PROJECT REPORT - ICLR15

Develop a security system for main entrance
- Can able change password
- Use suitable sensors and actuators for opening and closing of gate
- Based on the object size gate wants to open

Submitted by –
- Pratik Khandare -11020052

Date

16/12/2022

Course title

Microprocessors and Microcontrollers laboratory

Content

Page no-

- About the Project and Board used 02
- About Board 02
- Pin configuration 03
- Main components used 04
- Code 05
- Pin connections 10
- Schematic connections 10
- Procedure 11
- Conclusion 12
- Summary 13

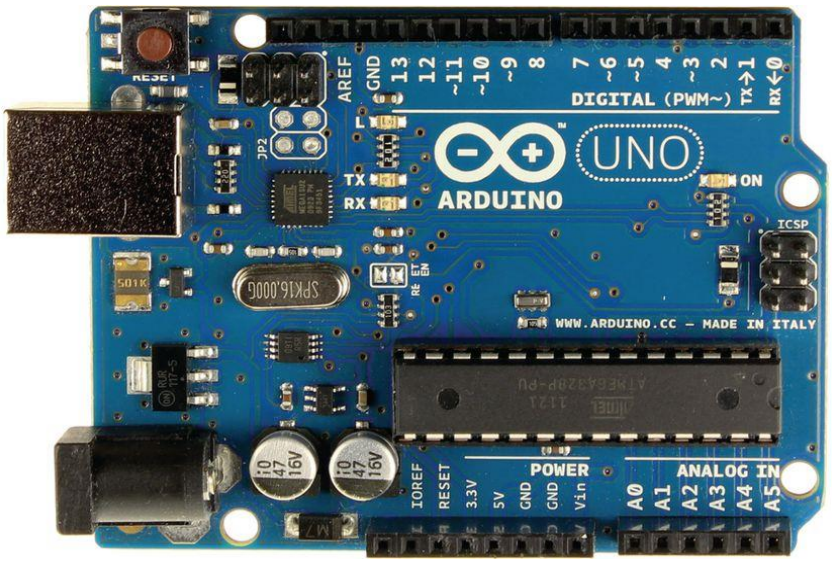
About Project

The project aims for security system for the main entrance. Various features are employed such as setting and changing of password, Automatic opening and closing of gate based on object size. The project can be called as innovation in the field of home automation.

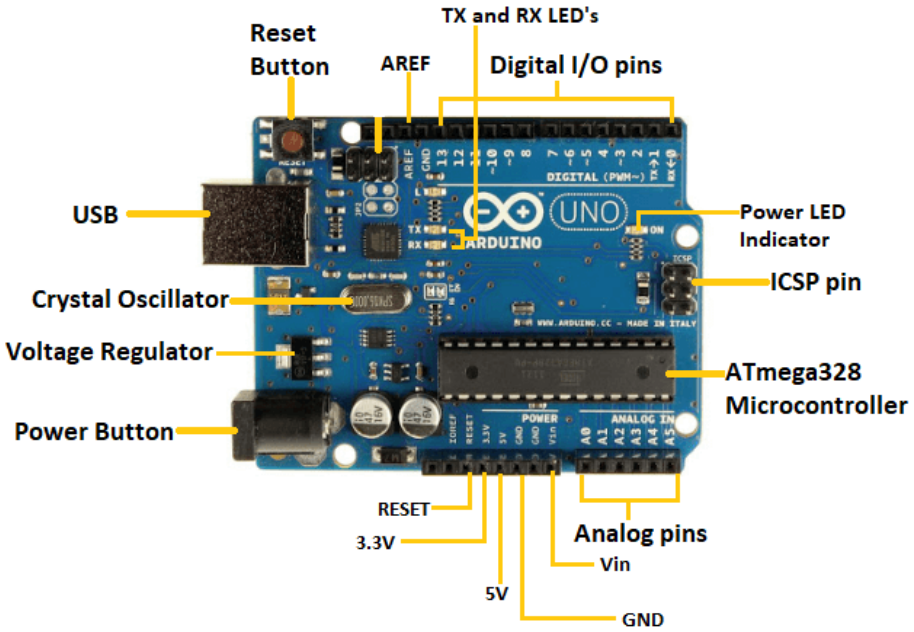
Board used: Arduino Uno Board

About board

The Arduino UNO is a standard board of Arduino. It was also the first USB board released by Arduino. It is considered as the powerful board used in various projects. Arduino.cc developed the Arduino UNO board. Arduino UNO is based on an ATmega328P microcontroller. It is easy to use compared to other boards, such as the Arduino Mega board, etc. The board consists of digital and analog Input/Output pins (I/O), shields, and other circuits. The Arduino UNO includes 6 analog pin inputs, 14 digital pins, a USB connector, a power jack, and an ICSP (In-Circuit Serial Programming) header. It is programmed based on IDE, which stands for Integrated Development Environment. It can run on both online and offline platforms.



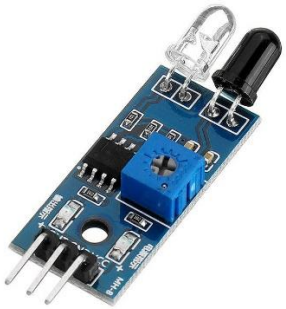
Pin configuration



- **ICSP pin** - The In-Circuit Serial Programming pin allows the user to program using the firmware of the Arduino board.
- **Power LED Indicator**- The ON status of LED shows the power is activated. When the power is OFF, the LED will not light up.
- **Digital I/O pins**- The digital pins have the value HIGH or LOW. The pins numbered from D0 to D13 are digital pins.
- **TX and RX LED's**- The successful flow of data is represented by the lighting of these LED's.
- **AREF-** The Analog Reference (AREF) pin is used to feed a reference voltage to the Arduino UNO board from the external power supply.
- **Reset button**- It is used to add a Reset button to the connection.
- **USB**- It allows the board to connect to the computer. It is essential for the programming of the Arduino UNO board.
- **Crystal Oscillator**- The Crystal oscillator has a frequency of 16MHz, which makes the Arduino UNO a powerful board.
- **Voltage Regulator**- The voltage regulator converts the input voltage to 5V.
- **GND**- Ground pins. The ground pin acts as a pin with zero voltage.
- **Vin**- It is the input voltage.
- **Analog Pins**- The pins numbered from A0 to A5 are analog pins. The function of Analog pins is to read the analog sensor used in the connection. It can also act as GPIO (General Purpose Input Output) pins.

Main Components Used

1. Arduino board
2. Breadboard
3. 5 V Servomotor
4. IR sensor
5. Buzzer
6. 4*4 Membrane Keypad



Code

```
#include<Keypad.h>
#include<Servo.h>

#define IR_1 10
#define IR_2 11
#define BUZZ A2

Servo Gate1;
Servo Gate2;

const byte ROWS = 4; //four rows
const byte COLS = 4; //four columns

char RIG_PIN[10] = "1234";
char ENT_PIN[10], NEW_PIN[10];
int i = 0;
char key = 0;
char keys[ROWS][COLS] =
{
  {'1','2','3','A'},
  {'4','5','6','B'},
  {'7','8','9','C'},
  {'*','0','#','D'}
};

byte rowPins[ROWS] = {9, 8, 7, 6}; //connect to the row pinouts of the keypad
byte colPins[COLS] = {5, 4, 3, 2}; //connect to the column pinouts of the keypad

//Create an object of keypad
Keypad keypad = Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS );

void setup()
{
  Serial.begin(9600);
  Gate1.attach(12);
  Gate2.attach(13);
  pinMode(IR_1,INPUT);
  pinMode(IR_2,INPUT);
  pinMode(BUZZ,OUTPUT);
  Serial.println("To change password, press #");
  Serial.print("Enter Password : ");
  Gate1.write(0);
  Gate2.write(180);
}

void Open_Gate()
{
  int pos1,pos2;
```

```
  delay(2000);
  bool Value_1 = digitalRead(IR_1);
  bool Value_2 = digitalRead(IR_2);
  if((Value_1 == HIGH && Value_2 == LOW)||(Value_1 == LOW && Value_2 == HIGH))
  {
   Serial.println("Half Gate Open");
   for(pos1=0,pos2=180;pos1<=75,pos2>=105;pos1++,pos2--)
   {
    Gate1.write(pos1);
    Gate2.write(pos2);
    delay(35);
   }
   delay(4000);

   for(pos1=75,pos2=105;pos1>=0,pos2<=180;pos1--,pos2++)
   {
    Gate1.write(pos1);
    Gate2.write(pos2);
    delay(35);
   }
  }
  else if(Value_1 == LOW && Value_2 == LOW)
  {
   Serial.println("Full Gate Open");
   for(pos1=0,pos2=180;pos1<=135,pos2>=45;pos1++,pos2--)
   {
    Gate1.write(pos1);
    Gate2.write(pos2);
    delay(35);
   }
   delay(6000);

   for(pos1=135,pos2=45;pos1>=0,pos2<=180;pos1--,pos2++)
   {
    Gate1.write(pos1);
    Gate2.write(pos2);
    delay(35);
   }
  }
  else
  {
   Serial.println("Entrant Can't be Identified.");
   Serial.println("Please try again");
  }
}

void Beep()
{
  tone(BUZZ,1000);
  delay(500);
```

```
  noTone(BUZZ);
}

void Correct_Beep()
{
  delay(1000);
  tone(BUZZ,700);
  delay(200);
  noTone(BUZZ);
  delay(200);
  tone(BUZZ,700);
  delay(200);
  noTone(BUZZ);
  tone(BUZZ,700);
  delay(200);
  noTone(BUZZ);
}

void Wrong_Beep()
{
  delay(1000);
  tone(BUZZ,1200);
  delay(600);
  noTone(BUZZ);
  delay(600);
  tone(BUZZ,1200);
  delay(600);
  noTone(BUZZ);
  delay(600);
  tone(BUZZ,1200);
  delay(600);
  noTone(BUZZ);
}

void Change_Pass()
{
  int j=0;
  Serial.print("\nEnter your current Password : ");
  while(j<4)
  {
    char key = keypad.getKey();
    if(key)
    {
      Serial.print("*");
      ENT_PIN[j++] = key;
      Beep();
    }
    key=0;
  }
  Serial.println("");
```

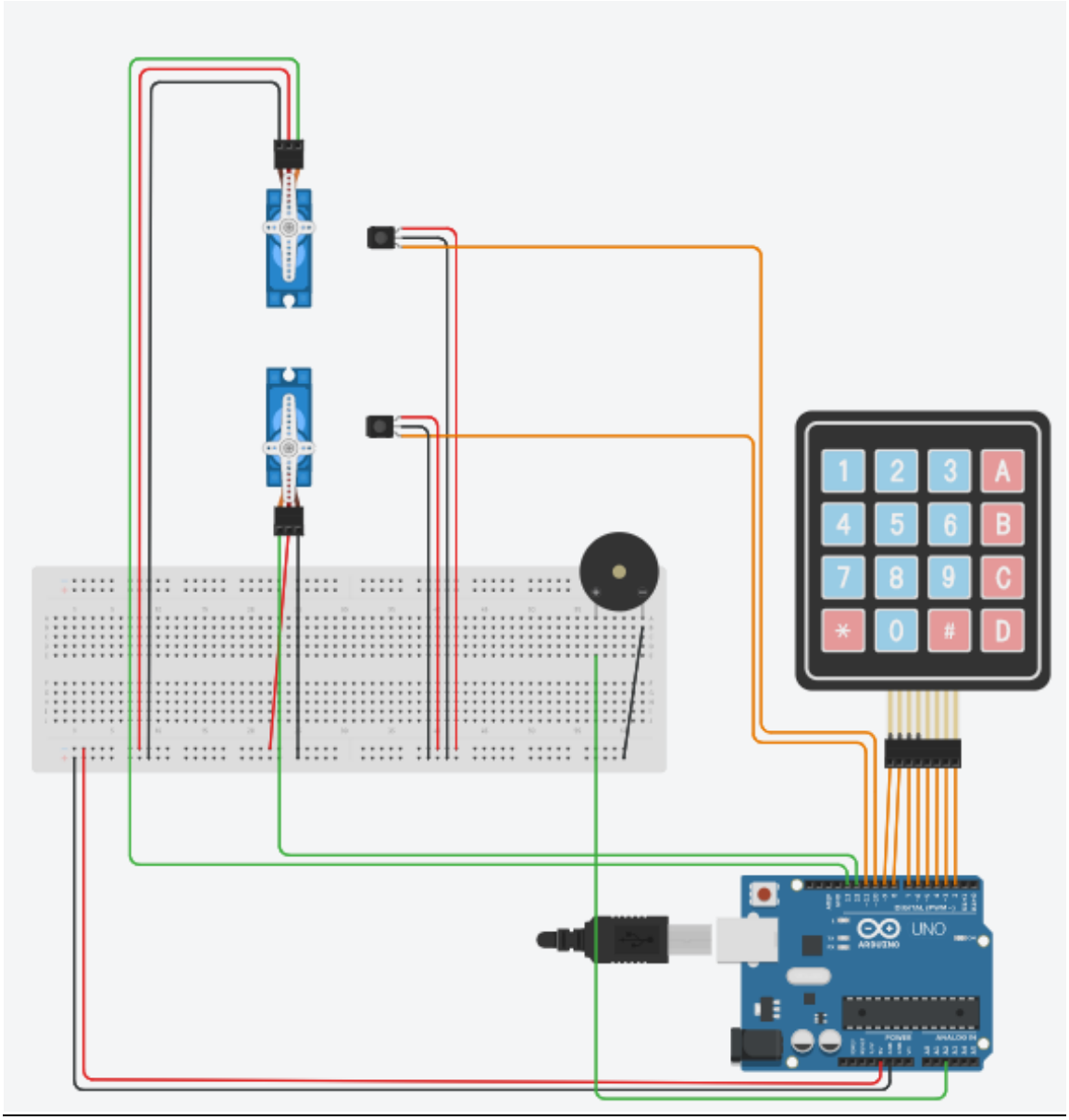
```
  delay(500);
  if(strncmp(RIG_PIN,ENT_PIN, 4) == 0)
  {
    j=0;
    Serial.print("Enter New Password:");
    while(j<4)
    {
      char key= keypad.getKey();
      if(key)
      {
        Serial.print("*");
        NEW_PIN[j++]=key;
        Beep();
      }
    }
    strncpy(RIG_PIN,NEW_PIN,4);
    delay(1000);
    Correct_Beep();
    Serial.print("\nPassword Successfully changed!");
    delay(1000);
  }
  else
  {
    Serial.println("Wrong Password");
    Serial.println("Better Luck Again");
    Delay(1000);
    Wrong_Beep();
    delay(1000);
  }
  key = 0;
  Serial.print("\nEnter Password : ");
}

void loop()
{
  char key = keypad.getKey();// Read the key
  if(key == '#')
  {
    Beep();
    Change_Pass();
  }
  if(key)
  {
    Serial.print("*");
    ENT_PIN[i++] = key;
    Beep();
  }
  if(i==4)
  {
    if(strncmp(ENT_PIN,RIG_PIN,4)==0)
```

```
  {
    Serial.println("");
    Serial.println("Access Granted");
    Correct_Beep();
    Open_Gate();
    i=0;
    Serial.print("Enter Password : ");
  }
  else
  {
    Serial.println("");
    Serial.println("Access Denied");
    Wrong_Beep();
    i=0;
    Serial.print("Enter Password : ");
  }
 }
}
```

Pin connections

1. Pins 2 to 9: Membrane Keypad connections - Inputs
2. Pins 10 and 11: Infrared Sensor - Inputs
3. Pins 12 and 13: Servo Motors – Outputs
4. Pin A2: Buzzer – Output
5. 5V Pin: To all Vcc pins in sensors and actuators.
6. GND Pin: To all ground pins in sensors and actuators.

Schematic connections



Procedure

1] Connect the Arduino board with your laptop and setup Arduino ide on your laptop where you will upload the code.

2] Upload the code to Arduino board through IDE.

3] The setup is ready; Now enters the object near the gate. near the gate enter the password

4] The object has to enter the password in the keypad.

5] In case you want to change the password, click on ‘ # ’ on the keypad.

6] Enter the password on the keypad.

7] Now, the gate will open as per the object size. If the object is small, it will open to 75 degrees whereas if the object is large it will open to 135 degrees.

8] Enter the object inside the gate and it will close at the delay of 6 sec.

9] In case of wrong password, Buzzer will start making noise.

10] The project was successfully demonstrated.

Conclusion

As far as the technology is developing, security is important in various field. To ensure good security system, the above project can be helpful. Automation in open and closing is new aspect that we can use in upcoming time. The password security system can again be further developed by adding the camera-based surveillance. And we can open and close door directly. Many innovations are further to be introduced in this project. As conclusion of this project, this is the good project for the security system and automation in door opening and closing system.

Summary

The project is basically based on security system and automatic opening and closing of the gate. The use of components is depicted in the above diagram. The opening and closing of gate are done by servo motor and IR sensors are used to detect the object and its size. The keypad is used to provide password to the system and thus, the door will open. The security in this case is ideal. The buzzer is fixed at the breadboard to get signal if the password is inputted incorrectly. There are various application of this project and various advancement can be done here. The project depicts the new technology that must enter for security and automation purposes. The limitation for this project is very less. Some recommendation for this project is object length measurement and incase of encryption the security system may fall. But overall, it is good project in case of learning as well as in day-to-day purpose.