

**North South University**

Department of Electrical & Computer Engineering

Summer 2023

**CSE331**

Microprocessor Interfacing & Embedded System

Project Report

***‘Implementation of an Encryption Table Using Micro controller’***

## Faculty: Dr. Dihan Md. Nuruddin Hasan (DMH) Section: 1

**Group: 11**

**Names and IDs of the Group Members:**

|  |  |
| --- | --- |
| MD Nazmul Islam Srabon | 2021227042 |
| Mia Mahmudul Houque Shaon | 2014314042 |
| Asif Rahman | 2012691042 |
| Tamim Shadman | 2021522642 |
| Ahanaf Akif Pervez | 1821145042 |
| Ibrahim Rashel | 1731627642 |

# Table of Contents

1. [Objective 3](#_TOC_250030)
2. [General Description 4](#_TOC_250029)
   1. [Arduino UNO R3 **4**](#_TOC_250028)
   2. [Arduino Mega 2560 **5**](#_TOC_250027)
   3. [Arduino IDE **5**](#_TOC_250026)
   4. [Proteus 8 Pro **6**](#_TOC_250025)
   5. [Logisim **6**](#_TOC_250024)
3. [Equipment Used 7](#_TOC_250023)
4. [Method of Derivation 8](#_TOC_250022)
   1. [Truth Table **8**](#_TOC_250021)
   2. [Karnaugh Map **9**](#_TOC_250020)
   3. [Derived Boolean Output Expression from Karnaugh Map **11**](#_TOC_250019)
5. [Circuit Diagram with Values of Electrical Components 11](#_TOC_250018)
   1. [Logisim Circuit **11**](#_TOC_250017)
   2. [Proteus Circuit **12**](#_TOC_250016)
6. [Circuit Operation Principles 13](#_TOC_250015)
7. [Program Flow Chart 15](#_TOC_250014)
8. [Arduino Program 16](#_TOC_250013)
   1. [Arduino Program for both Arduino UNO R3 and Mega using Arduino Libraries **16**](#_TOC_250012)
   2. [Hex Code **18**](#_TOC_250011)
      1. [Hex Codes generated for use on Arduino UNO R3 **18**](#_TOC_250010)
      2. Hex Code generated for use on Arduino Mega 2560 **21**
9. [Result Analysis and Comparison Among The Boards Used 25](#_TOC_250009)
   1. [Hardware Comparison Among Boards Used **25**](#_TOC_250008)
   2. [Power Consumption Comparison **26**](#_TOC_250007)
   3. [Memory Size and Execution Time Comparison **27**](#_TOC_250006)
10. [Conclusion 28](#_TOC_250005)
11. [Questions and Answers 29](#_TOC_250004)
    1. [Arduino UNO **29**](#_TOC_250003)
    2. [Arduino Mega 2560 **30**](#_TOC_250002)

[12.References 32](#_TOC_250001)

[13 Resources 32](#_TOC_250000)

## Objective

The project's main aim was to develop a 4-bit encryption system that obfuscates its output through the use of Boolean expressions. The encryption algorithm was devised based on a Truth Table that maps inputs to their respective outputs. A 4-bit input is provided to a microcontroller, which encrypts the output according to the Truth Table using Boolean logic expressions. The encrypted result is displayed through 4 LEDs, each representing a single bit. There are a total of 16 potential encryption combinations, ranging from 0 to 15 (corresponding to binary values from 0000 to 1111). For each of these combinations, the circuit's current draw is measured.

The primary objective of this experiment is to quantify the current consumption for all feasible logic combinations applied to the input and the overall system. This evaluation is conducted using two different microcontroller boards, each with distinct specifications. A comparison is made between these boards in terms of program size, memory utilization, and execution time.

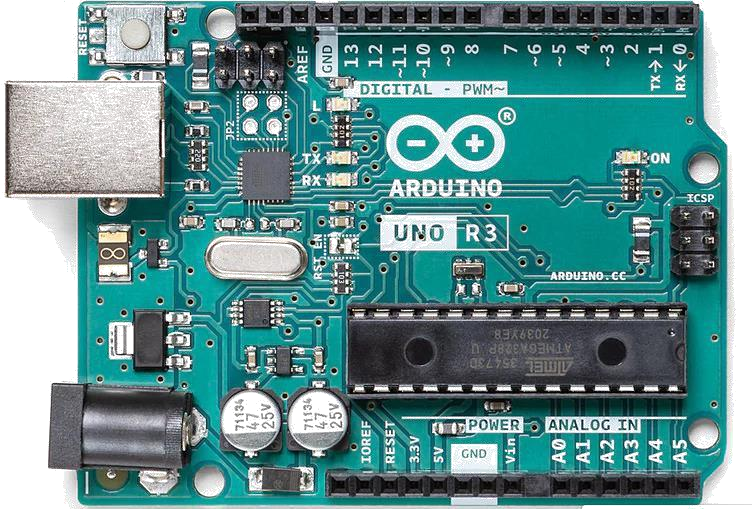
To achieve this, the HEX file generated by the programmed Arduino IDE code is imported into a circuit schematic designed in Proteus for system simulation. A virtual terminal is employed to observe the program's execution. The encryption system is executed on two distinct boards in Proteus: the Arduino UNO, which employs the ATmega328p microcontroller, and the Simulino UNO, which also employs the ATmega328p microcontroller. Both boards execute the encryption program using their respective programming methods, allowing for a performance and memory comparison to ensure the results are consistent and trustworthy.

# General Description

## Arduino UNO R3

The Arduino UNO R3 is a microcontroller board featuring the ATmega328P microcontroller. It is equipped with a 16 MHz ceramic resonator (CSTCE16M0V53-R0), 6 analog inputs, 14 digital input/output pins (with 6 offering PWM capabilities), a USB port, a power jack, an ICSP header, and a reset button. This board operates at a clock frequency of 16 MHz and boasts 32 KB of flash memory along with 2 KB of SRAM. The Arduino UNO R3 supports serial communication through USB and interfaces like UART, SPI, and I2C.

This board can execute programs written in standard C/C++ and employs libraries and headers designed for Arduino within the Arduino IDE. Its user-friendly nature, compact form factor, and versatility make it a highly sought-after choice for prototyping purposes.



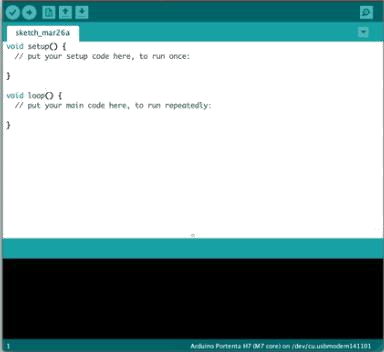
*Figure: Arduino UNO Rev3*

## Arduino Mega 2560



*Figure: Arduino Mega 2560*

The Arduino Mega 2560 is powered by the Atmega2560 microcontroller and is purpose-built for handling substantial projects that require a multitude of input and output ports. It stands out with an impressive array of 54 digital IO pins and 16 analog pins. Furthermore, among the digital IO pins, 15 are equipped to support Pulse Width Modulation (PWM), allowing it to manage multiple components simultaneously for data input and output. Key features of the Arduino Mega 2560 include built-in serial communication through USB and compatibility with various interfaces like UART, SPI, and I2C. It boasts a substantial 256 KB of memory for storing extensive code, in addition to 8 KB of SRAM and 4 KB of EEPROM. Similar to the Arduino UNO, the Arduino Mega 2560 operates at a clock frequency of 16 MHz

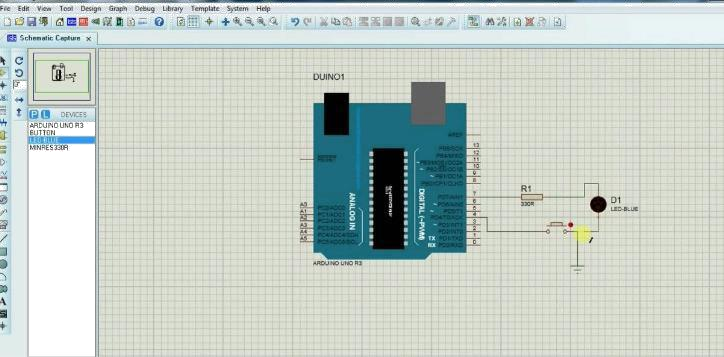
* 1. **Arduino IDE**

*Figure: Arduino IDE*

The Arduino IDE is an integrated development environment designed for programming and creating projects on Arduino boards using C/C++. It offers a user-friendly environment with a built-in text editor for code creation, compilation, and uploading to Arduino boards. The IDE includes a message area, text console, a toolbar with commonly used functions, and a menu system. In essence, it provides an accessible interface for writing, compiling, and transferring code to Arduino boards. The IDE also comes with a collection of libraries and functions that simplify the process of connecting code to the boards.

Programs created within the Arduino IDE are referred to as "sketches," and they can be effortlessly uploaded to the board with a single button click using the IDE's upload feature.

## Proteus 8 Pro



*Figure: Proteus 8 Pro Interface*

Proteus 8 Pro is a widely used and multifaceted software for designing electronic circuits, conducting simulations, and arranging PCB layouts. Its user-friendly interface, robust simulation capabilities, and vast collection of electronic components have made it a favored option for both electronics enthusiasts and industry experts.

## Logisim

*Figure: Logisim Interface*



Logisim is an open-source application designed for the creation and simulation of logic circuits. It is suitable for constructing simple as well as complex educational circuits and is primarily intended for educational purposes. The software offers a rich assortment of commonly used digital components for circuit design and simulation. It features an intuitive interface that allows users to effortlessly drag and drop components and observe real-time simulations to visualize output changes in response to input adjustments.

# Equipment Used

### Single pole double throw switch:

This is an electrical switch with a single input terminal (pole) that can establish a connection with one of two output terminals (throws).

### 10 kΩ Resistor:

This electronic component offers an electrical resistance of 10,000 ohms to impede the flow of electric current.

### Red LED:

This is a semiconductor component that generates red light when an electric current flows through it.

### Arduino UNO REV3:

This widely utilized microcontroller board is a common choice for electronics projects and prototyping. It is founded on the ATmega328P microcontroller and belongs to the Arduino board family.

### Arduino Mega 2560:

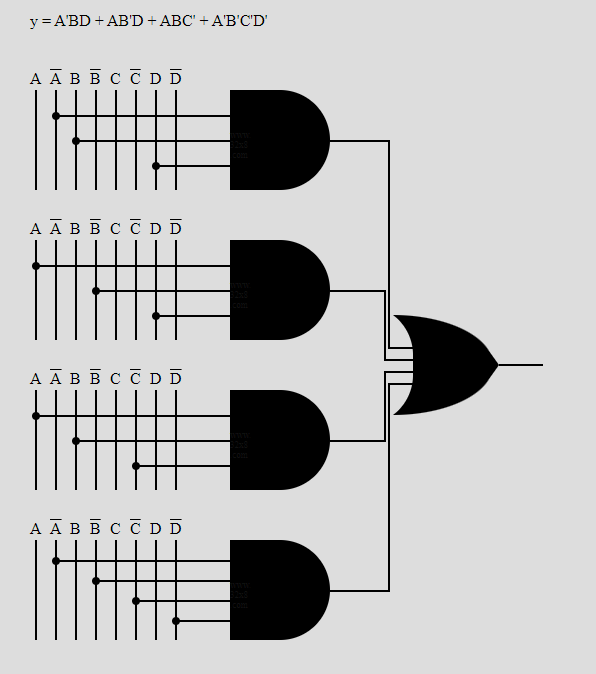
This expansive and adaptable microcontroller board possesses more extensive flash memory and EEPROM compared to other Arduino boards. It relies on the ATmega2560 microcontroller and delivers similar capabilities but in a larger physical size and with greater capacity.

# Method of Derivation

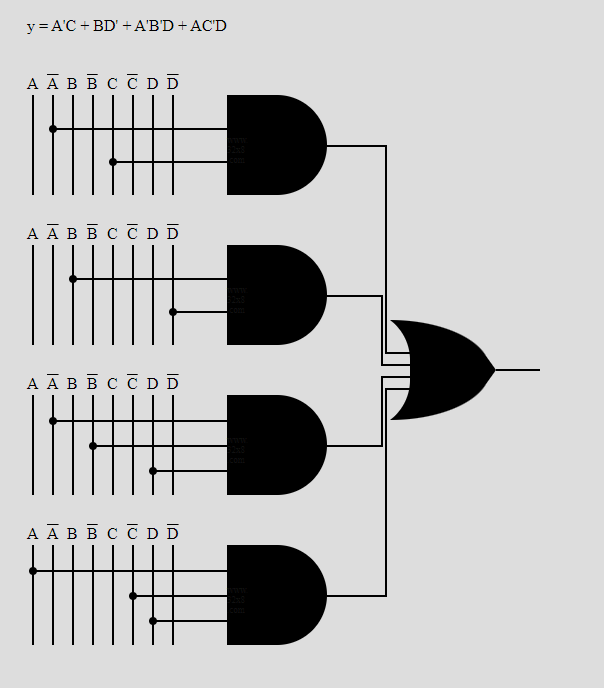
## Truth Table

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Input** | | | | **Output** | | | |
| **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** |

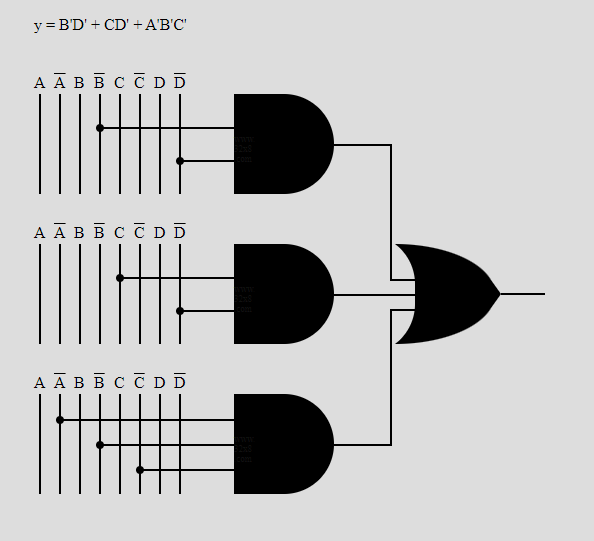
## Karnaugh Map



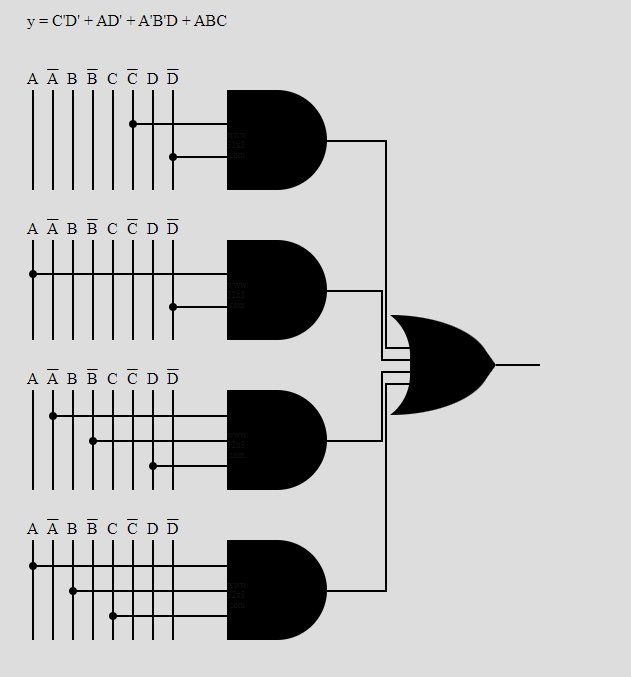
*Figure: K map Derivation for O0*



*Figure: K map Derivation for O1*



*Figure: K map Derivation for O2*



*Figure: K map Derivation for O3*

## Derived Boolean Output Expression from Karnaugh Map

The Final Derived Expression for each output from the K maps is:

**1. O0 (I3, I2, I1, I0) = I0’ I1 I3 + I0 I1’ I3 + I0 I1 I2’ + I0’ I1’ I2’ I3’**

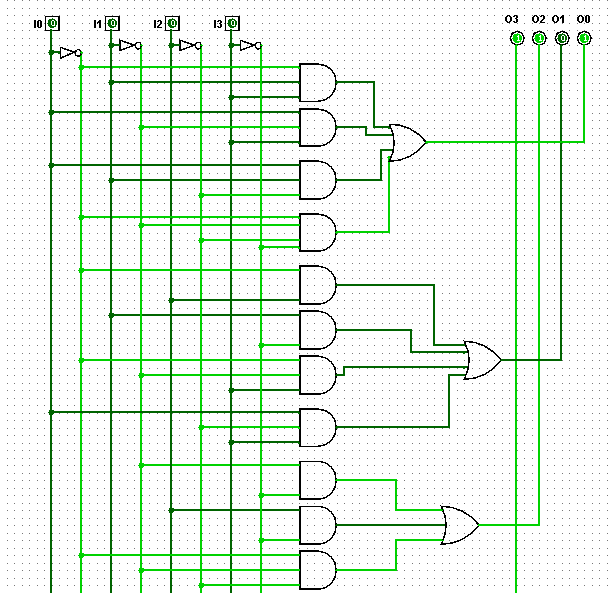
**2. O1 (I3, I2, I1, I0) = I0’ I2 + I1 I3’ + I0’ I1’ I3 + I0 I2’ I3**

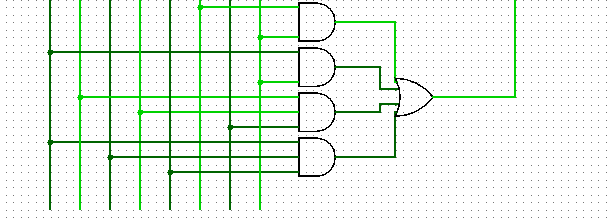
**3. O2 (I3, I2, I1, I0) = I1’ I3’ + I2 I3’ + I0’ I1’ I2’**

**4. O3 (I3, I2, I1, I0) = I2’ I3’ + I0 I3’ + I0’ I1’ I3 + I0 I1 I2**

## Circuit Diagram with Values of Electrical Components

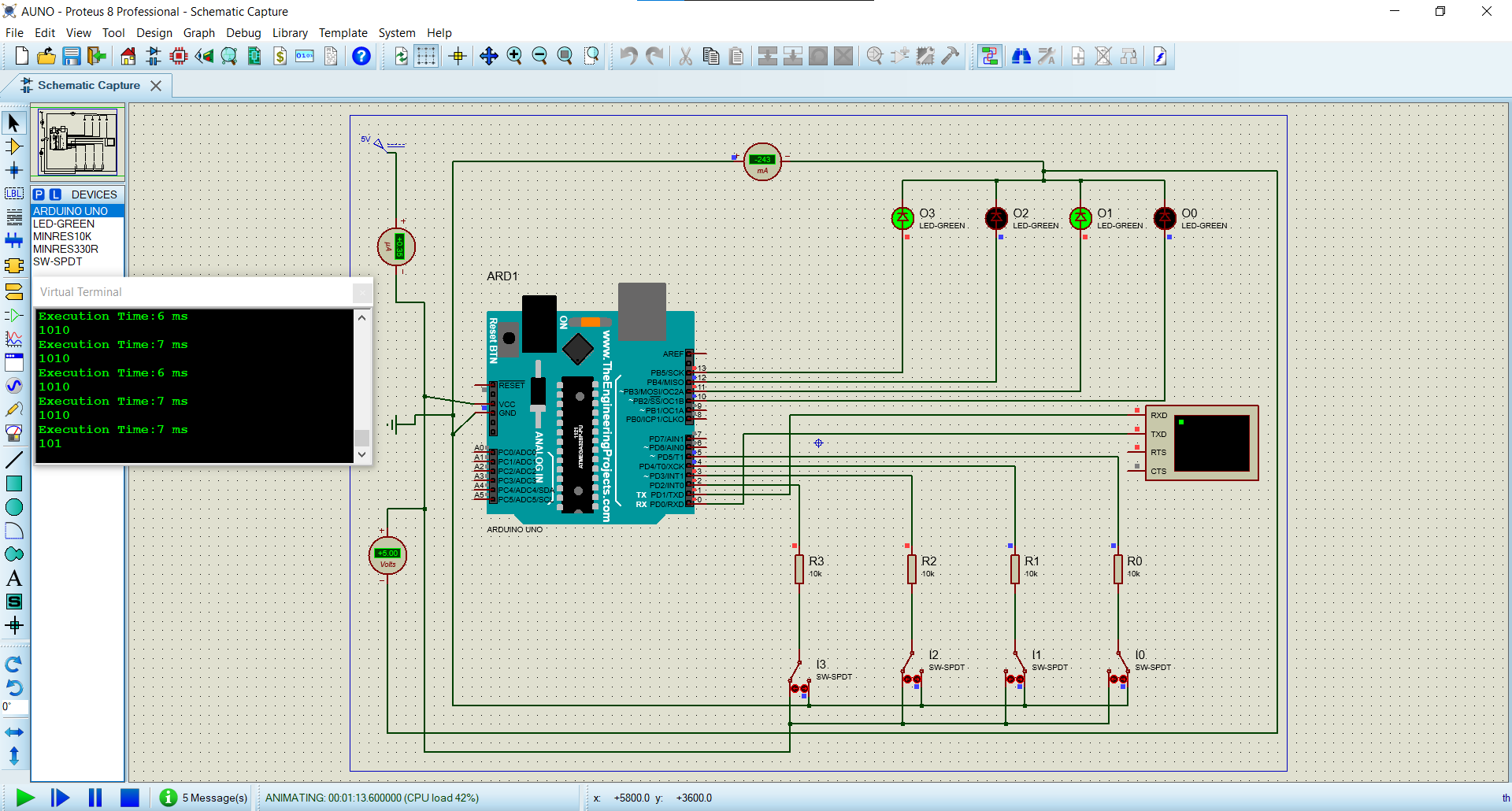
### Logisim Circuit

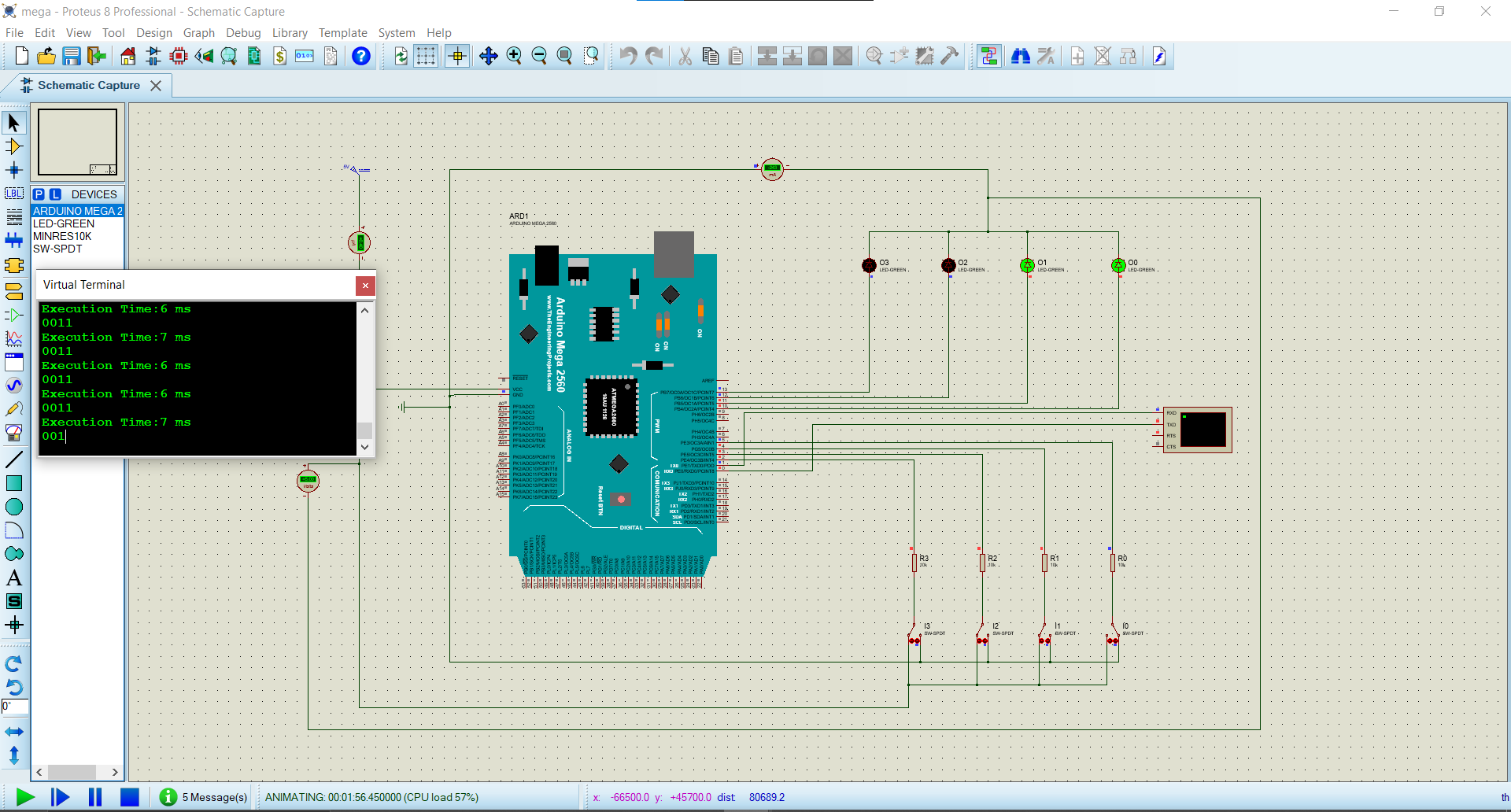
****

****

*Figure: Logisim Simulation*

## Proteus Circuit

 *Figure: Proteus Simulation Demo for Arduino Uno*

 *Figure: Proteus Simulation Demo for Arduino Mega 2560*

# Circuit Operation Principles

For our circuit operation the output equation from the K-map is as follows:

**1. O0 (I3, I2, I1, I0) = I0’ I1 I3 + I0 I1’ I3 + I0 I1 I2’ + I0’ I1’ I2’ I3’**

**2. O1 (I3, I2, I1, I0) = I0’ I2 + I1 I3’ + I0’ I1’ I3 + I0 I2’ I3**

**3. O2 (I3, I2, I1, I0) = I1’ I3’ + I2 I3’ + I0’ I1’ I2’**

**4. O3 (I3, I2, I1, I0) = I2’ I3’ + I0 I3’ + I0’ I1’ I3 + I0 I1 I2**

As a demo to test whether these expression is correct for our input bits, we designed a logic circuit on Logisim and simulated using four input and output bits and using the help AND, OR, and NOT gates to achieve the output equations.

Then, with the help of Proteus 8 Professional software, we built our main Hardware Circuit. We have used the following components:

**Arduino Uno**: We have used an Arduino Uno Model as our first board to test.

We will connect the Vcc pin of Arduino to a power supply of 5V. And we will connect the ground pin of the Arduino to a Ground component.

**Arduino Mega**: Used as the second board to test and compare. Connection of Vcc is same as with Arduino Uno.

**Resistors:** We have used a total of 4 resistors of 10k Ohms. These are connected with the SPDT switches.

**Single Pole Double Throw Switches**: We used four of these Switches with our Arduino Digital pins. They are arranged as follows:

### Arduino UNO Digital Pin No. 5 SPDT Switch I0

**Arduino UNO Digital Pin No. 4 SPDT Switch I1**

### Arduino UNO Digital Pin No. 3 SPDT Switch I2

**Arduino UNO Digital Pin No. 2 SPDT Switch I3**

The Resistors are connected in series with the pins and SPDT switches while the other pole of the switches is connected to the GND. Hence, one side of the switch acts as open while the other acts as closed and provides current flow and power to the pins.

**LED's**: We have connected Four LED's to the Digital Pins of Arduino to receive Output through those Pins. They are arranged in such way:

### Arduino UNO Digital Pin No. 10 LED O0

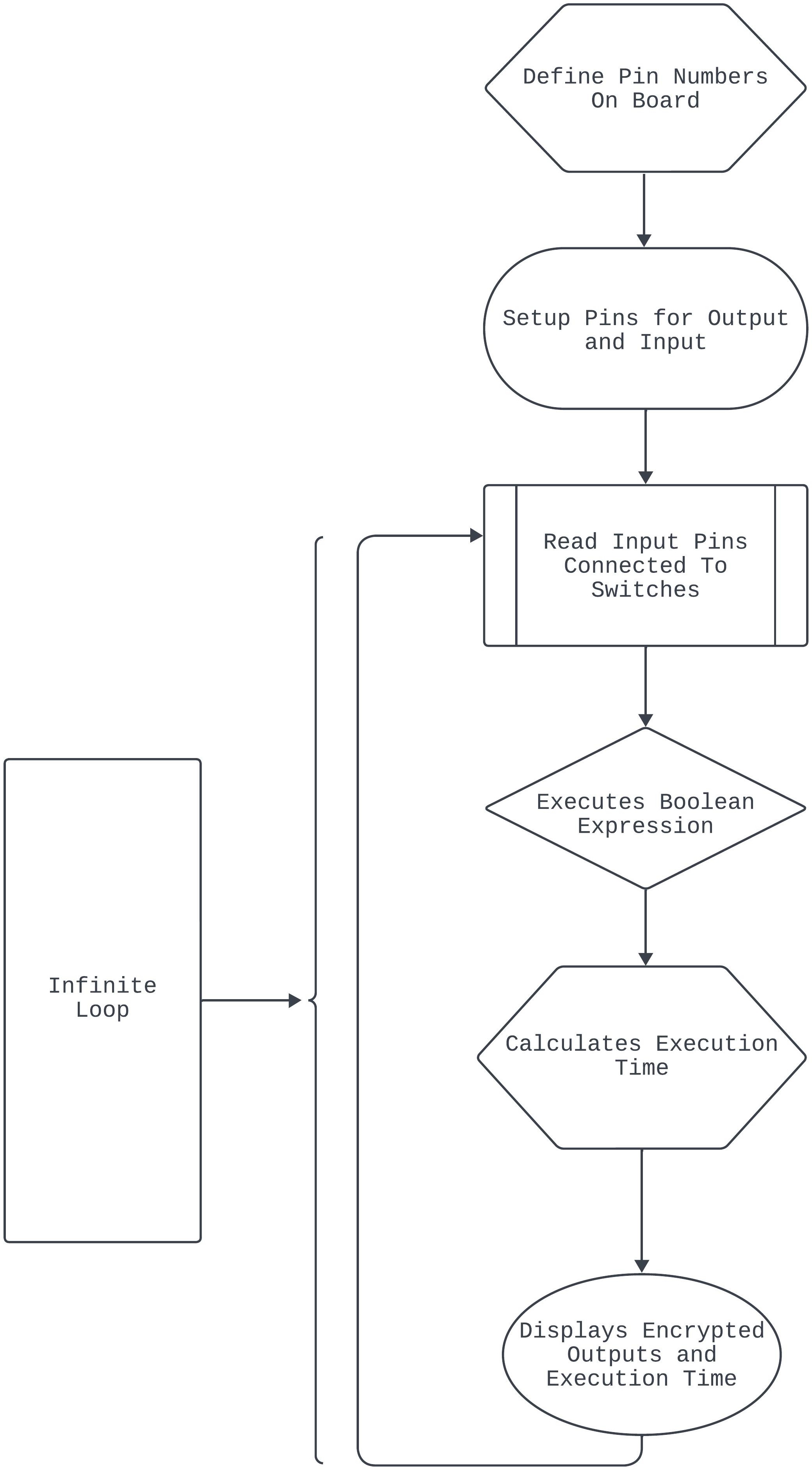
**Arduino UNO Digital Pin No. 11 LED O1**

### Arduino UNO Digital Pin No. 12 LED O2

**Arduino UNO Digital Pin No. 13 LED O3**

Finally, a Virtual Terminal is used to monitor the execution time to compare performance between Register Level Programming and C- programming. This execution time is analyzed in the Results Analysis section. The Hex file exported from Arduino IDE for both types of programming is imported on the board and simulated.

# Program Flow Chart



*Figure : Program Loop Flowchart*

# Arduino Program

## Arduino Program for both Arduino UNO R3 and Mega using Arduino Libraries

**Arduino Code:**

// Output LED pin variables

int O0LED = 10; // O0 LED connected to pin 10

int O1LED = 11; // O1 LED connected to pin 11

int O2LED = 12; // O2 LED connected to pin 12

int O3LED = 13; // O3 LED connected to pin 13

// Input switch pin variables

int I0Switch = 5; // Switch O0 connected to pin 5

int I1Switch = 4; // Switch O1 connected to pin 4

int I2Switch = 3; // Switch O2 connected to pin 3

int I3Switch = 2; // Switch O3 connected to pin 2

// Define constants for LED states

const boolean LED\_ON = HIGH;

const boolean LED\_OFF = LOW;

void setup() {

  // Set switch pins as inputs

  pinMode(I0Switch, INPUT);

  pinMode(I1Switch, INPUT);

  pinMode(I2Switch, INPUT);

  pinMode(I3Switch, INPUT);

  // Set LED pins as outputs

  pinMode(O0LED, OUTPUT);

  pinMode(O1LED, OUTPUT);

  pinMode(O2LED, OUTPUT);

  pinMode(O3LED, OUTPUT);

  Serial.begin(9600);

}

void loop() {

  unsigned long startTime = millis();

  // Read switch states

  boolean I0State = digitalRead(I0Switch);

  boolean I1State = digitalRead(I1Switch);

  boolean I2State = digitalRead(I2Switch);

  boolean I3State = digitalRead(I3Switch);

  // Calculate LED states using K-map equations

  boolean O0State = (!I0State & I1State & I3State) |

                    (I1State & !I1State & I3State) |

                    (I0State & !I2State & I1State) |

                    (!I3State & !I2State & !I1State & !I0State);

  boolean O1State = (I2State & !I0State) |

                    (!I3State & I1State) |

                    (!I0State & !I1State & I3State) |

                    (I0State & !I2State & I3State);

  boolean O2State = (!I3State & !I1State) |

                    (!I3State & I2State) |

                    (!I2State & !I1State & !I0State);

  boolean O3State = (!I3State & !I2State) |

                    (!I3State & I0State) |

                    (!I1State & I3State & !I0State) |

                    (I1State & I2State & I0State);

  // Update LEDs

  digitalWrite(O0LED, O0State);

  digitalWrite(O1LED, O1State);

  digitalWrite(O2LED, O2State);

  digitalWrite(O3LED, O3State);

  // Print LED states to the serial monitor

  Serial.print(O3State);

  Serial.print(O2State);

  Serial.print(O1State);

  Serial.println(O0State);

  unsigned long endTime = millis();

  Serial.print("Code Execution Time: ");

  Serial.print(endTime - startTime);

  Serial.println(" ms");

}

# Hex Code

## Hex Codes generated for use on Arduino UNO R3

**Arduino Library and C programming Hex Code:**

:100000000C9462000C948A000C948A000C948A0070

:100010000C948A000C948A000C948A000C948A0038

:100020000C948A000C948A000C948A000C948A0028

:100030000C948A000C948A000C948A000C948A0018

:100040000C94C3020C948A000C9433030C940D039B

:100050000C948A000C948A000C948A000C948A00F8

:100060000C948A000C948A000000000024002700F1

:100070002A0000000000250028002B0000000000DE

:1000800023002600290004040404040404040202DA

:100090000202020203030303030301020408102007

:1000A0004080010204081020010204081020000012

:1000B0000008000201000003040700000000000027

:1000C0000000CA0411241FBECFEFD8E0DEBFCDBFB1

:1000D00011E0A0E0B1E0E2E4FAE002C005900D9288

:1000E000A033B107D9F721E0A0E3B1E001C01D9230

:1000F000A63DB207E1F710E0C2E6D0E004C02197C8

:10010000FE010E941905C136D107C9F70E94650397

:100110000C941F050C940000833081F028F481308A

:1001200099F08230A9F008958730A9F08830C9F09D

:100130008430B1F4809180008F7D03C08091800075

:100140008F7780938000089584B58F7784BD08955C

:1001500084B58F7DFBCF8091B0008F778093B00006

:1001600008958091B0008F7DF9CFCF93DF93282F32

:1001700030E0F901E255FF4F8491F901E656FF4F57

:10018000D491F901EA57FF4FC491CC23A1F081111A

:100190000E948C00EC2FF0E0EE0FFF1FE458FF4FA1

:1001A000A591B491EC91ED2381E090E009F480E019

:1001B000DF91CF91089580E090E0FACF1F93CF9325

:1001C000DF93282F30E0F901E255FF4F8491F901C8

:1001D000E656FF4FD491F901EA57FF4FC491CC2363

:1001E000A9F0162F81110E948C00EC2FF0E0EE0F89

:1001F000FF1FEE58FF4FA591B4918FB7F894EC9183

:10020000111108C0D095DE23DC938FBFDF91CF9111

:100210001F910895DE2BF8CFCF93DF9390E0FC0180

:10022000E656FF4F24918A579F4FFC018491882303

:10023000D1F090E0880F991FFC01E859FF4FA5917C

:10024000B491FC01EE58FF4FC591D49161110EC0DD

:100250009FB7F8948C91E22FE0958E238C932881A0

:10026000E223E8839FBFDF91CF9108958FB7F89481

:10027000EC91E22BEC938FBFF6CFAF92BF92CF926F

:10028000DF92EF92FF920F931F93CF93DF936C0156

:100290007B018B01040F151FEB015E01AE18BF0837

:1002A000C017D10759F06991D601ED91FC910190E9

:1002B000F081E02DC6010995892B79F7C501DF9101

:1002C000CF911F910F91FF90EF90DF90CF90BF9053

:1002D000AF900895FC01538D448D252F30E0842F7D

:1002E00090E0821B930B541710F0CF96089501975E

:1002F0000895FC01918D828D981761F0A28DAE0F4B

:10030000BF2FB11D5D968C91928D9F5F9F73928FD1

:1003100090E008958FEF9FEF0895FC01918D828DFD

:10032000981731F0828DE80FF11D858D90E00895CA

:100330008FEF9FEF0895FC01918D228D892F90E022

:10034000805C9F4F821B91098F739927089589E3E1

:1003500091E00E949B0121E0892B09F420E0822F8B

:10036000089580E090E0892B29F00E94A701811177

:100370000C9400000895FC01A48DA80FB92FB11DA5

:10038000A35ABF4F2C91848D90E001968F739927CB

:10039000848FA689B7892C93A089B1898C918370A9

:1003A00080648C93938D848D981306C00288F389A2

:1003B000E02D80818F7D80830895EF92FF920F93CF

:1003C0001F93CF93DF93EC0181E0888F9B8D8C8D01

:1003D00098131AC0E889F989808185FF15C09FB7F5

:1003E000F894EE89FF896083E889F98980818370B8

:1003F000806480839FBF81E090E0DF91CF911F9167

:100400000F91FF90EF900895F62E0B8D10E00F5F87

:100410001F4F0F731127E02E8C8D8E110CC00FB65D

:1004200007FCFACFE889F989808185FFF5CFCE01F5

:100430000E94BB01F1CFEB8DEC0FFD2FF11DE35AB4

:10044000FF4FF0829FB7F8940B8FEA89FB89808178

:100450008062CFCFCF93DF93EC01888D8823B9F0F2

:10046000AA89BB89E889F9898C9185FD03C08081BF

:1004700086FD0DC00FB607FCF7CF8C9185FFF2CF3C

:10048000808185FFEDCFCE010E94BB01E9CFDF91D6

:10049000CF910895FC0101900020E9F73197AF0159

:1004A000481B590BBC0189E391E00C943D018F92EC

:1004B0009F92AF92BF920F931F93CF93DF93CDB7CD

:1004C000DEB7A1970FB6F894DEBF0FBECDBF19A25D

:1004D000423008F44AE08E010F5D1F4F842E912CAC

:1004E000B12CA12CA50194010E94F704E62FB901BB

:1004F000CA01EA30F4F4E05DD801EE938D01232BBC

:10050000242B252B79F790E080E0109719F0CD018E

:100510000E944A02A1960FB6F894DEBF0FBECDBF6F

:10052000DF91CF911F910F91BF90AF909F908F90CF

:100530000895E95CE1CFCF92DF92EF92FF92CF93E3

:10054000DF936B017C016DE289E391E00E94DD01A4

:10055000EC0166277727CB016C197D098E099F096D

:100560004AE00E9457028C0F9D1FDF91CF91FF90B0

:10057000EF90DF90CF900895BC01990F880B990BF5

:100580004AE00C9457021F920F920FB60F9211245B

:100590002F933F938F939F93AF93BF938091350198

:1005A00090913601A0913701B0913801309134011A

:1005B00023E0230F2D3758F50196A11DB11D20937F

:1005C00034018093350190933601A0933701B093A5

:1005D00038018091300190913101A0913201B091A8

:1005E00033010196A11DB11D80933001909331011B

:1005F000A0933201B0933301BF91AF919F918F913E

:100600003F912F910F900FBE0F901F90189526E8E5

:10061000230F0296A11DB11DD2CF1F920F920FB6CC

:100620000F9211242F933F934F935F936F937F9378

:100630008F939F93AF93BF93EF93FF9389E391E0E1

:100640000E94BB01FF91EF91BF91AF919F918F915C

:100650007F916F915F914F913F912F910F900FBEBE

:100660000F901F9018951F920F920FB60F921124A2

:100670002F938F939F93EF93FF93E0914901F09114

:100680004A018081E0914F01F091500182FD1BC031

:100690009081809152018F5F8F7320915301821757

:1006A00041F0E0915201F0E0E75CFE4F958F8093BE

:1006B0005201FF91EF919F918F912F910F900FBE5B

:1006C0000F901F9018958081F4CF789484B5826044

:1006D00084BD84B5816084BD85B5826085BD85B5E6

:1006E000816085BD80916E00816080936E00109264

:1006F00081008091810082608093810080918100DF

:10070000816080938100809180008160809380006F

:100710008091B10084608093B1008091B0008160CD

:100720008093B00080917A00846080937A008091F9

:100730007A00826080937A0080917A008160809351

:100740007A0080917A00806880937A001092C100CC

:1007500060E085E00E940C0160E084E00E940C01F2

:1007600060E083E00E940C0160E082E00E940C01E6

:1007700061E08AE00E940C0161E08BE00E940C01C4

:1007800061E08CE00E940C0161E08DE00E940C01B0

:10079000E0914901F0914A0182E08083E0914501B6

:1007A000F09146011082E0914701F09148018FECF1

:1007B000808310925101E0914D01F0914E0186E04D

:1007C0008083E0914B01F0914C01808180618083B6

:1007D000E0914B01F0914C01808188608083E09131

:1007E0004B01F0914C01808180688083E0914B0146

:1007F000F0914C0180818F7D808380E0C82E80E065

:10080000D82E8FB7F8948090350190903601A09043

:100810003701B09038018FBF85E00E94B5008C0190

:1008200077247394892B09F4712C84E00E94B5001D

:100830007C0166246394892B09F4612C83E00E9477

:10084000B500EC0155245394892B09F4512C82E016

:100850000E94B5009C0171E0892B09F470E0F1E081

:100860000115110509F0F0E0E1E0E114F10409F0EF

:10087000E0E091E0209709F090E0872D8923A1E046

:100880002115310509F0A0E00E291F29AE01402BEA

:10089000512B662D6723CF2FCE2BC623662D6823C1

:1008A000C62B242B352B61E0232B09F060E0C62BEF

:1008B000362D3A23252D2F23232B31E0012B09F050

:1008C00030E0132F1723122B8723182B052D0E2B07

:1008D0000A2381E0452B09F080E0082BD72DD92B86

:1008E000DA237F23E723DE2B76205720D5296C2FB0

:1008F0008AE00E94DE00612F8BE00E94DE00602F04

:100900008CE00E94DE006D2F8DE00E94DE008D2FB6

:1009100090E00E94BC02802F90E00E94BC02812FD8

:1009200090E00E94BC026C2F70E090E080E04AE012

:100930000E94570282E191E00E944A028FB7F89428

:100940004090350150903601609037017090380129

:100950008FBF85E191E00E944A02C301B20168198C

:1009600079098A099B094AE00E9457028BE291E0CB

:100970000E944A0282E191E00E944A02C114D1041D

:1009800009F43FCF0E94A701882309F43ACF0E94BF

:10099000000037CFE9E3F1E01382128288EE93E0A2

:1009A000A0E0B0E084839583A683B78384E091E0E0

:1009B0009183808385EC90E09587848784EC90E038

:1009C0009787868780EC90E0918B808B81EC90E01C

:1009D000938B828B82EC90E0958B848B86EC90E0FD

:1009E000978B868B118E128E138E148E0895A1E232

:1009F0001A2EAA1BBB1BFD010DC0AA1FBB1FEE1F99

:100A0000FF1FA217B307E407F50720F0A21BB30BE3

:100A1000E40BF50B661F771F881F991F1A9469F75F

:100A200060957095809590959B01AC01BD01CF01BB

:100A30000895EE0FFF1F0590F491E02D0994F894AE

:020A4000FFCFE6

:100A420000000000DD013D016A012A029B017901DB

:100A52008D010D0A00436F646520457865637574E6

:100A6200696F6E2054696D653A2000206D73000035

:00000001FF

## Hex Code generated for use on Arduino Mega 2560 Arduino Library and C programming Hex Code:

:100000000C9435010C9466010C9466010C94660105

:100010000C9466010C9466010C9466010C946601C4

:100020000C9466010C9466010C9466010C946601B4

:100030000C9466010C9466010C9466010C946601A4

:100040000C9466010C9466010C9466010C94660194

:100050000C9466010C9466010C9466010C940404E3

:100060000C9466010C9478040C944E040C94660174

:100070000C9466010C9466010C9466010C94660164

:100080000C9466010C9466010C9466010C94660154

:100090000C9466010C9466010C9466010C94660144

:1000A0000C9466010C9466010C9466010C94660134

:1000B0000C9466010C9466010C9466010C94660124

:1000C0000C9466010C9466010C9466010C94660114

:1000D0000C9466010C9466010C9466010C94660104

:1000E0000C9466010C94A7020C94C6010C94AF0109

:1000F0000C94C1010C94AB010C94B5020C94ED026C

:100100000C94B3010C94A1010C9432060C94C70119

:100110000C9493020C94B9010C943A030C9484024D

:100120000C94CB010C9494010C9490010C949B01C1

:100130000C94A5010C9498010C94BD010C948601BB

:100140000C9457020C948C0100002100240027001D

:100150002A002D00300033000101000004010701D6

:100160000A0100002200250028002B002E0031008B

:10017000340002010000050108010B01000020000D

:100180002300260029002C002F003200000100006F

:10019000030106010901050505050705080808080A

:1001A000020202020A0A080804040404010101010F

:1001B000010101010303030303030303040707070A

:1001C0000C0C0C0C0C0C0C0C0202020206060606AF

:1001D000060606060B0B0B0B0B0B0B0B010210207C

:1001E000200808102040102040800201020108046D

:1001F0000201010204081020408080402010080401

:100200000201800402018040201008040201080459

:10021000020101020408102040800102040810209D

:10022000408000000A0B02090C0D0E0807030401B0

:1002300000000000000000000000000000000000BE

:10024000000000000000000000000000000012118B

:10025000100000000000000000000000000000008E

:100260000000000000000000320611241FBECFEF86

:10027000D1E2DEBFCDBF00E00CBF12E0A0E0B2E0F3

:10028000E6E1FDE000E00BBF02C007900D92A03355

:10029000B107D9F722E0A0E3B2E001C01D92A63D6C

:1002A000B207E1F711E0C5E3D1E000E006C0219715

:1002B0000109802FFE010E948106C433D10780E02E

:1002C0000807A9F70E94AE040C9489060C9400005C

:1002D0008150823108F05AC0E82FF0E08827EC58AE

:1002E000FE4F8F4F0C9481069401980186018C017A

:1002F0009001C6019B01A101A501AB01AF01B301B2

:10030000B901BD01C601C101C701CB0180918000C7

:100310008F77809380000895809180008F7DF9CF42

:1003200080918000877FF5CF84B58F7784BD089555

:1003300084B58F7DFBCF8091B0008F778093B00024

:1003400008958091B0008F7DF9CF809190008F77D4

:10035000809390000895809190008F7DF9CF8091D7

:100360009000877FF5CF8091A0008F778093A000C9

:1003700008958091A0008F7DF9CF8091A000877FA4

:10038000F5CF809120018F7780932001089580918F

:1003900020018F7DF9CF80912001877FF5CFCF930A

:1003A000DF93282F30E0F901EE5DFD4F8491F901D4

:1003B000E452FE4FD491F901EA56FE4FC491CC238A

:1003C000A1F081110E946801EC2FF0E0EE0FFF1FF9

:1003D000E458FE4FA591B491EC91ED2381E090E0BB

:1003E00009F480E0DF91CF91089580E090E0FACFAA

:1003F0001F93CF93DF93282F30E0F901EE5DFD4F7F

:100400008491F901E452FE4FD491F901EA56FE4F6E

:10041000C491CC23A9F0162F81110E946801EC2F02

:10042000F0E0EE0FFF1FEE59FE4FA591B4918FB78C

:10043000F894EC91111108C0D095DE23DC938FBFA6

:10044000DF91CF911F910895DE2BF8CFCF93DF93EB

:1004500090E0FC01E452FE4F24918A569E4FFC012D

:1004600084918823D1F090E0880F991FFC01E85B0C

:10047000FE4FA591B491FC01EE59FE4FC591D49168

:1004800061110EC09FB7F8948C91E22FE0958E23F6

:100490008C932881E223E8839FBFDF91CF91089559

:1004A0008FB7F894EC91E22BEC938FBFF6CFAF921D

:1004B000BF92CF92DF92EF92FF920F931F93CF9351

:1004C000DF936C017B018B01040F151FEB015E01B3

:1004D000AE18BF08C017D10759F06991D601ED9148

:1004E000FC910190F081E02DC6011995892B79F7D7

:1004F000C501DF91CF911F910F91FF90EF90DF9099

:10050000CF90BF90AF900895FC01538D448D252F5F

:1005100030E0842F90E0821B930B541710F0CF969D

:10052000089501970895FC01918D828D981761F0CF

:10053000A28DAE0FBF2FB11D5D968C91928D9F5FE6

:100540009F73928F90E008958FEF9FEF0895FC01C5

:10055000918D828D981731F0828DE80FF11D858D78

:1005600090E008958FEF9FEF0895FC01918D228D0B

:10057000892F90E0805C9F4F821B91098F73992790

:10058000089589E392E00E94B50221E0892B09F4E5

:1005900020E0822F0895FC01A48DA80FB92FB11D72

:1005A000A35ABF4F2C91848D90E001968F739927A9

:1005B000848FA689B7892C93A089B1898C91837087

:1005C00080648C93938D848D981306C00288F38980

:1005D000E02D80818F7D80830895EF92FF920F93AD

:1005E0001F93CF93DF93EC0181E0888F9B8D8C8DDF

:1005F00098131AC0E889F989808185FF15C09FB7D3

:10060000F894EE89FF896083E889F9898081837095

:10061000806480839FBF81E090E0DF91CF911F9144

:100620000F91FF90EF900895F62E0B8D10E00F5F65

:100630001F4F0F731127E02E8C8D8E110CC00FB63B

:1006400007FCFACFE889F989808185FFF5CFCE01D3

:100650000E94CB02F1CFEB8DEC0FFD2FF11DE35A81

:10066000FF4FF0829FB7F8940B8FEA89FB89808156

:100670008062CFCFCF93DF93EC01888D8823B9F0D0

:10068000AA89BB89E889F9898C9185FD03C080819D

:1006900086FD0DC00FB607FCF7CF8C9185FFF2CF1A

:1006A000808185FFEDCFCE010E94CB02E9CFDF91A3

:1006B000CF91089580E090E0892B29F00E94C1023B

:1006C00081110E94000080E090E0892B49F080E0D9

:1006D00090E0892B29F00E94000081110E94000007

:1006E00080E090E0892B49F080E090E0892B29F0B0

:1006F0000E94000081110E94000080E090E0892BA0

:1007000049F080E090E0892B29F00E9400008111DF

:100710000C9400000895FC0101900020E9F7319746

:10072000AF01481B590BBC0189E392E00C945702BE

:100730008F929F92AF92BF920F931F93CF93DF93AD

:10074000CDB7DEB7A1970FB6F894DEBF0FBECDBF11

:1007500019A2423008F44AE08E010F5D1F4F842E2B

:10076000912CB12CA12CA50194010E945F06E62FCB

:10077000B901CA01EA30F4F4E05DD801EE938D01CD

:10078000232B242B252B79F790E080E0109719F08C

:10079000CD010E948B03A1960FB6F894DEBF0FBE69

:1007A000CDBFDF91CF911F910F91BF90AF909F90E0

:1007B0008F900895E95CE1CFCF92DF92EF92FF92A4

:1007C000CF93DF936B017C016DE289E392E00E949D

:1007D000ED02EC0166277727CB016C197D098E09A4

:1007E0009F094AE00E9498038C0F9D1FDF91CF91D3

:1007F000FF90EF90DF90CF900895BC01990F880B88

:10080000990B4AE00C9498031F920F920FB60F9227

:1008100011242F933F938F939F93AF93BF93809116

:10082000350290913602A0913702B0913802309192

:10083000340223E0230F2D3758F50196A11DB11D79

:10084000209334028093350290933602A0933702AE

:10085000B09338028091300290913102A09132021F

:10086000B09133020196A11DB11D80933002909387

:100870003102A0933202B0933302BF91AF919F91A6

:100880008F913F912F910F900FBE0F901F90189551

:1008900026E8230F0296A11DB11DD2CF1F920F9201

:1008A0000FB60F9211240BB60F922F933F934F93D5

:1008B0005F936F937F938F939F93AF93BF93EF93C8

:1008C000FF9389E392E00E94CB02FF91EF91BF91E9

:1008D000AF919F918F917F916F915F914F913F91D8

:1008E0002F910F900BBE0F900FBE0F901F90189579

:1008F0001F920F920FB60F9211240BB60F922F93E7

:100900008F939F93EF93FF93E0914902F0914A02F6

:100910008081E0914F02F091500282FD1DC09081D4

:10092000809152028F5F8F7320915302821741F0A2

:10093000E0915202F0E0E75CFD4F958F8093520208

:10094000FF91EF919F918F912F910F900BBE0F9080

:100950000FBE0F901F9018958081F2CF789484B5C8

:10096000826084BD84B5816084BD85B5826085BDAB

:1009700085B5816085BD80916E00816080936E0039

:10098000109281008091810082608093810080912B

:1009900081008160809381008091800081608093DC

:1009A00080008091B10084608093B1008091B0009C

:1009B00081608093B000809191008260809391006B

:1009C000809191008160809391008091900081607E

:1009D000809390008091A10082608093A10080911B

:1009E000A10081608093A1008091A000816080932C

:1009F000A0008091210182608093210180912101DA

:100A00008160809321018091200181608093200189

:100A100080917A00846080937A0080917A0082606D

:100A200080937A0080917A00816080937A0080912F

:100A30007A00806880937A001092C10060E085E0BF

:100A40000E94260260E084E00E94260260E083E0CB

:100A50000E94260260E082E00E94260261E08AE0B5

:100A60000E94260261E08BE00E94260261E08CE099

:100A70000E94260261E08DE00E942602E091490278

:100A8000F0914A0282E08083E0914502F0914602B3

:100A90001082E0914702F09148028FEC808310921F

:100AA0005102E0914D02F0914E0286E08083E09188

:100AB0004B02F0914C02808180618083E0914B0277

:100AC000F0914C02808188608083E0914B02F0912C

:100AD0004C02808180688083E0914B02F0914C024F

:100AE00080818F7D80838FB7F894C0903502D0903D

:100AF0003602E0903702F09038028FBF85E00E9406

:100B0000CF01EC0199249394892B09F4912C84E072

:100B10000E94CF018C0188248394892B09F4812CB5

:100B200083E00E94CF015C0177247394892B09F440

:100B3000712C82E00E94CF01AC0121E0892B09F4E5

:100B400020E071E0209709F070E031E00115110517

:100B500009F030E091E0A114B10409F090E0E92D32

:100B6000E92381E04115510509F080E00C2B1D2B94

:100B7000A02AB12A682D6223C72FC32BC623682D54

:100B80006E23C62B4A295B2961E0452B09F060E002

:100B9000C62B582D5823472D4723452B51E0012BB9

:100BA00009F050E0152F1223142BE2231E2B072DE2

:100BB000032B082341E0AB2809F040E0042B9929DE

:100BC0009823D22FD723D323D92B98207920D72924

:100BD0006C2F8AE00E94F801612F8BE00E94F801DF

:100BE000602F8CE00E94F8016D2F8DE00E94F801CB

:100BF0008D2F90E00E94FD03802F90E00E94FD0366

:100C0000812F90E00E94FD036C2F70E090E080E067

:100C10004AE00E94980382E192E00E948B038FB722

:100C2000F8948090350290903602A0903702B090F0

:100C300038028FBF85E192E00E948B03C501B401A9

:100C40006C197D098E099F094AE00E9498038BE286

:100C500092E00E948B0382E192E00E948B030E944B

:100C60005A0341CFE9E3F2E01382128288EE93E067

:100C7000A0E0B0E084839583A683B78384E092E00C

:100C80009183808385EC90E09587848784EC90E065

:100C90009787868780EC90E0918B808B81EC90E049

:100CA000938B828B82EC90E0958B848B86EC90E02A

:100CB000978B868B118E128E138E148E0895A1E25F

:100CC0001A2EAA1BBB1BFD010DC0AA1FBB1FEE1FC6

:100CD000FF1FA217B307E407F50720F0A21BB30B11

:100CE000E40BF50B661F771F881F991F1A9469F78D

:100CF00060957095809590959B01AC01BD01CF01E9

:100D00000895EE0FFF1F881F8BBF0790F691E02D0F

:060D10001994F894FFCFD6

:100D160000000000ED02570284023A03B502930276

:100D2600A7020D0A00436F646520457865637574F4

:100D3600696F6E2054696D653A2000206D7300005E

:00000001FF

# Result Analysis and Comparison Among The Boards Used

## Hardware Comparison Among Boards Used

|  |  |  |
| --- | --- | --- |
| **Hardware Specifications** | **Arduino Uno R3** | **Arduino Mega 2560** |
| **Micro controller Used** | ATmega328p | ATmega2560 |
| **Flash Memory** | 32 KB | 256KB |
| **EEPROM** | 1 KB | 4KB |
| **Clock Frequency** | 16 MHz | 16 MHz |
| **PWM pins** | 6 | 15 |
| **Analog Pins** | 6 | 16 |
| **Digital PIns** | 14 | 54 |
| **SRAM** | 2 KB | 8 KB |
| **Power** | 5V | 5V |

*Table : Hardware Specification of Arduino UNO R3 and Arduino Mega 2560*

## Power Consumption Comparison:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **InPut Bits** | **OutPut Bits** | **Current ( mA)** | **Voltage (V)** | **Power (mW)** |
| 0000 | 0100 | 365 | 5 | 1825 |
| 1000 | 1101 | 365 | 5 | 1825 |
| 0100 | 0010 | 243 | 5 | 1215 |
| 1100 | 1100 | 243 | 5 | 1215 |
| 0010 | 1011 | 243 | 5 | 1215 |
| 1010 | 1011 | 122 | 5 | 610 |
| 0110 | 1001 | 243 | 5 | 1215 |
| 1110 | 0000 | 243 | 5 | 1215 |
| 0001 | 0101 | 243 | 5 | 1215 |
| 1001 | 0011 | 243 | 5 | 1215 |
| 0101 | 1010 | 243 | 5 | 1215 |
| 1101 | 1111 | 122 | 5 | 610 |
| 0011 | 0100 | 365 | 5 | 1825 |
| 1011 | 1000 | 243 | 5 | 1215 |
| 0111 | 1110 | 356 | 5 | 1825 |
| 1111 | 0100 | 122 | 5 | 610 |

Table:Current Voltage and Power use for **Arduino UNO R3**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **InPut Bits** | **OutPut Bits** | **Current ( mA)** | **Voltage (V)** | **Power (mW)** |
| 0000 | 0100 | 365 | 5 | 1825 |
| 1000 | 1101 | 365 | 5 | 1825 |
| 0100 | 0010 | 243 | 5 | 1215 |
| 1100 | 1100 | 243 | 5 | 1215 |
| 0010 | 1011 | 243 | 5 | 1215 |
| 1010 | 1011 | 122 | 5 | 610 |
| 0110 | 1001 | 243 | 5 | 1215 |
| 1110 | 0000 | 243 | 5 | 1215 |
| 0001 | 0101 | 243 | 5 | 1215 |
| 1001 | 0011 | 243 | 5 | 1215 |
| 0101 | 1010 | 243 | 5 | 1215 |
| 1101 | 1111 | 122 | 5 | 610 |
| 0011 | 0100 | 365 | 5 | 1825 |
| 1011 | 1000 | 243 | 5 | 1215 |
| 0111 | 1110 | 356 | 5 | 1825 |
| 1111 | 0100 | 122 | 5 | 610 |

Table: Current Voltage and Power use for **Arduino Mega 2560**

Both Arduino UNO R3 and Mega show the same current draw, voltage, and Power consumption since both use the same number of components and resistors connected to it and both have same 5V power input via DC Generator. Increasing supplied power increases the current drawn.

## Memory Size and Execution Time Comparison

|  |  |  |
| --- | --- | --- |
|  | **Arduino UNO R3** | **Arduino Mega 2560** |
| **C Programming** | **C Programming** |
| **Program Memory Size** | 2674 bytes | 3398 bytes |
| **Local Variable Size Left** | 1834 bytes | 7978 bytes |
| **Global Variable Size** | 214 bytes | 214 bytes |
| **Program Execution Time** | 6-7 milliseconds | 6-7 milliseconds |

Table: Program memory Size, Variable Size and Program Execution Time using

**Arduino Uno R3** and **Arduino Mega 2560**

The program, when uploaded to an Arduino UNO R3, consumes 2674 bytes of program memory out of a maximum available of 32256 bytes. This represents only 8% of the program memory used. Additionally, the program uses 214 bytes for global variables out of a total of 2048 bytes allocated for variables. In terms of local variables, it consumes 1834 bytes.

In the case of an Arduino Mega 2560, the program occupies 3398 bytes of program memory out of a much larger 253952 bytes available, making up just 1% of the program memory used. It also utilizes 214 bytes for global variables out of an 8192-byte allocation, leaving 7978 bytes for local variables.

Furthermore, when utilizing C programming, the code runs with an execution time of 6-7 milliseconds. In this context, the program execution time refers to the duration from the moment the switch state is read to when the LED is illuminated.

# 10 Conclusion;

The main objective of this project was to assess the power consumption of different microcontroller boards, specifically the Arduino Uno R3 and Arduino Mega 2560, in a unique scenario involving 4-bit logic inputs.

The results of this experiment demonstrate that coding in C programming is notably more convenient, primarily due to the extensive libraries and functions available. In both cases, with both boards, the program's memory footprint is smaller, and it executes more quickly.

Based on these findings, it can be inferred that the data and outcomes of this project are dependable and trustworthy.

.

# Questions and Answers

## Arduino UNO

### What is the clock frequency of the micro controller used?

Answer: 16 MHz.

### What is the data bus width of the micro controller used?

Answer: 8 bit.

### What is the size of your hex file generated?

Answer: For default programming- 8KB, for Register Level Programming- 6KB

### Can the project be implemented by using interrupt?

Answer: Yes. The Arduino UNO pins enable the use of interrupts. External interrupts can be employed on digital pins 2 and 3, whereas Pin Change interrupts are available on a broader range of pins, including any of the 20 pins encompassing A0 to A5 and D0 to D13.

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

Answer: Yes. An infinite loop is essential for the primary routine. This is because we must continually supply input using switches, monitor the corresponding output, and continuously track the execution time for each input and output. Without an infinite loop, the program would run only once and then stop.

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

Answer: Since we are using SPDT switch, one terminal connects to ground and the other connects to input pins. There is no difference between level trigger and edge trigger in this project. The toggling switch provides either an ON-1(ON/HIGH) signal or ON-2 (OFF/LOW) signal.

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

Answer: The project involves the creation of an encryption system that transforms data from its input to output by applying a specific encryption algorithm. In this system, a 4-bit input corresponds to a distinct 4-bit output, effectively encrypting the original input data.

## Arduino Mega 2560

### What is the clock frequency of the micro controller used?

Answer: 16 MHz.

### What is the data bus width of the micro controller used?

Answer: 8 bit.

### What is the size of your hex file generated?

Answer: For default programming- 10KB, for Register Level Programming- 7KB

### Can the project be implemented by using interrupt?

Answer: Yes, the Arduino Mega 2560 accommodates the use of interrupts. It provides the capability for up to 6 external interrupts, which can be utilized on digital pins 2, 3, 18, 19, 20, and 21. Similar to the UNO, Pin Change Interrupts can be employed on all available pins.

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

Answer: Yes, The primary routine should be structured as an endless loop. This is essential because we must consistently deliver input using switches, continuously monitor the corresponding output, and continually track the execution time for each input and output. Without an infinite loop, the program would execute only once and then come to a halt.

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

Answer: As we are utilizing a Single Pole Double Throw (SPDT) switch, one terminal links to ground, while the other connects to input pins. In this project, there is no distinction between a level-triggered and edge-triggered approach. The toggle switch offers either an ON-1 (ON/HIGH) signal or an ON-2 (OFF/LOW) signal.

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

Answer: The project involves an encryption circuit that takes input data and applies an encryption algorithm, resulting in a distinct 4-bit output corresponding to the original 4-bit input, effectively encrypting the provided information.

## 12.References:

1. Materials provided by Dr. Dihan Md. Nuruddin Hassan (DMH).
2. Differentiate between arduino uno and arduino mega - Semiconductor for You (semiconductorforu.com)
3. Tinkercad + Arduino Lesson 4: Blink an LED (DDR and PORT registers) - YouTube
4. Arduino Uno Vs Nano Vs Mega, Pinout, and technical Specifications (electroniclinic.com)

6. Arduino Comparison Guide - SparkFun Learn

## 13.Resources:

1. Project Materials Drive link:

[https://drive.google.com/drive/folders/1BPo7y-](https://drive.google.com/drive/folders/1BPo7y-ts3qASD2YVE2BX_ll3AU37RCXs?usp=drive_link) ts3qASD2YVE2BX\_ll3AU37RCXs?usp=drive\_link

1. Project Demonstration Video Drive link:

[https://drive.google.com/drive/folders/16KjSFGUHJxN5Bwj6nfHyR60\_](https://drive.google.com/drive/folders/16KjSFGUHJxN5Bwj6nfHyR60_ToxIYVUd?usp=drive_link) ToxIYVUd?usp=drive\_link