تدريب

الاسم : روان سامي جمال الدين علي الربيعي .

كلية : الهندسة – **جامعة** اسكندرية .

التخصص : هندسة الكمبيوتر و الاتصالات البرامج العلمية المتخصصة.

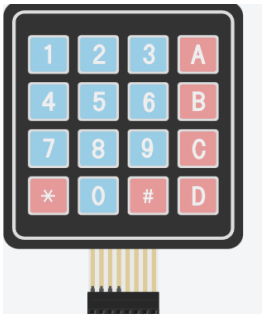
Smart Home Project with Tinkercad Simulation

1-Definition of a Smart Home:

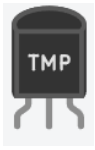
- A smart home refers to a convenient home setup where appliances and devices can be automatically controlled remotely from anywhere with an internet connection using a mobile app or any other networked device.
- Each function a house can do is controlled by a precise specific sensor so the errors resulting are very low and can be controlled.
- Therefore, Security and efficiency are the main reasons behind the increase in smart home technology use.

2-Components used in my project:

1-key pad: used as a smart lock for the house.

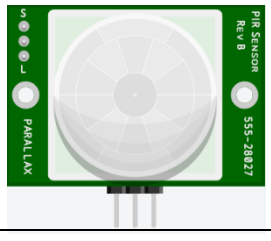


2-Temprature sensor: used to measure room temperature in “C”.



3-Smoke sensor: Combines with temperature sensor to make a whole fire detecting system.

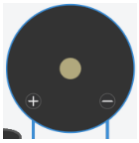
4-PIR sensor: used as a smart gate where is detects if there is a motion nearby the sensor and opens the gate automatically.



5-LDR sensor: used as a smart lamp to control light intensity produced.

6-Servo: used as a gate.

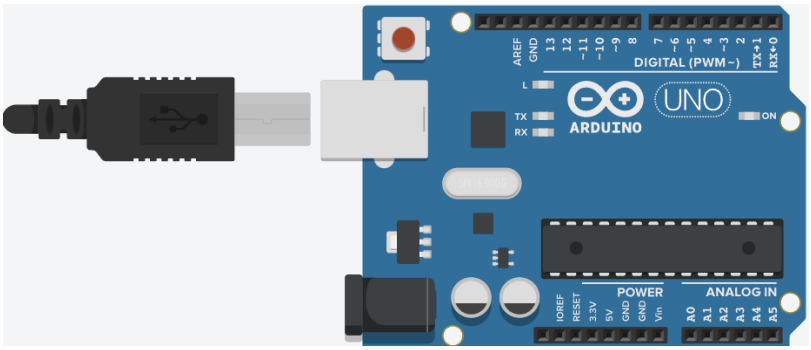
7-Buzzer: to produce an alarming tone.



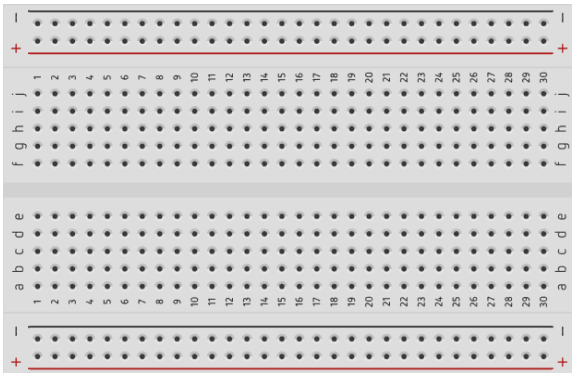
8-LEDs: to show open or closed states.

9-Resistence: 220 ohms to protects the LEDs from being burnt by 5v.

9-Arduino: provides breadboard with 5v and ground as well as it contains digital and analog pins used to connect my components with.



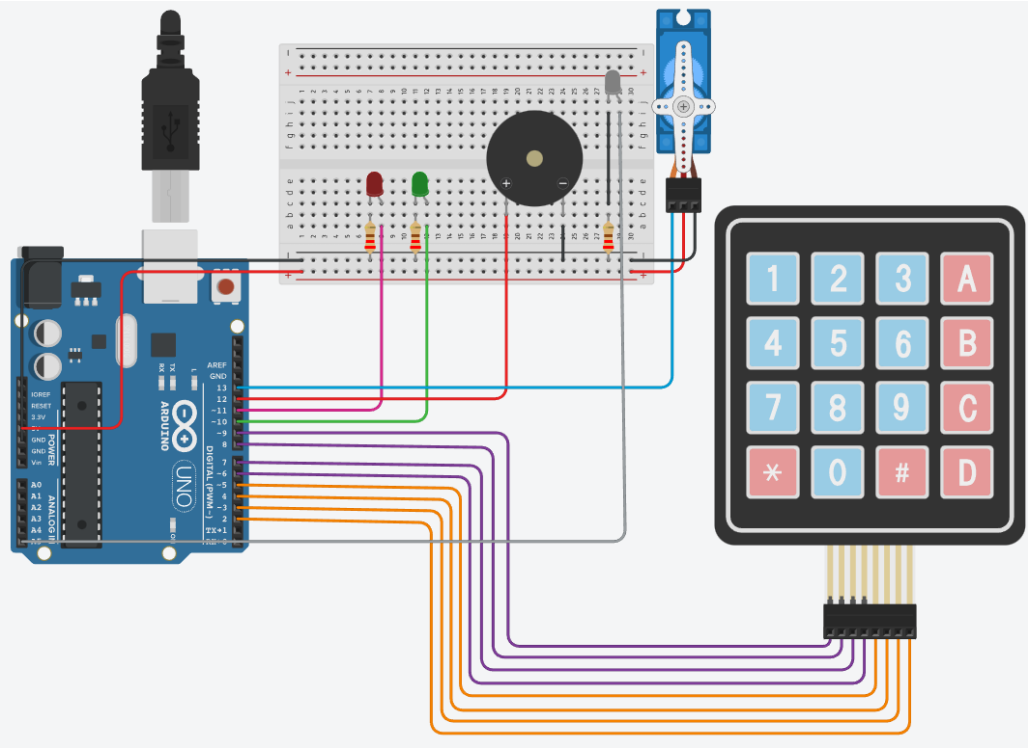
10-Breadboard: provides me with more nodes to connect more than one component at the same time.



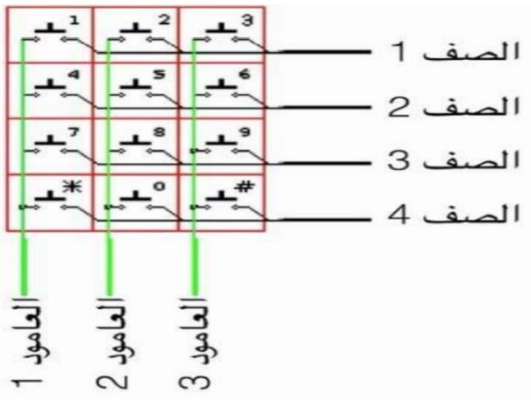
3-Operations that can by performed in my project:

1-Smart Security Lock:

My circuit design:



- Main component used here is the “keypad”.
- Pins from 1-4 are rows and from 5-8 are columns.



Function:

1-User enters the password directly on the keypad.

2-If the password entered is incorrect:

- Red LED flashes and “ACCESS DENIED” is printed in the serial monitor.
- Then buzzer gives a warning tone then asks the user to re-enter the password again.

3- If the password entered is correct:

- the green LED flashes and “ACCESS ALLOWED” is printed in the serial monitor.
- The servo arm starts moving slowly (as if it is the door’s house) then it gives enough time for the people to enter and closes again.
- White LED turns on (as if house light opens automatically).
- The compiler will not allow the user to enter any further passcodes.

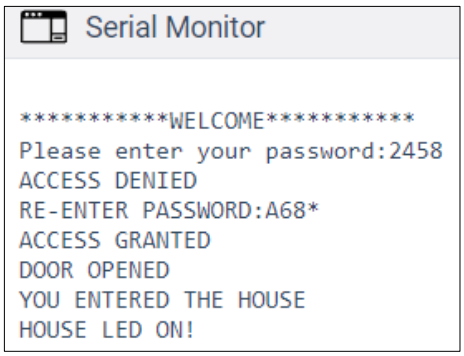
Code:

```
#include <Servo.h>

Servo myservo;

int pos = 0;

#include <Keypad.h>

const byte ROWS = 4; //four rows

const byte COLS = 4; //four columns

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}; //purple wires

byte colPins[COLS] = {5, 4, 3,2};  //orange wires

Keypad keypad = Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS );

String passcode= ""; //variable in which password is stored

int count=0;

int i;

int BUZZ = 12;
```

```
void setup(){
 Serial.begin(9600);
 Serial.println("");
 Serial.println("**********WELCOME**********");
 Serial.print("Please enter your password:");
 pinMode(11, OUTPUT); //red led
 pinMode(10, OUTPUT); //green led
 pinMode(BUZZ,OUTPUT);
 myservo.attach(13);
 pinMode(A5, OUTPUT);  //House led  }
void loop(){
 myservo.write(0);
 char key = keypad.getKey();
 if (key){
   Serial.print(key);
   count=count+1;
   passcode=passcode+key;
   if(count==4){
     if (passcode=="A68*") {
       Serial.println("");
       Serial.println("ACCESS GRANTED");
       for(i=0;i<6;i++){  //led blinks 7 times
         digitalWrite(10, HIGH); // set the LED on
         delay(100); // wait for 0.1 second
         digitalWrite(10, LOW);
         delay(100); }
       for (pos = 0; pos <= 90; pos++) {
         myservo.write(pos);
         digitalWrite(BUZZ,HIGH);
         delay(15); }
```

```
  digitalWrite(BUZZ,LOW);
  delay(1000);
  analogWrite(A5, 255);
  delay(600);
  Serial.println("DOOR OPENED");
  Serial.println("YOU ENTERED THE HOUSE");
  Serial.println("HOUSE LED ON!");
  for (pos = 90; pos >= 0; pos--) {
    myservo.write(pos);
    delay(15); }
  exit(0); }
 else{
  Serial.println("");
  Serial.println("ACCESS DENIED");
  for(i=0;i<3;i++){  //led blinks 4 times
    digitalWrite(11, HIGH); // set the LED on
    tone(BUZZ, 1250, 100);//rings the buzzer
    delay(100); // wait for 0.1 second
    tone(BUZZ, 1250, 200);//rings the buzzer
    delay(200); // wait for 0.25 second
    digitalWrite(11, LOW);
    delay(100); }
  tone(BUZZ, 1250, 600);//rings the buzzer
  delay(600); // wait for 0.6 second
  Serial.print("RE-ENTER PASSWORD:");
  passcode="";
  count=0;
 } } }}
```

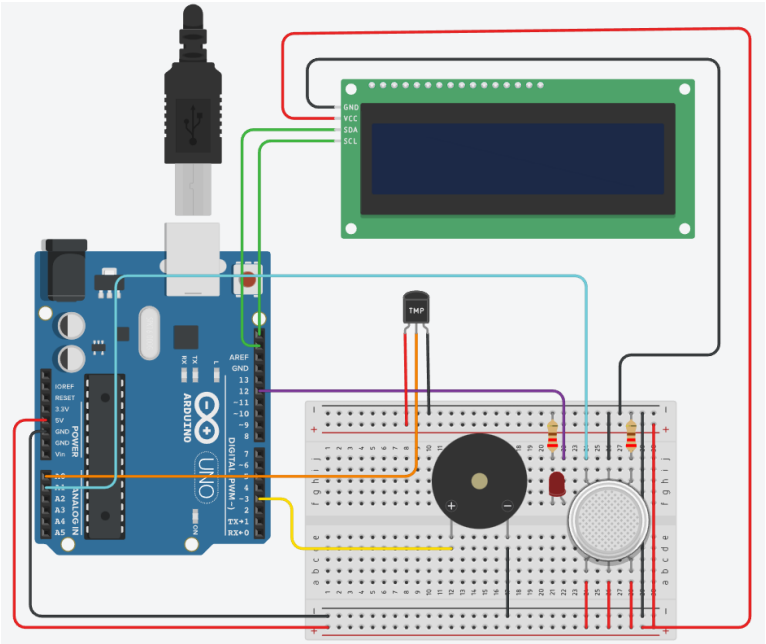
Serial monitor output:



Online simulation link: https://www.tinkercad.com/things/bnlydT5X1rm .

2-Fire Alarming System:

My circuit design:



- Main components used is temperature sensor, smoke sensor and LCD I2C.
  1. **LM35** is a temperature measuring device having an analog output voltage proportional to the temperature. It has 3-terminals used to measure surrounding temperature precisely ranging from -55 °C to 150 °C.

  2. **MQ2 Gas sensor** can detect LPG, Smoke, Alcohol concentrations in air. The analog output voltage provided by the sensor changes in proportional to the concentration of smoke/gas. The greater the gas concentration, the higher is the output voltage, while lesser gas concentration results in low output voltage.

3. **I2C protocol** has SCL that generates synchronized clock signal and SDA used to carry data which made an easier transformation of data to LCD than the LCD display without I2C.

Function:

- Temperature sensor measures atmosphere temperature and displays it in LCD and 4 cases can occur:

1- Temp >= 21 and temp<27 → GOOD WEATHER will be displayed on LCD.
2- Temp < 21 → HEATER ON will be displayed on LCD.
3- Temp >= 27 and temp <= 60 → FAN ON! will be displayed on LCD.
4- Temp > 60 here it checks the smoke sensor as well:
1-if smoke sensor reads < 30→ TEMP TOO HIGH -- FIRE EXPECTED will be displayed on LCD which means that no smoke occurs.
2-if smoke sensor reads >= 30→ FIRE DETECTED will be displayed on LCD and the buzzer produces an alarming tone indicating the presence of a fire.

Code:

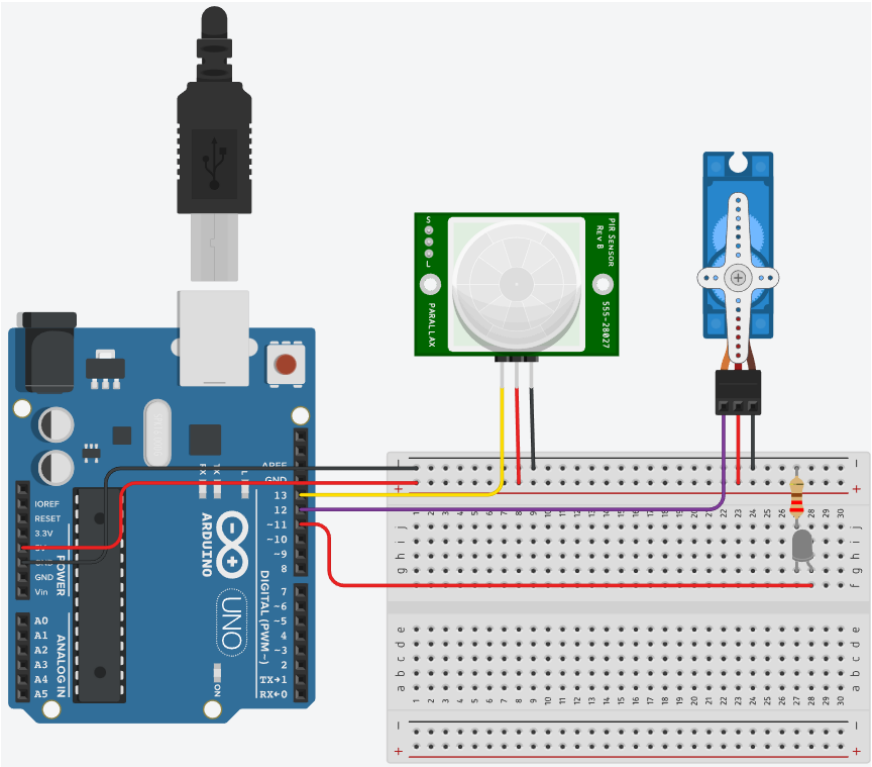
```
#include <LiquidCrystal_I2C.h>
#include <Adafruit_LiquidCrystal.h>
Adafruit_LiquidCrystal lcd(0);
#define tempPin A0
#define smoke A1
float Reading;
float temp;
int smokeValue;
int red = 12;
int buz=3;

void setup() {
 lcd.begin(16,2);
 lcd.setBacklight(1);
 pinMode(tempPin,INPUT);
 pinMode(red, OUTPUT);
 pinMode(buz, OUTPUT);
}

void loop() {
Reading = analogRead(tempPin);
temp=Reading/1024;
temp=(temp*5 -0.5)*100;
lcd.print("TEMP =   ");
```

```
lcd.setCursor(8,0);
lcd.print(temp);
lcd.print("  C");
lcd.setCursor(1,1);
//delay(3000);
  if(temp>=21 && temp<27){
  digitalWrite(red, LOW);
  lcd.print("GOOD WEATHER :)");
}
else if(temp>=27 && temp<=60)
{
  lcd.print("        ");
  digitalWrite(red, HIGH);
  lcd.print("FAN ON!");
  lcd.setCursor(0,0);
}
else if(temp>60){
  digitalWrite(red, LOW);
  lcd.print("                ");
  lcd.setCursor(0,1);
  lcd.println(" TEMP TOO HIGH  ");
  delay(500);
  lcd.setCursor(0,1);
  lcd.println(" FIRE  EXPECTED  ");
  delay(500);
  smokeValue = analogRead(smoke);
  while(smokeValue >=30){
     lcd.setCursor(0,1);
     lcd.println("FIRE  DETECTED!");
     tone(3,1000,100);
     delay(10);
     tone(3,1000,100);
     delay(10);
     smokeValue = analogRead(smoke);}
  lcd.setCursor(0,0);
}
else if (temp<21){
  lcd.clear();
  lcd.print("TEMP = ");
  lcd.setCursor(8,0);
  lcd.print(temp);
  lcd.print(" C");
  lcd.setCursor(3,1);
  digitalWrite(red, HIGH);
  lcd.print("HEATER ON");
  lcd.setCursor(0,0);  }}
```

Online simulation link: https://www.tinkercad.com/things/cM0yJ5LxXHQ .

3-Smart Gate:

My circuit design:



- Main component is PIR sensor.
  1- **PIR** stands for Pyroelectric Infrared Radial, and it detects the changes in the infrared light across certain distance and gives out an electrical signal as its output in response to a detected IR signal. It can detect any infrared emitting object such as human beings or animals whether it is the range of the sensor, or moves away from the range, or moves within the range of the sensor.

Function:

The value of PIR is taken:

1-If it equals 1 means the PIR detects a moving body therefore the servo arms moves acting as the rooms door

2-If it does not equal 1:

- If the servo arm was opened it closes.
- If it is closed it remains as it is.

Code:

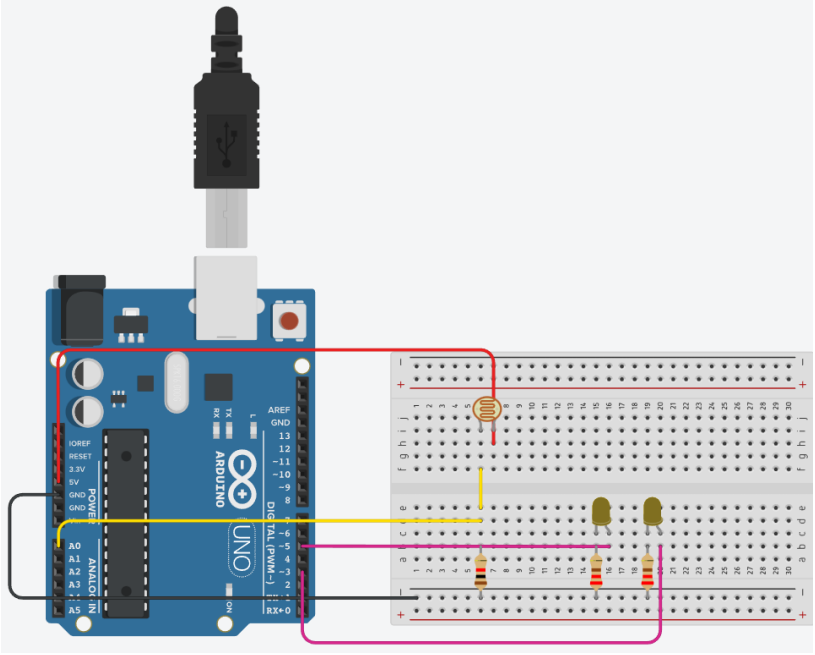
```
#include <Servo.h>
Servo myservo;
int pos = 0;
int pirPin = 13;
int pirValue;
int motionDetected = 0;
int led=11;
void setup(){
  Serial.begin(9600);
  pinMode(pirPin, INPUT);
  myservo.attach(12);
  myservo.write(pos);
  pinMode(led, OUTPUT);  }
void loop() {
  pirValue = digitalRead(pirPin);
  Serial.println(pirValue);
  if(pirValue = = 1) {
    motionDetected = 1;  }
  else  {
   if(pos!=0){
        Serial.println("3");
        for (pos = 90; pos >= 0; pos--) { //closing the door
          myservo.write(pos);
          delay(15); }}
        pos=0;  }
  while(motionDetected = = 1) {
    if(pos==0){
      Serial.println("1");
      for (pos = 0; pos <= 90; pos++) {  //opening the door
```

```
      myservo.write(pos);
      delay(15);  }
    digitalWrite(led,HIGH);  }
  else{
    Serial.println("2");
     myservo.write(90);
    motionDetected = 0;
    }  } }
```

Online simulation link: https://www.tinkercad.com/things/5NODoip4Bjk .

4-Smart light:

My circuit design:



- Main component is LDR sensor:
  1- An LDR is a component that has a variable resistance that changes with the light intensity that falls upon it.
  2- The most common type of LDR has a resistance that falls with an increase in the light intensity falling upon the device.
  Example: Daylight = 5000Ω and Dark = 20MΩ.

Function:

1-The resistance read by LDR is translated to brightness of the LED so :

- If resistance is low which means that we are daylight, therefore the LED doesn't emit any light
- As the resistance increases gradually which means that we are going to dark mode the intensity of the light produced by the LED starts to increase gradually as well.

Code:

```
#define ldr A0
#define led1 3
#define led3 5
int resistance;
int ledBrightness=0;
int threshold=400;
void setup(){
 Serial.begin(9600);
 pinMode(led1,OUTPUT);
 pinMode(led3,OUTPUT);
 pinMode(ldr,INPUT);
}
void loop(){
 resistance= analogRead(ldr);
 Serial.print("RESISTANCE : ");
 Serial.println(resistance);
 if(resistance<threshold){
   ledBrightness = map(resistance, 6, 508, 0, 128);
   analogWrite(led1,ledBrightness);
   analogWrite(led3,ledBrightness);
   Serial.println(ledBrightness);}
```

```
  else{
   ledBrightness = map(resistance, 508, 1023, 128, 255);
   analogWrite(led1,ledBrightness);
   analogWrite(led3,ledBrightness);
   Serial.println(ledBrightness);}
}
```

Online simulation link: https://www.tinkercad.com/things/fd7FxTY5CRT .

Real life Maquette: