3’+I0’I1’I2’



O3=I2’I3’+I0I3’+I0’I1’I3+I0I1I2

Kmap Expressions

00=I0I1’I3+I0’I1I3+I0I1I2’+I0’I1’I2’I3’

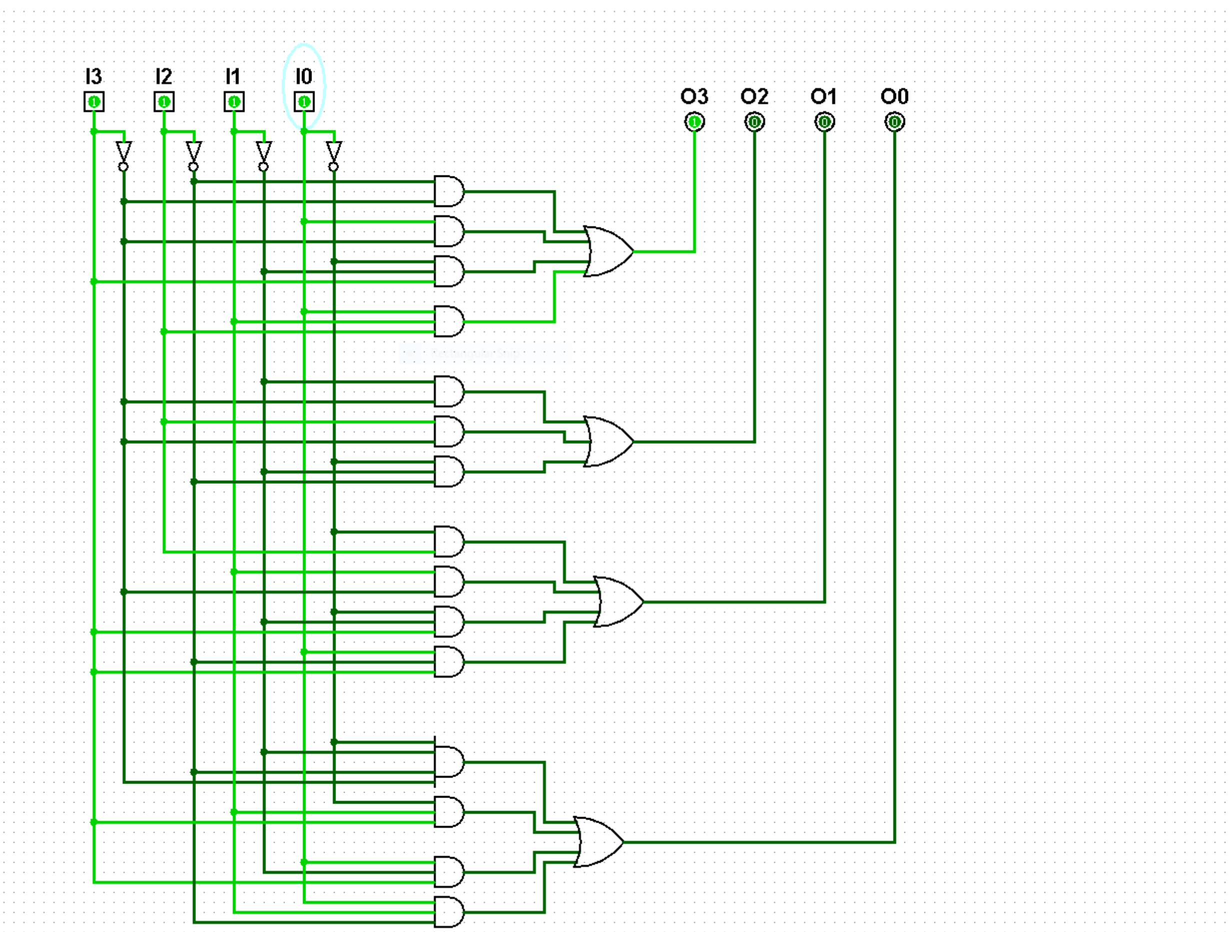
01=I1I3’+I0’I2+I1’I2’I3+I0I2’I3

02=I2I3’+I1’I3’+I0’I1’I2’

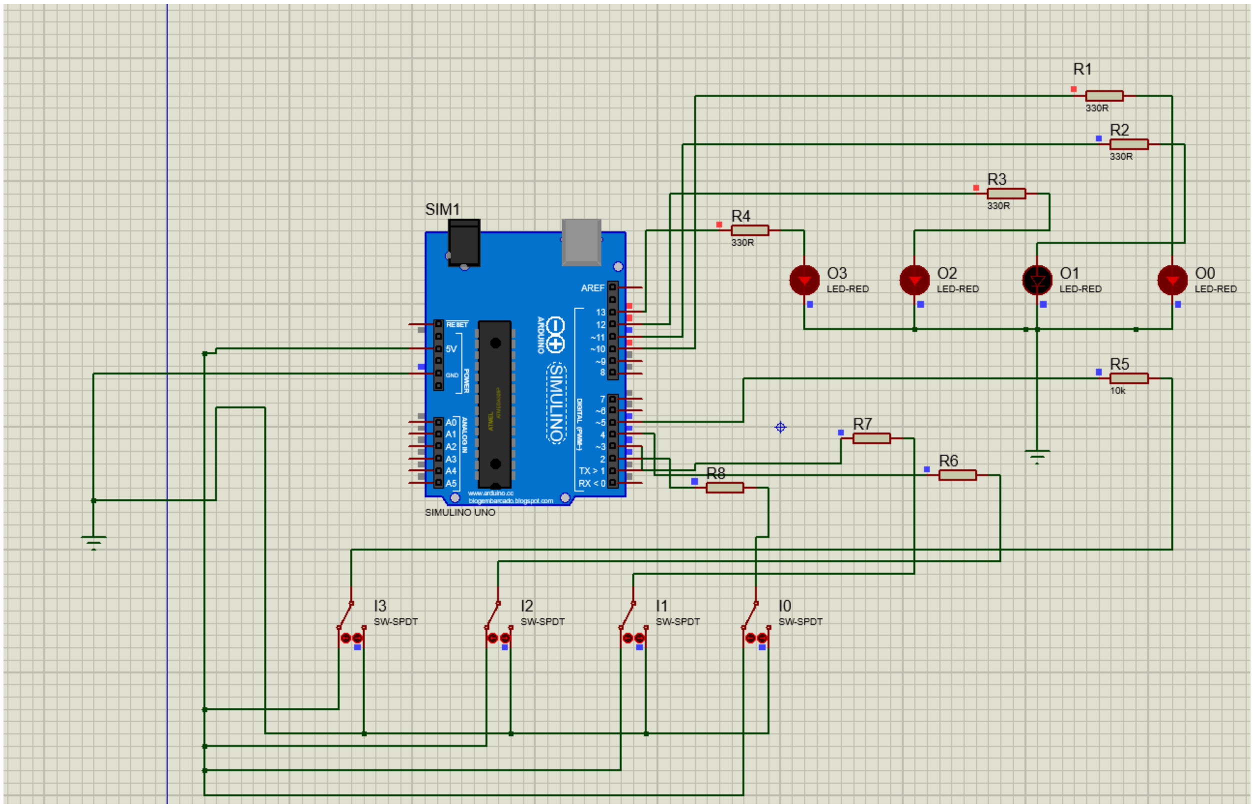
O3=I2’I3’+I0I3’+I0’I1’I3+I0I1I2

4. Circuit Diagram with Values of Electrical Components

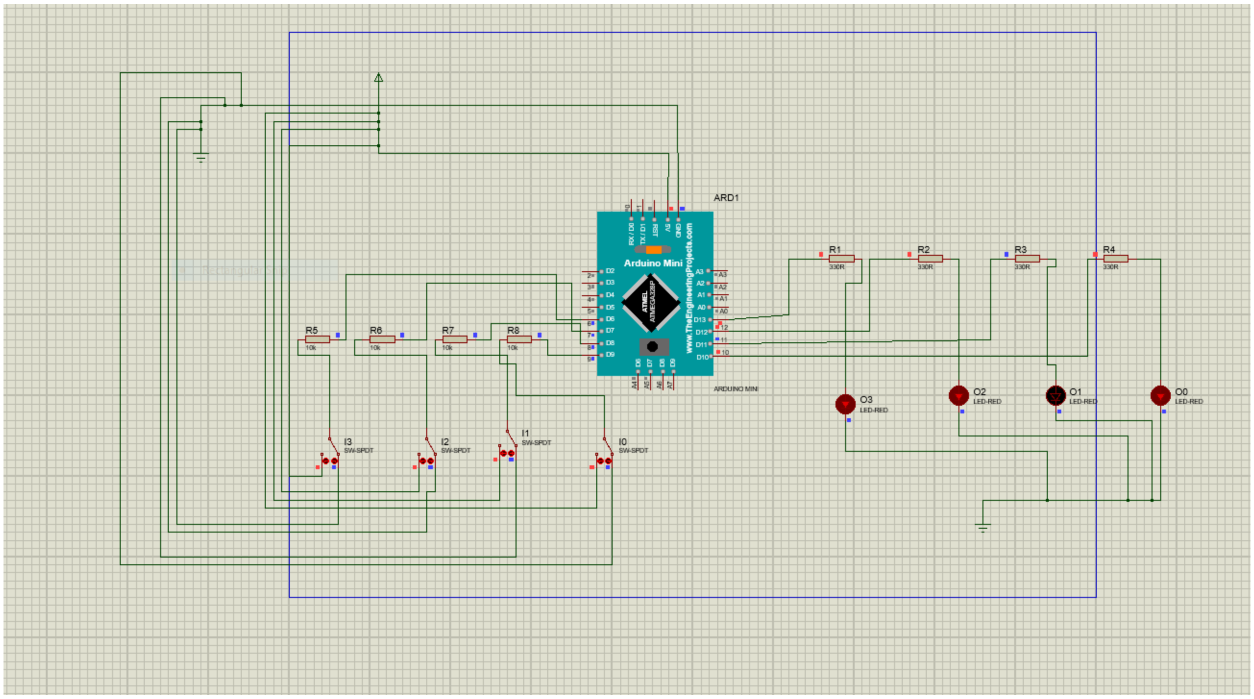
4.1 Logisim Circuit



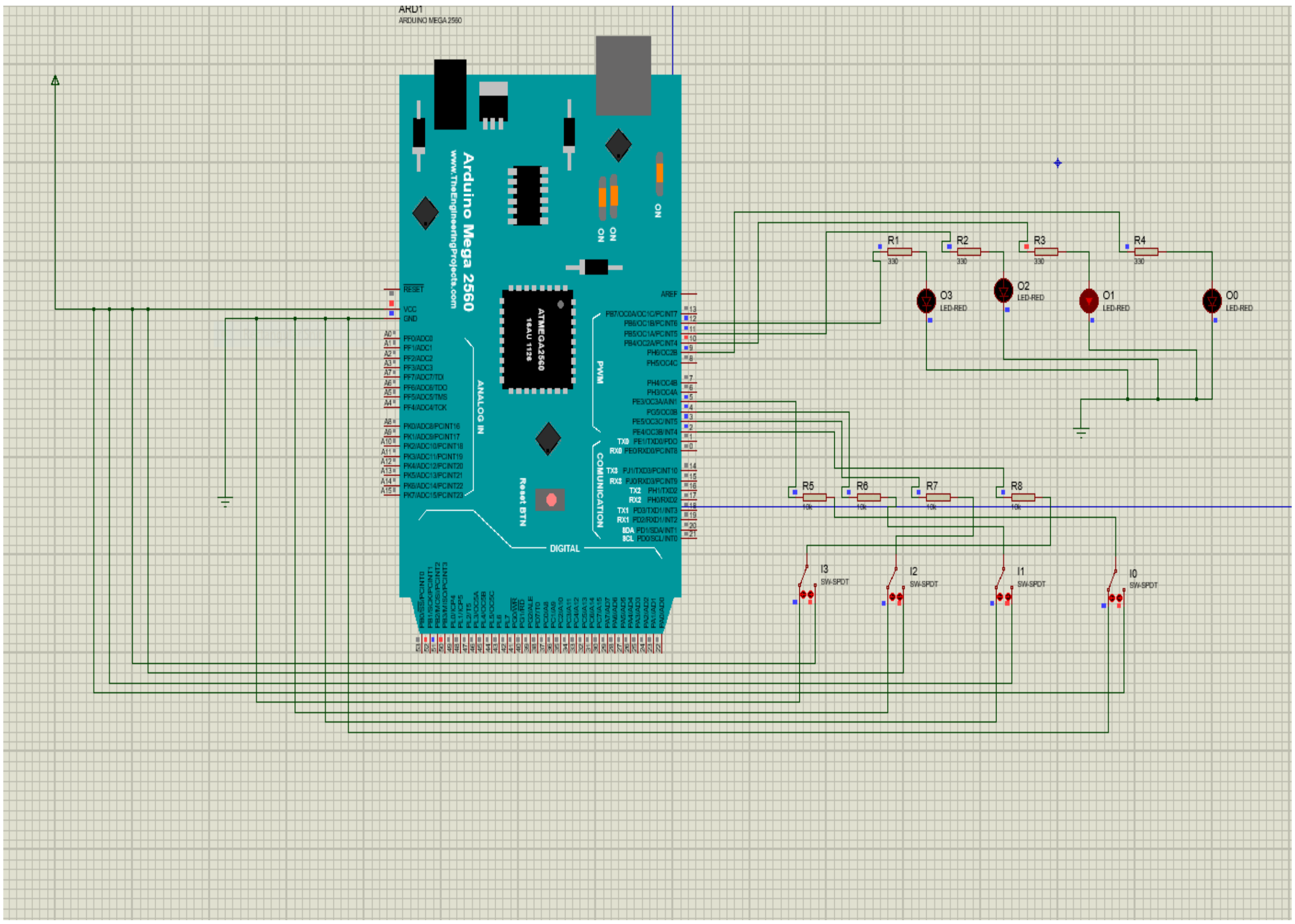
SIMULINO UNO



Arduino Pro mini



Arduino Mega 2560



Circuit Operation Principles

Firstly, the encryption table provided was solved using Karnaugh Map and we got 4 different equations for 4 outputs O0, O1 , O2, O3

The equations are:
O0 = **I3’ I2’ I1’ I0’ + I2’ I1 I0 + I3 I1’ I0 + I3 I1 I0’**
O1 = **I3’ I1 + I2 I0’ + I3 I2’ I1’ + I3 I2’ I0**
O2 = **I3’ I1’ + I2’ I1’ I0’ + I3’ I2**
O3 = **I3’ I2’ + I3’ I0 + I2 I1 I0 + I3 I1’ I0’**

Secondly, these equations were used to design the Circuit Diagram in Logisim.

Thirdly, using Proteus 8 Professional Software we built our main Hardware Circuit using three different Arduino boards.

We have used the following components:
1. **Arduino Uno, Arduino Pro mini, Arduino Mega :** Used three different Arduino boards as the microcontroller
We connected the power supply of 5V and the ground pin of the Arduino boards to a Ground component.
2. **Resistors:** We have used 8 resistors in total. We have used 330 ohm resistors for connection with the LEDs and 10k ohm resistors with the SP-DWT Switches.

3. **Single Pole Double Throw Switches:** We used four of these Switches with our Arduino Digital pins.
They are arranged in such a way for Arduino Uno
Arduino UNO Digital Pin No. 2 --------- SP-DWT Switch 1 (I0)
Arduino UNO Digital Pin No. 3 --------- SP-DWT Switch 2 (I1)
Arduino UNO Digital Pin No. 4 --------- SP-DWT Switch 3 (I2)
Arduino UNO Digital Pin No. 5 --------- SP-DWT Switch 4 (I3)
They are arranged in such a way for Arduino Pro mini
Arduino UNO Digital Pin No. 2 --------- SP-DWT Switch 1 (I0)
Arduino UNO Digital Pin No. 3 --------- SP-DWT Switch 2 (I1)
Arduino UNO Digital Pin No. 4 --------- SP-DWT Switch 3 (I2)
Arduino UNO Digital Pin No. 5 --------- SP-DWT Switch 4 (I3)
They are arranged in such a way for Arduino Uno
Arduino UNO Digital Pin No. 2 --------- SP-DWT Switch 1 (I0)
Arduino UNO Digital Pin No. 3 --------- SP-DWT Switch 2 (I1)
Arduino UNO Digital Pin No. 4 --------- SP-DWT Switch 3 (I2)
Arduino UNO Digital Pin No. 5 --------- SP-DWT Switch 4 (I3)

For the other two boards the arrangement is same according to the Digital Pins they are connected with.

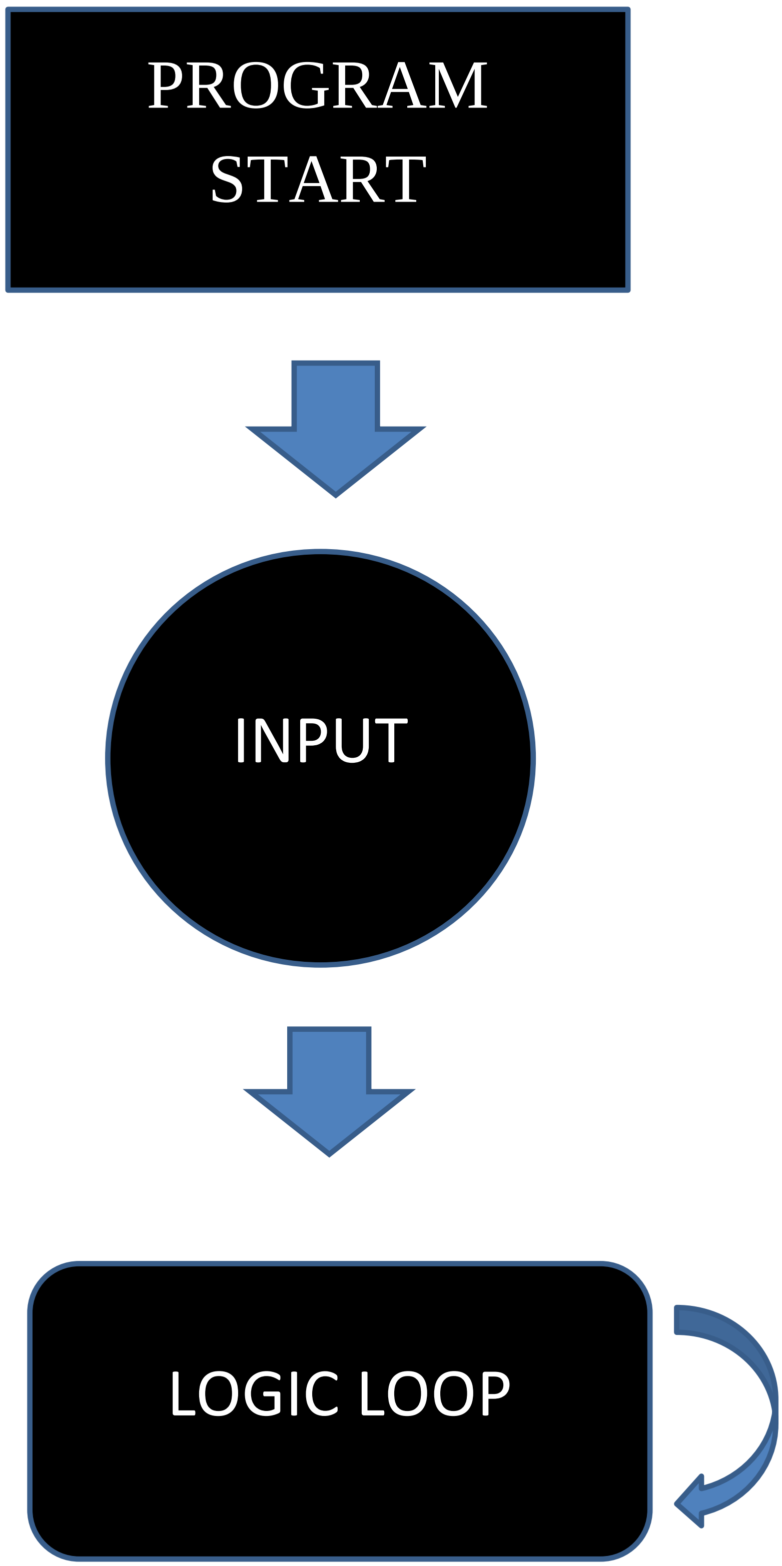
4. **LEDs:** We have connected Four LEDs to the Digital Pins of Arduino boards for output. They are arranged in the following order:
Arduino UNO Digital Pin No. 10 --------- LED1 (O0)
Arduino UNO Digital Pin No. 11 --------- LED2 (O1)
Arduino UNO Digital Pin No. 12 --------- LED3 (O2)
Arduino UNO Digital Pin No. 13 --------- LED4 (O3)

For the other two boards the arrangement of LED's is same according to the Digital Pins they are connected with.

After making the whole circuit we loaded the arduino code for each of the board and simulated the circuit.

Arduino Code:
We used Arduino IDE to write the Arduino code by using the K-maps equations we got and then generate the hex file and load it into the circuit in Proteus.

Program Flow chart



Arduino Program

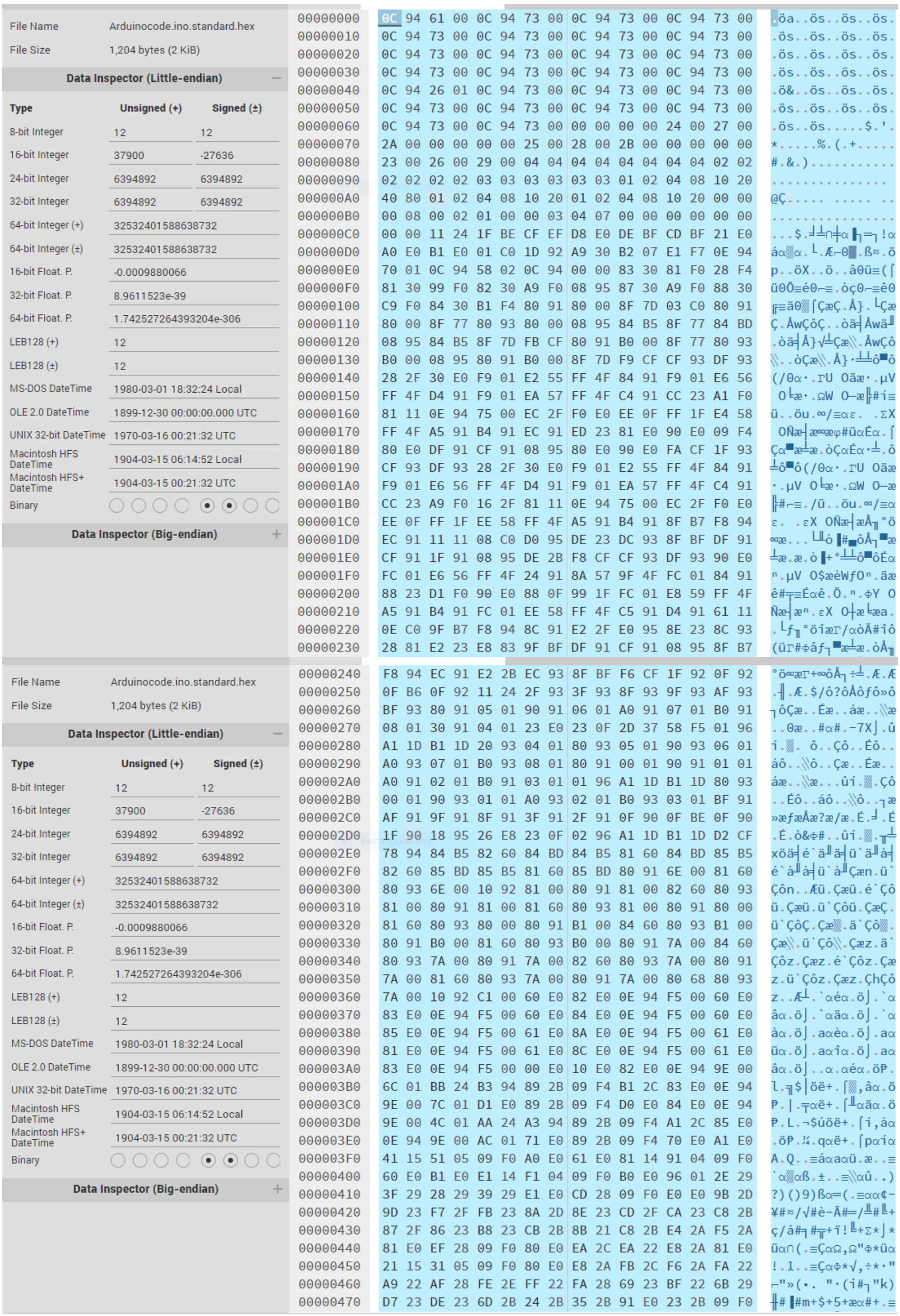
Arduino Uno

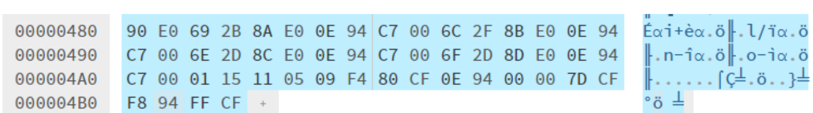
```
int I0 = 2;  //Input pins connected to digital pin
int I1 = 3;
int I2 = 4;
int I3 = 5;

int O0=10; //Output pins connected to digital pin
int O1=11;
int O2=12;
int O3=13;

void setup() {

  pinMode(I0,INPUT);  //Sets I0,I1,I2,I3 as INPUT
  pinMode(I1,INPUT);
  pinMode(I2,INPUT);
  pinMode(I3,INPUT);

  pinMode(O0,OUTPUT);  //Sets output pins as OUTPUT
  pinMode(O1,OUTPUT);
  pinMode(O2,OUTPUT);
```

```
  pinMode(O3,OUTPUT);

}

void loop() {
  boolean I0State = digitalRead(I0);  //Reads input pins
  boolean I1State = digitalRead(I1);
  boolean I2State = digitalRead(I2);
  boolean I3State = digitalRead(I3);

  boolean O0State;
  boolean O1State;
  boolean O2State;
  boolean O3State;

 O0State= (!I3State & !I2State & !I1State & !I0State)|(!I2State & I1State & I0State)|(I3State & !I1State & I0State)|(I3State & I1State& !I0State);
//Setting the kmap equations for each output pins
 O1State= (!I3State & I1State)|(I2State & !I0State)|(I3State & !I2State & !I1State)|(I3State & !I2State & I0State);
 O2State= (!I3State & !I1State)|(!I2State & !I1State & !I0State)|(!I3State & I2State);
 O3State= (!I3State & !I2State)|(!I3State & I0State)|(I2State & I1State & I0State)|(I3State & !I1State & !I0State);
```

```
 digitalWrite(O0,O0State);  //lighting of the LED's
 digitalWrite(O1,O1State);
 digitalWrite(O2,O2State);
 digitalWrite(O3,O3State);


}
```

Hex File


File Name Arduinocode.ino.standard.hex
File Size 1,204 bytes (2 KiB)
Data Inspector (Little-endian)
Type Unsigned (+) Signed (±)
8-bit Integer 12 12
16-bit Integer 37900 -27636
24-bit Integer 6394892 6394892
32-bit Integer 6394892 6394892
64-bit Integer (+) 32532401588638732
64-bit Integer (±) 32532401588638732
16-bit Float. P. -0.0009880066
32-bit Float. P. 8.9611523e-39
64-bit Float. P. 1.742527264393204e-306
LEB128 (+) 12
LEB128 (±) 12
MS-DOS DateTime 1980-03-01 18:32:24 Local
OLE 2.0 DateTime 1899-12-30 00:00:00.000 UTC
UNIX 32-bit DateTime 1970-03-16 00:21:32 UTC
Macintosh HFS DateTime 1904-03-15 06:14:52 Local
Macintosh HFS+ DateTime 1904-03-15 00:21:32 UTC
Binary
Data Inspector (Big-endian)
00000000 0C 94 61 00 0C 94 73 00 0C 94 73 00 0C 94 73 00
00000010 0C 94 73 00 0C 94 73 00 0C 94 73 00 0C 94 73 00
00000020 0C 94 73 00 0C 94 73 00 0C 94 73 00 0C 94 73 00
00000030 0C 94 73 00 0C 94 73 00 0C 94 73 00 0C 94 73 00
00000040 0C 94 26 01 0C 94 73 00 0C 94 73 00 0C 94 73 00
00000050 0C 94 73 00 0C 94 73 00 0C 94 73 00 0C 94 73 00
00000060 0C 94 73 00 0C 94 73 00 00 00 00 00 24 00 27 00
00000070 2A 00 00 00 00 00 25 00 28 00 2B 00 00 00 00 00
00000080 23 00 26 00 29 00 04 04 04 04 04 04 04 04 02 02
00000090 02 02 02 02 03 03 03 03 03 03 01 02 04 08 10 20
000000A0 40 80 01 02 04 08 10 20 01 02 04 08 10 20 00 00
000000B0 00 08 00 02 01 00 00 03 04 07 00 00 00 00 00 00
000000C0 00 00 11 24 1F BE CF EF D8 E0 DE BF CD BF 21 E0
000000D0 A0 E0 B1 E0 01 C0 1D 92 A9 30 B2 07 E1 F7 0E 94
000000E0 70 01 0C 94 58 02 0C 94 00 00 83 30 81 F0 28 F4
000000F0 81 30 99 F0 82 30 A9 F0 08 95 87 30 A9 F0 88 30
00000100 C9 F0 84 30 B1 F4 80 91 80 00 8F 7D 03 C0 80 91
00000110 80 00 8F 77 80 93 80 00 08 95 84 B5 8F 77 84 BD
00000120 08 95 84 B5 8F 7D FB CF 80 91 B0 00 8F 77 80 93
00000130 B0 00 08 95 80 91 B0 00 8F 7D F9 CF CF 93 DF 93
00000140 28 2F 30 E0 F9 01 E2 55 FF 4F 84 91 F9 01 E6 56
00000150 FF 4F D4 91 F9 01 EA 57 FF 4F C4 91 CC 23 A1 F0
00000160 81 11 0E 94 75 00 EC 2F F0 E0 EE 0F FF 1F E4 58
00000170 FF 4F A5 91 B4 91 EC 91 ED 23 81 E0 90 E0 09 F4
00000180 80 E0 DF 91 CF 91 08 95 80 E0 90 E0 FA CF 1F 93
00000190 CF 93 DF 93 28 2F 30 E0 F9 01 E2 55 FF 4F 84 91
000001A0 F9 01 E6 56 FF 4F D4 91 F9 01 EA 57 FF 4F C4 91
000001B0 CC 23 A9 F0 16 2F 81 11 0E 94 75 00 EC 2F F0 E0
000001C0 EE 0F FF 1F EE 58 FF 4F A5 91 B4 91 8F B7 F8 94
000001D0 EC 91 11 11 08 C0 D0 95 DE 23 DC 93 8F BF DF 91
000001E0 CF 91 1F 91 08 95 DE 2B F8 CF CF 93 DF 93 90 E0
000001F0 FC 01 E6 56 FF 4F 24 91 8A 57 9F 4F FC 01 84 91
00000200 88 23 D1 F0 90 E0 88 0F 99 1F FC 01 E8 59 FF 4F
00000210 A5 91 B4 91 FC 01 EE 58 FF 4F C5 91 D4 91 61 11
00000220 0E C0 9F B7 F8 94 8C 91 E2 2F E0 95 8E 23 8C 93
00000230 28 81 E2 23 E8 83 9F BF DF 91 CF 91 08 95 8F B7
File Name Arduinocode.ino.standard.hex
File Size 1,204 bytes (2 KiB)
Data Inspector (Little-endian)
Type Unsigned (+) Signed (±)
8-bit Integer 12 12
16-bit Integer 37900 -27636
24-bit Integer 6394892 6394892
32-bit Integer 6394892 6394892
64-bit Integer (+) 32532401588638732
64-bit Integer (±) 32532401588638732
16-bit Float. P. -0.0009880066
32-bit Float. P. 8.9611523e-39
64-bit Float. P. 1.742527264393204e-306
LEB128 (+) 12
LEB128 (±) 12
MS-DOS DateTime 1980-03-01 18:32:24 Local
OLE 2.0 DateTime 1899-12-30 00:00:00.000 UTC
UNIX 32-bit DateTime 1970-03-16 00:21:32 UTC
Macintosh HFS DateTime 1904-03-15 06:14:52 Local
Macintosh HFS+ DateTime 1904-03-15 00:21:32 UTC
Binary
Data Inspector (Big-endian)
00000240 F8 94 EC 91 E2 2B EC 93 8F BF F6 CF 1F 92 0F 92
00000250 0F B6 0F 92 11 24 2F 93 3F 93 8F 93 9F 93 AF 93
00000260 BF 93 80 91 05 01 90 91 06 01 A0 91 07 01 B0 91
00000270 08 01 30 91 04 01 23 E0 23 0F 2D 37 58 F5 01 96
00000280 A1 1D B1 1D 20 93 04 01 80 93 05 01 90 93 06 01
00000290 A0 93 07 01 B0 93 08 01 80 91 00 01 90 91 01 01
000002A0 A0 91 02 01 B0 91 03 01 01 96 A1 1D B1 1D 80 93
000002B0 00 01 90 93 01 01 A0 93 02 01 B0 93 03 01 BF 91
000002C0 AF 91 9F 91 8F 91 3F 91 2F 91 0F 90 0F BE 0F 90
000002D0 1F 90 18 95 26 E8 23 0F 02 96 A1 1D B1 1D D2 CF
000002E0 78 94 84 B5 82 60 84 BD 84 B5 81 60 84 BD 85 B5
000002F0 82 60 85 BD 85 B5 81 60 85 BD 80 91 6E 00 81 60
00000300 80 93 6E 00 10 92 81 00 80 91 81 00 82 60 80 93
00000310 81 00 80 91 81 00 81 60 80 93 81 00 80 91 80 00
00000320 81 60 80 93 80 00 80 91 B1 00 84 60 80 93 B1 00
00000330 80 91 B0 00 81 60 80 93 B0 00 80 91 7A 00 84 60
00000340 80 93 7A 00 80 91 7A 00 82 60 80 93 7A 00 80 91
00000350 7A 00 81 60 80 93 7A 00 80 91 7A 00 80 68 80 93
00000360 7A 00 10 92 C1 00 60 E0 82 E0 0E 94 F5 00 60 E0
00000370 83 E0 0E 94 F5 00 60 E0 84 E0 0E 94 F5 00 60 E0
00000380 85 E0 0E 94 F5 00 61 E0 8A E0 0E 94 F5 00 61 E0
00000390 81 E0 0E 94 F5 00 61 E0 8C E0 0E 94 F5 00 61 E0
000003A0 83 E0 0E 94 F5 00 00 E0 10 E0 82 E0 0E 94 9E 00
000003B0 6C 01 BB 24 B3 94 89 2B 09 F4 B1 2C 83 E0 0E 94
000003C0 9E 00 7C 01 D1 E0 89 2B 09 F4 D0 E0 84 E0 0E 94
000003D0 9E 00 4C 01 AA 24 A3 94 89 2B 09 F4 A1 2C 85 E0
000003E0 0E 94 9E 00 AC 01 71 E0 89 2B 09 F4 70 E0 A1 E0
000003F0 41 15 51 05 09 F0 A0 E0 61 E0 81 14 91 04 09 F0
00000400 60 E0 B1 E0 E1 14 F1 04 09 F0 B0 E0 96 01 2E 29
00000410 3F 29 28 29 39 29 E1 E0 CD 28 09 F0 E0 E0 9B 2D
00000420 9D 23 F7 2F FB 23 8A 2D 8E 23 CD 2F CA 23 C8 2B
00000430 87 2F 86 23 B8 23 CB 2B 8B 21 C8 2B E4 2A F5 2A
00000440 81 E0 EF 28 09 F0 80 E0 EA 2C EA 22 E8 2A 81 E0
00000450 21 15 31 05 09 F0 80 E0 E8 2A FB 2C F6 2A FA 22
00000460 A9 22 AF 28 FE 2E FF 22 FA 28 69 23 BF 22 6B 29
00000470 D7 23 DE 23 6D 2B 24 2B 35 2B 91 E0 23 2B 09 F0


00000480 90 E0 69 2B 8A E0 0E 94 C7 00 6C 2F 8B E0 0E 94
00000490 C7 00 6E 2D 8C E0 0E 94 C7 00 6F 2D 8D E0 0E 94
000004A0 C7 00 01 15 11 05 09 F4 80 CF 0E 94 00 00 7D CF
000004B0 F8 94 FF CF

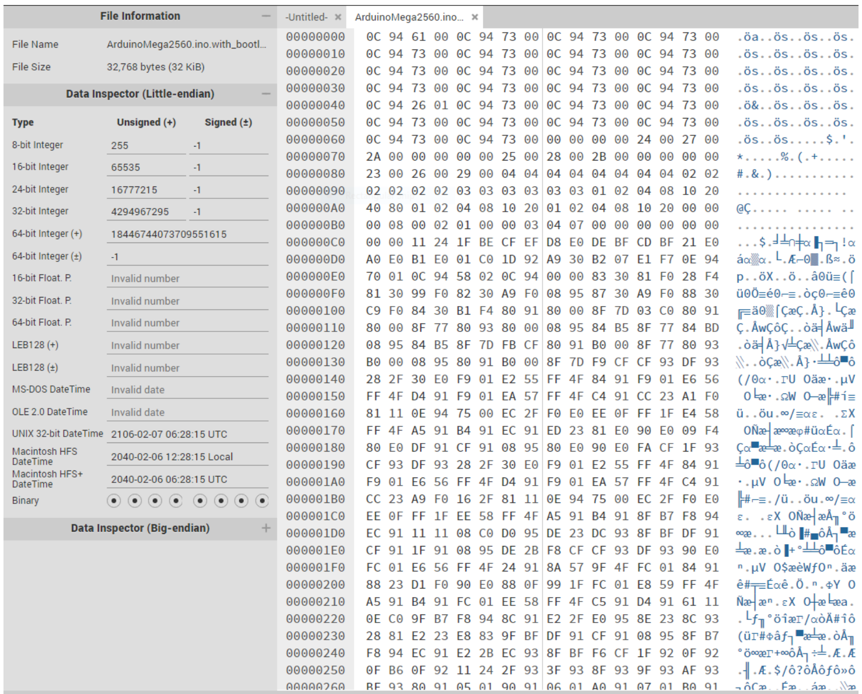
Arduino Mega

```
int I0 = 5;  //Input pins connected to digital pin

int I1 = 4;

int I2 = 3;

int I3 = 2;


int O0=9; //Output pins connected to digital pin

int O1=10;

int O2=11;

int O3=12;


void setup() {


  pinMode(I0,INPUT);  //Sets I0,I1,I2,I3 as INPUT

  pinMode(I1,INPUT);

  pinMode(I2,INPUT);

  pinMode(I3,INPUT);


  pinMode(O0,OUTPUT);  //Sets output pins as OUTPUT

  pinMode(O1,OUTPUT);

  pinMode(O2,OUTPUT);

  pinMode(O3,OUTPUT);
```

```
}

void loop() {
  boolean I0State = digitalRead(I0);  //Reads input pins
  boolean I1State = digitalRead(I1);
  boolean I2State = digitalRead(I2);
  boolean I3State = digitalRead(I3);

  boolean O0State;
  boolean O1State;
  boolean O2State;
  boolean O3State;

 O0State= (!I3State & !I2State & !I1State & !I0State)|(!I2State & I1State & I0State)|(I3State & !I1State & I0State)|(I3State & I1State& !I0State); //Setting the kmap equations for each output pins
 O1State= (!I3State & I1State)|(I2State & !I0State)|(I3State & !I2State & !I1State)|(I3State & !I2State & I0State);
 O2State= (!I3State & !I1State)|(!I2State & !I1State & !I0State)|(!I3State & I2State);
 O3State= (!I3State & !I2State)|(!I3State & I0State)|(I2State & I1State & I0State)|(I3State & !I1State & !I0State);
```

```
 digitalWrite(O0,O0State);  //lighting of the LED's
 digitalWrite(O1,O1State);
 digitalWrite(O2,O2State);
 digitalWrite(O3,O3State);


}
```

Hex File


File Information
File Name
ArduinoMega2560.ino.with_bootl...
File Size
32,768 bytes (32 KiB)
Data Inspector (Little-endian)
Type
Unsigned (+)
Signed (±)
8-bit Integer
255
-1
16-bit Integer
65535
-1
24-bit Integer
16777215
-1
32-bit Integer
4294967295
-1
64-bit Integer (+)
18446744073709551615
64-bit Integer (±)
-1
16-bit Float. P.
Invalid number
32-bit Float. P.
Invalid number
64-bit Float. P.
Invalid number
LEB128 (+)
Invalid number
LEB128 (±)
Invalid number
MS-DOS DateTime
Invalid date
OLE 2.0 DateTime
Invalid date
UNIX 32-bit DateTime
2106-02-07 06:28:15 UTC
Macintosh HFS DateTime
2040-02-06 12:28:15 Local
Macintosh HFS+ DateTime
2040-02-06 06:28:15 UTC
Binary
Data Inspector (Big-endian)
-Untitled-
ArduinoMega2560.ino...


-Untitled-
ArduinoMega2560.ino...

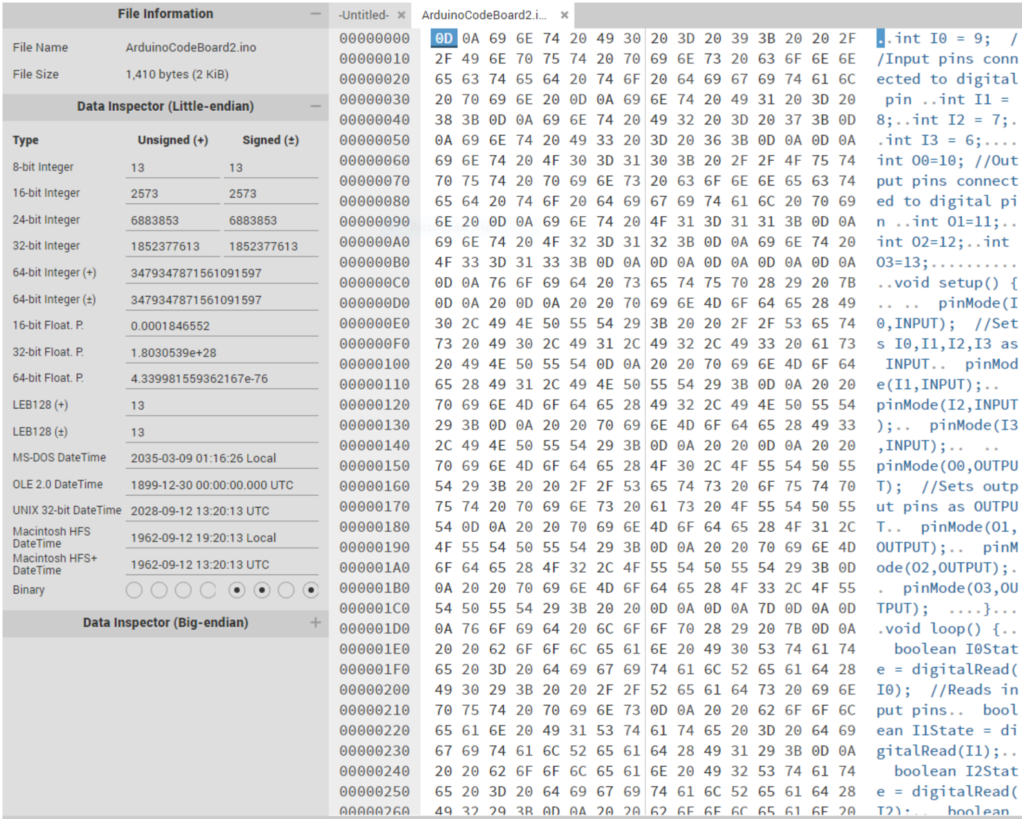
```
00007DB0  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
00007DC0  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
00007DD0  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
00007DE0  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
00007DF0  FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
00007E00  11 24 84 B7 14 BE 81 FF F0 D0 85 E0 80 93 81 00
00007E10  82 E0 80 93 C0 00 88 E1 80 93 C1 00 86 E0 80 93
00007E20  C2 00 80 E1 80 93 C4 00 8E E0 C9 D0 25 9A 86 E0
00007E30  20 E3 3C EF 91 E0 30 93 85 00 20 93 84 00 96 BB
00007E40  B0 9B FE CF 1D 9A A8 95 81 50 A9 F7 CC 24 DD 24
00007E50  88 24 83 94 B5 E0 AB 2E A1 E1 9A 2E F3 E0 BF 2E
00007E60  A2 D0 81 34 61 F4 9F D0 08 2F AF D0 02 38 11 F0
00007E70  01 38 11 F4 84 E0 01 C0 83 E0 8D D0 89 C0 82 34
00007E80  11 F4 84 E1 03 C0 85 34 19 F4 85 E0 A6 D0 80 C0
00007E90  85 35 79 F4 88 D0 E8 2E FF 24 85 D0 08 2F 10 E0
00007EA0  10 2F 00 27 0E 29 1F 29 00 0F 11 1F 8E D0 68 01
00007EB0  6F C0 86 35 21 F4 84 E0 90 D0 80 E0 DE CF 84 36
00007EC0  09 F0 40 C0 70 D0 6F D0 08 2F 6D D0 80 E0 C8 16
00007ED0  80 E7 D8 06 18 F4 F6 01 B7 BE E8 95 C0 E0 D1 E0
00007EE0  62 D0 89 93 0C 17 E1 F7 F0 E0 CF 16 F0 E7 DF 06
00007EF0  18 F0 F6 01 B7 BE E8 95 68 D0 07 B6 00 FC FD CF
00007F00  A6 01 A0 E0 B1 E0 2C 91 30 E0 11 96 8C 91 11 97
00007F10  90 E0 98 2F 88 27 82 2B 93 2B 12 96 FA 01 0C 01
00007F20  87 BE E8 95 11 24 4E 5F 5F 4F F1 E0 A0 38 BF 07
00007F30  51 F7 F6 01 A7 BE E8 95 07 B6 00 FC FD CF 97 BE
00007F40  E8 95 26 C0 84 37 B1 F4 2E D0 2D D0 F8 2E 2B D0
00007F50  3C D0 F6 01 EF 2C 8F 01 0F 5F 1F 4F 84 91 1B D0
00007F60  EA 94 F8 01 C1 F7 08 94 C1 1C D1 1C FA 94 CF 0C
00007F70  D1 1C 0E C0 85 37 39 F4 28 D0 8E E1 0C D0 85 E9
00007F80  0A D0 8F E0 7A CF 81 35 11 F4 88 E0 18 D0 1D D0
00007F90  80 E1 01 D0 65 CF 98 2F 80 91 C0 00 85 FF FC CF
00007FA0  90 93 C6 00 08 95 80 91 C0 00 87 FF FC CF 80 91
00007FB0  C0 00 84 FD 01 C0 A8 95 80 91 C6 00 08 95 E0 E6
00007FC0  F0 E0 98 E1 90 83 80 83 08 95 ED DF 80 32 19 F0
00007FD0  88 E0 F5 DF FF CF 84 E1 DE CF 1F 93 18 2F E3 DF
00007FE0  11 50 E9 F7 F2 DF 1F 91 08 95 80 E0 E8 DF EE 27
00007FF0  FF 27 09 94 FF FF FF FF FF FF FF FF FF FF 04 04
00008000
```

Arduino Pro Mini

```
int I0 = 9;  //Input pins connected to digital pin
int I1 = 8;
int I2 = 7;
int I3 = 6;

int O0=10; //Output pins connected to digital pin
int O1=11;
int O2=12;
int O3=13;

void setup() {

  pinMode(I0,INPUT);  //Sets I0,I1,I2,I3 as INPUT
  pinMode(I1,INPUT);
  pinMode(I2,INPUT);
  pinMode(I3,INPUT);

  pinMode(O0,OUTPUT);  //Sets output pins as OUTPUT
  pinMode(O1,OUTPUT);
  pinMode(O2,OUTPUT);
  pinMode(O3,OUTPUT);

}
```

```
void loop() {

  boolean I0State = digitalRead(I0);  //Reads input pins

  boolean I1State = digitalRead(I1);

  boolean I2State = digitalRead(I2);

  boolean I3State = digitalRead(I3);


  boolean O0State;

  boolean O1State;

  boolean O2State;

  boolean O3State;


 O0State= (!I3State & !I2State & !I1State & !I0State)|(!I2State & I1State & I0State)|(I3State & !I1State & I0State)|(I3State & I1State& !I0State);    //Setting the kmap equations for each output pins

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

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

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



 digitalWrite(O0,O0State);  //lighting of the LED's

 digitalWrite(O1,O1State);

 digitalWrite(O2,O2State);

 digitalWrite(O3,O3State);
```

Hex File


File Information
File Name
ArduinoCodeBoard2.ino
File Size
1,410 bytes (2 KiB)
Data Inspector (Little-endian)
Type
Unsigned (+)
Signed (±)
8-bit Integer 13 13
16-bit Integer 2573 2573
24-bit Integer 6883853 6883853
32-bit Integer 1852377613 1852377613
64-bit Integer (+) 3479347871561091597
64-bit Integer (±) 3479347871561091597
16-bit Float. P. 0.0001846552
32-bit Float. P. 1.8030539e+28
64-bit Float. P. 4.339981559362167e-76
LEB128 (+) 13
LEB128 (±) 13
MS-DOS DateTime 2035-03-09 01:16:26 Local
OLE 2.0 DateTime 1899-12-30 00:00:00.000 UTC
UNIX 32-bit DateTime 2028-09-12 13:20:13 UTC
Macintosh HFS DateTime 1962-09-12 19:20:13 Local
Macintosh HFS+ DateTime 1962-09-12 13:20:13 UTC
Binary
Data Inspector (Big-endian)
-Untitled-
ArduinoCodeBoard2.i...
00000000 0D 0A 69 6E 74 20 49 30 20 3D 20 39 3B 20 20 2F ..int I0 = 9; /
00000010 2F 49 6E 70 75 74 20 70 69 6E 73 20 63 6F 6E 6E /Input pins conn
00000020 65 63 74 65 64 20 74 6F 20 64 69 67 69 74 61 6C ected to digital
00000030 20 70 69 6E 20 0D 0A 69 6E 74 20 49 31 20 3D 20 pin ..int I1 =
00000040 38 3B 0D 0A 69 6E 74 20 49 32 20 3D 20 37 3B 0D 8;..int I2 = 7;.
00000050 0A 69 6E 74 20 49 33 20 3D 20 36 3B 0D 0A 0D 0A .int I3 = 6;....
00000060 69 6E 74 20 4F 30 3D 31 30 3B 20 2F 2F 4F 75 74 int O0=10; //Out
00000070 70 75 74 20 70 69 6E 73 20 63 6F 6E 6E 65 63 74 put pins connect
00000080 65 64 20 74 6F 20 64 69 67 69 74 61 6C 20 70 69 ed to digital pi
00000090 6E 20 0D 0A 69 6E 74 20 4F 31 3D 31 31 3B 0D 0A n ..int O1=11;..
000000A0 69 6E 74 20 4F 32 3D 31 32 3B 0D 0A 69 6E 74 20 int O2=12;..int
000000B0 4F 33 3D 31 33 3B 0D 0A 0D 0A 0D 0A 0D 0A 0D 0A O3=13;..........
000000C0 0D 0A 76 6F 69 64 20 73 65 74 75 70 28 29 20 7B ..void setup() {
000000D0 0D 0A 20 0D 0A 20 20 70 69 6E 4D 6F 64 65 28 49 .. .. pinMode(I
000000E0 30 2C 49 4E 50 55 54 29 3B 20 20 2F 2F 53 65 74 0,INPUT); //Set
000000F0 73 20 49 30 2C 49 31 2C 49 32 2C 49 33 20 61 73 s I0,I1,I2,I3 as
00000100 20 49 4E 50 55 54 0D 0A 20 20 70 69 6E 4D 6F 64 INPUT.. pinMod
00000110 65 28 49 31 2C 49 4E 50 55 54 29 3B 0D 0A 20 20 e(I1,INPUT);..
00000120 70 69 6E 4D 6F 64 65 28 49 32 2C 49 4E 50 55 54 pinMode(I2,INPUT
00000130 29 3B 0D 0A 20 20 70 69 6E 4D 6F 64 65 28 49 33);.. pinMode(I3
00000140 2C 49 4E 50 55 54 29 3B 0D 0A 20 20 0D 0A 20 20 ,INPUT);.. ..
00000150 70 69 6E 4D 6F 64 65 28 4F 30 2C 4F 55 54 50 55 pinMode(O0,OUTPU
00000160 54 29 3B 20 20 2F 2F 53 65 74 73 20 6F 75 74 70 T); //Sets outp
00000170 75 74 20 70 69 6E 73 20 61 73 20 4F 55 54 50 55 ut pins as OUTPU
00000180 54 0D 0A 20 20 70 69 6E 4D 6F 64 65 28 4F 31 2C T.. pinMode(O1,
00000190 4F 55 54 50 55 54 29 3B 0D 0A 20 20 70 69 6E 4D OUTPUT);.. pinM
000001A0 6F 64 65 28 4F 32 2C 4F 55 54 50 55 54 29 3B 0D ode(O2,OUTPUT);.
000001B0 0A 20 20 70 69 6E 4D 6F 64 65 28 4F 33 2C 4F 55 . pinMode(O3,OU
000001C0 54 50 55 54 29 3B 20 20 0D 0A 0D 0A 7D 0D 0A 0D TPUT); ....}...
000001D0 0A 76 6F 69 64 20 6C 6F 6F 70 28 29 20 7B 0D 0A .void loop() {..
000001E0 20 20 62 6F 6F 6C 65 61 6E 20 49 30 53 74 61 74   boolean I0Stat
000001F0 65 20 3D 20 64 69 67 69 74 61 6C 52 65 61 64 28 e = digitalRead(
00000200 49 30 29 3B 20 20 2F 2F 52 65 61 64 73 20 69 6E I0); //Reads in
00000210 70 75 74 20 70 69 6E 73 0D 0A 20 20 62 6F 6F 6C put pins.. bool
00000220 65 61 6E 20 49 31 53 74 61 74 65 20 3D 20 64 69 ean I1State = di
00000230 67 69 74 61 6C 52 65 61 64 28 49 31 29 3B 0D 0A gitalRead(I1);..
00000240 20 20 62 6F 6F 6C 65 61 6E 20 49 32 53 74 61 74   boolean I2Stat
00000250 65 20 3D 20 64 69 67 69 74 61 6C 52 65 61 64 28 e = digitalRead(
00000260 49 32 29 3B 0D 0A 20 20 62 6F 6F 6C 65 61 6E 20 I2);.. boolean

```
000001B0  0A 20 20 70 69 6E 4D 6F 64 65 28 4F 33 2C 4F 55  .  pinMode(O3,OU
000001C0  54 50 55 54 29 3B 20 20 0D 0A 0D 0A 7D 0D 0A 0D  TPUT);  ....}...
000001D0  0A 76 6F 69 64 20 6C 6F 6F 70 28 29 20 7B 0D 0A  .void loop() {..
000001E0  20 20 62 6F 6F 6C 65 61 6E 20 49 30 53 74 61 74    boolean I0Stat
000001F0  65 20 3D 20 64 69 67 69 74 61 6C 52 65 61 64 28  e = digitalRead(
00000200  49 30 29 3B 20 20 2F 2F 52 65 61 64 73 20 69 6E  I0);  //Reads in
00000210  70 75 74 20 70 69 6E 73 0D 0A 20 20 62 6F 6F 6C  put pins..  bool
00000220  65 61 6E 20 49 31 53 74 61 74 65 20 3D 20 64 69  ean I1State = di
00000230  67 69 74 61 6C 52 65 61 64 28 49 31 29 3B 0D 0A  gitalRead(I1);..
00000240  20 20 62 6F 6F 6C 65 61 6E 20 49 32 53 74 61 74    boolean I2Stat
00000250  65 20 3D 20 64 69 67 69 74 61 6C 52 65 61 64 28  e = digitalRead(
00000260  49 32 29 3B 0D 0A 20 20 62 6F 6F 6C 65 61 6E 20  I2);..  boolean 
00000270  49 33 53 74 61 74 65 20 3D 20 64 69 67 69 74 61  I3State = digita
00000280  6C 52 65 61 64 28 49 33 29 3B 0D 0A 20 20 0D 0A  lRead(I3);..  ..
00000290  20 20 62 6F 6F 6C 65 61 6E 20 4F 30 53 74 61 74    boolean O0Stat
000002A0  65 3B 0D 0A 20 20 62 6F 6F 6C 65 61 6E 20 4F 31  e;..  boolean O1
000002B0  53 74 61 74 65 3B 0D 0A 20 20 62 6F 6F 6C 65 61  State;..  boolea
000002C0  6E 20 4F 32 53 74 61 74 65 3B 0D 0A 20 20 62 6F  n O2State;..  bo
000002D0  6F 6C 65 61 6E 20 4F 33 53 74 61 74 65 3B 0D 0A  olean O3State;..
000002E0  0D 0A 20 4F 30 53 74 61 74 65 3D 20 28 21 49 33  .. O0State= (!I3
000002F0  53 74 61 74 65 20 26 20 21 49 32 53 74 61 74 65  State & !I2State
00000300  20 26 20 21 49 31 53 74 61 74 65 20 26 20 21 49   & !I1State & !I
00000310  30 53 74 61 74 65 29 7C 28 21 49 32 53 74 61 74  0State)|(!I2Stat
00000320  65 20 26 20 49 31 53 74 61 74 65 20 26 20 49 30  e & I1State & I0
00000330  53 74 61 74 65 29 7C 28 49 33 53 74 61 74 65 20  State)|(I3State 
00000340  26 20 21 49 31 53 74 61 74 65 20 26 20 49 30 53  & !I1State & I0S
00000350  74 61 74 65 29 7C 28 49 33 53 74 61 74 65 20 26  tate)|(I3State &
00000360  20 49 31 53 74 61 74 65 26 20 21 49 30 53 74 61   I1State& !I0Sta
00000370  74 65 29 3B 20 20 20 20 2F 2F 53 65 74 74 69 6E  te);    //Settin
00000380  67 20 74 68 65 20 6B 6D 61 70 20 65 71 75 61 74  g the kmap equat
00000390  69 6F 6E 73 20 66 6F 72 20 65 61 63 68 20 6F 75  ions for each ou
000003A0  74 70 75 74 20 70 69 6E 73 0D 0A 20 4F 31 53 74  tput pins.. O1St
000003B0  61 74 65 3D 20 28 21 49 33 53 74 61 74 65 20 26  ate= (!I3State &
000003C0  20 49 31 53 74 61 74 65 29 7C 28 49 32 53 74 61   I1State)|(I2Sta
000003D0  74 65 20 26 20 21 49 30 53 74 61 74 65 29 7C 28  te & !I0State)|(
000003E0  49 33 53 74 61 74 65 20 26 20 21 49 32 53 74 61  I3State & !I2Sta
000003F0  74 65 20 26 20 21 49 31 53 74 61 74 65 29 7C 28  te & !I1State)|(
00000400  49 33 53 74 61 74 65 20 26 20 21 49 32 53 74 61  I3State & !I2Sta
00000410  74 65 20 26 20 49 30 53 74 61 74 65 29 3B 0D 0A  te & I0State);..
```


```
00000330  53 74 61 74 65 29 7C 28 49 33 53 74 61 74 65 20  State)|(I3State 
00000340  26 20 21 49 31 53 74 61 74 65 20 26 20 49 30 53  & !I1State & I0S
00000350  74 61 74 65 29 7C 28 49 33 53 74 61 74 65 20 26  tate)|(I3State &
00000360  20 49 31 53 74 61 74 65 26 20 21 49 30 53 74 61   I1State& !I0Sta
00000370  74 65 29 3B 20 20 20 20 2F 2F 53 65 74 74 69 6E  te);    //Settin
00000380  67 20 74 68 65 20 6B 6D 61 70 20 65 71 75 61 74  g the kmap equat
00000390  69 6F 6E 73 20 66 6F 72 20 65 61 63 68 20 6F 75  ions for each ou
000003A0  74 70 75 74 20 70 69 6E 73 0D 0A 20 4F 31 53 74  tput pins.. O1St
000003B0  61 74 65 3D 20 28 21 49 33 53 74 61 74 65 20 26  ate= (!I3State &
000003C0  20 49 31 53 74 61 74 65 29 7C 28 49 32 53 74 61   I1State)|(I2Sta
000003D0  74 65 20 26 20 21 49 30 53 74 61 74 65 29 7C 28  te & !I0State)|(
000003E0  49 33 53 74 61 74 65 20 26 20 21 49 32 53 74 61  I3State & !I2Sta
000003F0  74 65 20 26 20 21 49 31 53 74 61 74 65 29 7C 28  te & !I1State)|(
00000400  49 33 53 74 61 74 65 20 26 20 21 49 32 53 74 61  I3State & !I2Sta
00000410  74 65 20 26 20 49 30 53 74 61 74 65 29 3B 0D 0A  te & I0State);..
```


```
00000420  20 4F 32 53 74 61 74 65 3D 20 28 21 49 33 53 74   O2State= (!I3St
00000430  61 74 65 20 26 20 21 49 31 53 74 61 74 65 29 7C  ate & !I1State)|
00000440  28 21 49 32 53 74 61 74 65 20 26 20 21 49 31 53  (!I2State & !I1S
00000450  74 61 74 65 20 26 20 21 49 30 53 74 61 74 65 29  tate & !I0State)
00000460  7C 28 21 49 33 53 74 61 74 65 20 26 20 49 32 53  |(!I3State & I2S
00000470  74 61 74 65 29 3B 0D 0A 20 4F 33 53 74 61 74 65  tate);.. O3State
00000480  3D 20 28 21 49 33 53 74 61 74 65 20 26 20 21 49  = (!I3State & !I
00000490  32 53 74 61 74 65 29 7C 28 21 49 33 53 74 61 74  2State)|(!I3Stat
000004A0  65 20 26 20 49 30 53 74 61 74 65 29 7C 28 49 32  e & I0State)|(I2
000004B0  53 74 61 74 65 20 26 20 49 31 53 74 61 74 65 20  State & I1State 
000004C0  26 20 49 30 53 74 61 74 65 29 7C 28 49 33 53 74  & I0State)|(I3St
000004D0  61 74 65 20 26 20 21 49 31 53 74 61 74 65 20 26  ate & !I1State &
000004E0  20 21 49 30 53 74 61 74 65 29 3B 0D 0A 0D 0A 20   !I0State);.... 
000004F0  0D 0A 20 64 69 67 69 74 61 6C 57 72 69 74 65 28  .. digitalWrite(
00000500  4F 30 2C 4F 30 53 74 61 74 65 29 3B 20 20 2F 2F  O0,O0State);  //
00000510  6C 69 67 68 74 69 6E 67 20 6F 66 20 74 68 65 20  lighting of the 
00000520  4C 45 44 27 73 0D 0A 20 64 69 67 69 74 61 6C 57  LED's.. digitalW
00000530  72 69 74 65 28 4F 31 2C 4F 31 53 74 61 74 65 29  rite(O1,O1State)
00000540  3B 0D 0A 20 64 69 67 69 74 61 6C 57 72 69 74 65  ;.. digitalWrite
00000550  28 4F 32 2C 4F 32 53 74 61 74 65 29 3B 0D 0A 20  (O2,O2State);.. 
00000560  64 69 67 69 74 61 6C 57 72 69 74 65 28 4F 33 2C  digitalWrite(O3,
00000570  4F 33 53 74 61 74 65 29 3B 0D 0A 0D 0A 0D 0A 7D  O3State);......}
00000580  0D 0A                                            ..
```

Question/Answers

Arduino Uno:

I) Clock frequency of the microcontroller used-

Arduino Uno is a popular microcontroller development board based on 8-bit ATmega328P microcontroller. Arduino Uno has an inbuilt clock frequency up to 8Mhz whereas an external crystal frequency 16MHz is also available.

II) Data bus width of the microcontroller-The data buses are 32-bit wide. The external address bus is 24 bits wide. This will allow you to access up to 16 MB of memory. It also has 8 chip selects.

III) Size of hex file generated-3.32 KB (3,409 bytes)

IV) Can the project be implemented using interrupt -

Yes the project can be implemented using interrupt As previously stated on Arduino Uno you can only use pin 2 and 3 for interrupts. On the Arduino Uno pins 2 and 3 are capable of generating interrupts and they correspond to interrupt vectors 0 and 1 respectively.

V) Is the main routine required to be an infinite loop - Yes

VI) Is there any difference between level triggered and edge triggered operation for the given project-

The main difference between edge and level triggering is that in edge triggering, the output of the sequential circuit changes during the high voltage period or low voltage period while, in level triggering, the output of the sequential circuit changes during transits from the high voltage to low voltage or low voltage to high voltage.

vii) Is the project referring encryption or decryption for input to output-For input it's referring to encryption.

Arduino Pro mini:

I) Clock frequency of the microcontroller used-

8 MHz (3.3V versions) or 16 MHz (5V versions)

II) Data bus width of the microcontroller-The data buses are 32-bit wide. The external address bus is 24 bits wide. This will allow you to access up to 16 MB of memory. It also has 8 chip selects.

III) Size of hex file generated-3.32 KB (3,409 bytes)

IV) Can the project be implemented using interrupt -

Yes the project can be implemented using external interrupt As previously stated on Arduino Uno you can only use pin 2 and 3 for interrupts. On the Arduino Uno pins 2 and 3 are capable of generating interrupts and they correspond to interrupt vectors 0 and 1 respectively.

V) Is the main routine required to be an infinite loop - Yes

VI) Is there any difference between level triggered and edge triggered operation for the given project-

The main difference between edge and level triggering is that in edge triggering, the output of the sequential circuit changes during the high voltage period or low voltage period while, in level triggering, the output of the sequential circuit changes during transits from the high voltage to low voltage or low voltage to high voltage.

vii) Is the project referring encryption or decryption for input to output-For input it's referring to encryption.

Arduino Mega 2560

I) Clock frequency of the microcontroller used-

The controller used in this board is ATmega2560 has a clock speed of 16 MHz .

II) Data bus width of the microcontroller-

Many chips like the Mega1284 and 2560 actually address 128 or 256k of flash, that means their address bus is **17 or 18 bits**. Yet they are called an 8-bit processor because the data bus is 8 bits. Some 16-bit processors like the 68000 had 24-bit address busses

III) Size of hex file generated-3.32 KB (3,409 bytes)

IV) Can the project be implemented using interrupt - **External Interrupts:** 2 (interrupt 0), 3 (interrupt 1), 18 (interrupt 5), 19 (interrupt 4), 20 (interrupt 3), and 21 (interrupt 2). These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the attach Interrupt() function for details.

V) Is the main routine required to be an infinite loop - Yes

VI) Is there any difference between level triggered and edge triggered operation for the given project-

The main difference between edge and level triggering is that in edge triggering, the output of the sequential circuit changes during the high voltage period or low voltage period while, in level triggering, the output of the sequential circuit changes during transits from the high voltage to low voltage or low voltage to high voltage.

vii) Is the project referring encryption or decryption for input to output-For input it's referring to encryption.

Power Consumption

I)Maximum and minimum power consumption sequence for a specific bit sequence for a specific board(In case of multiple cases, report all. exclusively comment on the input-output pair that will draw the maximum power assuming a single power supply is used for the system (compulsory)

II) Comparison of maximum and minimum power consumption sequence between the three selected boards –
Arduino Uno:
Minimum Operating voltage -2.7V
Maximum operating voltage-6V

Arduino Pro Mini:
3.35 -12 V (3.3V model) or 5 - 12 V (5V model)

Arduino Mega 2560
Minimum Operating voltage -5V
Maximum operating voltage-6-20 V

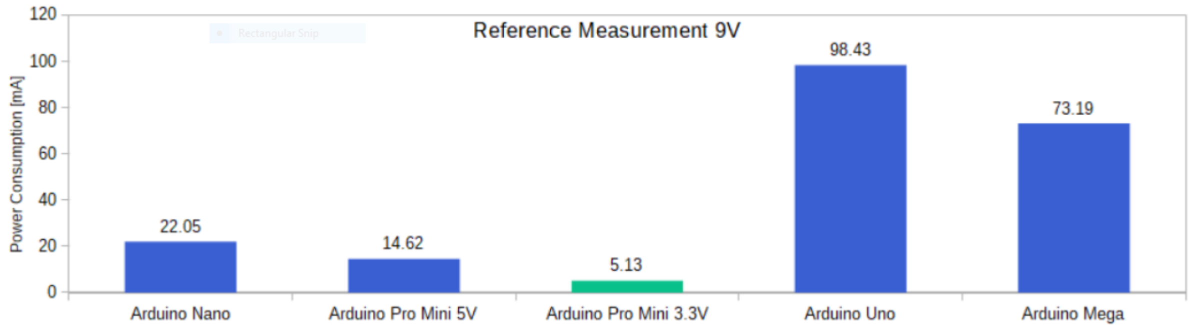
III) Comparison of power consumption between the three selected boards in idle mode (battery on with no switching)-

IV) Discussion on variation of power consumption among the three selected boards based based on the technical parameters provided in slide –

Arduino Uno-The power supply of an Arduino Uno can be between 7V-12V.The current consumption lies between 45mA-80mA while the current consumption deep is 35mA.

Arduino Pro mini- The Arduino Uno board draws about 42 mA assuming no power draw from sensors or other components needed in your system. With a minimum supply voltage of 7 volts, the power consumption of the board is therefore 0.29 Watts.
The lowest current consumption has the Arduino Pro Mini with 1.58mA

Arduino Mega 2560-With no pin IO,we can expect a load between 50mA and 75 mA.It has an upperlimit of 200 mA.



Current reading with no code and code burnt

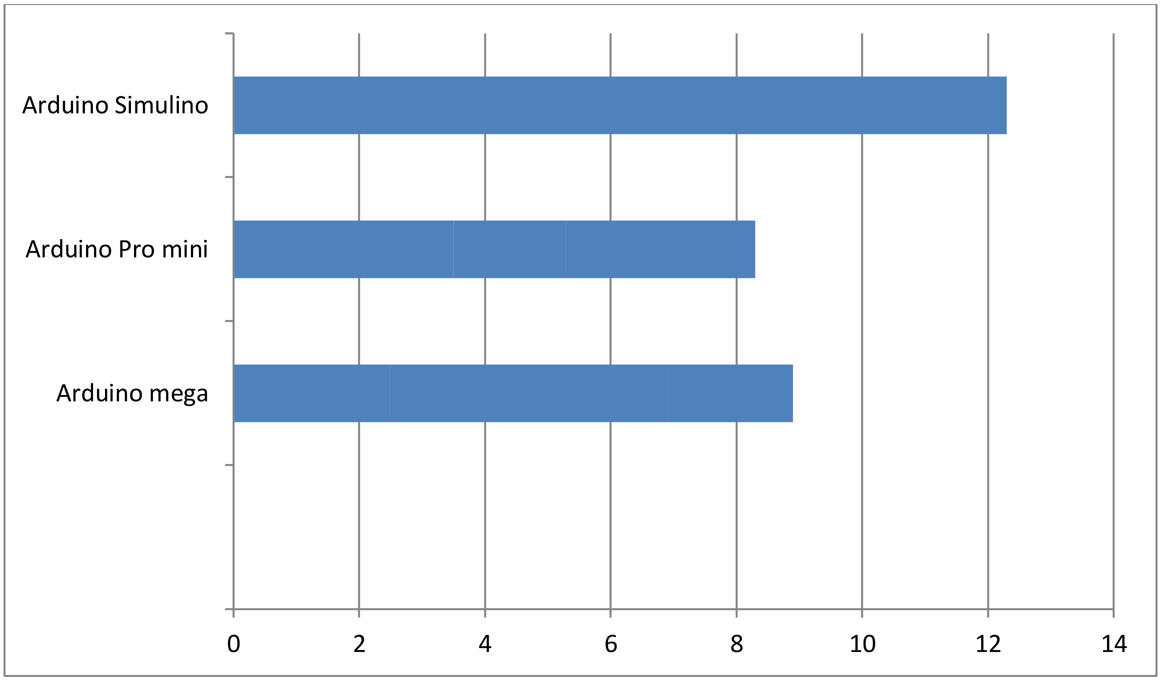
Board	Current/mA(No code)	Current/mA(Code Burnt)
Arduino simulino	9	135.9
Arduino Mega	10	44.5
Arduino Pro mini	16	35

Sequences that give maximum reading for each board

Board	Maximum	Minimum
Arduino Simulino	1000	1101
Arduino Mega	1111	1010
Arduino Pro mini	1000	1010

Power Consumption chart for each board

Power Consumption



Discussion

In the group project we had to Implement a given encryption table using microcontroller using single pole, double throw switch to configure the inputs for high and low conditions and LEDs to represent the corresponding output statuses. We used the software Proteus to build the software circuits for the three boards and implemented one of the software circuit using hardware equipments.We used Arduino Uno, Breadboard, Double pin switch,LED’s and wires. The implementation of the circuit in the software was hassle free but while implementing the hardware we faced a series of problems regarding some of the hardware equipment’s.

- The switches we initially used were single pin switches so therefore it was hard to deduce when it the switch is ON/OFF. We had to rely on the output. Again when we shifted to double pin switch the problem remained the same. The switches had degrading quality as well as it was hard to insert them in the breadboard.
- The LED’s we used as output only worked for an hour/two and got dimmer with time. All the LED’s we used weren’t of good quality so we often got confused of the problem was in our circuit or in the LED’s.
- Lastly, we had little knowledge of the hardware therefore we were at a dead end multiple times and it required a lot of time than expected to build one simple circuit.

Conclusion

Overall, the project provided us with a tremendous learning opportunity by allowing us to learn about various software programs and get our hands on hardware that we otherwise would not have had access to.