IOT AND APPLICATION (3160716)

VVP ENGINEERING COLLEGE

SUBMITTED BY: **DISHEN MAKWANA**

Department of Computer Engineering

180470107035

G2

V. V. P. Engineering College, Rajkot

Department of Computer Engineering

Vision of the Department

- Transforming students into globally efficient professionals with moral values.

Mission of the Department

- To provide a strong foundation of computer engineering through effective teaching learning process.
- To enhance industry linkage & alumni network for better placement and real-world exposure.
- To provide various opportunities & platforms for all round development of students & encourage them for value-based practices.

V. V. P. Engineering College, Rajkot

Department of Computer Engineering

Course Outcomes

IOT and Applications (3160716)

STUDENTS WILL BE ABLE TO

1) Explain the Architecture, Basic protocols of IOT and its application.
2) Develop basic applications using Arduino and Raspberry Pi.
3) Demonstrate concepts of IOT with cloud and Fog computing.
4) Describe Security issues of IOT.

V. V. P. Engineering College
Rajkot

CERTIFICATE

This is Certify that

Mr. **DISHEN MAKWANA** Enrollment No **180470107035** Branch **Computer Engineering** Semester **6th** has satisfactory completed the course in the subject **IOT and Application** within the four walls of V. V. P. Engineering College, Rajkot.

Date of Submission: 01/05/2021

Dr. Kamal Sutaria
Staff in charge

INDEX

Practical Name	page
1) Study of Arduino IDE and installation .	**6**
2) Study ISIS Protuous circuit simulation tool for arduino and installation.	**10**
3) Study arduino UNO architecture and Write arduino program to turn on and off the LED with digital pin 13 and do simulation using ISIS Proteus.	**22**
4) **(a)** Write arduino code to turn on and off 8 LEDs with digital pins D0 to D7 of arduino UNO. **(b)** Write arduino code to turn ON and OFF 4 LEDs (LED0 ….LED7) one at a time . (First 4 LED ON and LAST 4 LED OFF after 1 second First 4 LED OFF and Last 4 LED ON). **(c)** Write arduino code to Turn ON and OFF LEDs alternate from left to right(LED0 to LED7). **(d)** Write arduino code to Turn ON and OFF LEDs from right to left (LED7 to LED0). **(e)** Write arduino code to Turn ON and OFF LED in left and right side pattern. (ON LED1 & 8 after a second LED2 & 7 after a second LED3 & 6 after a second LED 4 & 5). **(f)** Write arduino code for Dancing LED pattern. (1 LED Toggle ,8 LEDs Toggle , 4 LED ON and OFF , Alternat LED pattern , Left side pattern , right side pattern and left and right side pattern.	**25**
5) Write arduino Program to control LED with digital pin 13 using PUSH button and do simulation using ISIS Proteus.	**34**
6) **(a)** Write arduino program to print 0 to 9 on Seven Segment display and do simulation using ISIS Proteus. **(b)** Write arduino program to print 0 to 99 on 2 DIGIT Seven Segment display and do simulation using ISIS Proteus	**36**
7) **(a)** Write arduino program to display your name on LCD screen and do simulation using ISIS proteus. **(b)** Write arduino program to display numbers from 0 to 9on LCD Screen and do simulation using ISIS proteus.	**39**

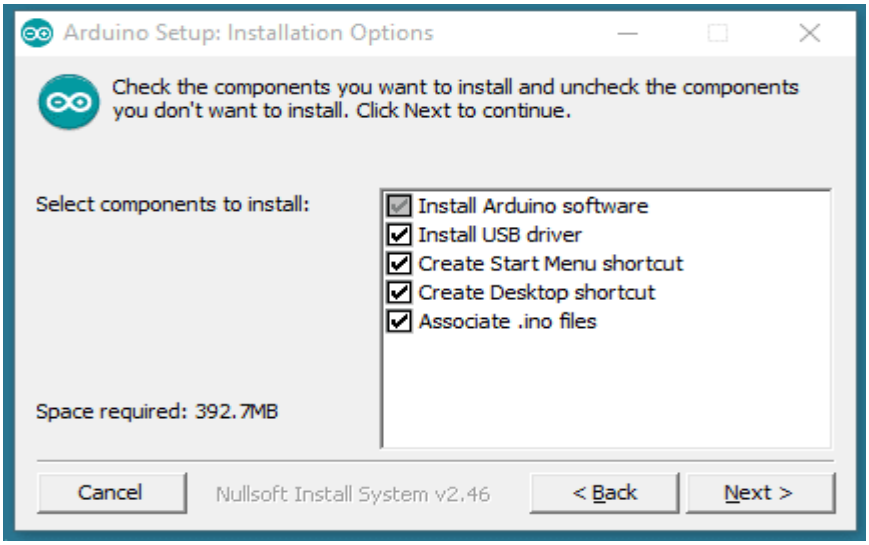
(8) **(a)** Write arduino code to print "Hello World" on serial monitor and do simulation on ISIS Proteus. **(b)** Write arduino code to receive data serially from user and display LED based on users input (from 1 to 8). (Example:- if user enter 4 , LED 4 should blink. **(c)** Write arduino code to transmit and receive data serially. **(d)** Write arduino code to transmit "Good morning" serially if switch is pressed and transmit "Bad Morning" if switch is open , do simulation using proteus.	**42**
(9) (a) Write arduino code to transmit "Hello World – Code to demonstrate BT Communication" string on serial monitor using Bluetooth. Do simulation on ISIS Proteus. **(b)** Write arduino code to blink LED using Bluetooth communication and do simulation on ISIS Proteus. **(C)** Write arduino code to blink LED based on User's choice using Bluetooth communication and do simulation on ISIS Proteus	**47**
(10) (a) Write arduino code to rotate Stepper Motor and do interfacing & simulation with arduino on ISIS Proteus. **(b)** Write arduino code to make LED ON based on Capacitive touch sensor. Do interfacing & simulation with arduino on ISIS Proteus. **(c)** Write arduino code to make Buzzer ON based on Flame sensor . Do interfacing & simulation with arduino on ISIS Proteus.	**51**
(11) (a) Write arduino code to make LED and Buzzer ON based on Vibration sensor. Do interfacing & simulation with arduino on ISIS Proteus. **(b)** Write arduino code to make LED/Buzzer ON based on GAS sensor. Do interfacing & simulation with arduino on ISIS Proteus. **(c)** Write arduino code to make LED/Buzzer ON based on Flex sensor. Do interfacing & simulation with arduino on ISIS Proteus. **(d)** Write arduino code to make LED/Buzzer ON based on PIR sensor. Do interfacing & simulation with arduino on ISIS Proteus.	**55**
(12) (a) Write arduino code to send SMS using GSM module and do simulation using ISIS proteus. **(b)** Write arduino code to print current location using GPS. Do interfacing & simulation with arduino on ISIS Proteus.	**60**
(13) Create arduino based IOT project for Password Based Door looking system.	**64**
(14) Create arduino based IOT project for smart street light	**72**

Assignment - 1

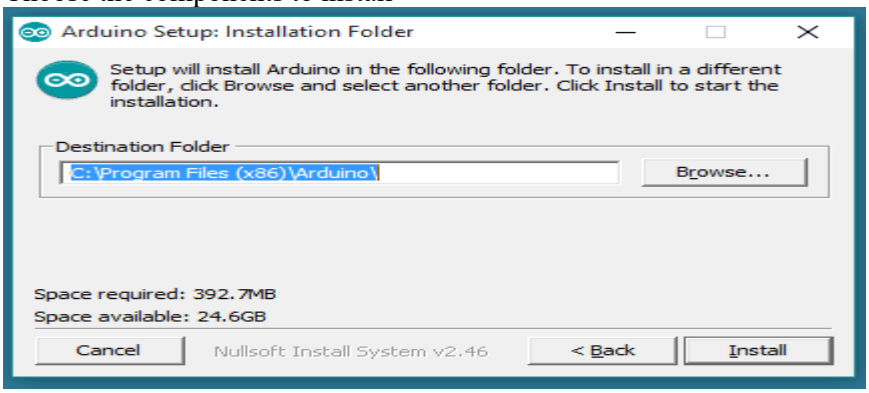
1) Study of Arduino IDE and installation.

Download the Arduino Software (IDE)
Get the latest version from the download page. You can choose between the Installer (.exe) and the Zip packages. We suggest you use the first one that installs directly everything you need to use the Arduino Software (IDE), including the drivers. With the Zip package you need to install the drivers manually. The Zip file is also useful if you want to create a portable installation.

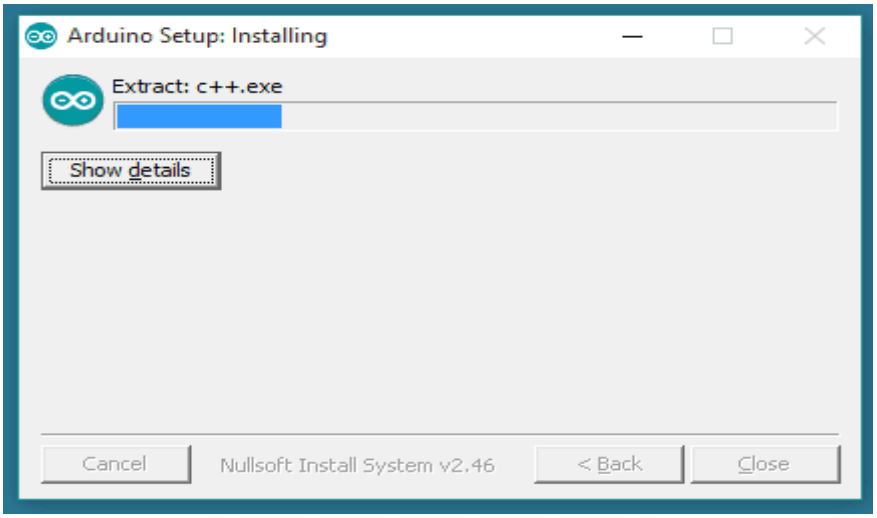
When the download finishes, proceed with the installation and please allow the driver installation process when you get a warning from the operating system.



Choose the components to install



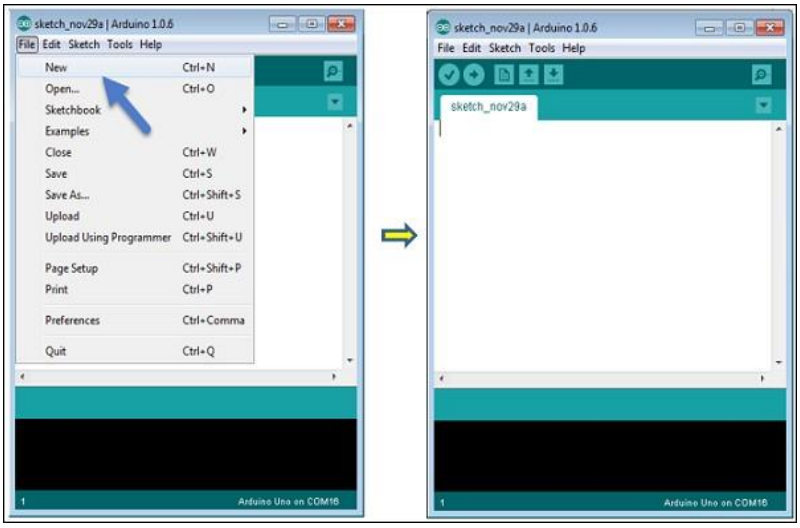
Choose the installation directory (we suggest to keep the default one)

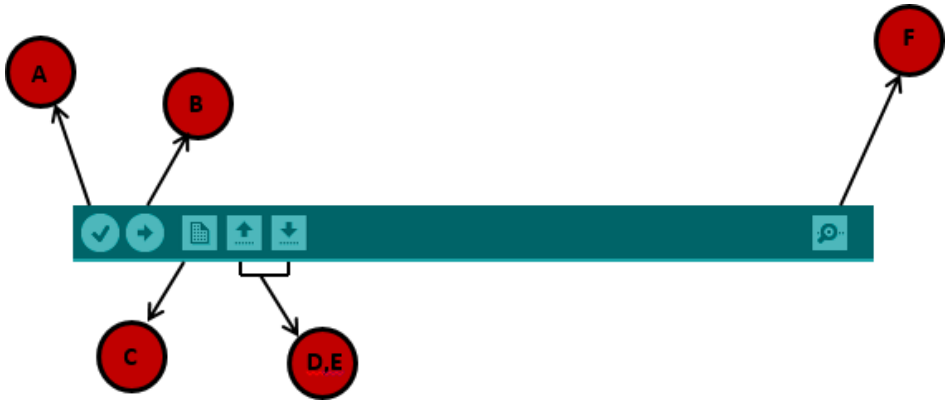


The process will extract and install all the required files to execute properly the Arduino Software (IDE).

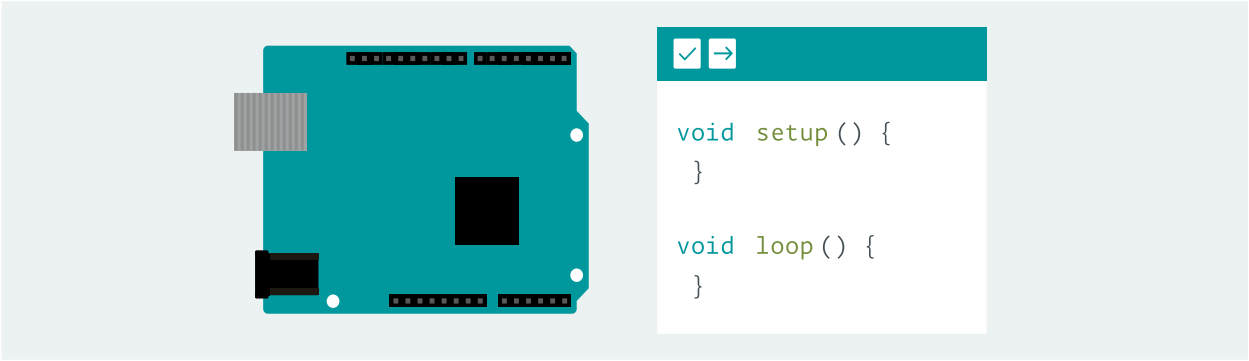
Arduino is an open-source electronics platform based on easy-to-use hardware and software. Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online. You can tell your board what to do by sending a set of instructions to the microcontroller on the board. To do so you use the Arduino programming language (based on Wiring), and the Arduino Software (IDE), based on Processing.

- Go to the Arduino download page and download the latest version of the Arduino software for Windows.
- When the download is finished, un-zip it and open up the Arduino folder to confirm that yes, there are indeed some files and sub-folders inside. The file structure is important so don't be moving any files around unless you really know what you're doing.
- Power up your Arduino by connecting your Arduino board to your computer with a USB cable (or FTDI connector if you're using an Arduino pro). You should see an LED labeled 'ON' light up.





A − Used to check if there is any compilation error. B − Used to upload a program to the Arduino board. C − Shortcut used to create a new sketch.
D − Used to directly open one of the example sketch.
E − Used to save your sketch.
F − Serial monitor used to receive serial data from the board and send the serial data to the board.
Now, simply click the "Upload" button in the environment. Wait a few seconds; you will see the RX and TX LEDs on the board, flashing. If the upload is successful, the message "Done uploading" will appear in the status bar.



Microcontrollers

- Digital Pins: How the pins work and what it means for them to be configured as inputs or outputs.
- Analog Input Pins: Details about the analog-to-digital conversion and other uses of the pins.
- PWM: How the analogWrite() function simulates an analog output using pulse-width modulation.
- Memory: The various types of memory available on the Arduino board.

Programming

- Language Reference: Discover the what each term of the Arduino code language means.
- Variables: Understand how to define and use variables in a Sketch.
- Functions: Learn how to define and use functions in a Sketch.
- Library: Using and installing Arduino Libraries.
- Cores: Need to add a new board to the Arduino Software? Install the relate core and manage it.
- Vidor HDL basics: Learn the basics of FPGAs programming language.

Assignment - 2

2) Study ISIS Protuous circuit simulation tool for arduino and installation.

Proteus is famous electric circuit simulation software. You can easily simulate any electronics circuit in this software. Not only that you can also analyze the data using some tools like oscilloscope, voltmeter, ammeter, signal analyzer etc. The software was developed by Labcenter Electronics. Though there are also a lot of simulation software are available but almost all of them Labcenter Proteus Simulation Software is the most popular. Do you know why this simulation software is very popular to engineers and circuit lover? Because it has some unique feature like this:

1. The easy and user friendly environment

2. A large collection of components and library

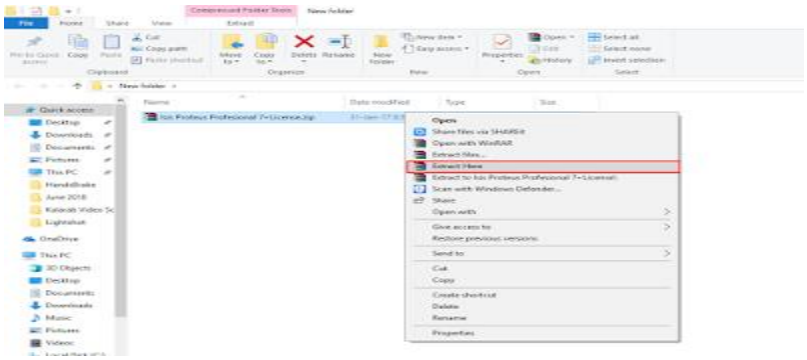
3. Real time simulation with data analysis.

4. The most important and unique feature of this software is, you can convert your designed circuit to pcb as like as professionals.

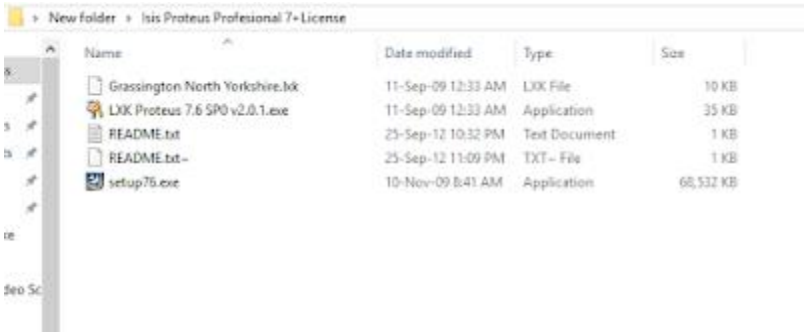
If you have downloaded the software successfully than you are ready to install proteus 7 professional.

Procedure of Installation of Proteus 7 professional:

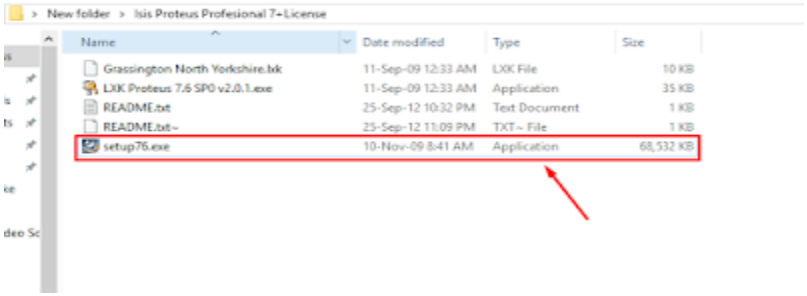
Step-1: Unzip the rar file you have downloaded from the above link. For this you must have unzip software (ex: Winrar) installed in your computer. Now right-click on the rar file and select extract here. The file will automatically unzip.



If you have successfully unzip proteus rar file than you will see 5 files like this:

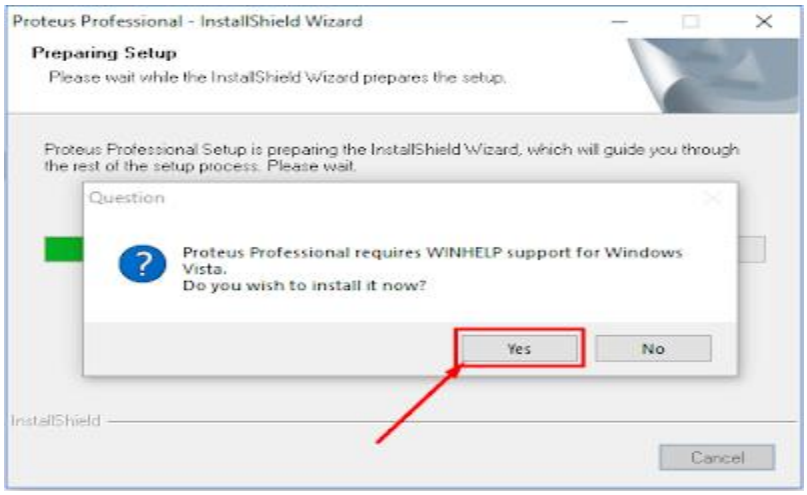


Step-2: Now double click on the stup76.exe file.

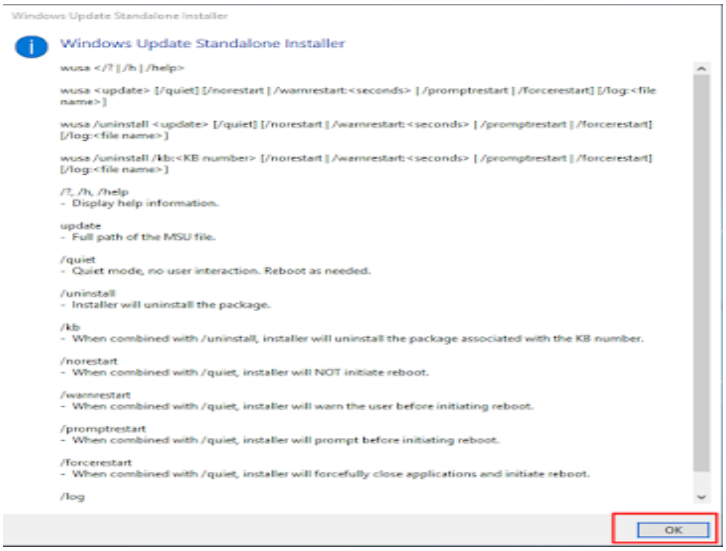


You will see a popup window asking you do you want to install this? Click Yes.

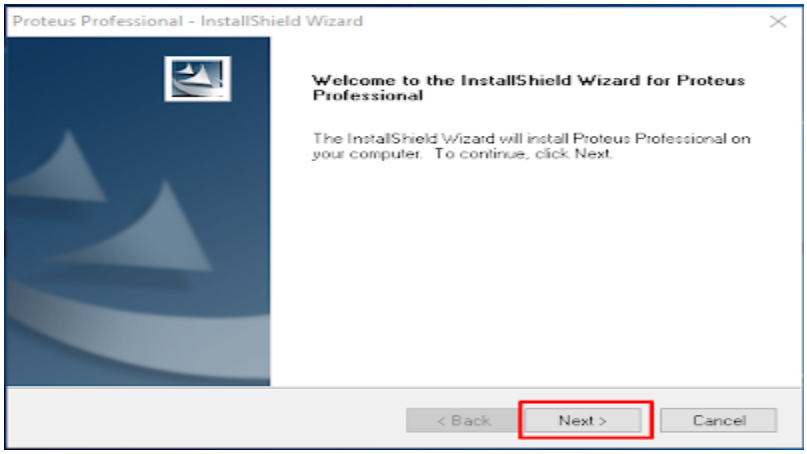
Then a new dialogue box will open. Click on the yes tab again.



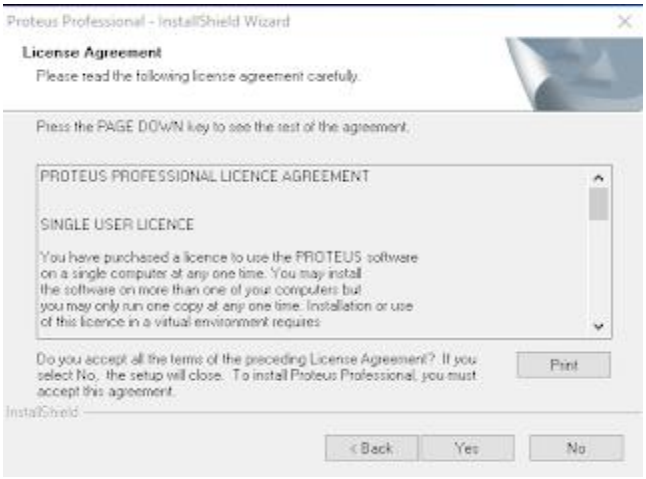
Now a Dialogue box name ‘Windows Updated Standalone Installer’ will open. Just click on OK.



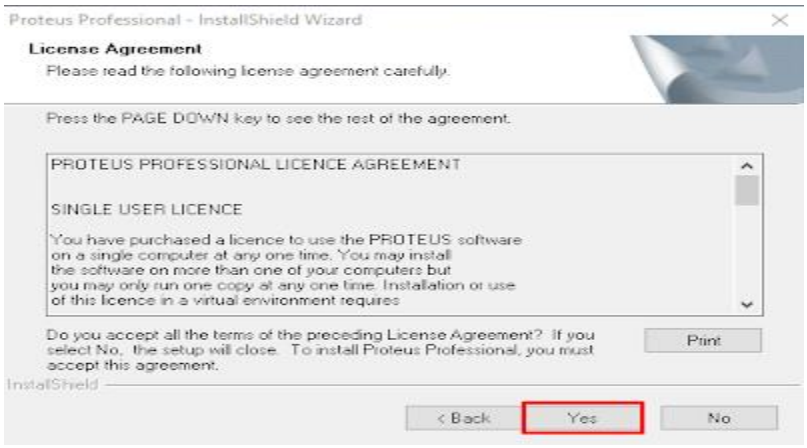
Click on Next.



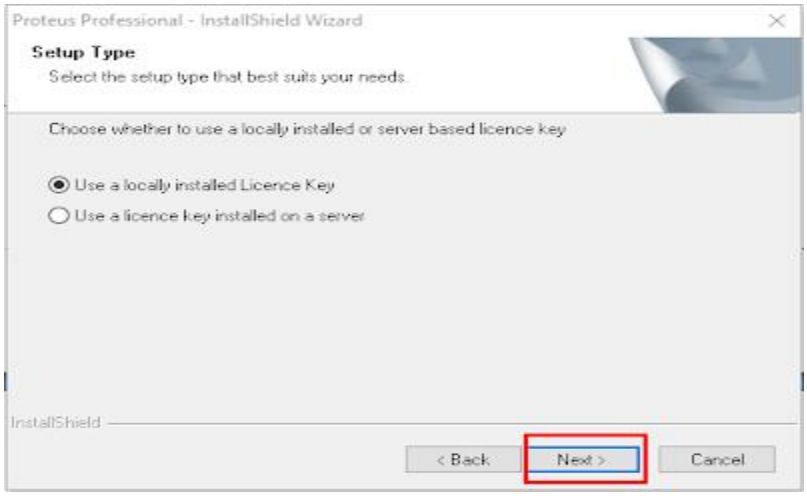
Click on Yes



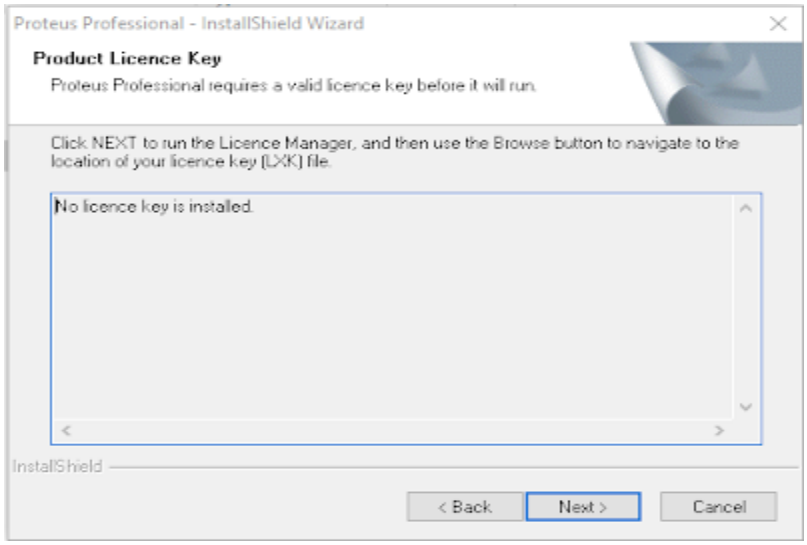
Again Click on Yes

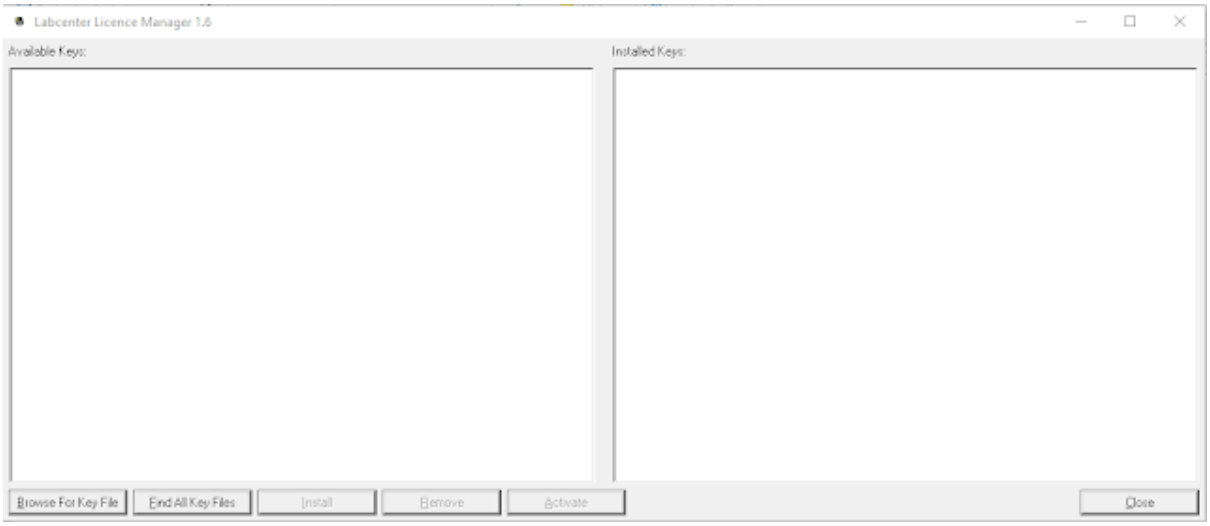


Now you will see a dialogue box with the message ‘No License Key is Installed’. So that we need to install a license key which I have already provided in the zip file. Here just click on Next.

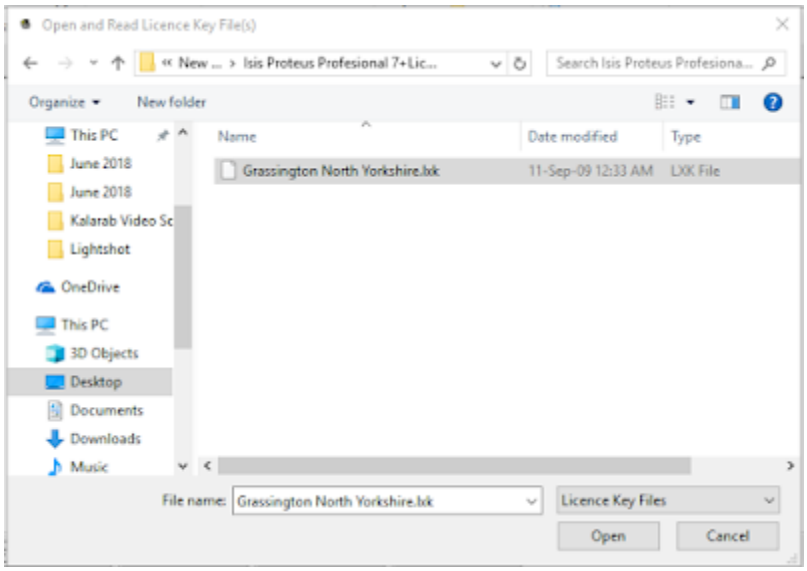


A new window will open with the name ‘Labcenter License Manager’. Here we should now select our license file. For this just click on ‘Browse For Key File’.

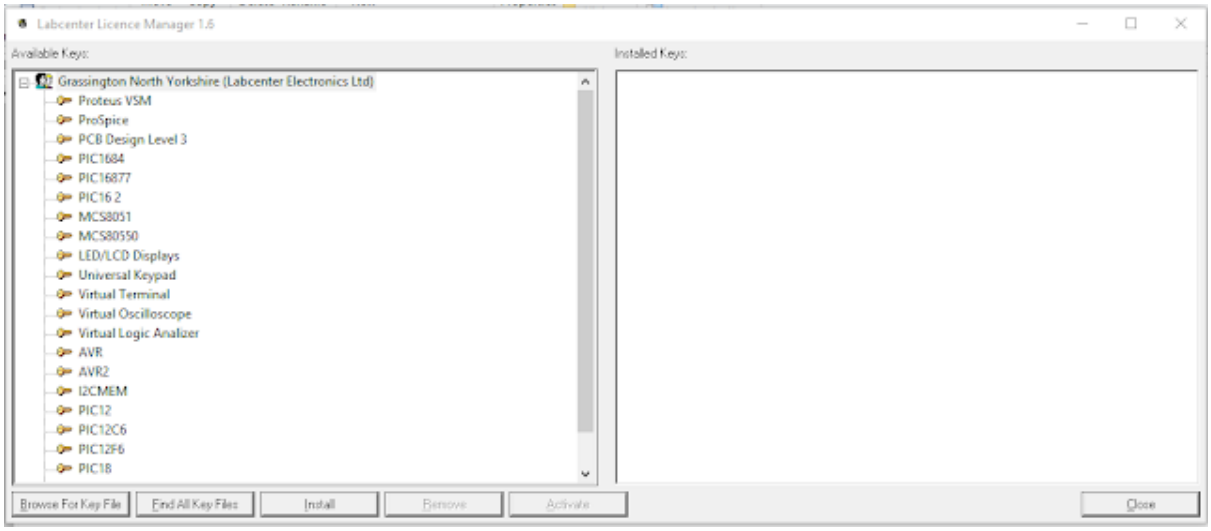




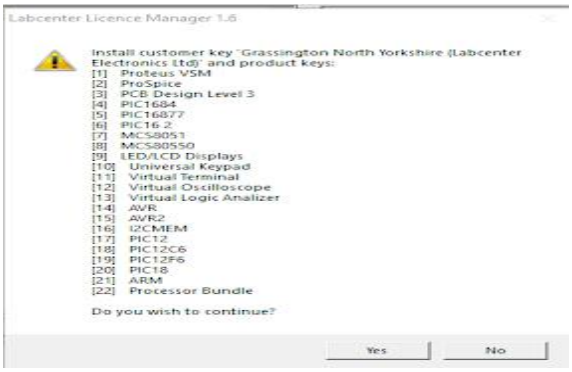
Now Browse the folder where you have extracted the zip file. You will see a file name ‘Grassington North Yorkshire’ inside the folder. Select the file and click on



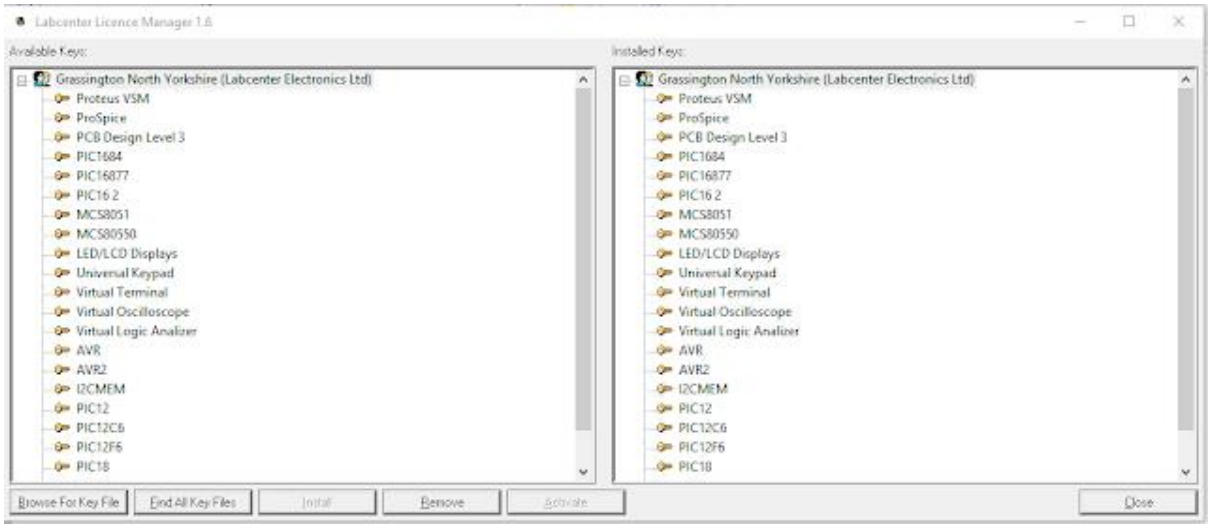
The License file is now showing under the Available Keys section. Now just click on ‘Install’ to install the license key.



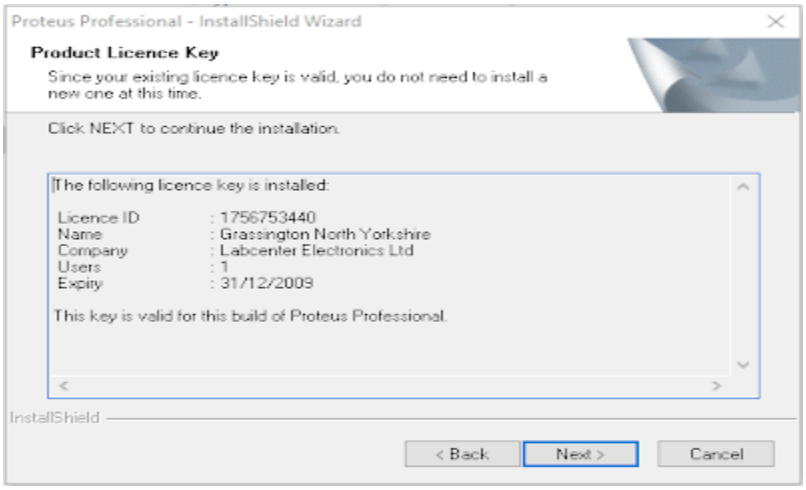
A dialogue box will open. Just click on Yes.



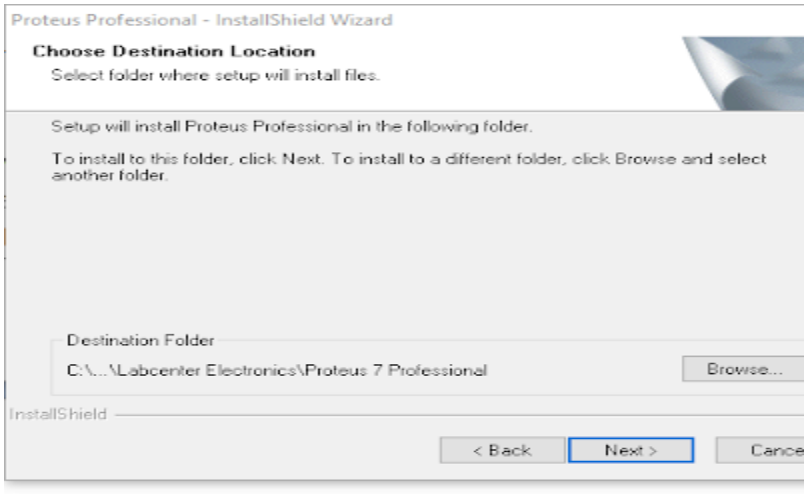
The license file will be installed. Now just click on ‘Close’.



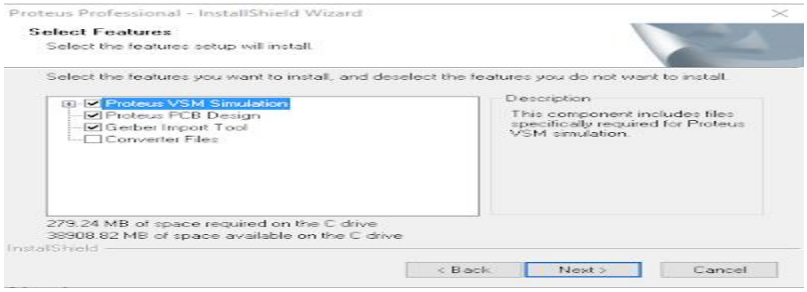
Ok. Now we will returned the previous step and now we will see that 'No License Key is Installed' messages is replaced with 'The following License key is installed'. Now, just click on Next.



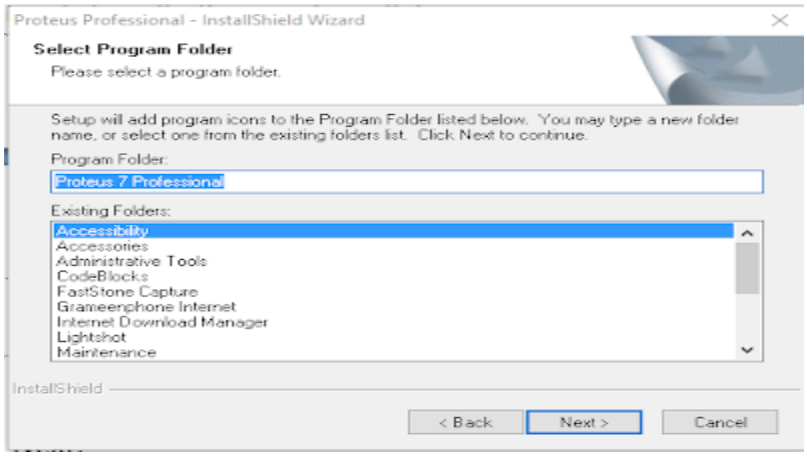
Click on Next.



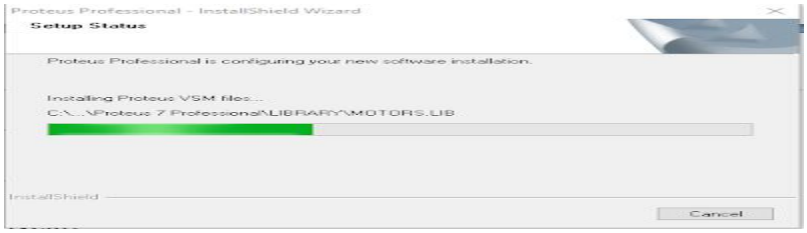
Click on Next.



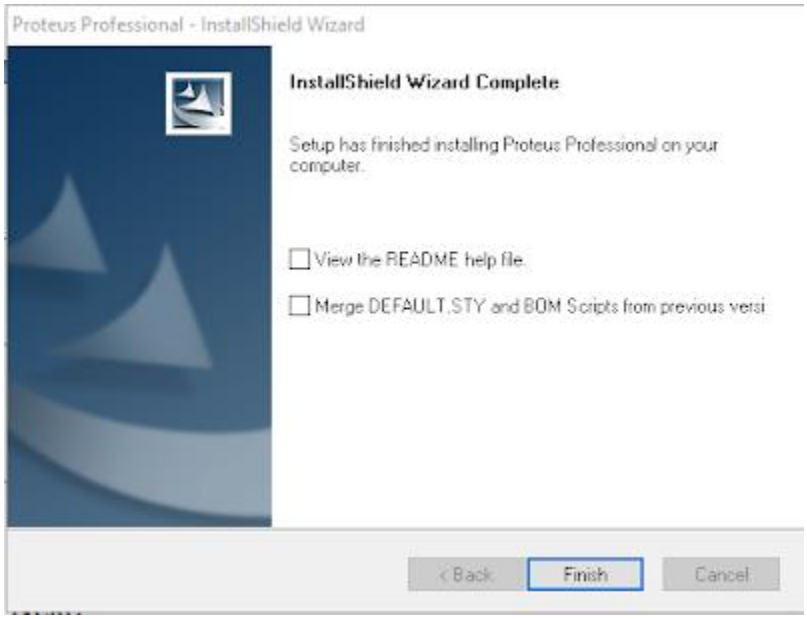
Click on Next.



The setup will started to install. Wait for some time until it completed.

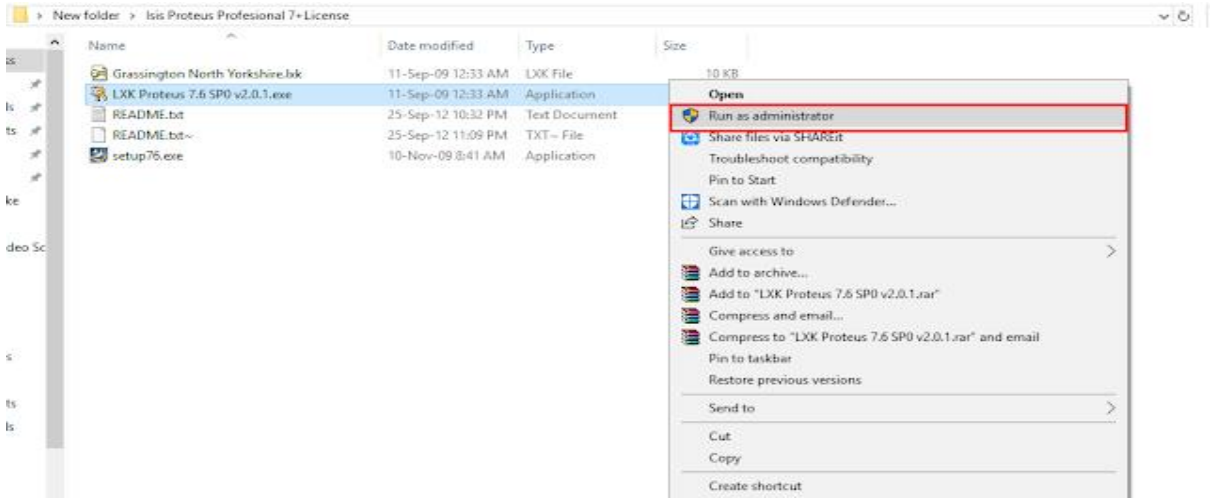


After completing the installation process will see a window like below. Uncheck the both option and click on Finish.

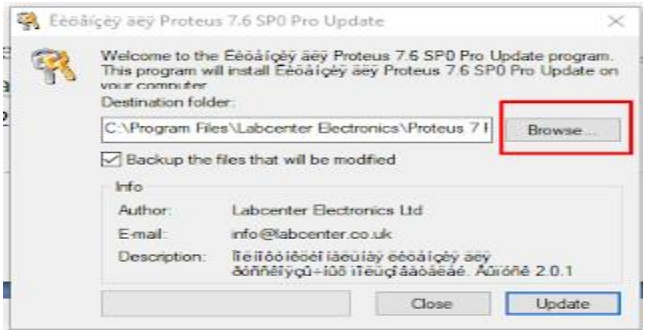


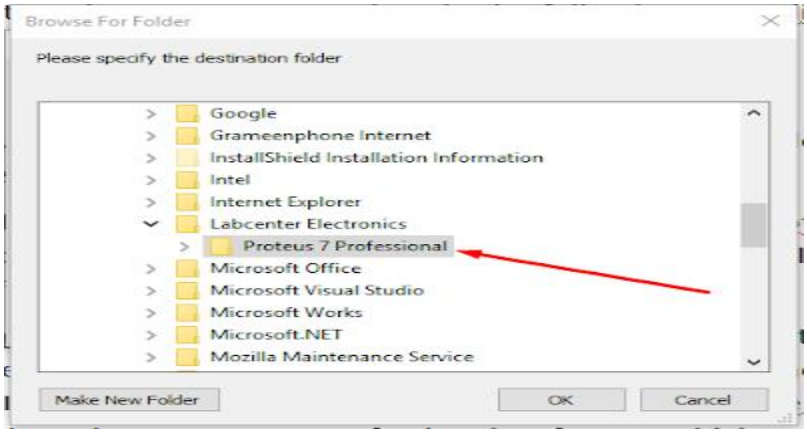
Step-3: Our Proteus ISIS 7 Professional has already installed successfully. But we have an another little step. You already noticed that we have installed a license file. Now we need to update the license file to use the Proteus 7 Professional properly.

Go to the folder again where you have extracted the zip file. Now right click on the file name ‘LXK Proteus’ and click on ‘Run as Administrator’.



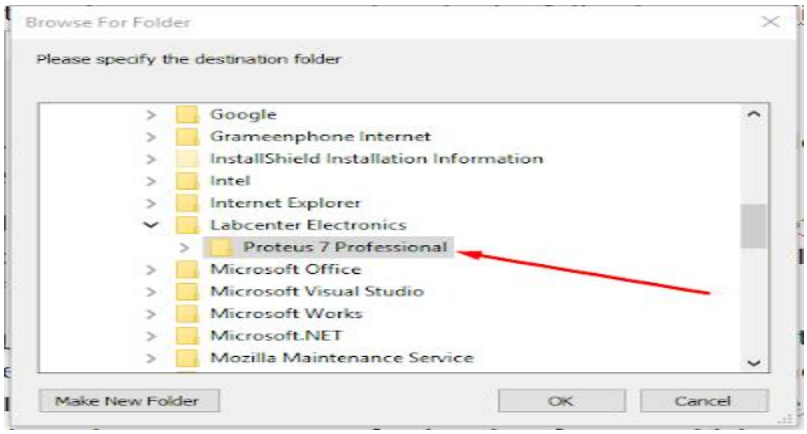
A new window will open like this.



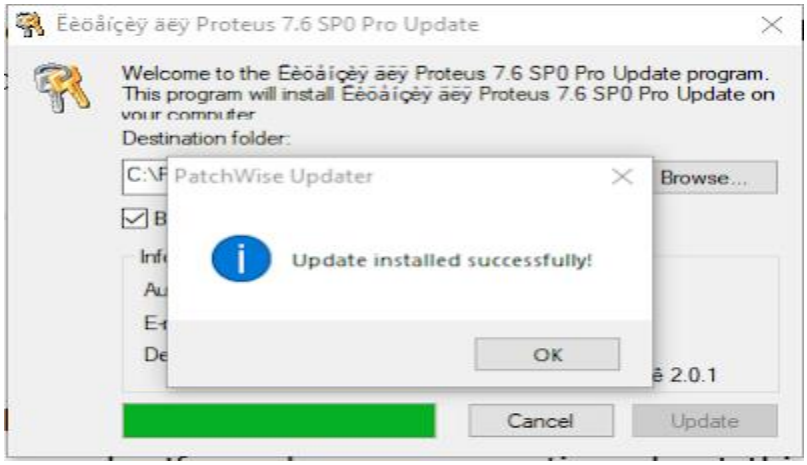


A new window will open to browse folder. Here just go to the following path: Local Disk (C:) > Program Files (x86) > Labcenter Electronics > Proteus 7 Professional. Now just select the 'Proteus 7 Professional' folder and click on Ok.

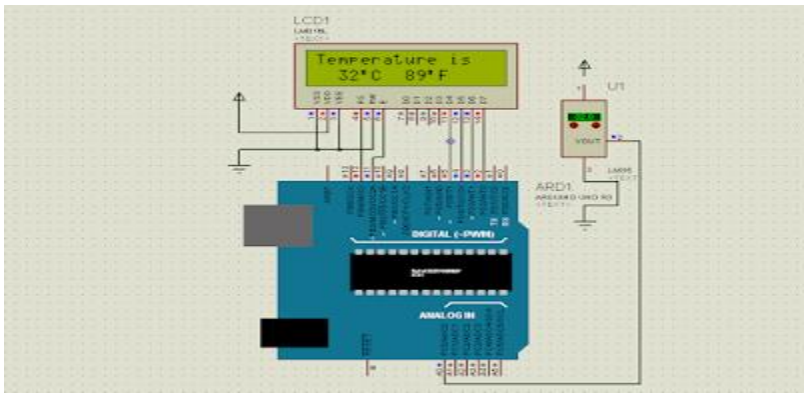
Now for the both 32 Bit and 64 Bit user, just click on Update.



You will see that Update Installed Succesfully. Just Click Ok.

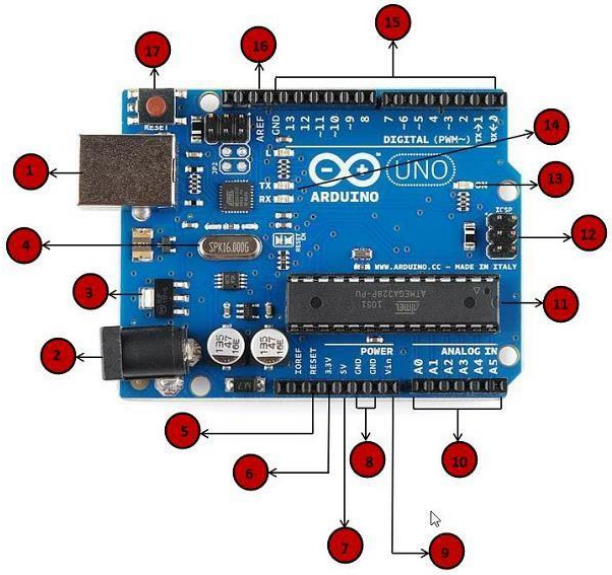


Proteus is the best simulation software for arduino and microcontroller related circuit and embedded system. Look the screenshot given below:



Assignment - 3

3) Study Arduino UNO architecture and Write arduino program to turn on and off the LED with digital pin 13 and do simulation using ISIS Proteus.



1)Power USB

Arduino board can be powered by using the USB cable from your computer. All you need to do is connect the USB cable to the USB connection (1).

2)Power (Barrel Jack)

Arduino boards can be powered directly from the AC mains power supply by connecting it to the Barrel Jack (2).

3)Voltage Regulator

The function of the voltage regulator is to control the voltage given to the Arduino board and stabilize the DC voltages used by the processor and other elements.

4)Crystal Oscillator

The crystal oscillator helps Arduino in dealing with time issues. How does Arduino calculate time? The answer is, by using the crystal oscillator. The number printed on top of the Arduino crystal is 16.000H9H. It tells us that the frequency is 16,000,000 Hertz or 16 MHz.

5,17) Arduino Reset

You can reset your Arduino board, i.e., start your program from the beginning. You can reset the UNO board in two ways. First, by using the reset button (17) on the board. Second, you can connect an external reset button to the Arduino pin labelled RESET (5).

6,7,8,9) Pins (3.3, 5, GND, Vin)

- **3.3V (6)** − Supply 3.3 output volt
- **5V (7)** − Supply 5 output volt
- Most of the components used with Arduino board works fine with 3.3 volt and 5 volt.
- **GND (8)** (Ground) − There are several GND pins on the Arduino, any of which can be used to ground your circuit.
- **Vin (9)** − This pin also can be used to power the Arduino board from an external power source, like AC mains power supply.

10)Analog pins

The Arduino UNO board has six analog input pins A0 through A5. These pins can read the signal from an analog sensor like the humidity sensor or temperature sensor and convert it into a digital value that can be read by the microprocessor.

11)Main microcontroller

Each Arduino board has its own microcontroller (11). You can assume it as the brain of your board. The main IC (integrated circuit) on the Arduino is slightly different from board to board. The microcontrollers are usually of the ATMEL Company. You must know what IC your board has before loading up a new program from the Arduino IDE. This information is available on the top of the IC. For more details about the IC construction and functions, you can refer to the data sheet.

13) Power LED indicator

This LED should light up when you plug your Arduino into a power source to indicate that your board is powered up correctly. If this light does not turn on, then there is something wrong with the connection.

14) TX and RX LEDs

On your board, you will find two labels: TX (transmit) and RX (receive). They appear in two places on the Arduino UNO board. First, at the digital pins 0 and 1, to indicate the pins responsible for serial communication. Second, the TX and RX led (13). The TX led flashes with different speed while sending the serial data. The speed of flashing

depends on the baud rate used by the board. RX flashes during the receiving process.

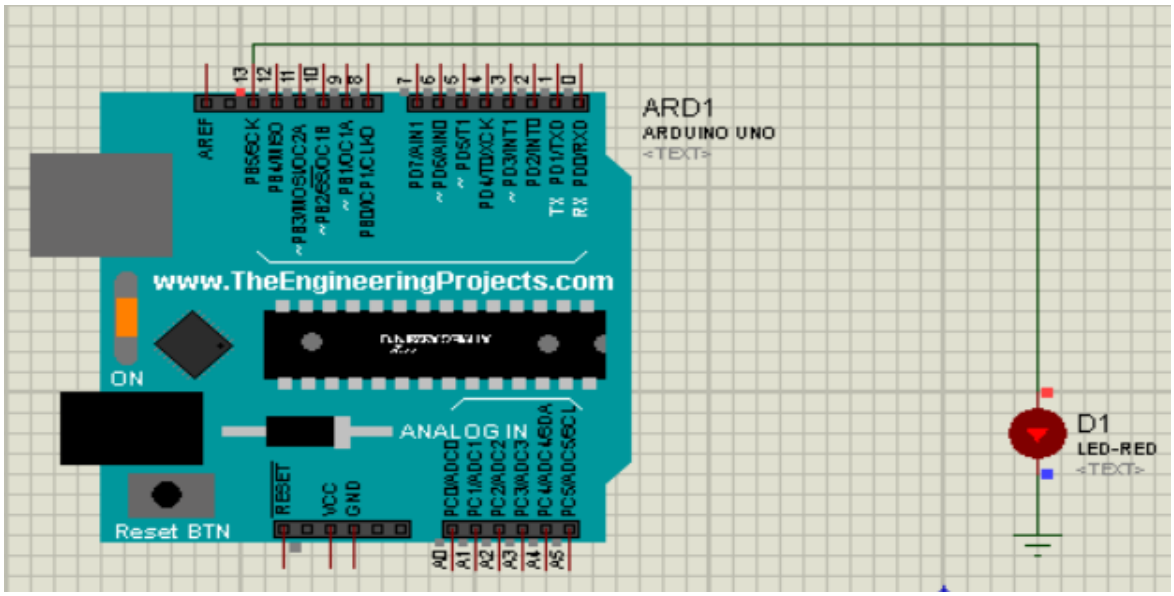
15) Digital I/O

The Arduino UNO board has 14 digital I/O pins (15) (of which 6 provide PWM (Pulse Width Modulation) output. These pins can be configured to work as input digital pins to read logic values (0 or 1) or as digital output pins to drive different modules like LEDs, relays, etc. The pins labeled “~” can be used to generate PWM.

Example:

```
void setup()
```

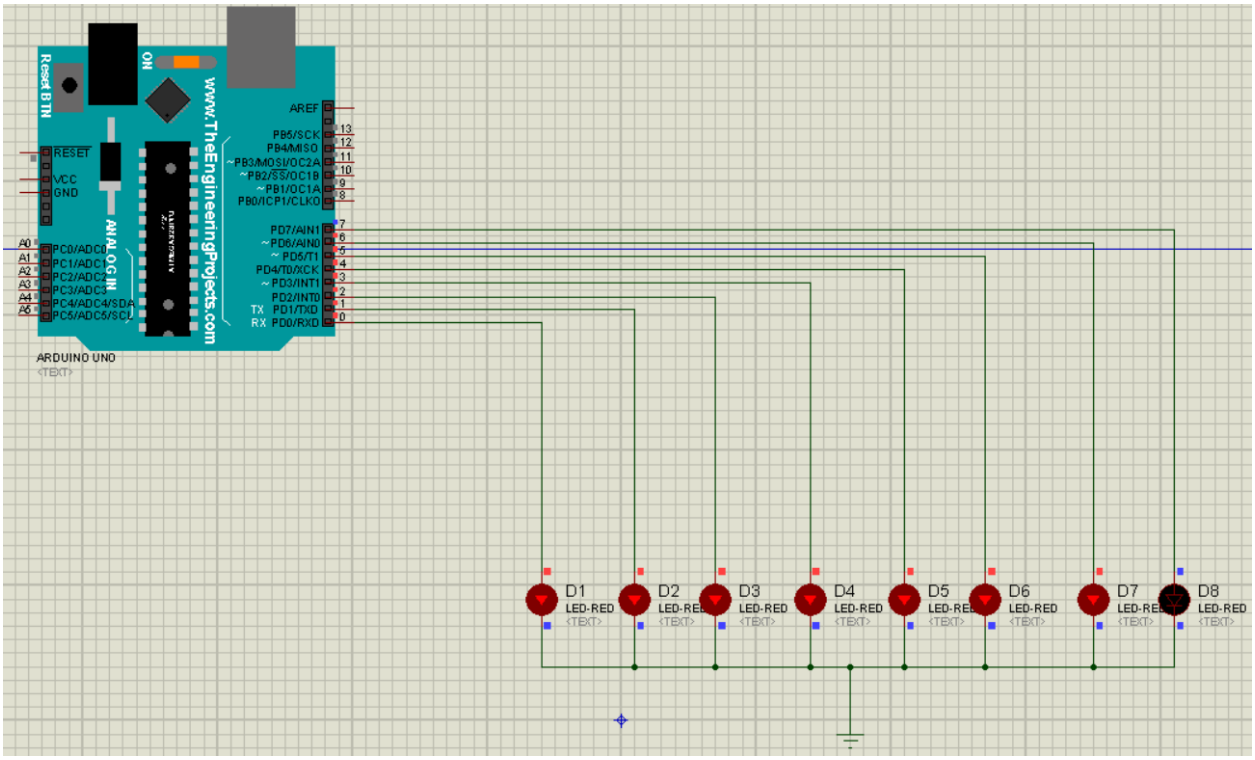
```
{
  pinMode(LED_BUILTIN, OUTPUT);
}
void loop()
{
  digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000);                     // wait for a second
  digitalWrite(LED_BUILTIN, LOW);  // turn the LED off by making the voltage LOW
  delay(1000);                     // wait for a second
}
```



Assignment – 4

a) Write arduino code to turn on and off 8 LEDs with digital pins D0 to D7 of arduino UNO.

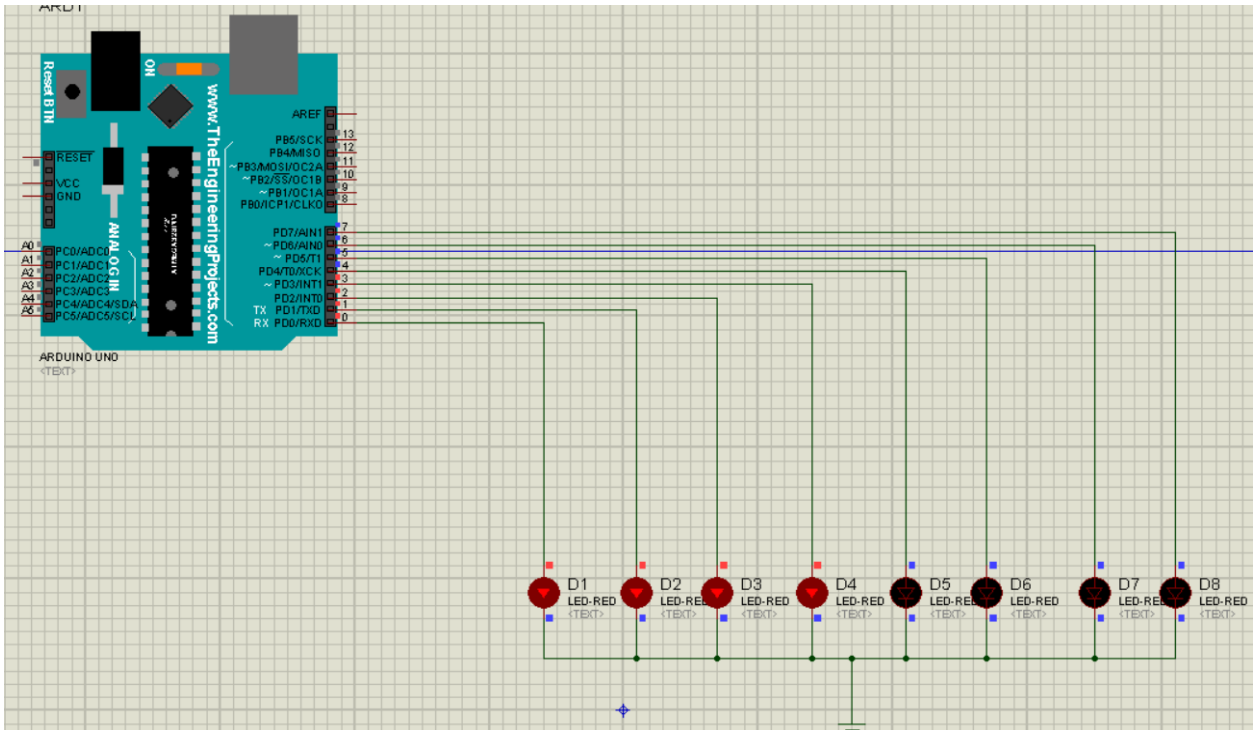
```
void setup()
{
   int i;
   for (int i = 0; i < 8; i++)
   {
      pinMode(i, OUTPUT);
   }
}
void loop()
{
   for (int i = 0; i < 8; i++)
   {
      digitalWrite(i, HIGH);
      delay(500);
   }
}
```



b) Write arduino code to turn ON and OFF 4 LEDs (LED0 ….LED7) one at a time . (First 4 LED ON and LAST 4 LED OFF after 1 second First 4 LED OFF and Last 4 LED ON).

```
void setup()
{
  for (int i = 0; i < 8; i++)
  {
    pinMode(i, OUTPUT);
  }
}
void loop()
{
  PORTD = 0xf0;
```

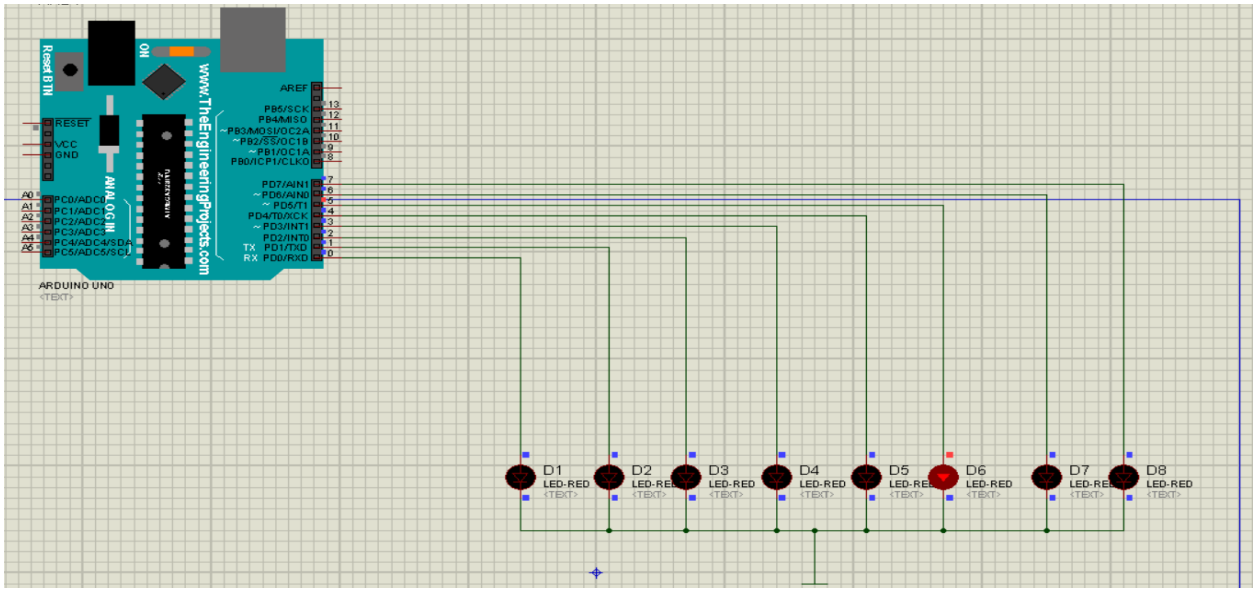
```
    delay(500);
    PORTD = 0x0f;
    delay(500);
}
```



c) Write arduino code to Turn ON and OFF LEDs alternate from left to right . (LED0 to LED7).

```
void setup()
{
    for (int i = 0; i < 8; i++)
    {
        pinMode(i, OUTPUT);
    }
}
void loop()
```

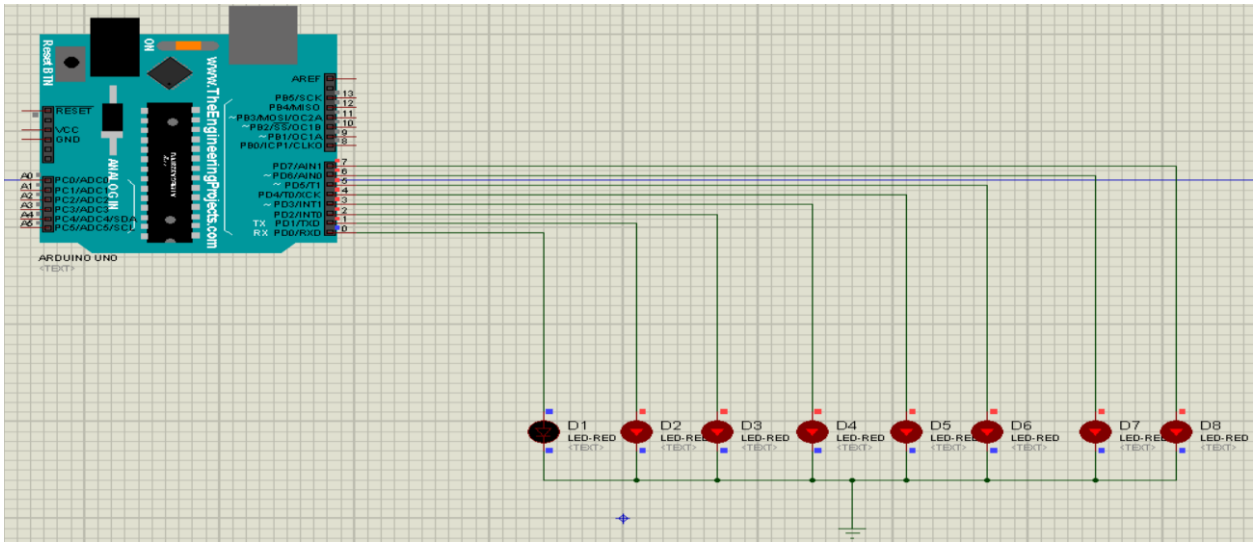
```
{
   int x = 129;
   for (int j = 0; j < 8; j++)
   {
      PORTD = x;
      delay(500);
      x = x << 1;
   }
}
```



d) Write arduino code to Turn ON and OFF LEDs from right to left . (LED7 to LED0).

```
void setup()
{
   int i;
   for (int i = 0; i < 8; i++)
   {
      pinMode(i, OUTPUT);
```

```
    }
}
void loop()
{
    int i;
    char x = 128;
    for (int i = 0; i < 8; i++)
    {
        PORTD = x;
        x = x >> 1;
        delay(300);
    }
}
```

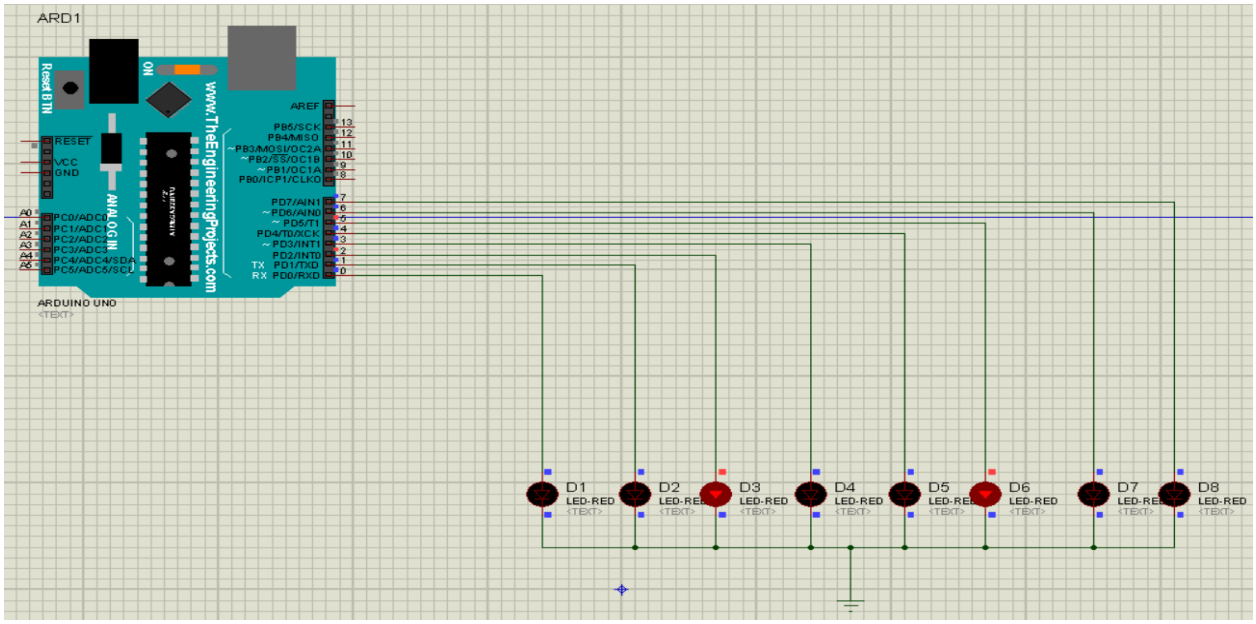


e) Write arduino code to Turn ON and OFF LED in left and right side pattern. (ON LED1 & 8 after a second LED2 & 7 after a second LED3 & 6 after a second LED 4 & 5).

```
void setup()
```

```
{
   for (int i = 0; i < 8; i++)
   {
      pinMode(i, OUTPUT);
   }
}
void loop()
{
   int i;
   for (int i = 0; i < 8; i++)
   {
      //    PORTD = 129;
      PORTD = 0X81;
      delay(250);
      //    PORTD = 66;
      PORTD = 0X42;
      delay(250);
      //    PORTD = 36;
      PORTD = 0X24;
      delay(250);
      //    PORTD = 24;
      PORTD = 0X18;
      delay(250);
      //    PORTD = 36;
      PORTD = 0X24;
      delay(250);
      //    PORTD = 66;
      PORTD = 0X42;
      delay(250);
      //    PORTD = 129;
```

```
    PORTD = 0X81;
    delay(250);
  }    }
```

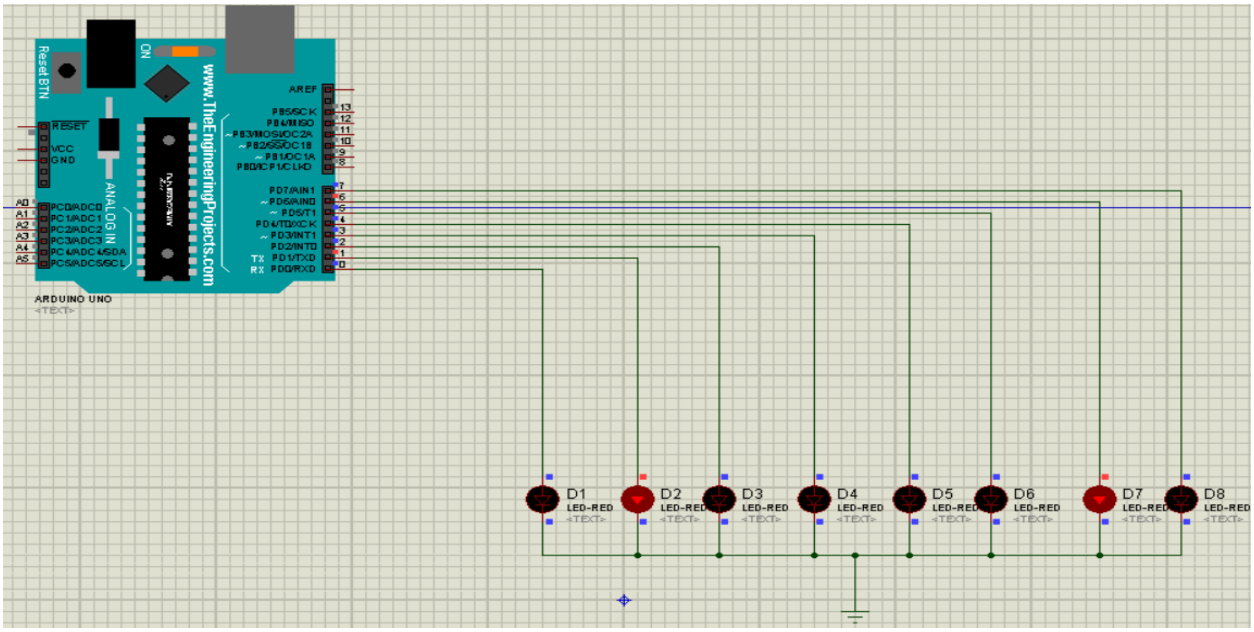


f) Write arduino code for Dancing LED pattern. (1 LED Toggle , 8 LEDs Toggle , 4 LED ON and OFF , Alternat LED pattern , Left side pattern , right side pattern and left and right side pattern.

```
void setup()
{
  for (int i = 0; i < 8; i++)
  {
    pinMode(i, OUTPUT);
  }
}

void loop()
{
```

```
PORTD = 0xFF;
delay(500);
PORTD = 0x00;
delay(500);
PORTD = 0x0f;
delay(1000);
PORTD = 0xf0;
delay(1000);

int x = 129;
for (int j = 0; j < 8; j++)
{ }
x = 129;
PORTD = x;
delay(500);
x = x << 1;
PORTD = x;
delay(500);
x = x >> 1;

int k = 0;
for (int j = 7; j >= 0; j--)
{
   digitalWrite(k, HIGH);
   digitalWrite(j, HIGH);
   delay(500);
   digitalWrite(k, LOW);
   digitalWrite(j, LOW);
   delay(500);
   k++;
```

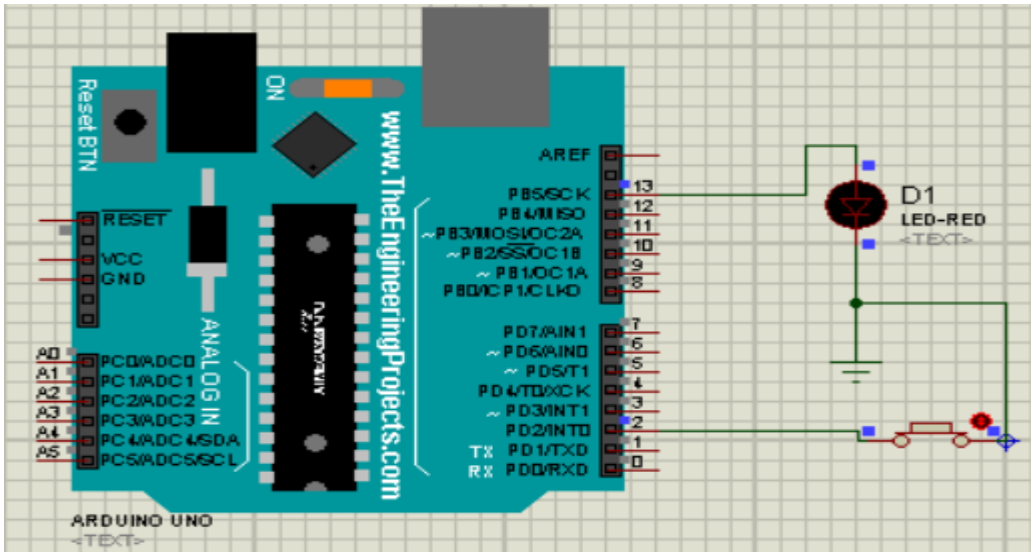
```
    }
}
```


Reset BTN
ON
RESET
VCC
GND
ANALOG IN
www.TheEngineeringProjects.com
AREF
ARDUINO UNO
D1
LED-RED
D2
LED-RED
D3
LED-RED
D4
LED-RED
D5
LED-RED
D6
LED-RED
D7
LED-RED
D8
LED-RED

Assignment – 5

Write arduino Program to control LED with digital pin 13 using PUSH button and do simulation using ISIS Proteus.

```
const int buttonPin = 2; // the number of the pushbutton pin
const int ledPin = 13;   // the number of the LED pin
int buttonState = 0;     // variable for reading the pushbutton status

void setup()
{
   pinMode(ledPin, OUTPUT);
   pinMode(buttonPin, INPUT);
}

void loop()
{
   buttonState = digitalRead(buttonPin);
   if (buttonState == HIGH)
   {
      digitalWrite(ledPin, HIGH);
   }
   else
   {
      digitalWrite(ledPin, LOW);
   }
}
```

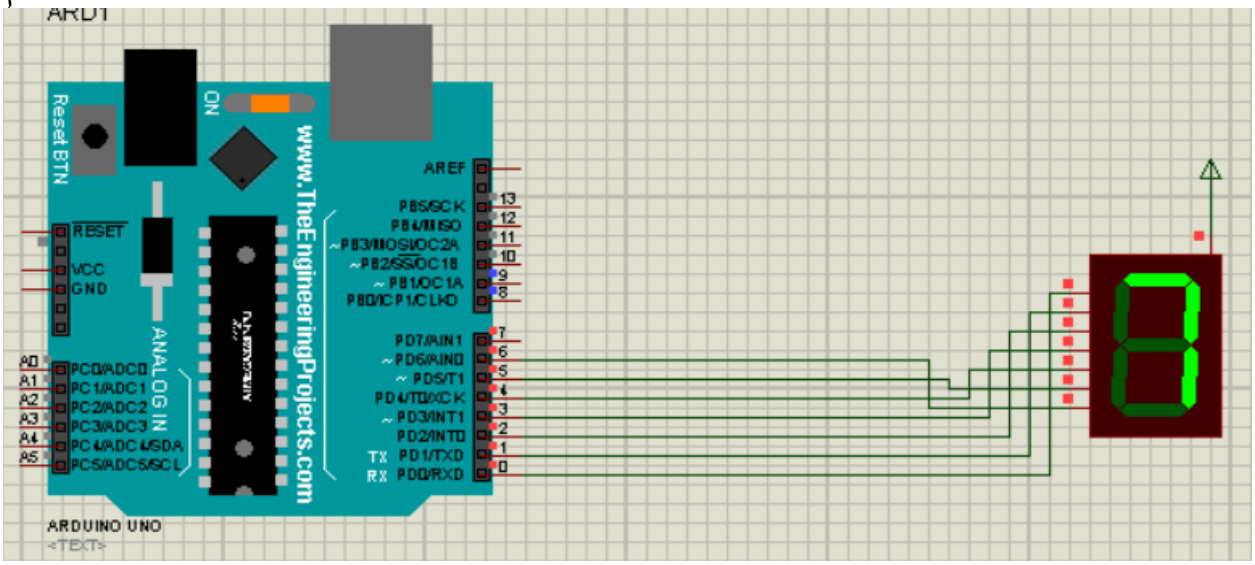

Reset BTN
ON
www.TheEngineeringProjects.com
RESET
VCC
GND
ANALOG IN
A0
A1
A2
A3
A4
A5
PC0/ADC0
PC1/ADC1
PC2/ADC2
PC3/ADC3
PC4/ADC4/SDA
PC5/ADC5/SCL
AREF
PB5/SCK
PB4/MISO
~PB3/MOSI/OC2A
~PB2/SS/OC1B
~PB1/OC1A
PB0/ICP1/CLKO
PD7/AIN1
~PD6/AIN0
~PD5/T1
PD4/T0/XCK
~PD3/INT1
PD2/INT0
TX PD1/TXD
RX PD0/RXD
13
12
11
10
9
8
7
6
5
4
3
2
1
0
D1
LED-RED
<TEXT>
ARDUINO UNO
<TEXT>

Assignment – 6

a) Write arduino program to print 0 to 9 on Seven Segment display and do simulation using ISIS Proteus.

```
void setup()
{
   int i;
   for (int i = 0; i < 10; i++)
   {
      pinMode(i, OUTPUT);
   }
}
char num[] = {0xc0, 0xf9, 0xa4, 0xb0, 0x99, 0x92, 0x82, 0xf8, 0x80, 0x90};
void loop()
{
   for (int i = 0; i < 10; i++)
   {
      PORTD = num[i];
      delay(400);
   }
```

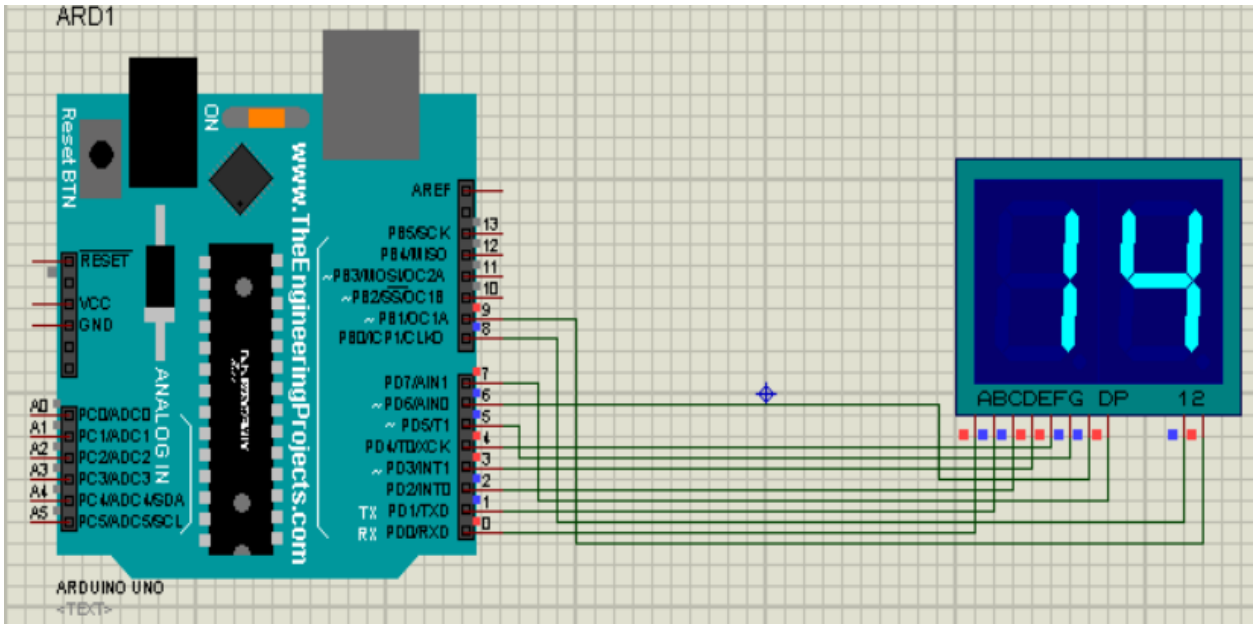
}



b) Write arduino program to print 0 to 99 on 2 DIGIT Seven Segment display and do simulation using ISIS Proteus.

```
#define E1 8
#define E2 9
void setup()
{
  for (int i = 0; i < 10; i++)
  {
    pinMode(i, OUTPUT);
  }
}
char num[] = {0xc0, 0xf9, 0xa4, 0xb0, 0x99, 0x92, 0x82, 0xf8, 0x80, 0x90};
void loop()
{
  for (char i = 0; i < 100; i++)
  {
    display2segment(i);
```

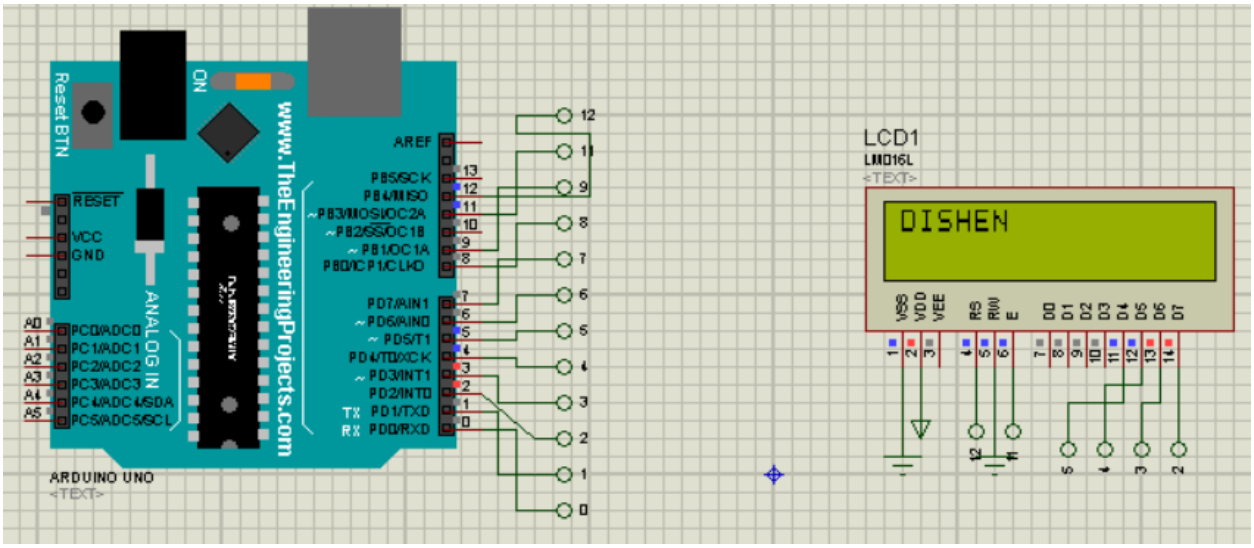
```
    }
}
void display2segment(char s)
{
    for (int d = 0; d < 100; d++)
    {
        digitalWrite(E2, LOW);
        digitalWrite(E1, HIGH);
        PORTD = num[s / 10];
        delay(2);

        digitalWrite(E1, LOW);
        digitalWrite(E2, HIGH);
        PORTD = num[s % 10];
        delay(8);
    }     }
```



Assignment – 7

a) Write arduino program to display your name on LED screen and do simulation using ISIS proteus.

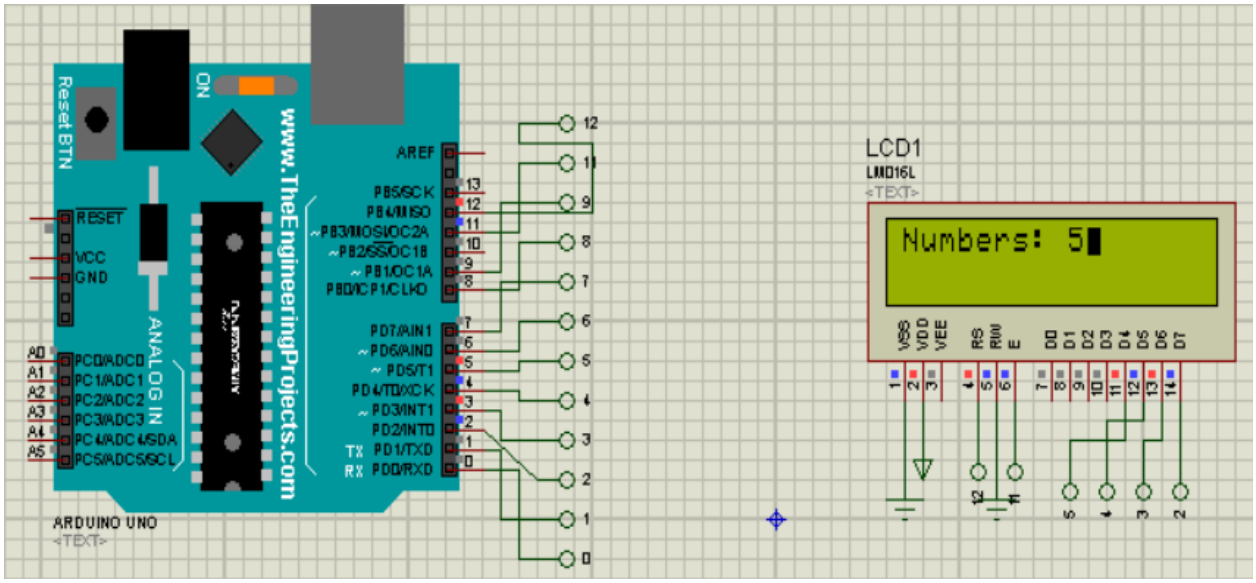
```
#include <LiquidCrystal.h>
const int rs = 12, en = 11, d4 = 5, d5 = 4, d6 = 3, d7 = 2;
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
void setup()
{
   // set up the LCD's number of columns and rows:
   lcd.begin(16, 2);
   // Print a message to the LCD.
   lcd.print("DISHEN");
}
void loop()
{
   // Turn off the display:
   lcd.noDisplay();
   delay(500);
   // Turn on the display:
   lcd.display();
   delay(500);
}
```



b) Write arduino program to display numbers from 0 to 9on LED screen and do simulation using ISIS proteus.

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(12,11, 5, 4, 3, 2);
void setup() {
    lcd.begin(16, 2);
}
void loop()
{
   lcd.setCursor(0, 0);
   lcd.print("Numbers:");
   for (int i = 0; i <= 9; i++)
   {
      lcd.setCursor(9, 0);
      lcd.blink();
      delay(500);
      lcd.print(i);
```

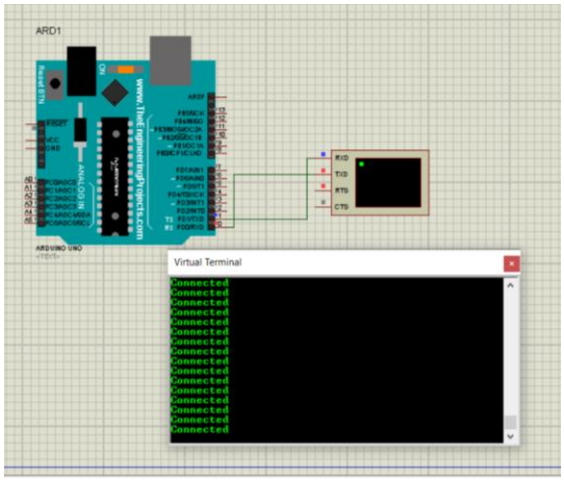
```
        delay(1000);
    }
}
```


Reset BTN
ON
RESET
VCC
GND
ANALOG IN
www.TheEngineeringProjects.com
ARDUINO UNO
<TEXT>
AREF
PB5/SCK
PB4/MISO
PB3/MOSI/OC2A
PB2/SS/OC1B
PB1/OC1A
PB0/ICP1/CLKO
PD7/AIN1
PD6/AIN0
PD5/T1
PD4/T0/XCK
PD3/INT1
PD2/INT0
TX PD1/TXD
RX PD0/RXD
PC0/ADC0
PC1/ADC1
PC2/ADC2
PC3/ADC3
PC4/ADC4/SDA
PC5/ADC5/SCL
LCD1
LM016L
<TEXT>
Numbers: 5
VSS
VDD
VEE
RS
RW
E
D0
D1
D2
D3
D4
D5
D6
D7

Assignment – 8

(a) Write arduino code to print “Hello World” on serial monitor and do simulation on ISIS Proteus.

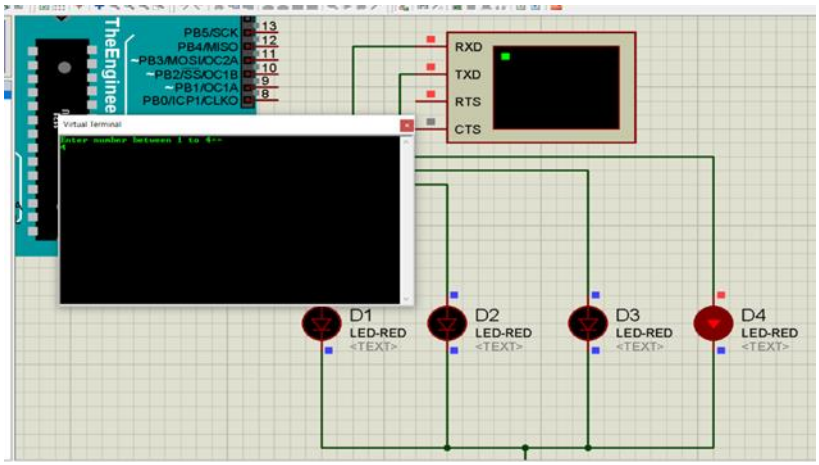
```
void setup()
{
  Serial.begin(9600);
}
void loop()
{
  Serial.println("Hello From DISHEN MAKWANA");
}
```



(b) Write arduino code to receive data serially from user and display LED based on users input (from 1 to 4). (Example:- if user enter 4 , LED 4 should blink.

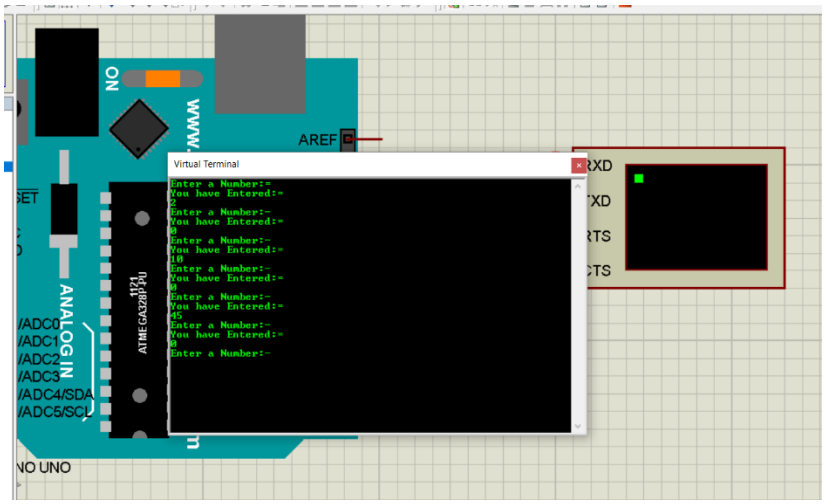
```
void setup()
{
  for (int i = 2; i < 6; i++)
```

```
    {
      pinMode(i, OUTPUT);
    }
    Serial.begin(9600);
    Serial.println("Enter number between 1 to 4==");
}
void loop()
{
    while (Serial.available() > 0)
    {
      int k = Serial.parseInt();
      Serial.println(k);
      if (k >= 1 && k <= 4)
      {
        digitalWrite(k + 1, HIGH);
        delay(2000);
        digitalWrite(k + 1, LOW);
        delay(2000);
      }
    }
}
```



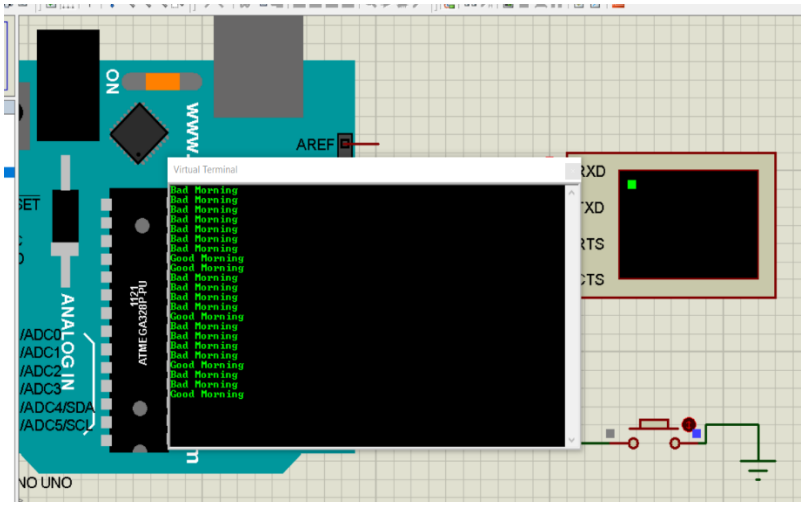
(c) Write arduino code to transmit and receive data serially.

```
void setup()
{
   Serial.begin(9600);
   Serial.println("Enter a Number:=");
}
void loop()
{
   while (Serial.available() > 0)
   {
      int k = Serial.parseInt();
      Serial.println("You have Entered:=");
      Serial.println(k);
      delay(500);
      Serial.println("Enter a Number:-");
   }
}
```



(d) Write arduino code to transmit “Good morning” serially if switch is pressed and transmit “Bad Morning” if switch is open , do simulation using proteus.

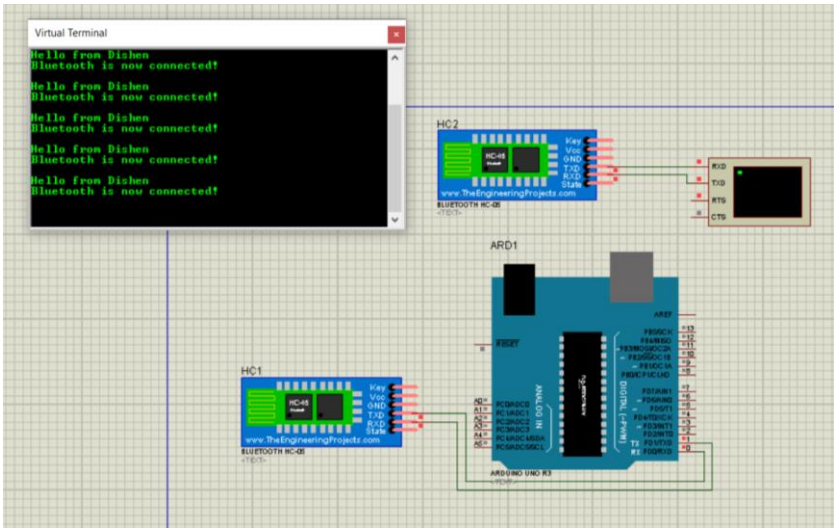
```
const int buttonPin = 2;
int buttonState = 0;
void setup()
{
   Serial.begin(9600);
   pinMode(buttonPin, INPUT);
}
void loop()
{
   buttonState = digitalRead(buttonPin);
   if (buttonState == HIGH)
   {
      Serial.println("Good Morning");
   }
   else
   {
      Serial.println("Bad Morning");
   }
   delay(500);
}
```


Virtual Terminal
Bad Morning
Bad Morning
Bad Morning
Bad Morning
Bad Morning
Bad Morning
Good Morning
Good Morning
Bad Morning
Bad Morning
Bad Morning
Bad Morning
Good Morning
Bad Morning
Bad Morning
Bad Morning
Bad Morning
Good Morning
Bad Morning
Bad Morning
Good Morning
AREF
ANALOG IN
RXD
TXD
RTS
CTS

Assignment – 9

(a) Write arduino code to transmit “Hello World – Code to demonstrate BT Communication” string on serial monitor using Bluetooth. Do simulation on ISIS Proteus.

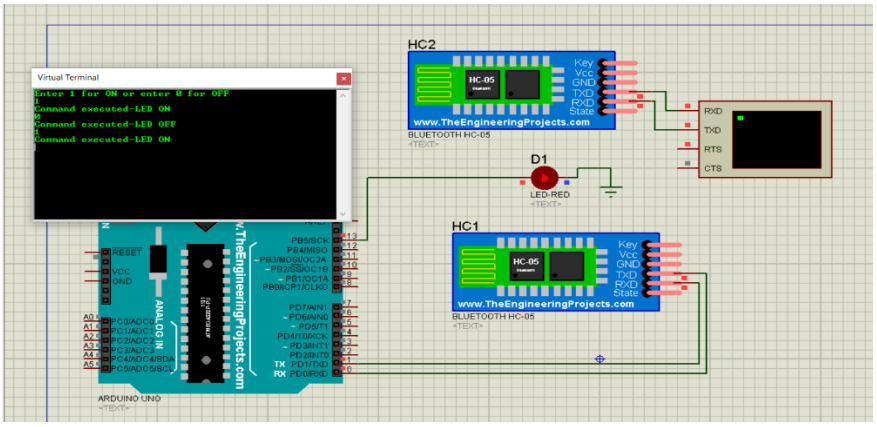
```
void setup()
{
  Serial.begin(9600);
}
void loop()
{
  Serial.println("Hello from Dishen");
  Serial.println("Bluetooth is connected");
  delay(1000); }
```



(b) Write arduino code to blink LED using Bluetooth communication and do simulation on ISIS Proteus.

```
void setup()
```

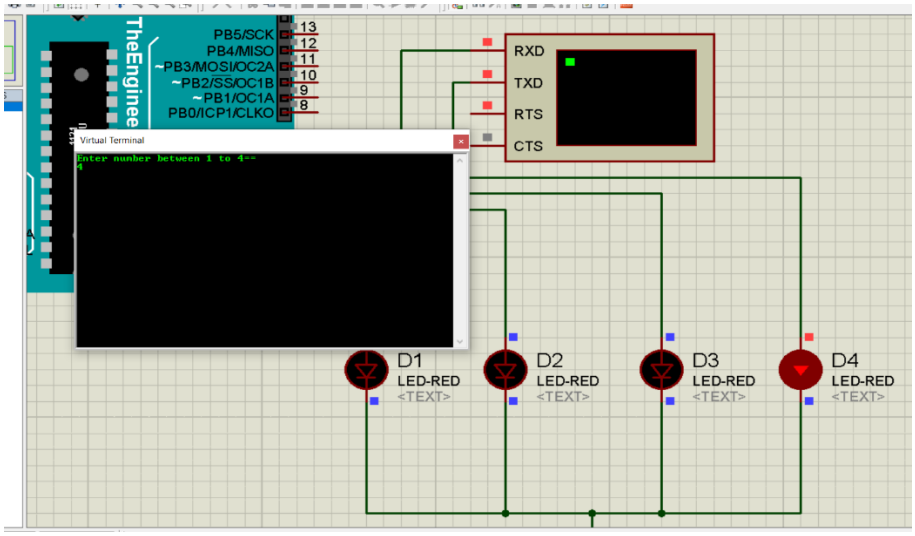
```
{
  Serial.begin(9600);
  pinMode(13, OUTPUT);
  Serial.println("Enter 1 for ON or enter 0 for OFF");
}
void loop()
{
  if (Serial.available())
  {
    int k = Serial.parseInt();
    Serial.println(k);
    if (k == 1)
    {
      digitalWrite(13, HIGH);
      Serial.println("Command executed-LED ON");
      delay(500);
    }
    else
    {
      digitalWrite(13, LOW);
      Serial.println("Command executed-LED OFF");
      delay(500);
    }
  }
}
```



(c) Write arduino code to blink LED based on User's choice using Bluetooth communication and do simulation on ISIS Proteus.

```
void setup()
{
  for (int i = 2; i < 6; i++)
  {
    pinMode(i, OUTPUT);
  }
  Serial.begin(9600);
  Serial.println("Enter number between 1 to 4==");
}
void loop()
{
  while (Serial.available() > 0)
  {
    int k = Serial.parseInt();
    Serial.println(k);
    if (k >= 1 && k <= 4)
    {
      digitalWrite(k + 1, HIGH);
```

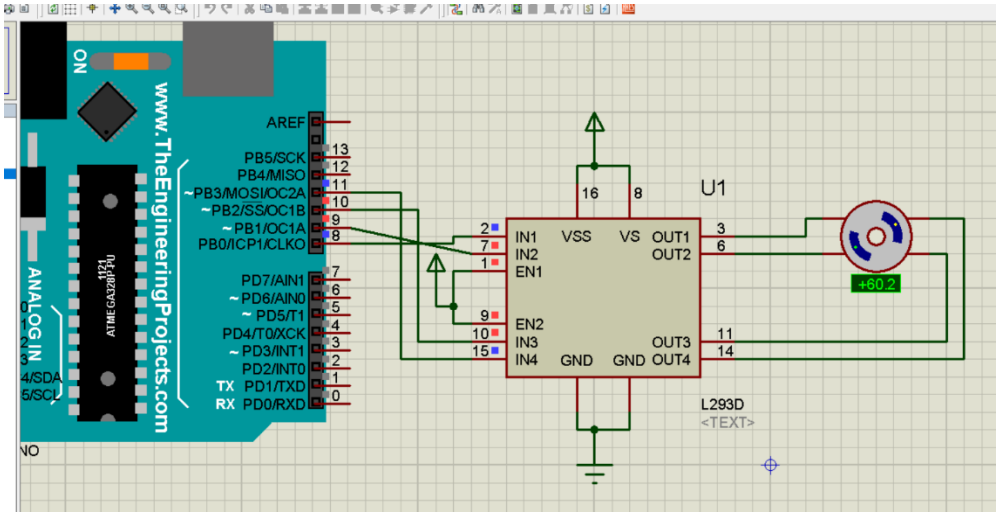
```
            delay(2000);
            digitalWrite(k + 1, LOW);
        }
    }
}
```



Assignment – 10

(a)Write arduino code to rotate Stepper Motor and do interfacing & simulation with arduino on ISIS Proteus.

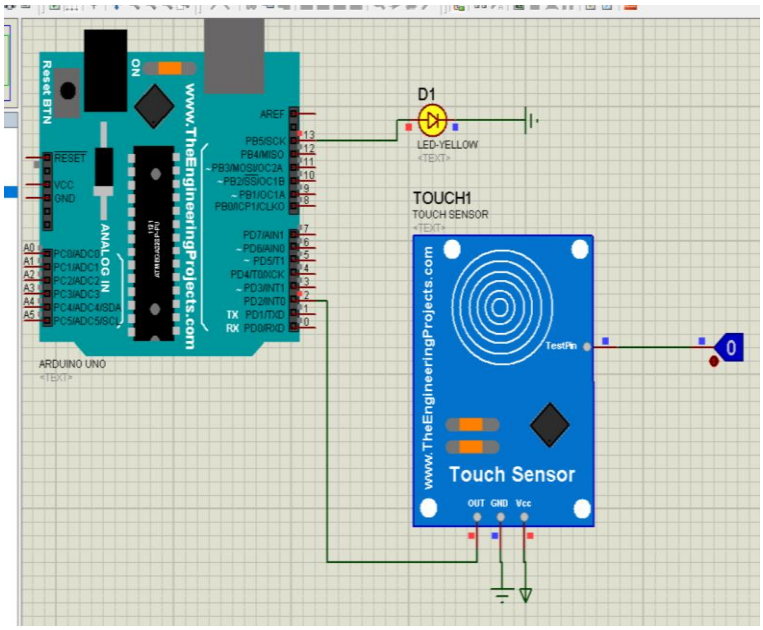
```
#include <Stepper.h>
int stepsPerRevolution = 20;
Stepper myStepper(stepsPerRevolution, 8, 9, 10, 11);
void setup()
{
   // set the speed at 40 rpm:
   myStepper.setSpeed(40);
}
void loop()
{
   // step one revolution  in one direction:
   myStepper.step(stepsPerRevolution);
   delay(20);
   myStepper.step(~stepsPerRevolution);
   delay(20);
}
```



(b) Write arduino code to make LED ON based on Capacitive touch sensor. Do interfacing & simulation with arduino on ISIS Proteus.

```
const int buttonPin = 2;
const int ledPin = 13;
int buttonState = 0;
void setup()
{
   pinMode(ledPin, OUTPUT);
   pinMode(buttonPin, INPUT);
}
void loop()
{
   buttonState = digitalRead(buttonPin);
   if (buttonState == HIGH)
   {
      digitalWrite(ledPin, HIGH);
   }
   else
```

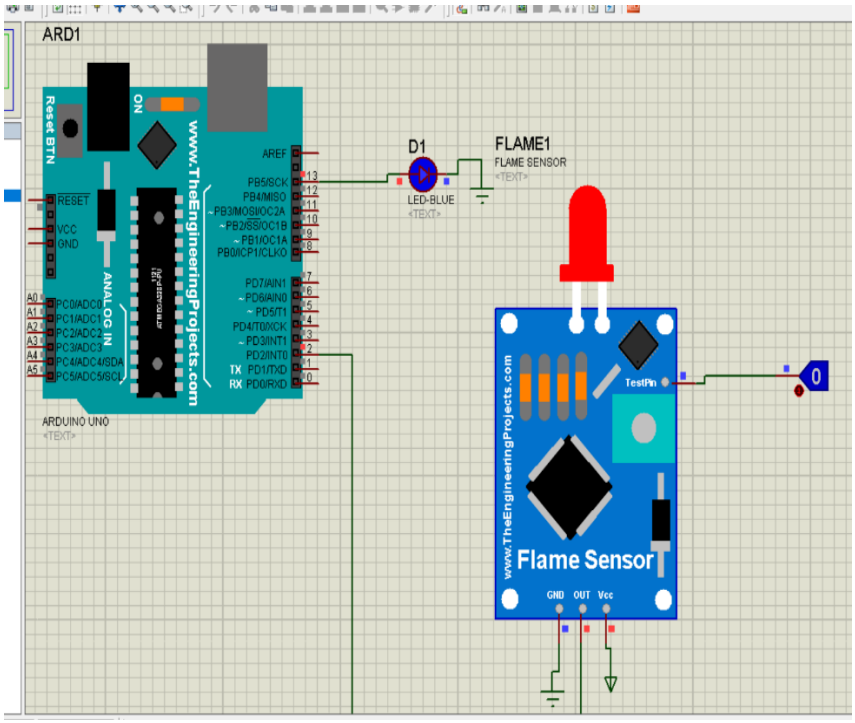
```
  {
    digitalWrite(ledPin, LOW);
  }
}
```



(c) Write arduino code to make LED/BUZZER ON based on Flame sensor. Do interfacing & simulation with arduino on ISIS Proteus.

```
const int buttonPin = 2;
const int ledPin = 13;
int buttonState = 0;
void setup()
{
  pinMode(ledPin, OUTPUT);
  pinMode(buttonPin, INPUT);
}
void loop()
{
```

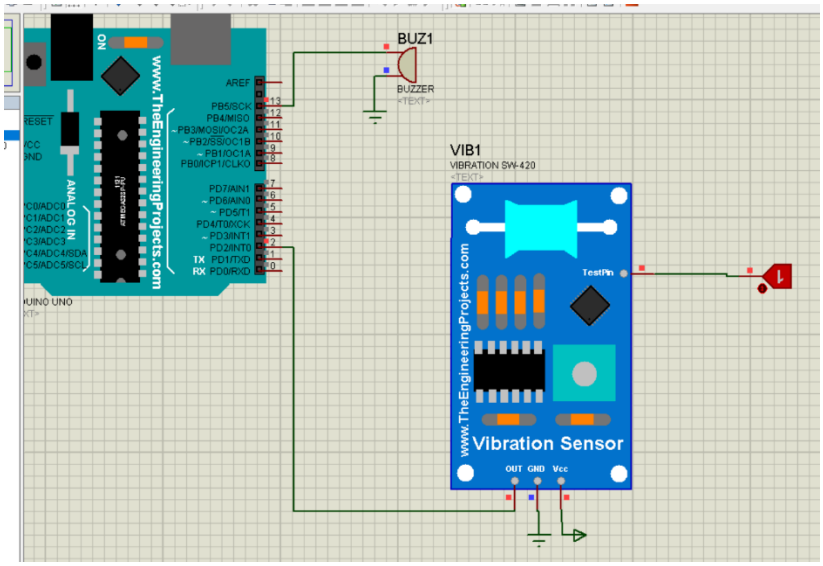
```
    buttonState = digitalRead(buttonPin);
    if (buttonState == HIGH)
    {
       digitalWrite(ledPin, HIGH);
    }
    else
    {
       digitalWrite(ledPin, LOW);
    }
}
```



Assignment – 11

(a)Write arduino code to make LED/BUZZER ON based on Vibration sensor. Do interfacing & simulation with arduino on ISIS Proteus.

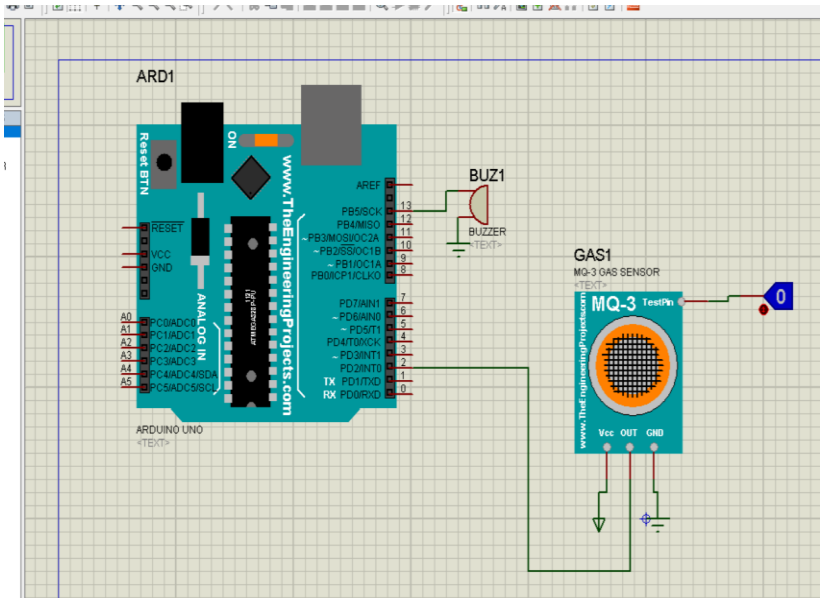
```
const int buttonPin = 2;
const int ledPin = 13;
int buttonState = 0;
void setup()
{
   pinMode(ledPin, OUTPUT);
   pinMode(buttonPin, INPUT);
}
void loop()
{
   buttonState = digitalRead(buttonPin);
   if (buttonState == HIGH)
   {
      digitalWrite(ledPin, HIGH);
   }
   else
   {
      digitalWrite(ledPin, LOW);
   }
}
```



b)Write arduino code to make LED/Buzzer ON based on GAS sensor. Do interfacing & simulation with arduino on ISIS Proteus.

```
int Gas = 2;     // the number of the pushbutton pin
int ledPin = 13; // the number of the LED pin
int GasState = 0; // variable for reading the pushbutton status

void setup()
{
   pinMode(ledPin, OUTPUT);
   pinMode(Gas, INPUT_PULLUP);
}
void loop()
{
   GasState = digitalRead(Gas);
   if (GasState == HIGH)
   {
      digitalWrite(ledPin, HIGH);
```

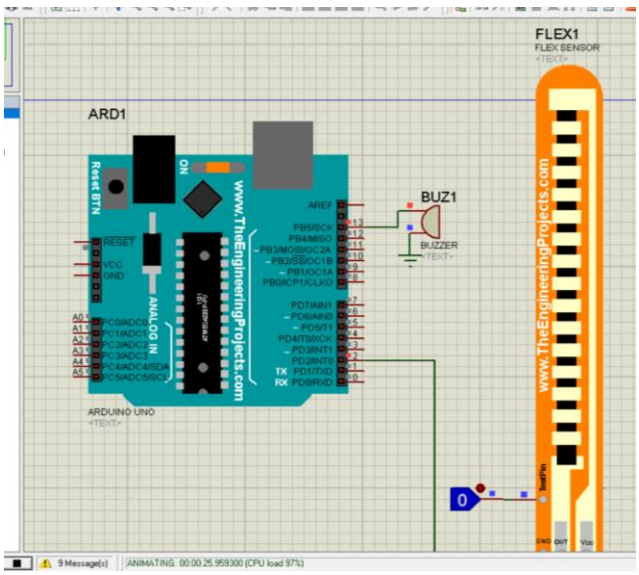
```
    }
    else
    {
       digitalWrite(ledPin, LOW);
    }
}
```



c)Write arduino code to make LED/Buzzer ON based on Flex sensor. Do interfacing & simulation with arduino on ISIS Proteus.

```
int flex = 2;      // the number of the pushbutton pin
int ledPin = 13;   // the number of the LED pin
int flexState = 0; // variable for reading the pushbutton status
void setup()
{
   pinMode(ledPin, OUTPUT);
   pinMode(flex, INPUT_PULLUP);
}
void loop()
```

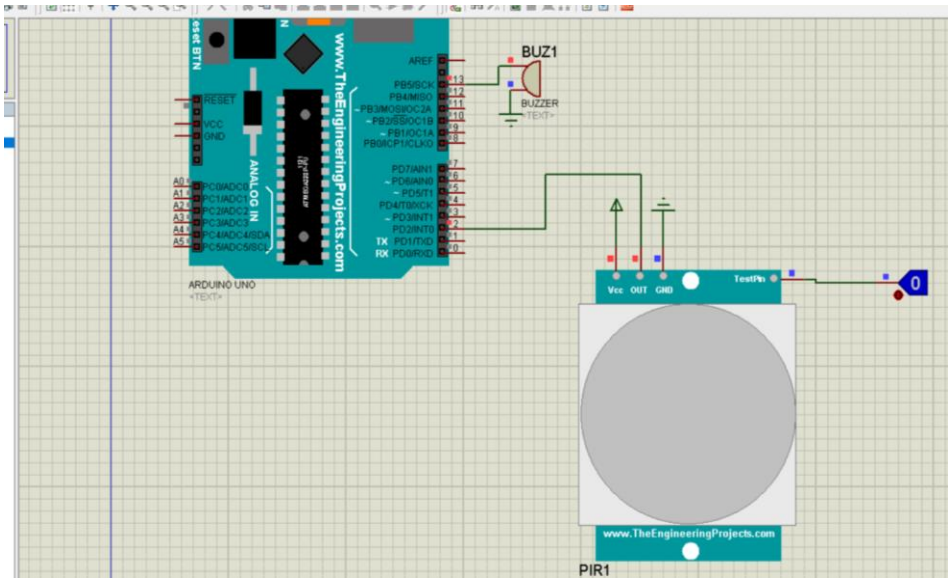
```
{
   flexState = digitalRead(flex);
   if (flexState == HIGH)
   {
      digitalWrite(ledPin, HIGH);
   }
   else
   {
      digitalWrite(ledPin, LOW);
   }
}
```



(d)Write arduino code to make LED/Buzzer ON based on PIR sensor. Do interfacing & simulation with arduino on ISIS Proteus.

```
int PIR = 2;      // the number of the pushbutton pin
int ledPin = 13;  // the number of the LED pin
int PIRState = 0; // variable for reading the pushbutton status
void setup()
```

```
{
   pinMode(ledPin, OUTPUT);
   pinMode(PIR, INPUT_PULLUP);
}
void loop()
{
   PIRState = digitalRead(PIR);
   if (PIRState == HIGH)
   {
      digitalWrite(ledPin, HIGH);
   }
   else
   {
      digitalWrite(ledPin, LOW);
   }
}
```

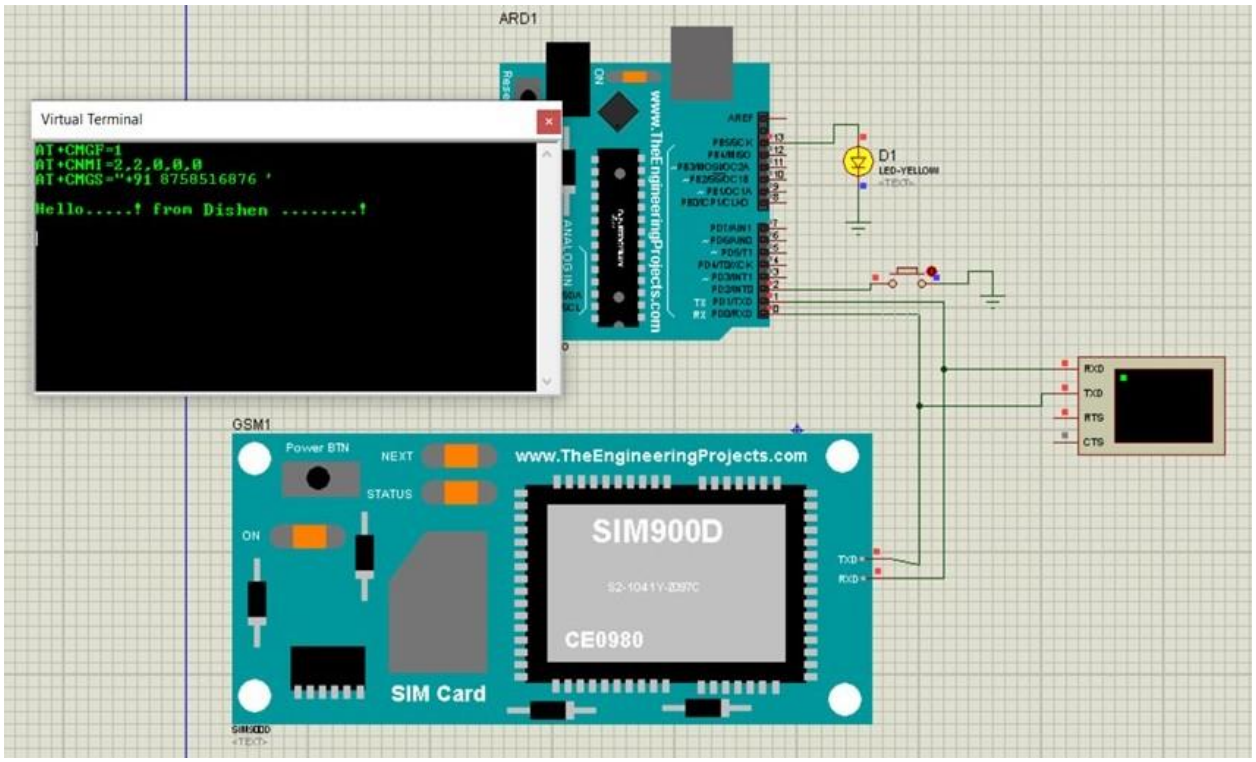


Assignment – 12

(a) Write arduino code to send SMS using GSM module and do simulation using ISIS proteus.

```
int buttonPin = 2;
#define led 13
void setup()
{
   pinMode(buttonPin, INPUT_PULLUP);
   pinMode(led, OUTPUT);
   Serial.begin(9600); // Setting the baud rate of Serial Monitor (Arduino)
   delay(100);
   Serial.println("AT+CMGF=1"); //Sets the GSM Module in Text Mode Attention Command
   delay(100);
   Serial.println("AT+CNMI=2,2,0,0,0"); // AT Command to receive a live SMS-New message Indication-(mode-empty-BM-DS-Buffer)
   delay(1000);
   digitalWrite(led, HIGH);
}
void loop()
{
   int buttonState = digitalRead(buttonPin);
   if (buttonState == LOW)
   {
      sendmessage();
      digitalWrite(led, HIGH);
   }
}
void sendmessage()
```

```
{
  Serial.println("AT+CMGS=\"+91 8758516876\"\r"); // Message Send
  delay(1000);
  Serial.println("Hello.....! from Dishen.....!"); // The SMS text you want to send
  delay(100);
  Serial.println((char)26); // ctrl+z to terminate message
  delay(1000);
}
```

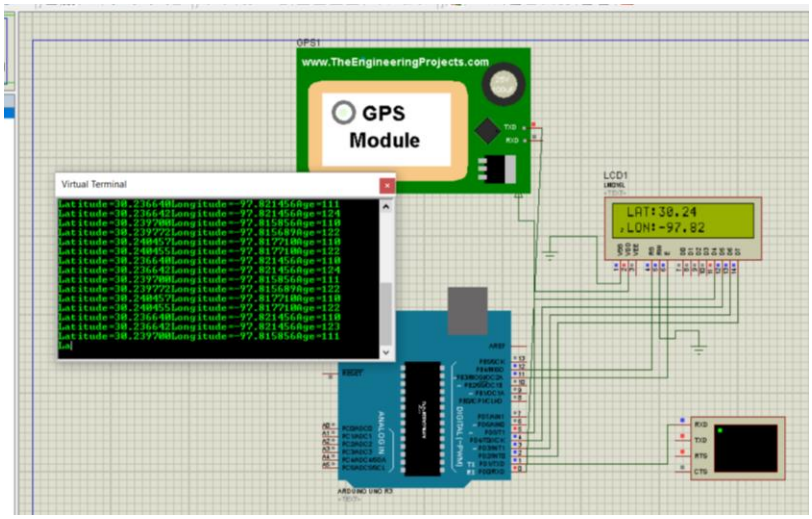


b) Write arduino code to print current location using GPS. Do interfacing & simulation with arduino on ISIS Proteus.

```
#include <TinyGPS.h>
#include <LiquidCrystal.h>
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
TinyGPS gps;
```

```
void setup()
{
  Serial.begin(9600);
  Serial.print("Welcome to Demonstrate GPS");
  lcd.begin(16, 2);
}
void loop()
{
  while (Serial.available())
  {
    char c = Serial.read();
    if (gps.encode(c))
    {
      float la, lo;
      unsigned long age;
      gps.f_get_position(&la, &lo, &age);
      Serial.print("Latitude=");
      Serial.print(la, 6);
      Serial.print("Longitude=");
      Serial.print(lo, 6);
      Serial.print("Age=");
      Serial.print(age, 6);
      Serial.println();
      lcd.setCursor(1, 0);
      lcd.print("LAT:");
      lcd.setCursor(5, 0);
      lcd.print(la);
      lcd.setCursor(0, 1);
      lcd.print(",LON:");
      lcd.setCursor(5, 1);
```

```
            lcd.print(lo);
        }
    }
}
```


www.TheEngineeringProjects.com
GPS
Module
Virtual Terminal
LCD1
LAT:30.24
LON:-97.82

Assignment – 13

(a) Create arduino based IOT project for Password Based Door looking system.

```
#include <LiquidCrystal.h>
#include <Servo.h>
#include <Keypad.h>
Servo myservo;
int pos = 0; // position of servo motor
LiquidCrystal lcd(A4, A5, A3, A2, A1, A0);
const byte rows = 4;
const byte cols = 3;
char key[rows][cols] = {
   {'1', '2', '3'},
   {'4', '5', '6'},
   {'7', '8', '9'},
   {'*', '0', '#'}};
byte rowPins[rows] = {0, 1, 2, 3};
byte colPins[cols] = {4, 5, 6};
Keypad keypad = Keypad(makeKeymap(key), rowPins, colPins, rows, cols);
char *password = "0123";
int currentposition = 0;
void setup()
{
   displayscreen();
   myservo.attach(9); //Servo motor connection
   lcd.begin(16, 2);
}
void loop()
```

```
{
  if (currentposition == 0)
  {
    displayscreen();
  }
  int l;
  char code = keypad.getKey();
  if (code != NO_KEY)
  {
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("PASSWORD:");
    lcd.setCursor(7, 1);
    lcd.print(" ");
    lcd.setCursor(7, 1);
    for (l = 0; l <= currentposition; ++l)
    {
      lcd.print("*");
      //keypress();
    }
    if (code == password[currentposition])
    {
      ++currentposition;
      if (currentposition == 4)
      {
        unlockdoor();
        currentposition = 0;
      }
    }
    else
```

```
        {
          incorrect();
          currentposition = 0;
        }
    }
}
//------------------ Function 1- OPEN THE DOOR--------------//
void unlockdoor()
{
   delay(900);
   lcd.setCursor(0, 0);
   lcd.println(" ");
   lcd.setCursor(1, 0);
   lcd.print("Access Granted");
   lcd.setCursor(4, 1);
   lcd.println("WELCOME!!");
   lcd.setCursor(15, 1);
   lcd.println(" ");
   lcd.setCursor(16, 1);
   lcd.println(" ");
   lcd.setCursor(14, 1);
   lcd.println(" ");
   lcd.setCursor(13, 1);
   lcd.println(" ");

   for (pos = 180; pos >= 0; pos -= 5) // open the door
   {
      myservo.write(pos);
      delay(5);
   }
```

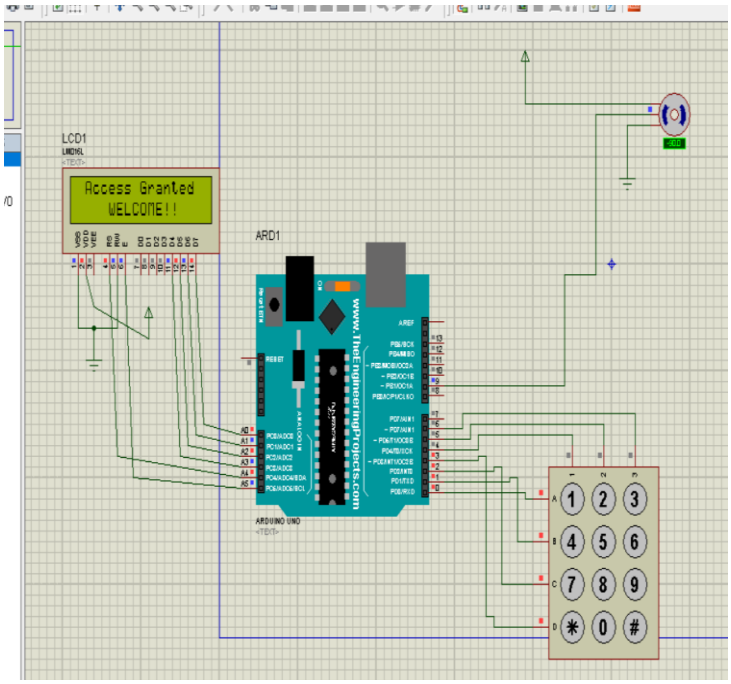
```
    delay(2000);
    delay(1000);
    counterbeep();
    delay(1000);

    for (pos = 0; pos <= 180; pos += 5) // close the door
    {                                   // in steps of 1 degree
      myservo.write(pos);
      delay(15);
      currentposition = 0;
      lcd.clear();
      displayscreen();
    }
  }
  //---------------------Function 2- Wrong code--------------//
  void incorrect()
  {
    delay(500);
    lcd.clear();
    lcd.setCursor(1, 0);
    lcd.print("CODE");
    lcd.setCursor(6, 0);
    lcd.print("INCORRECT");
    lcd.setCursor(15, 1);
    lcd.println(" ");
    lcd.setCursor(4, 1);
    lcd.println("GET AWAY!!!");

    lcd.setCursor(13, 1);
    lcd.println(" ");
```

```
    Serial.println("CODE INCORRECT YOU ARE UNAUTHORIZED");
    delay(3000);
    lcd.clear();
    displayscreen();
  }
  //-------Function 3 - CLEAR THE SCREEN--------------------/
  void clearscreen()
  {
    lcd.setCursor(0, 0);
    lcd.println(" ");
    lcd.setCursor(0, 1);
    lcd.println(" ");
    lcd.setCursor(0, 2);
    lcd.println(" ");
    lcd.setCursor(0, 3);
    lcd.println(" ");
  }
  //------------Function 4 - DISPLAY FUNCTION--------------------//
  void displayscreen()
  {
    lcd.setCursor(0, 0);
    lcd.println("*ENTER THE CODE*");
    lcd.setCursor(1, 1);
    lcd.println("TO OPEN DOOR!!");
  }
  //--------------Function 5 - Count down------------------//
  void counterbeep()
  {
    delay(1200);
    lcd.clear();
```

```
lcd.setCursor(2, 15);
lcd.println(" ");
lcd.setCursor(2, 14);
lcd.println(" ");
lcd.setCursor(2, 0);
delay(200);
lcd.println("GET IN WITHIN:::");
lcd.setCursor(4, 1);
lcd.print("5");
delay(200);
lcd.clear();
lcd.setCursor(2, 0);
lcd.println("GET IN WITHIN:");
delay(1000);
lcd.setCursor(2, 0);
lcd.println("GET IN WITHIN:");
lcd.setCursor(4, 1); //2
lcd.print("4");
delay(100);
lcd.clear();
lcd.setCursor(2, 0);
lcd.println("GET IN WITHIN:");
delay(1000);

lcd.setCursor(2, 0);
lcd.println("GET IN WITHIN:");
lcd.setCursor(4, 1);
lcd.print("3");
delay(100);
lcd.clear();
```

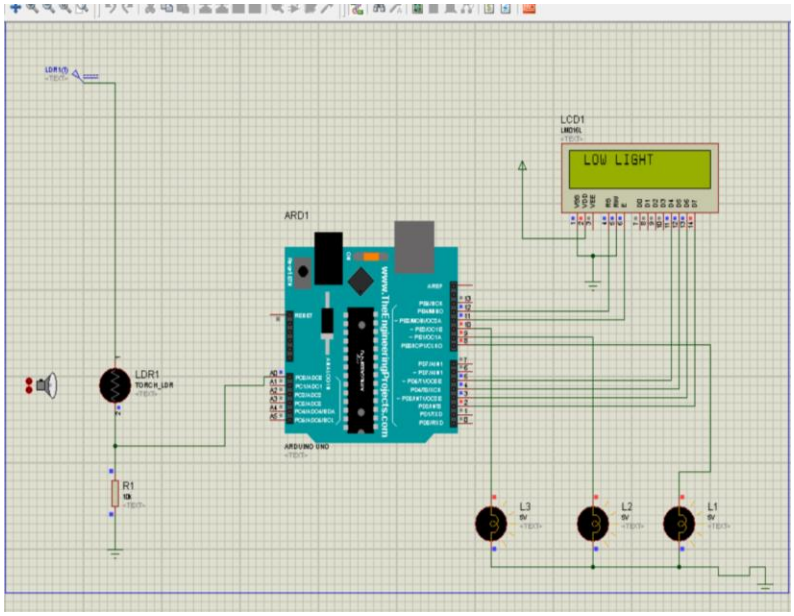
```
lcd.setCursor(2, 0);
lcd.println("GET IN WITHIN:");
delay(1000);

lcd.setCursor(2, 0);
lcd.println("GET IN WITHIN:");
lcd.setCursor(4, 1);
lcd.print("2");
delay(100);
lcd.clear();
lcd.setCursor(2, 0);
lcd.println("GET IN WITHIN:");
delay(1000);

lcd.setCursor(4, 1);
lcd.print("1");
delay(100);
lcd.clear();
lcd.setCursor(2, 0);
lcd.println("GET IN WITHIN::");
delay(1000);
delay(40);
lcd.clear();
lcd.setCursor(2, 0);
lcd.print("RE-LOCKING");
delay(500);
lcd.setCursor(12, 0);
lcd.print(".");
delay(500);
lcd.setCursor(13, 0);
```

```
    lcd.print(".");
    delay(500);
    lcd.setCursor(14, 0);
    lcd.print(".");
    delay(400);
    lcd.clear();
    lcd.setCursor(4, 0);
    lcd.print("LOCKED!");
    delay(440);
}
```



Assignment – 14

(a) Create arduino based IOT project for smart street light.

```
#include <LiquidCrystal.h>
int a, b, c;
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
void setup()
{
   pinMode(8, OUTPUT);
   pinMode(9, OUTPUT);
   pinMode(10, OUTPUT);
   lcd.clear();
   lcd.begin(16, 2);
   lcd.setCursor(1, 0);
   lcd.print("Street Light");
   lcd.setCursor(1, 1);
   lcd.print("Project");
   delay(1000);
   lcd.clear();
}
void loop()
{
   a = analogRead(A0);
   b = map(a, 0, 1023, 0, 255);
   if (b > 110)
   {
      lcd.setCursor(1, 0);
      lcd.print("NORMAL LIGHT");
      digitalWrite(8, LOW);
```

```
        digitalWrite(9, LOW);
        digitalWrite(10, LOW);
    }
    if (b < 110)
    {
        lcd.setCursor(1, 0);
        lcd.print("LOW LIGHT");
        digitalWrite(8, HIGH);
        digitalWrite(9, HIGH);
        digitalWrite(10, HIGH);
    }
}
```